

**.Net programuotojo 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 (.Net programuotojo 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:

Vytautas Gaidys

Virginija Putnaitė

Laura Vaišnorė

Rita Daukšienė

Jaroslav Grablevski

Jolita Bačkienė

# Modulis „Įvadas į profesiją“

## *TESTAS ĮSIVERTINTI GEBĖJIMAMS PRIEŠ PRADEDANT MOKYTIS*

1. Kiek bitų sudaro vieną baitą?
   1. 1 bitas
   2. 8 bitai
   3. Atvirkščiai – tai keli baitai sudaro vieną bitą
   4. 2 bitai
2. Kurio dešimtainės skaičiavimo sistemos skaičiaus dvejetainė išraiška yra 10110111?
   1. 138
   2. 183
   3. 191
3. Kokia galėjo būti kintamojo C reikšmė, jeigu, įvykdžius programos fragmentą, buvo išspausdintas skaičius 4?

A := 2; R := 9; C :=?;

while A < R do

if A > C

then R := R - 5

else A := A + 2;

Write(A);

* 1. 4
  2. 17
  3. 3

1. Kiek baitų sudaro kilobaitą
   1. 1000
   2. 1024
   3. 1000000
   4. 8096
2. Kas yra IP adresas?
   1. kompiuterio identifikatorius IP tinkluose
   2. interneto vartotojo registracijos adresas
   3. failo kietajame diske sektoriaus numeris
3. Duota funkcija. Koks bus funkcijos kintamojo x rezultatas, kai bus įvesti skaitmenys: a = 3, b = 42, c = 5?

class Foo {

public void myFoo(int a, int b, int c) {

int x = (a + b) + (a % c) - b;

}

}

* 1. 4
  2. 8
  3. 6

1. Kas būdinga atvirajai programinei įrangai (angl. *open source*)?
   1. nemokamai galima naudotis tik ribotu šios programinės įrangos funkcionalumu
   2. ši įranga yra nemokama. Jos pirminį tekstą (angl. *source code*) galima keisti
   3. šia įranga galima naudotis laisvai tik su autoriaus leidimu
2. Su kuria natūraliojo skaičiaus X reikšme loginio pateikto reiškinio reikšmė bus TRUE?
3. ((x > 18) or (x <> 19)) and (x = 18)
   1. 18
   2. 19
   3. 20
4. Operacinė sistema - tai:
   1. programų komplektas, valdantis kompiuterio aparatinės įrangos darbą ir atliekantis sąsajos tarp kompiuterio ir jo naudotojo vaidmenį
   2. duomenų rinkinys, susistemintas ir sutvarkytas taip, kad juo būtų galima patogiai naudotis; duomenys gali būti įvairūs: tekstai, paveikslai, garsai
   3. programa (ar programų komplektas), valdanti kurį nors kompiuterio įtaisą, pavyzdžiui klaviatūrą, diskų įtaisą, pelę, skenerį
   4. visuma programų, padedančių kompiuterio naudotojui atlikti praktinius darbus: rašyti raštus, piešti piešinius, kurti muziką, sudaryti įvairių duomenų lenteles, braižyti grafikus ir pan
5. Ar galima turėti interneto svetainės adresą nelotyniškomis raidėmis, pvz. www.ačiū.lt ?
   1. taip
   2. ne
   3. tik gavus valstybinės kalbos komisijos leidimą
6. Programos naudotojo licencija – tai:
   1. naudojimosi programine įranga instrukcija
   2. programos įdiegimo failas su dokumentacija
   3. kodas, kuris įvedamas programos diegimo metu
   4. leidimas, suteikiantis teisę naudoti programą pagal gamintojo (autoriaus) sąlygas
7. Kuris iš šių šešioliktainių skaičių atitinka dešimtainį skaičių 64?
   1. 3F
   2. 40
   3. 20
8. Kokia yra pagrindinė HTTP protokolo paskirtis?
   1. vaizdo suspaudimo protokolas, skirtas informacijai perduoti internetu
   2. informacijos šifravimo nuo kibernetinių atakų būdas
   3. pagrindinis metodas informacijai pasauliniame tinkle (WWW) pasiekti
9. Kuris teiginys apie HTTPS protokolą yra teisingas?
   1. spartesnė HTTP protokolo versija skirta vaizdui ar garsui perduoti internetu naudojanti informacijos suspaudimo algoritmus
   2. HTTP protokolo praplėtimas, skirtas informacijai perduoti saugiai
   3. HTTP protokolo pirmtakas ir šiuo metu nebenaudojamas
10. Kas yra HTML?
    1. HTTPS protokole naudojamas šifravimo algoritmas
    2. žymių kalba, skirta internetiniams puslapiams kurti ir atvaizduoti
    3. populiarios HotMail pašto tarnybos trumpinys, tapęs elektroninių laiškų perdavimo standartu

# Modulis „Informacinių sistemų projektavimas ir kūrimas (.NET)“

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

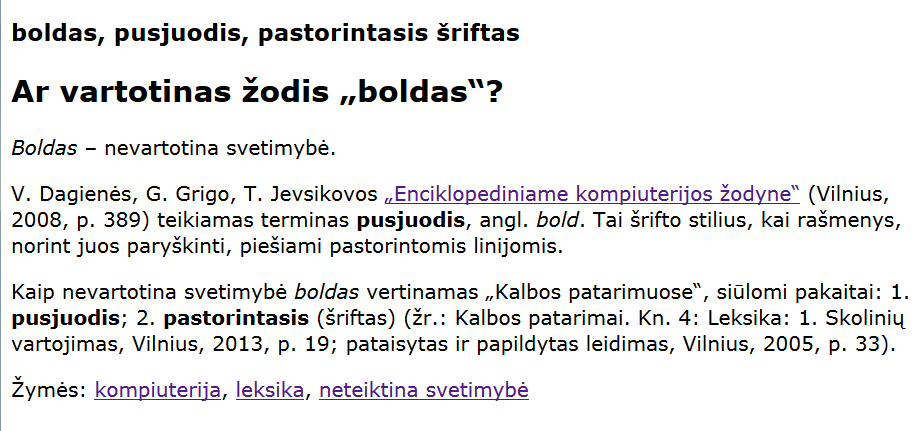
### Mokymosi rezultatas. Pateikti internetinio puslapio turinį naudojant kompiuterinę žymėjimo kalbą.

### Tema. Pagrindinės HTML (HTML 5) kalbos žymės.

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

UŽDUOTIES SĄLYGA: naudojant HTML antraščių, pastraipos, teksto formatavimo, nuorodų elementus, sukurti 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

### Tema. HTML kalbos turinio žymės.

#### *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.* PATEIKITE 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.

### Tema. CSS pagrindai.

#### *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>

1. 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>

1. 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.

### Tema. CSS tinklalapio maketo kūrimo technikos.

#### *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.

### Tema. JavaScript kalbos pagrindai.

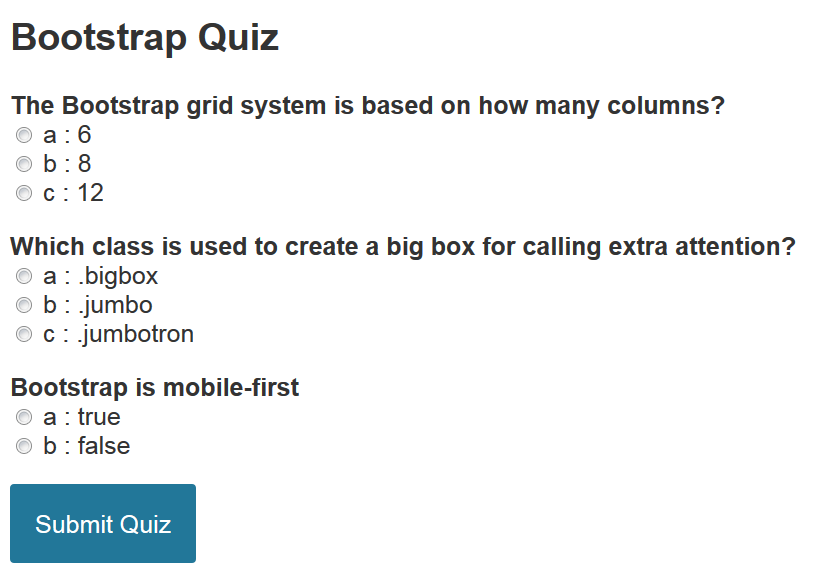
#### *1 užduotis.* SUKURKITE TESTĄ, TIKRINANTĮ IR PATEIKIANTĮ TEISINGUS ATSAKYMUS.

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:**



### Tema. Dinaminis manipuliavimas naudojant DOM API.

#### *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>

* 1. 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 ;

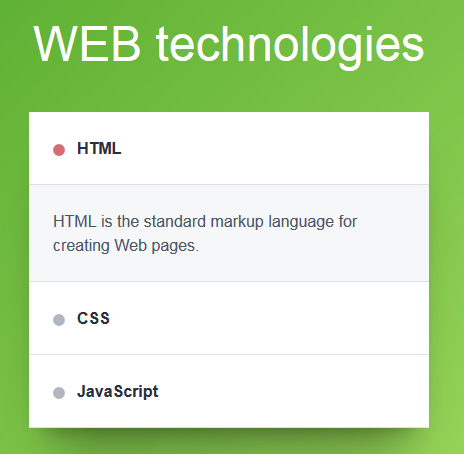
}

**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 nuspaudus 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>

1. 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;

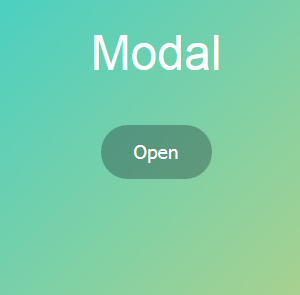
background: transparent;

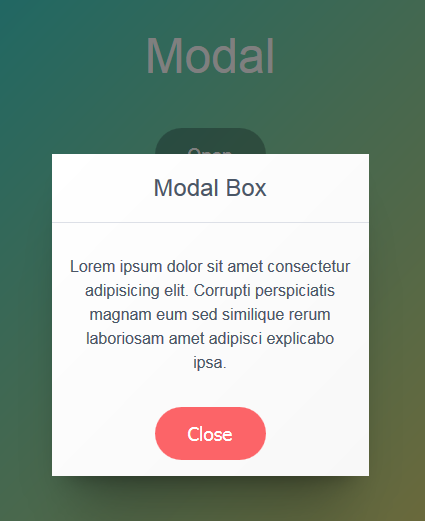
}

**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>

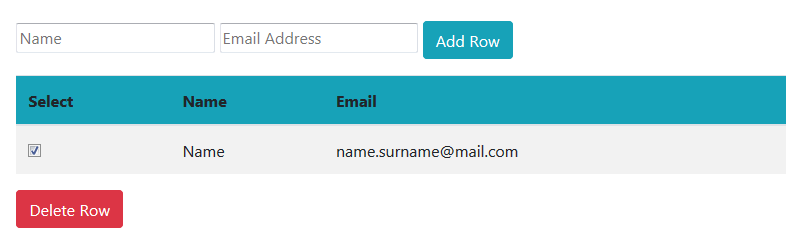
</body>

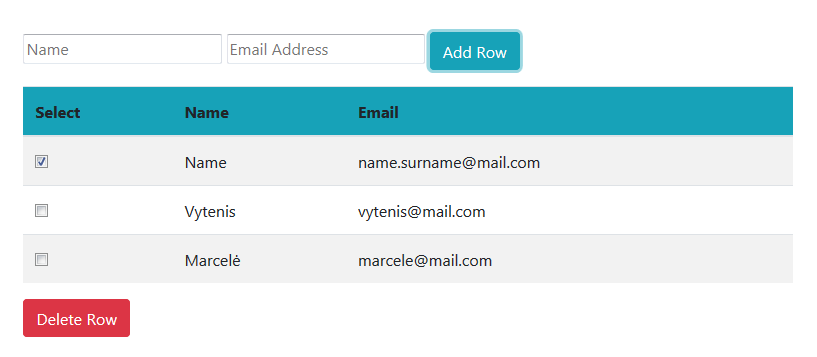
</html>

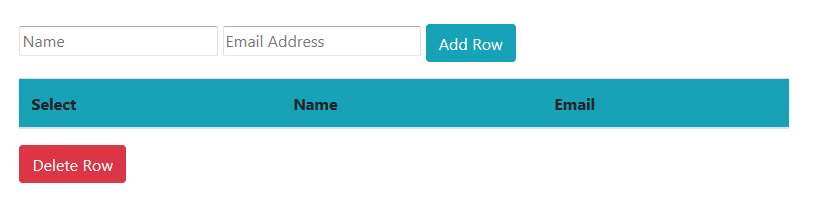
**Reikalavimai:**

1. HTML dokumente užrašyti nuorodas į skripto failus.
2. Naudojant selektorius, įvykius, metodus darbui su CSS klasėmis.

**Pavyzdys:**







## Kompetencija. Taikyti programinės įrangos kūrimui naudojamus informatikos principus ir metodus.

### Mokymosi rezultatas. Suprasti skaičiavimo sistemas.

### Tema. Skaičiavimo sistemos.

*1 užduotis.* APRAŠYKITE KAIP NUMERAVO SENOVĖS BABILONIEČIAI, EGIPTIEČIAI, GRAIKAI, SLAVAI.

*2 užduotis.* SURAŠYKITE KOKIOSE ŽMOGAUS VEIKLOS SRITYSE SKAIČIAVIMO SISTEMOS TURI DIDŽIAUSIĄ REIKŠMĘ.

*3 užduotis.* SURAŠYKITE VISAS JUMS ŽINOMAS SKAIČIAVIMO SISTEMAS (POZICINĖS).

*4 užduotis.* UŽRAŠYKITE DEŠIMTAINIUS SKAIČIUS 125, 256, 2019 DVEJETAINIAIS SKAIČIAIS.

*5 užduotis.* UŽRAŠYKITE DEŠIMTAINIAIS SKAIČIAIS ŠIUOS DVEJETAINĖS SISTEMOS SKAIČIUS: 101100110001, 1010101, 001110011.

*6 užduotis.* UŽRAŠYKITE DEŠIMTAINIUS SKAIČIUS 14, 56, 2019 AŠTUNTAINIAIS SKAIČIAIS.

*7 užduotis.* UŽRAŠYKITE DEŠIMTAINIAIS SKAIČIAIS ŠIUOS AŠTUNTAINĖS SISTEMOS SKAIČIUS: 77, 66, 32.

*8 užduotis.* UŽRAŠYKITE DEŠIMTAINIUS SKAIČIUS 62, 216, 2019 ŠEŠIOLIKTAINIAIS SKAIČIAIS.

*9 užduotis.* UŽRAŠYKITE DEŠIMTAINIAIS SKAIČIAIS ŠIUOS ŠEŠIOLIKTAINĖS SISTEMOS SKAIČIUS: 1D5F, A9BC, FF.

### Tema. Informacijos matavimo vienetai.

*10 užduotis.* UŽRAŠYKITE VISUS JUMS ŽINOMUS INFORMACIJOS KIEKIO MATAVIMO VIENETUS.

*11 užduotis.* TURITE 16GB USB ATMINTUKĄ. PASKAIČIUOKITE KIEK TOKS ATMINTUKAS TURI MEGABAITŲ (MB), KILOBAITŲ (KB) IR BAITŲ (B).

Paskaičiuokite kiek galite patalpinti į tokią laikmeną bitų informacijos.

*12 užduotis.* JŪSŲ INTERNETO PASLAUGŲ TIEKĖJAS PER MĖNESĮ UŽTIKRINA JUMS 8589934592 BITŲ DUOMENŲ KIEKĮ.

Kiek nuotraukų galite persiųsti kiekvieną mėnesį, jei vienos nuotraukos dydis bus 5mb?

#### *13 užduotis.* JŪSŲ INTERNETO GREITAVEIKA 100MBPS. KIEK LAIKO UŽTRUKS 100MB FAILO SIUNTIMAS?

#### Laikome, kad interneto greitaveika nesikeičia ir yra pastovi.

#### *14 užduotis.* KOKIA TURI BŪTI JŪSŲ INTERNETO GREITAVEIKA, KAD PER 5 MINUTES GALIMA BŪTŲ PERDUOTI 600MB INFORMACIJOS?

#### *15 užduotis.* JUMS REIKIA PERDUOTI 100GB DUOMENŲ.

#### Jūsų tinklo greitaveika yra 300mbps. Tinkle dirba kiti kompiuteriai. Įvertinkite esamą situaciją ir nuspręskite, ar galima bus persiųsti pateiktą duomenų kiekį per 5 valandas.

### Mokymosi rezultatas. Taikyti algoritmų ir logikos mokslo pagrindus programuojant.

### Tema. Logikos mokslo pagrindai.

#### *1* *užduotis.* FORMALIZUOKITE PATARLĘ: „DEGTINEI Į TROBĄ ĮEINANT, PROTAS IŠEINA LAUKAN“ TEIGINIŲ LOGIKOS PRIEMONĖMIS.

#### (užduotis iš Raškinis, A., Karoblis, G. knygos „Logikos užduočių pratybos”).

#### *2 užduotis*. FORMALIZUOKITE PATARLĘ: „DEGTINĖ IR ALUS GIMDO VARGUS“ TEIGINIŲ LOGIKOS PRIEMONĖMIS.

#### (užduotis iš Raškinis, A., Karoblis, G. knygos „Logikos užduočių pratybos”).

#### *3 užduotis.* UŽRAŠYKITE DE MORGANO TAISYKLĖS KONJUNKCIJAI SIMBOLINĘ IŠRAIŠKĄ.

#### Kaip ta išraiška turi skambėti skaitoma? Pateikite savo sugalvotą interpretacijos pavyzdį (užduotis iš Raškinis, A., Karoblis, G. knygos „Logikos užduočių pratybos”).

*4 užduotis*. UŽRAŠYKITE DE MORGANO TAISYKLĖS KONJUNKCIJAI SIMBOLINĘ IŠRAIŠKĄ.



Kaip ta išraiška turi skambėti skaitoma? Pateikite savo sugalvotą interpretacijos pavyzdį.

#### (užduotis iš Raškinis, A., Karoblis, G. knygos „Logikos užduočių pratybos”).

### Tema. Algoritmai ir algoritmavimas.

#### *5 užduotis.* PARAŠYKITE PROGRAMĄ BET KOKIA JUMS ŽINOMA PROGRAMAVIMO KALBA IR/ARBA PAVAIZDUOKITE ALGORITMĄ, KURIS APSKAIČIUOJA SKAIČIŲ MASYVO SUMĄ IR VIDURKĮ.

#### Papildoma sąlyga: turi būti panaudotas iteratyvus algoritmas.

#### *6 užduotis.* PARAŠYKITE PROGRAMĄ BET KOKIA JUMS ŽINOMA PROGRAMAVIMO KALBA IR/ARBA PAVAIZDUOKITE ALGORITMĄ, KURIS SURANDA MAŽIAUSIĄ MASYVO SKAIČIŲ.

#### Papildoma sąlyga: turi būti panaudotas rekursyvus algoritmas.

#### *7 užduotis*. PARAŠYKITE PROGRAMĄ BET KOKIA JUMS ŽINOMA PROGRAMAVIMO KALBA IR/ARBA PAVAIZDUOKITE ALGORITMĄ, KURIS SKAIČIUOJA 120 FIBONAČIO SKAIČIŲ.

#### Papildoma sąlyga: turi būti naudojamas dinaminis programavimas.

#### *8 užduotis.* PARAŠYKITE PROGRAMĄ BET KOKIA JUMS ŽINOMA PROGRAMAVIMO KALBA IR/ARBA PAVAIZDUOKITE ALGORITMĄ, KURIS NUSTATO AR RINKINYJE (SET) YRA ELEMENTŲ RINKINYS (SUBSET) LYGUS NURODYTAI SUMAI.

#### Papildoma sąlyga: turi būti naudojamas dinaminis programavimas.

#### *9 užduotis*. PARAŠYKITE PROGRAMĄ BET KOKIA JUMS ŽINOMA PROGRAMAVIMO KALBA IR/ARBA PAVAIZDUOKITE ALGORITMĄ, KURIS SURŪŠIUOJA SVEIKŲJŲ SKAIČIŲ MASYVĄ.

#### Papildoma sąlyga: turi būti panaudotas greito rūšiavimo algoritmas.

#### *10 užduotis.* PARAŠYKITE PROGRAMĄ BET KOKIA JUMS ŽINOMA PROGRAMAVIMO KALBA IR/ARBA PAVAIZDUOKITE ALGORITMĄ, KURIS SURŪŠIUOJA SVEIKŲJŲ SKAIČIŲ MASYVĄ.

#### Papildoma sąlyga: turi būti panaudotas burbulo algoritmas.

#### *11 užduotis*. PARAŠYKITE PROGRAMĄ BET KOKIA JUMS ŽINOMA PROGRAMAVIMO KALBA IR/ARBA PAVAIZDUOKITE ALGORITMĄ, KURIS SIMBOLIŲ EILUTĘ ATSPAUSDINA ATVIRKŠČIAI.

#### Papildoma sąlyga: turi būti panaudota lifo duomenų struktūra (stack).

#### *12 užduotis*. PARAŠYKITE PROGRAMĄ BET KOKIA JUMS ŽINOMA PROGRAMAVIMO KALBA IR/ARBA PAVAIZDUOKITE ALGORITMĄ, KURIS RANDA MAŽIAUSIĄ SKAIČIŲ AIBĖJE.

#### Papildoma sąlyga: turi būti panaudota dvejetainio medžio (binary tree) struktūra.

#### *13 užduotis*. ĮVERTINKITE KURIE IŠ ALGORITMŲ IR KOKIAIS ATVEJAIS YRA NAUDINGESNI IR GREIČIAU VEIKIA:

- algoritmai su rekursija;

- algoritmai be rekursijos.

### Mokymosi rezultatas. Taikyti programinio kodo dizaino modelius programuojant.

### Tema. Dizaino šablonai.

#### *1* *užduotis*. UŽPILDYKITE PATEIKTĄ LENTELĘ, NURODYDAMI JUMS ŽINOMUS DIZAINO MODELIUS/ŠABLONUS (ANGL. *DESIGN PATTERNS*) IR PRISKIRDAMI VIENAI IŠ KATEGORIJŲ.

|  |  |  |
| --- | --- | --- |
| **Dizaino šablono rūšys** | | |
| Creational | Structural | Behavioral |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

#### *2 užduotis.* APRAŠYKITE PASIRINKTINAI TRIS DIZAINO ŠABLONUS.

#### Šablonai turi būti skirtingų rūšių. Pateikite pavyzdžių kur šablonas gali būti taikomas.

### Tema. Dizaino šablonų taikymas.

#### *3 užduotis.* NAUDOJANT UML MODELIAVIMO KALBĄ SUKURKITE KLASIŲ DIAGRAMĄ „ABSTRAKČIOS GAMYKLOS“ ŠABLONUI (ABSTRACT FACTORY PATTERN).

#### *4 užduotis.* NAUDOJANT UML MODELIAVIMO KALBĄ SUKURKITE KLASIŲ DIAGRAMĄ „DEKORATORIAUS“ ŠABLONUI (DECORATOR PATTERN).

#### *5 užduotis*. NAUDOJANT UML MODELIAVIMO KALBĄ SUKURKITE KLASIŲ DIAGRAMĄ „STRATEGIJOS“ ŠABLONUI (STRATEGY PATTERN).

#### *6 užduotis.* PARINKITE DIZAINO ŠABLONĄ PATEIKTAM GYVENIMO PAVYZDŽIUI: GAMYKLA GAMINA TRIJŲ MODELIŲ AUTOMOBILIUS.

#### Kiekvienas automobilis turi savo unikalius parametrus. Kiekvienas automobilis gaminamas pagal savo brėžinį. Visi automobiliai gaminami vienoje gamykloje.

#### *7 užduotis.* PARINKITE DIZAINO ŠABLONĄ PATEIKTAM GYVENIMO PAVYZDŽIUI: TURIME PAŠTO SERVERĮ.

#### Serveris moka siųsti elektroninius laiškus. Laiškai būna skirti įvairiems adresatams. Priklausomai nuo adresato prie laiško yra pridėdama tam tikra tarnybinė informacija.

#### *8 užduotis.* PARAŠYKITE .NET PROGRAMĄ, KURI REALIZUOTŲ „ADAPTER“ ŠABLONĄ.

#### *9 užduotis.* PARAŠYKITE .NET PROGRAMĄ, KURI REALIZUOTŲ „TEMPLATE METHOD“ ŠABLONĄ.

#### *10 užduotis.* APGALVOKITE IR NUSPRĘSKITE KOKĮ DIZAINO ŠABLONĄ GALIMA BŪTŲ PANAUDOTI APRAŠANT PAUKŠČIŲ KLASES.

### Mokymosi rezultatas. Naudoti informacinių sistemų kūrimo principus ir metodus programinės įrangos projektavime ir programavime.

### Tema. Daugiasluoksnė programų architektūra ir MVC struktūra.

#### *1 užduotis*. APRAŠYKITE KLIENTO-SERVERIO (CLIENT SERVER) PROGRAMINĖS ĮRANGOS MODELĮ.

#### Pateikite pavyzdžių kur tokia architektūra yra naudojama.

#### *2 užduotis*. APRAŠYKITE MODELIO-VAIZDO-KONTROLERIO (MODEL VIEW CONTROLLER, MVC) PROGRAMINĖS ĮRANGOS MODELĮ.

#### Pateikite pavyzdžių kur tokia architektūra yra naudojama.

#### *3 užduotis*. NAUDOJANT MVC ARCHITEKTŪROS MODELĮ, SUKURKITE PAPRASTĄ DARBUOTOJŲ VALDYMO PROGRAMĄ.

#### Programoje turi būti panaudoti: modelis, vaizdas ir kontroleris. Darbuotojai turi savo unikalų numerį (id), vardą (name), amžių (age), darbo stažą (experience).

#### *4 užduotis*. NAUDOJANT PASIRINKTĄ KARKASĄ SUKURKITE PAPRASTĄ WEB VARTOTOJŲ REGISTRACIJOS PROGRAMĄ.

#### Registracijos formos laukai: vardas, pavardė, vartotojo vardas (slapyvardis), telefonas, el. Paštas, slaptažodis.

### Tema. Pagrindiniai programavimo principai.

#### *5 užduotis*. APRAŠYKITE SOLID IR DRY PROGRAMAVIMO PRINCIPUS.

#### *6 užduotis*. APRAŠYKITE SEPARATION OF CONCERNS IR CODE REUSE PROGRAMAVIMO PRINCIPUS.

#### *7 užduotis*. NAUDOJANT FUNKCINIO PROGRAMAVIMO PRINCIPUS REALIZUOKITE GREITO RŪŠIAVIMO ALGORITMĄ.

#### Programa turi mokėti rūšiuoti sveikų skaičių sąrašą.

#### *8 užduotis*. NAUDOJANT FUNKCINIO PROGRAMAVIMO PRINCIPUS REALIZUOKITE SĄRAŠO GRUPAVIMO, ELEMENTŲ SKAIČIAVIMO IR RŪŠIAVIMO ALGORITMĄ .

#### Programa turi dirbti su vardų sąrašu.

#### *9 užduotis*. KOKIAIS ATVEJAIS FUNKCINIS PROGRAMAVIMAS YRA EFEKTYVESNIS UŽ OBJEKTINĮ?

## Kompetencija. Kurti tipinę programinę įrangą.

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

### Tema. Visual Studio aplinka.

*1 užduotis.*  ĮDIEKITE VISUAL STUDIO INTEGRUOTĄ PROGRAMAVIMO APLINKĄ SU “.NET CORE CROSS-PLATFORM DEVELOPMENT” IR “ASP.NET AND WEB DEVELOPMENT” MODULIAIS.

*2 užduotis.*  PATIKRINKITE AR TEISINGAI ĮDIEGTAS PROGRAMAVIMO SU .NET CORE KARKASU PRIEMONĖS, GAUNANT INFORMACIJĄ APIE KARKASĄ KOMANDINĖJE EILUTĖJE.

*3 užduotis.*  NAUDOJANT VISUAL STUDIO IDE SUKURKITE .NET CORE PROGRAMĄ KONSOLEI, KURI IŠVEDA PASISVEIKINIMĄ SU PROGRAMOS AUTORIUMI IR JO VARDĄ.

*4 užduotis.*  PROGRAMĄ IŠ *3* *užduoties* SUKOMPILIUOKITE IR PALEISKITE NAUDOJANT KOMANDINĘ EILUTĘ.

### Tema. C# programavimo pagrindai ir sintaksė.

*5 užduotis. #5.*  NAUDOJANT VISUAL STUDIO IDE SUKURKITE.NET CORE PROGRAMĄ KONSOLEI, KURI TURI TURĖTI MAIN METODĄ IR METODĄ PAVADINIMU PRINTTEXT, KURIS ATSPAUSDINA TEKSTĄ, PERDUODAMĄ PARAMETRU.

Main metode kviečiamas PrintText metodas ir atspausdinamas laisvai pasirenkamas tekstas.

*6 užduotis.* PAPILDYKITE PROGRAMĄ „*5 užduotis* #5“ VIENO IR KELIŲ EILUČIŲ TIPŲ KOMENTARAIS.

*7 užduotis.* SUKURKITE PROGRAMĄ, KURIOJE YRA SĄLYGOS SAKINYS, KURIS ATSPAUSDINTŲ PRANEŠIMĄ JEIGU KINTAMOJO REIKŠMĖ DIDESNĖ UŽ 100, JEIGU NE – TADA ATSPAUSDINTŲ KITĄ PRANEŠIMĄ.

*8 užduotis.* SUKURKITE PROGRAMĄ, KURIOJE YRA SĄLYGOS SAKINYS, KURIS PATIKRINTŲ IR ATSPAUSDINTŲ PRANEŠIMĄ „SKAIČIUS YRA 0...100 TARPE“ JEIGU KINTAMOJO REIKŠMĖ YRA ATITINKAMAME RĖŽYJE.

Kitu atveju atspausdintų pranešimą „Skaičiaus rėžyje nėra“.

*9 užduotis.* SUKURKITE DU *INT* TIPO KINTAMUOSIUS (PVZ. *A* IR *B*) IR PRISKIRTI JIEMS REIKŠMES.

Panaudoti aritmetinius, loginius, jungtinius priskyrimo, inkremento, dekremento operatorius. Kiekvienai užduoties daliai sukurti po naują kintamąjį. Kintamojo reikšmes atspausdinti konsolėje.

*10 užduotis.* SUKURKITE PROGRAMĄ, KURI PAPRAŠO ĮVESTI VARDĄ IR PATEIKIA IŠVESTĮ „LABAS, [VARDAS]“, PAGAL ĮVESTĄ VARDĄ.

*11 užduotis.* SUKURKITE PROGRAMĄ SU METODU, KURIS PRIIMA SKAIČIŲ IR PANAUDOJA BOOL IŠRAIŠKĄ KODE, KURIS NURODO AR SKAIČIUS YRA TEIGIAMAS.

*12 užduotis.* SUKURKITE PROGRAMĄ SU METODU, KURIS GRĄŽINA BOOL DUOMENŲ TIPĄ, PAVADINIMU NUMBERISNEGATIVE.

Metodas turi patikrinti ar skaičius neigiamas ir grąžinti rezultatą. Panaudoti IF sakinį metode.

*13 užduotis.* SUKURKITE PROGRAMĄ SU METODU, KURIS PATIKRINA AR PADUOTAS SKAIČIUS YRA TRIŽENKLIS.

Nepamiršti neigiamų skaičių.

*14 užduotis.* SUKURKITE PROGRAMĄ SU METODU, KURIS PRIIMA TRIS *INT* TIPO PARAMETRUS (SK – SKAIČIUS PATIKRINIMUI, r1, r2 – SKAIČIAI NUSAKANTYS RĖŽIUS) IR GRĄŽINA *BOOL* DUOMENŲ TIPĄ.

* Patikrinti ar r1 < r2 – jeigu neatitinka sąlygos, atspausdinti pranešimą ir baigti darbą panaudojant return
* Patikrinti ir atspausdinti pranešimus ar sk yra tarp r1 r2 ar ne
* Atitinkamai pagal rezultatą ar skaičius tarp r1 ir r2 grąžinti false arba true
* Papildomai patikrinti su neigiamais skaičiais.

*15 užduotis.* SUKURKITE PROGRAMĄ SU METODU, KURISNAUDOJANT CIKLĄ ATSPAUSDINA Į KONSOLĘ NELYGINIUS SKAIČIUS NUO 1 IKI 10.

*16 užduotis.* PARAŠYKITE FAKTORIALO METODĄ NAUDOJANT REKURSIJĄ.

*17 užduotis.* PARAŠYKITE FIBINAČIO SEKOS METODĄ NAUDOJANT REKURSIJĄ.

*18 užduotis.* SUKURKITE PROGRAMĄ, KURIOJE SUKURIAMI ŠIO TIPO KINTAMIEJI IR ATSPAUSDINAMI KONSOLĖS LANGE: BOOL, BYTE , CHAR, INT, FLOAT, DOUBLE, LONG, STRING.

Panaudoti var raktažodį su anksčiau paminėtais duomenų tipais.

*19 užduotis.* MASYVO, KURIS TURI REIKŠMES {1,66,2,8,33,5,2}:

* atspausdinti kiekvieną skaičių naudojant ciklą;
* rasti masyvo narių sumą;
* rasti masyvo narių vidurkį;
* rasti mažiausia masyvo skaičių;
* rasti didžiausią masyvo skaičių.

Rezultatus atvaizduoti.

*20 užduotis.* SUKURKITE PROGRAMĄ, KURI IŠ PRADŽIŲ PAPRAŠYTŲ ĮVESTI TIEK SKAIČIŲ KIEK REIKĖS ĮVESTI (Į KOKIO DYDŽIO MASYVĄ) IR TADA IŠ KOMANDINĖS EILUTĖS NUSKAITYTŲ VISUS SKAIČIUS.

Pabaigus skaitymą, atspausdinti masyvo narius, visų narių sumą, vidurkį ir sandaugą.

*21 užduotis.* SUKURKITE PAIEŠKOS MASYVE METODĄ, KURIS PRIIMTŲ SVEIKŲ SKAIČIŲ MASYVĄ IR SVEIKĄ SKAIČIŲ KAIP PARAMETRUS.

Grąžintų indeksą masyve (ieškoto skaičiaus vietą masyve) jeigu skaičius rastas, o jeigu ne – tada -1. Metodo veikimą pademonstruoti.

*22 užduotis.* SUKURKITE PROGRAMĄ SU METODU , KURIS PRIIMA DU SKAIČIUS IR GRĄŽINA JŲ SUMĄ.

Perkrauti metodą, kuris priimtų tris skaičius ir paskaičiuotų jų sumą

### Mokymosi rezultatas. Kurti nesudėtingą programinį kodą C# programavimo kalba.

### Tema. Darbas su duomenimis.

### 

*1 užduotis* #*1.* SUKURKITE PROGRAMĄ, KURI TURI KLASĘ, APRAŠANČIĄ DARBUOTOJĄ, KURIO SAVYBĖS YRA VARDAS, GIMIMO DATA, PAREIGOS IR AMŽIUS (TIK SKAITOMA SAVYBĖ, GAUNAMA IŠ DABARTINĖS DATOS ATĖMUS GIMIMO DATĄ).

*2 užduotis* #*2.* SUKURKITE PROGRAMĄ, KURI PAPRAŠO ĮVESTI DU SKAIČIUS IR ATSPAUSDINA JŲ SUMĄ

.

Panaudoti *parse* metodą, kuris paverstų vartotojo įvestį į skaičių.

*3 užduotis*#*3.* SUKURKITE PROGRAMĄ, KURI PAPRAŠYTŲ VARTOTOJO ĮVESTIES IR SU *TRYPARSE* PATIKRINTI AR ĮVESTIS TEISINGA, JEI NETEISINGA, PROGRAMĄ UŽBAIGTI PANAUDOJUS *RETURN* RAKTAŽODĮ.

Parašyti *switch* sakinį, kuris tikrintų įvesties sąlygas 1,2,3.

Jei įvestas skaičius pagal sąlygą, tą skaičių atspausdinti, priešingu atveju, atspausdinti „Įvestas nežinomas skaičius“.

*4 užduotis*#*4.* SUKURKITE PROGRAMĄ SU MENIU, KURI ATSPAUSDINTŲ MENIU PUNKTUS SKAIČIŲ ARITMETINĖMS FUNKCIJOMS: SKAIČIŲ SUDĖTIS, DAUGYBA, KĖLIMAS KVADRATU.

Programa turi paprašyti vartotojo pasirinkti meniu punktą. Punkto pasirinkimus įgyvendinti per *switch* sakinį. Kiekviena *switch* sakinio sąlyga, turi nukreipti į atskirą metodą, kuris vykdomas atitinkama logika. Skaičių sudėtis ir daugyba turi paprašyti vartotoją įvesti du skaičius ir atspausdinti jų rezultatą. Skaičiaus kėlimas kvadratu prašo įvesti vieną skaičių. Jei įvestis neteisinga, programą užbaigti.

*5 užduotis*#*5.* PARAŠYKITE PROGRAMĄ, KURI SUKURIA FAILĄ SU TEKSTU, JĮ NUSKAITO IR TURINĮ IŠVEDA Į EKRANĄ.

*6 užduotis*#*6.* PARAŠYKITE PROGRAMĄ, KURI PRIE EGZISTUOJANČIO FAILO PRIDEDA PAPILDOMO TEKSTO, O NE JĮ PERRAŠO.

*7 užduotis*#*7.* PARAŠYKITE PROGRAMĄ, KURI PATIKRINA AR FAILAS EGZISTUOJA IR JĮ IŠTRINA.

*8 užduotis*#*8.* PARAŠYKITE PROGRAMĄ, KURI SUKURIA FAILĄ SU NURODYTU KIEKIU EILUČIŲ (KIEKVIENOJE EILUTĖJE SAUGOMAS EILUTĖS INDEKSAS IR VARTOTOJO ĮVESTAS TEKSTAS).

Failą nuskaityti ir išvesti paskutinę jo eilutę.

### Tema. LINQ, .NET Generics ir kolekcijos.

*9 užduotis*#*9.* SUKURKITE PROGRAMĄ, KURI TURI SWAP METODĄ, LEIDŽIANTĮ SUKEISTI REIKŠMES VIETOMIS, T. Y. KVIEČIAME SWAP(A,B), KUR A=2, B=3, O METODAS PADARO B=3 IR A=2.

Sukurkite generic metodo versiją.

*10užduotis*#*10.* SUKURKITE SVEIKŲ SKAIČIŲ MASYVĄ IŠ VARTOTOJO ĮVESTIES IR ATSPAUSDINTI JO NARIŲ SUMĄ.

Masyvo dydį nurodo vartotojas.

*11 užduotis#11.* SUKURKITE PROGRAMĄ, KURI PRAŠO VARTOTOJO VESTI SVEIKUOSIUS SKAIČIUS TOL, KOL ŠIS NEĮVES 0, TUOMET IŠVESKITE TIEK SKAIČIŲ KIEK VARTOTOJAS ĮVEDĖ, JŲ SUMĄ IR ARITMETINĮ VIDURKĮ.

Nulis į skaičių sąrašą nepridedamas.

*12 užduotis*#*12.* SUKURKITE PROGRAMĄ, KURI TELEFONO NUMERIŲ IR VARDŲ SAUGOJIMUI NAUDOJA DICTIONARY KLASĘ, KUR TELEFONO NUMERIS YRA DUOMENŲ POROS RAKTAS, O VARDAS POROS REIKŠMĖ.

Vartotojui suteikiama galimybė pridėti naujų įrašų, kol vietoje numerio neįvedamas nulis. Įvedus „0“ vietoje tel. numerio, išvedama visa telefonų knygą ir baigiamas darbas.

*13 užduotis*#*13.* SUKURKITE PROGRAMĄ, KURI SUKURIA SVEIKŲ SKAIČIŲ NUO 1 IKI 10 DUOMENŲ MASYVĄ , NAUDOJANT LINQ UŽKLAUSĄ ,IŠVEDAMI TIK LYGINIAI SKAIČIAI.

*14 užduotis#14.* SUKURKITE PROGRAMĄ, KURI SURŪŠIUOJA DUOTĄ MIESTO PAVADINIMŲ SĄRAŠĄ PAGAL ABĖCĖLĘ.

*15 užduotis*#*15.* SUKURKITE PROGRAMĄ, KURI „VARTOTOJAI“ ĮVESTAI TEKSTO EILUTEI, IŠVEDA ŽODŽIUS NE TRUMPESNIUS, NEI NURODĖ VARTOTOJAS. PVZ.: TEKSTUI „GERA DIENA“ IR „5“ IŠVEDAMAS TIK VIENAS ŽODIS „DIENA“.

*16 užduotis*#*16.* PARAŠYKITE LAMBDA IŠRAIŠKĄ, KURI DUOTĄ SVEIKĄ SKAIČIŲ PAKELIA KVADRATU IR PADEMONSTRUOKITE JOS VEIKIMĄ.

*17 užduotis*#*17.* PARAŠYKITE LAMBDA IŠRAIŠKĄ, KURI DUOTAM SVEIKAM SKAIČIUI GRĄŽINTŲ *TRUE*, JEI SKAIČIUS YRA LYGINIS IR *FALSE* KITU ATVEJU.

Pademonstruoti jos veikimą linq užklausoje su sveikų skaičių masyvu nuo 0 iki 9 išvedant ekrane tik lyginius masyvo skaičius.

### Tema. .NET karkaso komponentai.

*18 užduotis.* SUKURKITE METŲ MĖNESIŲ REIKŠMIŲ ENUMERACIJĄ , KUR SAUSIS YRA 1, O GRUODIS – 12.

Vartotojui leisti įvesti skaičių ir išvesti skaičių atitinkantį mėnesį iš enumeracijos.

*19 užduotis.* SUKURKITE IR PANAUDOKITE METODĄ, KURIS ESAMAI DATAI GRĄŽINA KELIAS REIKŠMES: IŠKART – METUS, MĖNESĮ IR DIENĄ – NAUDOJANT VALUETUPLE KLASĘ.

*20 užduotis.* SUKURKITE KLASĘ VARTOTOJO INFORMACIJAI SAUGOTI (VARDUI, PAVARDEI, AMŽIUI, EL. PAŠTO ADRESUI).

Šiai klasei sukurti dekonstrukcijos metodą, kuris būtų naudojamas vartotojo klasės objektui dekonstruoti.Pademonstruoti dekonstruktoriaus veikimą.

### Tema. Kodavimo standartai.

*21 užduotis.* SUKURKITE SU 2 METODAIS INTERFEISĄ, ATITINKANTĮ MICROSOFT PAVADINIMŲ GAIRES.

*22 užduotis.*  SUKURKITE SU 3 SAVYBĖMIS IR 2 METODAIS KLASĘ, ATITINKANČIĄ MICROSOFT PAVADINIMŲ GAIRES.

*23 užduotis.* ĮGYVENDINTINKITE SINGLETON PROJEKTAVIMO ŠABLONĄ PROGRAMOS KONFIGŪRACIJAI LAIKYTI: DUOMENŲ BAZĖS PRISIJUNGIMO INFORMACIJAI, MAKSIMALIAM KLIENTŲ SKAIČIUI.

*24 užduotis.* SUKURKITE PROGRAMĄ SU METODU, KURIS GRĄŽINA DVIEJŲ SVEIKŲ SKAIČIŲ SUMĄ IR JAM PRIDĖKITE METODO SANTRAUKĄ.

### Tema. MVC (RAZOR engine) technologijos panaudojimas vartotojo sąsajai programuoti.

*25 užduotis.* APIBRĖŽKITE MVC ŠABLONĄ, ESMINIUS JO KOMPONENTUS IR JŲ ATSAKOMYBES.

*26 užduotis.* APIBŪDINKITE RAZOR VARIKLIO PASKIRTĮ.

*27 užduotis.* APIBRĖŽKITE KAS YRA RAZOR PAGES.

*28 užduotis.* APIBRĖŽKITE KAS YRA RESTFUL SERVISAS IR KAS UŽTIKRINA, KAD SERVISĄ GALIMA LAIKYTI RESTFUL.

*29 užduotis.* APIBRĖŽKITE RICHARDSON MATURITY MODEL IR JO LYGIUS.

*30 užduotis.* APIBRĖŽKITE, KAS YRA TARPINĖ PROGRAMINĖ ĮRANGA (ANGL. *MIDDLEWARE*) ASP.NET CORE KARKASO KONTEKSTE.

*31 užduotis.* APIBRĖŽKITE KAM ASP.NET CORE KARKASE YRA NAUDOJAMA STARTUP KLASĖ, JOS ATSAKOMYBES.

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

### Tema. Objektinis programavimas.

*1 užduotis.* SUKURKITE KLASĘ KOORDINATĖS TAŠKUI POINT.

Klasė turi turėti savybes x ir y koordinatei bei spalvai saugoti.

* Sukurti kelis objektus su skirtingomis koordinatėmis, reikšmes atspausdinti konsolėje
* Sukurti konstruktorius:
  + Be parametrų
  + Su x, y
  + Su x, y ir spalva
  + Su spalva

*2 užduotis.* SUKURKITE NAUJĄ PROJEKTĄ MATHEMATICS.

Projekte sukurti statinę klasę *Calculator*. Joje sukurti metodus: *Add*, kuris sudėtų du skaičius *Subtract*, kuris paskaičiuotų dviejų skirtumą *Multiply*, kuris paskaičiuotų dviejų skaičių daugybą *Divide*, kuris padalintų du skaičius. Klasės metodus iškviesti *Main* metode konsolės projekte.

*3 užduotis.* SUKURKITE KLASĘ AUTOMOBILIO INFORMACIJAI SAUGOTI.

Klasė turi turėti atributus:

* rida (angl. mileage)
* kuro kiekis (angl. fuel amount)
* pavadinimas(angl. name)
* maksimali kuro talpa (angl. max fuel amount). Reikšmė turi būti 50 visiems automobiliams.

Sukurti metodus:

* Važiuoti (angl. drive). Šis metodas turi didinti ridą po 10 ir mažinti kurą po 1. Jei neužtenka, kuro važiuoti, turi atspausdinti pranešimą
* Pildyti kurą (angl. refuel). Šis metodas turi didinti kurą pagal paduotą parametrą. Jei pildoma daugiau nei maksimalus kiekis, turi būti atspausdintas atitinkamas pranešimas

Atributas maksimali kuro talpa turi būti paslėptas ir nekeičiamas. Atributai: rida, pavadinimas, kuro kiekis, turi būti pasiekiami, bet jų reikšmių keitimas turi būti apribotas tik automobilio klasėje.

*4 užduotis.* SUKURKITE KLASĘ GYVŪNAS SU ATRIBUTAIS VARDAS IR ŠEIMA IR METODĄ PASISVEIKINTI, KURIS ATSPAUSDINTŲ TEKSTĄ PAGAL ESAMUS ATRIBUTUS: “LABAS AŠ Y IŠ X GIMINĖS”.

Perrašyti numatytąjį konstruktorių ir sukurti parametrizuotą konstruktorių, kuris inicijuoja visus klasės atributus

* Sukurti vaikinę klasę šuo. Klasė turėtų papildomą atributą, lojimų skaičius ir metodą loti (angl. bark), kuris atspausdintų tekstą ir didintų atributo lojimų skaičius reikšmę vienetu, kai metodas yra iškviečiamas. Į tėvinį konstruktorių perduoti šeimą, vaikinės klasės konstruktorius turi priimti vieną parametrą.
* Sukurti vaikinę klasę katė. Klasė turėtų papildomą atributą, miauksėjimų skaičius ir metodą miaukti, kuris atspausdintų tekstą ir didintų atributo miauksėjimų skaičius reikšmę vienetu, kai metodas yra iškviečiamas. Į tėvinį konstruktorių perduoti šeimą, vaikinės klasės konstruktorius turi priimti vieną parametrą.
* Sukurti vaikinę klasę driežas. Klasė turėtų papildomą atributą, pagautų musių skaičius ir metodą gaudyti musę, kuris atspausdintų tekstą ir didintų atributo , pagautų musių skaičius reikšmę vienetu, kai metodas yra iškviečiamas. Į tėvinį konstruktorių perduoti šeimą, vaikinės klasės konstruktorius turi priimti vieną parametrą.

Sukurtas klases inicijuoti Main() metode. Pademonstruoti kaip klasės paveldi visus tėvinių klasių metodus ir atributus.

*5 užduotis.* PADEMONSTRUOKITE POLIMORFIZMO GEBĖJIMUS SUKURIANT PROGRAMĄ SU ATITINKAMOMIS KLASĖMIS.

* Sukurti tėvinę klasę figūra, kuri turi statinį atributą saugoti sukurtų figūrų skaičiui. Šis atributas turėtų keisti reikšmę, kai bus sukuriama nauja klasės esybė. Atributas turi būti tik keičiamas klasės viduje, kad jį pasikeisti, sukurti statinį metodą, kuris gražintų atributo reikšmę.
* Sukurti klasę apskritimas, kuri praplėstų figūros klasę ir turėtų papildomą atributą spindulys. Šiai klasei sukurti metodus, kurie paskaičiuotų plotą ir perimetrą. Sukurti parametrizuotą konstruktorių, kuris priskirtų spinduliui reikšmę.
* Sukurti klasę cilindras, kuri praplėstų apskritimo klasę ir turėtų papildomą atributą aukštis. Taip pat turi metodą, kuris skaičiuoja plotą ir metodą kuris skaičiuoja tūrį, perimetrą ir šoninį plotą. Sukurti konstruktorių, kuris priskirtų atributų reikšmes.
* Sukurti klasę kvadratas, kuri praplėstų figūros klasę ir kuri turėtų atributą kraštinės ilgis. Turėtų metodus, kurie paskaičiuotų perimetrą ir plotą. Sukurti konstruktorių, kuris priskirtų atributų reikšmes.
* Sukurti klasę kubas, kuri praplėstų klasę kvadratas ir turėtų metodus, kurie paskaičiuotų plotą, perimetrą, tūrį.

Atliekant užduotis stengtis naudoti polimorfizmą ir base raktažodį ir kuo mažiau kartoti kodo. Perimetro ir ploto metodus pažymėti *virtual* ir perrašyti vaikinėje klasėje.

Užduotys atliktos teisingai, jei teisingai atlieka skaičiavimus ir jei kiekviena sukurta figūra padidina statinį figūrų skaičių. *Pi* reikšmei gauti naudokite *Math.PI* konstantą.

*6 užduotis.* SUKURKITE ABSTRAKČIĄ KLASĘ DARBUOTOJAS, KURI TURĖTŲ ATRIBUTĄ VARDAS IR KONSTRUKTORIŲ, KURIS INICIJUOJA ATRIBUTŲ REIKŠMES IR ABSTRAKTŲ *VOID* TIPO METODĄ DOWORK().

Sukurti klases: programuotojas, vadybininkas, kurios paveldėtų klasę darbuotojas.

Įgyvendinti abstrakčios klasės metodus.

*7 užduotis.* ATLIKTĄ *5 užduotį* „POLIMORFIZMO“ TEMOJE MODIFIKUOKITE: KLASĘ FIGŪRA PADARANT ABSTRAKČIA IR JOJE SUKURTI METODUS PLOTAS IR PERIMETRAS, KURIE BŪTŲ ABSTRAKTŪS.

Atlikti reikiamus pataisymus vaikinėse klasėse.

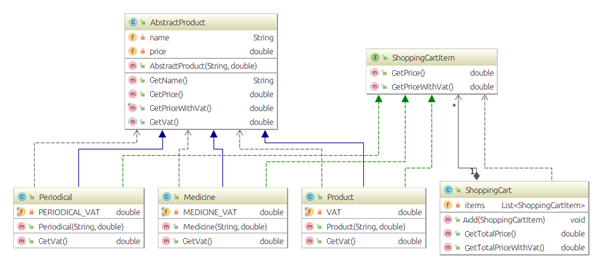
*8 užduotis.* SUKURKITE INTERFEISĄ GEOMETRINEI FIGŪRAI SU PLOTO IR PERIMETRO SKAIČIAVIMO METODAIS.

Sukurti klasę, įgyvendinančią sukurtą interfeisą.

### Tema. Pagrindinės UML klasių diagramos.

*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 C# KLASES NUBRAIŽYKITE UML KLASIŲ DIAGRAMĄ.

public class Shape

{

private string color;

public Shape(string color)

{

this.color = color;

}

public override string ToString() => $"Shape[color={color}]";

public virtual double GetArea()

{

Console.WriteLine("Shape unknown! Cannot compute area!");

return 0;

}

}

public class Rectangle : Shape

{

private int length;

private int width;

public Rectangle(string color, int length, int width) : base(color)

{

this.length = length;

this.width = width;

}

public override string ToString() => $"Rectangle[length={length},width={width},{base.ToString()}]";

public override double GetArea() => length \* width;

}

public class Triangle : Shape

{

private int @base;

private int height;

public Triangle(string color, int @base, int height) : base(color)

{

this.@base = @base;

this.height = height;

}

public override string ToString()=> $"Triangle[base={@base}, height={height}, {base.ToString()}]";

public override double GetArea() => 0.5 \* @base \* height;

}

### 3.3.3. Tema. Objektinio programavimo gerųjų praktikų (SOLID principų) taikymas .NET.

*11 užduotis.* APRAŠYKITE SOLID (SRP, OCP, LSP, ISP, DIP) PROGRAMAVIMO PRINCIPUS.

*12 užduotis.* APRAŠYKITE KAM SKIRTI INVERSION OF CONTROL (IOC) KONTEINERIAI IR KOKIE JIE YRA.

*13 užduotis.* NAUDOJANTIS PATEIKTA UML KLASIŲ DIAGRAMA SUPROGRAMUOKITE SOLID PRINCIPUS ATITINKANTĮ KAVOS APARATO KODĄ IR PROGRAMĄ, NAUDOJANČIĄ ŠIAS KLASES IR DEPENDENCY CONSTRUCTOR INJECTION.

### Mokymosi rezultatas. Naudoti daugiagijo programavimo .NET karkase priemones.

### Tema. Daugiagijis (Multithreaded) programavimas.

*1 užduotis.* SUKURKITE METODĄ, KURIS NURODYTĄ KIEKĮ KARTŲ (PVZ. 100) Į EKRANĄ IŠVEDA ESAMOS GIJOS ID.

Sukurti giją, kuri panaudotų sukurtąjį metodą ir ją paleisti. Metodą paleisti be gijos sinchroniškai. Stebėti gijų atliekamą darbą ir patvirtinti, kad į konsolę gijos rezultatus išveda lygiagrečiai.

*2 užduotis.* PANAUDOKITE METODĄ IŠ „THREAD DARBAS SU GIJOMIS“ *1 užduoties.*

Metodą paleisti su *Task.Run* ir *Task.Factory.StartNew* panaudojant lambda išraiškas vietoj *ThreadStart* delegatų.

*3 užduotis.* SUKURKITE KONSOLINĘ PROGRAMĄ SU METODU, KURIS 100 KARTŲ PADIDINTŲ BENDRAI PASIEKIAMĄ INT TIPO KINTAMĄJĮ VIENETU.

Paleisti 5 gijas naudojant *Task.Run*, kurios kviestų int kintamojo didinimo metodą. Įsitikinti, kad vietoj laukto rezultato 500, gautas kitas skaičius. Ištaisyti šį *thread-unsafe* kodą panaudojant *lock* sakinį.

### Tema. Asinchroninis programavimas naudojant .Net aplinką.

*4 užduotis.* SUKURKITE METODUS ASINCHRONINIAM ILGAI TRUNKANČIŲ UŽDUOČIŲ VYKDYMUI IMITUOTI, KURIAME PANAUDOTI *AWAIT TASK.DELAY() ASYNC* METODAI.

Metodas kaip parametrą priima laiką milisekundėmis, kuris ir panaudotas metodo trukmei nustatyti (t. y. panaudotas await *Task.Delay()* parametrui).

Sukurti papildomą *async* metodą, kuris triskart kviestų asinchroninį metodą su parametrais 2, 3, 5 sekundėms ir truktų ne ilgiau nei ilgiausiai trunkančio asinchroninio metodas (t. y. 5 sekundės).

Kviesti antrąjį metodą ir matavimui naudojant *Stopwatch* klasę parodyti kviestų metodų parametrus ir bendrą vykdymo trukmę.

*5 užduotis.* SUKURKITE ASYNC METODĄ, KURIS SUKELTŲ KLAIDĄ.

Kviesti sukurtą metodą asinchroniškai ir teisingai pagauti bei apdoroti klaidą.

*6 užduotis.* MODIFIKUOTI *4 užduotyje* PARAŠYTĄ INKAPSULIUOJANTĮ METODĄ TAIP, KAD REIKĖTŲ LAUKTI NE KIEKVIENOS UŽDUOTIS ATSKIRAI, O BENDRAI VISŲ TRIJŲ UŽDUOČIŲ PABAIGOS.

### Mokymosi rezultatas. Testuoti programinę įrangą naudojant su .NET karkasu suderinamus testavimo įrankius ir metodus.

### Tema. Programų testavimas.

*1 užduotis.* SUKURKITE KLASĘ SU VIENU TEKSTINIU LAUKU, KURIS PADUODAMAS PER KONSTRUKTORIŲ IR METODU, KURIS GRĄŽINA TĄ LAUKĄ.

Klasei parašyti testą, kuris patikrina ar kuriant klasės instanciją paduotas tekstas yra tas pats, kurį grąžina klasė naudojantis vieninteliu klasės metodu.

*2 užduotis.* SU MS TEST TESTAVIMO KARKASU SUKURKITE MODULIŲ TESTŲ PROJEKTĄ.

Projekto klasėje sukurti:

* Testą, kuris patikrina ar įvykdytos sudėties operacijos 2+2 suma 4. Panaudoti *Assert.AreEqual()* metodą
* Testą, kurio bloke būtų metama klaida. Ant testo metodo uždėti *ExpectedException* atributą
* Testą, kuris yra nesėkmingas. Pvz. Neteisinga sudėties operacija

*3 užduotis.* SUKURKITE KLASĘ TODOITEM, KURI TURĖTŲ ATRIBUTUS: PAVADINIMAS, STATUSAS.

Pradinis statusas turi būti „notstarted“

Klasei parašyti modulių testus. Klasės metodai:

* Start() – metodas turi pakeisti statusą į „Doing“. Turi būti statuso patikrinimas, kad galima pradėti darbus tik, kurie yra nepradėti arba turi statusą „NotStarted“. Jei blogas statusas mesti klaidą.
* Done() – metodas turi pakeisti statusą į „Done“. Turi būti statuso patikrinimas, kad galima pradėti darbus tik, kurie yra nepradėti arba turi statusą „Doing“. Jei blogas statusas mesti klaidą.
* Archive() – metodas turi pakeisti statusą į „Archived“. Turi būti statuso patikrinimas, kad galima pradėti darbus tik, kurie yra nepradėti arba turi statusą „Done“. Jei blogas statusas mesti klaidą.

*4 užduotis.* SUKURKITE METODĄ, KURIS VARTOTOJO PAPRAŠYTŲ ĮVESTI NORIMO MASYVO DYDĮ IR VISUS JO NARIUS.

Po to paprašytų vartotojo įvesti norimo masyvo nario indeksą. Programos kodą patalpinti *try* bloke. Sukurti kelis *catch* blokus, kurie apdorotų visas galimas kodo klaidas. *Catch* blokai:

* *OverflowException* – nurodamas netinkamas masyvo dydis
* *IndexOutOfRangeException* – nurodomas neegzistuojantis masyvo indeksas
* *FormatException* – vartotojas neteisingai įveda įversti ir *int.Parse* metodas meta klaidą.

*5 užduotis.* SUKURKITE METODĄ „LOGIN“, KURIS PAPRAŠYTŲ VARTOTOJĄ ĮVESTI PRISIJUNGIMĄ IR SLAPTAŽODĮ.

Jei vartotojo įvestis neatitinka sugalvotų duomenų, tai mesti klaidą su pranešimu „Blogi prisijungimo duomenys“, jei duomenys teisingi spausdinti „Sveiki prisijungę“. Metodą panaudoti *Main*() metode tarp *try-catch* blokų.

*6 užduotis.* SUKURKITE KLASĘ BEAR., KURI TURI ATRIBUTUS: RŪŠIS, AR MIEGA.

* Sukurti klasę medus, kuris turi atributus rūšis, ar šviežias.
* Meškai sukurti metodą - duoti medaus, kurio parametras yra medus. Šis metodas turi patikrinti ar medus nėra grikių ar garstyčių rūšies. Taip pat ar medus šviežias. Jei taip yra, tada turi būti metama klaida, kad meška yra pikta. Jei meška miega, turi mesti klaidą, kad meška miega. Šioms klaidomis naudoti *Exception.*
* Sukurti savo klaidą kuri bus metama, kai meška yra miega, kuri turėtų atributą kaip garsiai meška riaumoja, kai yra pažadinta. Perrašyti *Message* atributą, kad būtų nurodyta kaip garsiai meška riaumoja.
* Sukurti klaidą, kuri per konstruktorių priima medų ir gali atspausdinti koks medus meškai nepatiko.
* Pritaikyti sukurtas klaidas.

### Tema. Programų derinimas.

*7 užduotis.* APIBRĖŽKITE, KAS YRA PROGRAMINĖS ĮRANGOS ŽURNALAI.

*8 užduotis.* ĮVARDINKITE, KOKIOS ŽURNALŲ PRIEMONĖS YRA SKIRTOS .NET PLATFORMAI.

*9 užduotis.* KONSOLINĖS APLIKACIJOS PROJEKTE, NAUDOJANT SERILOG ŽURNALO ĮRANKĮ, IŠVESKITE PAPRASTĄ INFORMACINĮ ŽURNALO PRANEŠIMĄ Į KONSOLĘ.

*10 užduotis.* APIBRĖŽKITE, KAS YRA KODO DERINIMO PROCESAS.

*11 užduotis.* ĮVARDINKITE, KOKIOS KODO DERINIMO PRIEMONES PATEIKIAMOS VISUAL STUDIO APLINKOJE.

*12 užduotis.* APIBRĖŽKITE, KAS YRA STATINĖ KODO ANALIZĖ.

*13 užduotis.* IŠVARDINKITE KELIS STATINĖS KODO ANALIZĖS ĮRANKIUS, SKIRTUS .NET PLATFORMAI.

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

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

### Tema. Reikalavimų programinei įrangai formatai.

*1 užduotis.* UŽRAŠYKITE VARTOTOJO PASAKOJIMO (USER STORY) FORMATĄ.

Pateikite pavyzdžių.

Yra pateikti funkciniai programinės įrangos reikalavimai. pabandykite suformuoti tris vartotojo pasakojimus (user stories)

|  |
| --- |
| **FR1. Sistemoje egzistuoja vartotojai**  Sistemoje egzistuoja vartotojai, kurie gali dirbti su sistema prisijungę. Prisijungimas vykdomas pateikiant vartotojo vardą ir slaptažodį.  **FR2. Sistemoje egzistuoja administratoriaus rolė**  Sistemoje egzistuoja speciali vartotojų rolė - administratoriai. Administratoriai gali:   * kurti vartotojų grupes (žiūrėti FR3), priskirti joms vartotojus * kurti dokumentų tipus ir priskirti grupes galinčias kurti dokumentus bei juos priimti * registruoti naujus vartotojus sistemoje * keisti esamų vartotojų slaptažodžius |

*2 užduotis.* APRAŠYKITE GERŲ VARTOTOJO PASAKOJIMO FORMATĄ (INVEST).

### Tema. Reikalavimų peržiūros procesas.

*3 užduotis.* APRAŠYKITE REIKALAVIMŲ PERŽIŪROS ŽINGSNIUS.

*4 užduotis.* APRAŠYKITE REIKALAVIMŲ PARUOŠIMO SPRINTUI KRITERIJUS (*DEFINITION OF READY*).

Pateikite pavyzdžių.

*5 užduotis.* APRAŠYKITE REIKALAVIMŲ BAIGTUMO APIBRĖŽIMĄ.

Pateikite pavyzdžių.

*6 užduotis.* APRAŠYKITE PAGRINDINIUS *SCRUM* PROCESO ĮVYKIUS.

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

### Tema. Funkciniai reikalavimai.

*1 užduotis.* APRAŠYKITE KAS YRA FUNKCINIAI REIKALAVIMAI.

Pateikite pavyzdžių.

*2 užduotis.* PATEIKTAM PAVYZDŽIUI („FUNKCINIAI KOMPIUTERINĖS PROGRAMOS REIKALAVIMAI“ POTEMĖS UŽDUOČIAI) PARAŠYKITE, KAIP BUS NUSTATOMA AR FUNKCINIS REIKALAVIMAS BUVO ĮVYKDYTAS AR NE.

### Tema. Nefunkciniai reikalavimai.

*3 užduotis.* APRAŠYKITE KAS YRA NEFUNKCINIAI REIKALAVIMAI.

Pateikite pavyzdžių.

*4 užduotis.* PATEIKTAM PAVYZDŽIUI („NEFUNKCINIAI (TECHNINIAI, SAUGOS) KOMPIUTERINĖS PROGRAMOS REIKALAVIMAI“ POTEMĖS UŽDUOČIAI) PARAŠYKITE KAIP BUS NUSTATOMA AR NEFUNKCINIS REIKALAVIMAS BUVO ĮVYKDYTAS AR NE.

#### 

# Modulis „Nesudėtingų duomenų bazių projektavimas ir kūrimas (.Net)“

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

### Mokymosi rezultatas. Pateikti reliacinės duomenų bazės schemą.

### Tema. Duomenų bazių projektavimas (CREATE TABLE sakinys, duomenų normalizavimas).

#### *1 užduotis*. SUPROJEKTUOKITE 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*. SUPROJEKTUOKITE 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?

### Mokymosi rezultatas. Pateikti nereliacinės (NoSQL) duomenų bazės schemą.

### Tema. NoSQL duomenų bazių tipai ir jų savybės.

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.

### Tema. NoSQL duomenų bazių valdymas.

#### *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.

## Kompetencija. Programiškai įgyvendinti ir administruoti duomenų bazes.

### Mokymosi rezultatas. Diegti ir valdyti duomenų bazių valdymo sistemą.

### 2.1.1. Tema. DBVS diegimas.

*1 užduotis.* ĮDIEGTI IR SUKONFIGŪRUOTI MS SQL SERVER EXPRESS EDITION IR SQL SERVER MANAGEMENT STUDIO (SSMS)

Autentifikacijos tipas turėtų būti Windows autentifikacija. Sukurto serverio instancijos pavadinimas turėtų būti jūsų vardas

### Tema. DBVS administravimas.

*2 užduotis.* SUKURKITE NAUJĄ SQL VARTOTOJĄ NAUDOJANT SQL SKRIPTĄ ARBA GRAFINĘ SSMS SĄSAJĄ IR SUTEIKITE JAM ADMINISTRATORIAUS TEISES.

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

### 2.2.1. Tema. Duomenų išrinkimas naudojant SQL select sakinį ir pagrindinius select elementus.

*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);

ATLIKITE UŽKLAUSAS IMPORTUOTOJE DB (NAUDOJAMOS PRIEMONĖS/APLINKOS – SQL DB 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.

### Tema. Duomenų išrinkimas naudojant sąryšius (SQL select su join).

#### *2 užduotis*. PANAUDOKITE TINKAMAS SĄLYGŲ IR GRUPAVIMO SQL KOMANDAS.

Duota:

1. 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 IMPORTUOTOJE DB (NAUDOJAMOS PRIEMONĖS/APLINKOS – SQL DB 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?

#17 punkto užklausą pataisykite taip, kad rodytų tik tuos projektus, kur dirba bent 3 darbuotojai.

### Mokymosi rezultatas. Kurti duomenis duomenų bazėje valdančią programinę įrangą.

### Tema. Duomenų bazių naudojimas programų sistemose naudojant ADO.NET sąsają.

*1 užduotis.* APIBŪDINKITE, KAS YRA ADO.NET IR KOKIAS PROGRAMAVIMO PRIEMONES JIS TEIKIA .NET KARKASE.

*2 užduotis.* SUKURKITE KONSOLINĘ PROGRAMĄ, KURI DUOMENŲ BAZĖS LENTELĖS DUOMENŲ IŠVEDIMUI NAUDOJA DATAREADER.

*3 užduotis.* SUKURKITE KONSOLINĘ PROGRAMĄ, KURI ĮRAŠO DUOMENŲ EILUTĘ Į DUOMENŲ BAZĖS LENTELĘ IR GRĄŽINA ĮRAŠYTOS EILUTĖS NUMERĮ.

Db lentelė turi turėti *auto-increment*

Naudoti insert komandą su parametrais.

*4 užduotis.* SUKURKITE KONSOLINĘ PROGRAMĄ, KURI DB LENTELĖS SUKŪRIMUI, PIRMŲ DUOMENŲ ĮVEDIMUI IR TŲ DUOMENŲ IŠVEDIMUI NAUDOJA EF6 KARKASĄ IR CODE FIRST METODĄ.

*5 užduotis.* SUKURKITE MIGRACIJĄ ANKSTESNĖS UŽDUOTIES SUKURTAI DUOMENŲ BAZEI: PRIDĖKITE LENTELEI LAUKĄ (SUGALVOTI SAVARANKIŠKAI) IR ĮVYKDYKITE SĖKMINGĄ MIGRACIJĄ.

# Modulis „Programavimo aplinkos ir kūrimo proceso valdymas (.NET)“

## Kompetencija. Naudoti tarnybinių stočių operacines sistemas.

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

## 

## Tema. Serverio operacinės sistemos diegimas.

*1 užduotis.* WINDOWS SERVER 2016 DESKTOP EXPERIENCE DIEGIMAS IR KONFIGŪRAVIMAS.

Vykdomas grafinės sąsajos pagalba.

Naudojamos priemonės (aplinkos): Diegimo aplinkai patogiausia naudoti Windows operacinę sistemą su virtualizacijos platforma kaip, kad vmware workstation, vmware player ar virtual box.

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

## 1.2.1. Tema. Pagrindinės serverio administravimo funkcijos ir komandos.

*1 užduotis.* WINDOWS SERVER 2016 SERVER CORE DIEGIMAS IR KONFIGŪRACIJA (POWERSHELL KOMANDINĖS EILUTĖS PAGALBA).

Susipažinome Windows server operacine sistema ir ją įdiegėme grafinio naudotojo sąsajos (GUI) pagalba. Sekantis žingsnis siekiant optimizuoti serverio resursus – įdiegti Windows Server Core leidimą. Diegimo procesas toks pat kaip ir prieš tai diegto Standard/Datacenter leidimo su „desktop experience“ pasirinkimu. Visgi šis leidimas administravimo ir konfigūracijos prasme neturi grafinės naudotojo sąsajos. Konfigūracija turi būti atliekama CMD ir Powershell komandinių eilučių pagalba.

Atlikimo/rezultato vertinimo kriterijai: įdiegta Windows Server 2016 Core operacinė sistema, sukonfigūruoti tinklo nustatymai (serveris gali pasiekti internetą), pakeistas serverio vardas, sudiegti windows atnaujinimai, sudiegtas ir sukonfigūruotas web serveris.

Papildomi informacijos šaltiniai:

<https://www.edx.org/xseries/microsoft-windows-server-2016>

<https://info.microsoft.com/TheUltimateGuideToWindowsServer2016.html?ls=Website>

## Mokymosi rezultatas. Valdyti programinius paketus.

## 1.3.1. Tema. Programiniai paketai.

*1 užduotis.* ĮDIEKITE WINDOWS SERVER 2016 HYPER-V ROLĘ.

Hyper-V manager pagalba paleisti du virtualius serverius, sukurti komunikaciją tarp jų.

*2 užduotis.* WINDOWS CONTAINERS / DOCKER DIEGIMAS IR KONFIGŪRACIJA.

## Mokymosi rezultatas. Naudoti Web serverio programinę įrangą HTTP bylų viešinimui.

## 1.4.1. Tema. Web serverio aplinka operacinėje sistemoje.

*1 užduotis.* ASMENINIAME KOMPIUTERYJE INSTALIUOKITE IR SUKONFIGŪRUOKITE INTERNET INFORMATION SERVICES (IIS) WEB PROGRAMINĘ ĮRANGĄ.

*2 užduotis.* IIS APLIKACIJOS ERDVIŲ KONFIGŪRAVIMAS.

IIS programinės įrangos pagalba sukurti aplikacijos erdves:

1. Application pool
   1. Suteikti savo pavadinimą, be .Net karkaso versijos ir „Classic” pipeline režimą
2. Tinklapis (angl. Site)
   1. C:\inetpub direktorijoje sukurti katalogą „aplikacija”. Kataloge patalpinti index.html failą su taisyklingu html turiniu, kuris bus paviešintas tinklapio pagalba.
   2. Sukurti naują tinklapį, priskirti savo sukurtą application pool ir nurodyti kelią į naujai sukurtą katalogą.
   3. Per naršyklę naviguoti iki tinklapio adreso, kuris buvo priskirtas.
3. Virtuali direktorija (angl. Virtual Directory)
   1. Susikurti naują katalogą „images” c:\intepub kataloge.
   2. Sukurti virtualią direktoriją ir paviešinti sukurtą katalogą.
   3. Sukurtame kataloge patalpinti paveiksliukus.
   4. Per naršyklę pasitikrinti ar paveiksliukai pasiekiami per priskirtą adresą.
4. Aplikacija (angl. Application)
   1. C:\ inetpub direktorijoje susikurti naują katalogą ir jame patalpinti index.html bylą su html turiniu.
   2. Sukurti aplikaciją savo tinklapyje ir nurodyti sukurto katalogo direktoriją.
   3. Per naršyklę įjungti aplikaciją.
   4. Pakoreguoti html bylą taip, kad joje būtų bent vienas paveiksliukas iš anksčiau sukurtos virtualios direktorijos. Per naršyklę pasitikrinti ar paveiksliukai pasiekiami per priskirtą adresą.
5. Programavimo aplinka Visual Studio ir IIS
   1. Sukurti naują .Net karkaso web projektą ir pakeisti nustatymus, kad projektas veiktų per lokalų IIS serverį.
   2. Sukurti virtualią direktoriją per Visual Studio.

## 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.

### Mokymosi rezultatas. Suprasti SCRUM proceso dalis ir komandos narių atsakomybes.

### Tema. Scrum procesas.

#### *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

### Mokymosi rezultatas. Analizuoti pateiktus reikalavimus, nustatant programos atitikimą reikalavimams.

### Tema. Programinės įrangos reikalavimų analizė.

#### *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).

### Mokymosi rezultatas. Naudoti projekto eigos valdymo principus.

### Tema. Projekto eigos valdymas.

#### *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.

### Mokymosi rezultatas. Sekti programavimo darbų vykdymą naudojant komandinio darbų planavimo sistemas.

### 3.2.1. Tema. Komandinio darbų planavimo sistemos.

#### *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.

### Tema. Programavimo darbų vykdymo sekimas.

#### *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.

### Mokymosi rezultatas. Vykdyti programinio kodo versijavimą naudojant programinio kodo versijavimo įrankius, tinkamus C# programavimo kalbai.

### Tema. Išeities kodo saugyklos.

#### *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 IR 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.

### Tema. Programinio kodo versijavimo vykdymas.

#### *9 užduotis.* NUKOPIJUOKITE KODO PAVYZDĮ IŠ NUOTOLINĖS GITHUB KODO SAUGYKLOS: <https://github.com/PacktPublishing/.NET-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 .NET 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. Ar C# kalba palaiko daugybinį paveldimumą (angl. *multiple inheritance*)?
   1. Taip
   2. Ne
2. Iš kurios .NET klasės paveldi visos kitos klasės?
   1. Object
   2. System.Net
   3. System.Object
   4. System
   5. System.Root
3. Kuri .NET kolekcija leidžia pasiekti elementus naudojant unikalų raktą?
   1. ListDictionary
   2. Stack
   3. Hashtable
   4. ArrayList
   5. StringCollection
4. Kam yra skirtos enumeracijos *enum*?
   1. Skirta inicializuoti kintamiesiems.
   2. Skirta apibrėžti konstantoms.
   3. Skirta apibrėžti kintamiesiems.
5. Ar *finally* blokas bus įvykdytas įvykus nenumatytai klaidai (angl. *exception*)?
   1. Taip
   2. Ne
6. Ar gali būti įvykdyti keli *catch* blokai vienam *try* blokui?
   1. Taip
   2. Ne
7. Ar .NET palaiko galimybę paveldėti iš kelių interfeisų?
   1. Taip
   2. Ne
8. Kurie žodžiai yra duomenų bazių ACID savybių rinkinio dalis (pasirinkti visus)?
   1. Isolated
   2. Connected
   3. Durable
   4. Atomic
   5. Integrity
   6. Consistent
   7. Applied
   8. Derivative
9. Kokį .NET tipą reiškia C# raktažodis *int*?
   1. System.Int16
   2. System.Int32
   3. System.Int64
   4. System.Int128
10. Kuris iš pateiktų sakinių teisingai apibrėžia dvimatį skaičių masyvą C# kalboje?
    1. Int[,] myArray;
    2. Int[][] myArray;
    3. Int[2] myArray
    4. System.Array[2] myArray;
    5. System.Array[,] myArray;
11. Jei metodas yra pažymėtas *protected internal* pasiekiamumo modifikatoriumi, kas gali jį pasiekti?
    1. Klasės, kurios yra toje pačioje asamblėjoje (angl. *assembly*) ir paveldi iš metodą apibrėžiančios klasės, t. y. atitinka abi sąlygas.
    2. Tik metodai, kurie yra toje pačioje klasėje kaip ir metodas klausime.
    3. *internal* metodai gali būti kviečiami tik naudojantis *Reflection* priemonėmis.
    4. Klasės, kurios yra toje pačioje asamblėjoje (angl. *assembly*), arba klasės, kurios paveldi iš metodą apibrėžiančios klasės, t. y. atitinka vieną iš šių sąlygų.
12. Kas yra *boxing*?
    1. Objekto inkapsuliavimas į *value* tipą.
    2. *value* tipo kopijos inkapsuliavimas įobjektą.
    3. Objekto kopijos inkapsuliavimas į *value* tipą.
    4. *value* tipo inkapsuliavimas į objektą.
13. Kuris kompiliatoriaus jungiklis sugeneruoja XML failą XML komentarams esantiems asamblėjoje (angl. *assembly*)?
    1. /text
    2. /doc
    3. /xml
    4. /help
    5. /xmlhelp
14. Kuris projektavimo šablonas yra išreikštas šiuo kodu?

|  |
| --- |
| public class A  {  private A instance;  private A() { }  public static A Instance  {  get  {  if (instance == null)  instance = new A();  return instance;  }  }  } |

* 1. Factory
  2. Abstract Factory
  3. Singleton
  4. Builder

1. Kurios iš operacijų negalima atlikti su ADO.NET DataSet klase?
   1. DataSet gali būti sinchronizuotas su duomenų baze.
   2. DataSet gali būti sinchronizuotas su RecordSet.
   3. DataSet gali būti konvertuotas į XML.
   4. Galima gauti duomenų reprezentacijos schemą iš DataSet objekto.
2. Kuris teiginys geriausiai apibūdina inkapsuliaciją objektinio programavimo kontekste?
   1. Objekto konvertavimas iš vieno tipo į kitą.
   2. Kelių tipų savybių ir metodų abstraktinimas į vieną bendrą.
   3. Objekto savybių gavimas iš tėvinio objekto.
   4. Interfeiso ir įgyvendinimo detalių atskyrimas (informacijos slėpimas).
3. Kaip šifruojamas MVC šablono akronimas?
   1. Model, Vision, Control
   2. Model, View, Controller
   3. Model, ViewData, Controller
   4. Model, Data, Controller
4. Kas yra modelis MVC šablone?
   1. Duomenys
   2. HTML turinys
   3. Verslo logika
   4. Atvaizdavimo variklis
5. Kuris iš nurodytų tipų yra *View* tipas ASP.NET karkase?
   1. Partial view
   2. Executable view
   3. Data view
   4. Designer view
6. Įterpkite trūkstamą žodį: \_\_\_\_\_\_ atributai gali būti naudojami duomenų validacijai ASP.NET karkase.
   1. DataAnnotations
   2. Fluent API
   3. DataModel
   4. HtmlHelper
7. Kuris yra numatytas kelias (angl. *route*) ASP.NET karkase?
   1. /{action}/{controller}/{id}
   2. {controller}/{id}
   3. {controller}/{action}/{id}
   4. {controller}/{action}
8. Kuri klasė yra skirta kelių (angl. *routes*) konfigūravimui?
   1. FilterConfig
   2. RegisterRouteConfig
   3. RouteConfig
   4. MVCRoutes
9. Kas yra .NET karkaso asamblėja (angl. *assembly*)?
   1. Visų .NET klasių rinkinys.
   2. .NET karkaso versijos informacija.
   3. .NET karkaso vystymo susirinkimas.
   4. Mažiausias platinamas .NET programos vienetas.
10. Ar galima perrašyti (angl. *override*) *private virtual* metodus?
    1. Taip
    2. Ne
11. Kuris iš .NET komponentų pašalina nebenaudojamus objektus iš *managed heap* atminties?
    1. Garbage Collector (GC).
    2. Just-in-Time (JIT).
    3. Common Language Runtime (CLR).
    4. Intermediate Language (IL).
12. Kaip vadinamas kintamasis, kuris apibrėžtas metodo viduje?
    1. Privatus
    2. Lokalus
    3. Statinis
    4. Linijinis
13. Ar abstrakti klasė gali turėti numatytą metodo implementaciją?
    1. Taip
    2. Ne
14. Koks skirtumas tarp *Convert.ToInt32* ir *Int.Parse* metodų?
    1. Abu metodai išduos *ArgumentNullException* klaidos padavus *null* reikšmę, kaip parametrą.
    2. *Convert.ToInt32* negali suvaldyti *null* reikšmių,išduos *ArgumentNullException* klaidą.
    3. *+Int.Parse* negali suvaldyti *null* reikšmių,išduos *ArgumentNullException* klaidą
    4. Abu metodai neišduos *ArgumentNullException* klaidos padavus *null* reikšmę, kaip parametrą.
15. Ar galima pakeisti kintamojo reikšmę derinimo metu?
    1. Taip
    2. Ne

# Užduočių atsakymai

# Modulis „Įvadas į profesiją“

Testo atsakymai

1. b

2. b

3. c

4. b

5. a

6. c

7. b

8. a

9. a

10. a

11. d

12. b

13. c

14. b

15. b

# Modulis „Informacinių sistemų projektavimas ir kūrimas (.NET)“

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

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

### Tema. Pagrindinės HTML (HTML 5) kalbos žymės.

#### *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>

### Tema. HTML kalbos turinio žymės.

#### *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“

* 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>

* 1. 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>

* 1. 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>

### 

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

### 

### Tema. CSS pagrindai.

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

Rezultatai:

1. indexEx.html

<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>

1. 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;

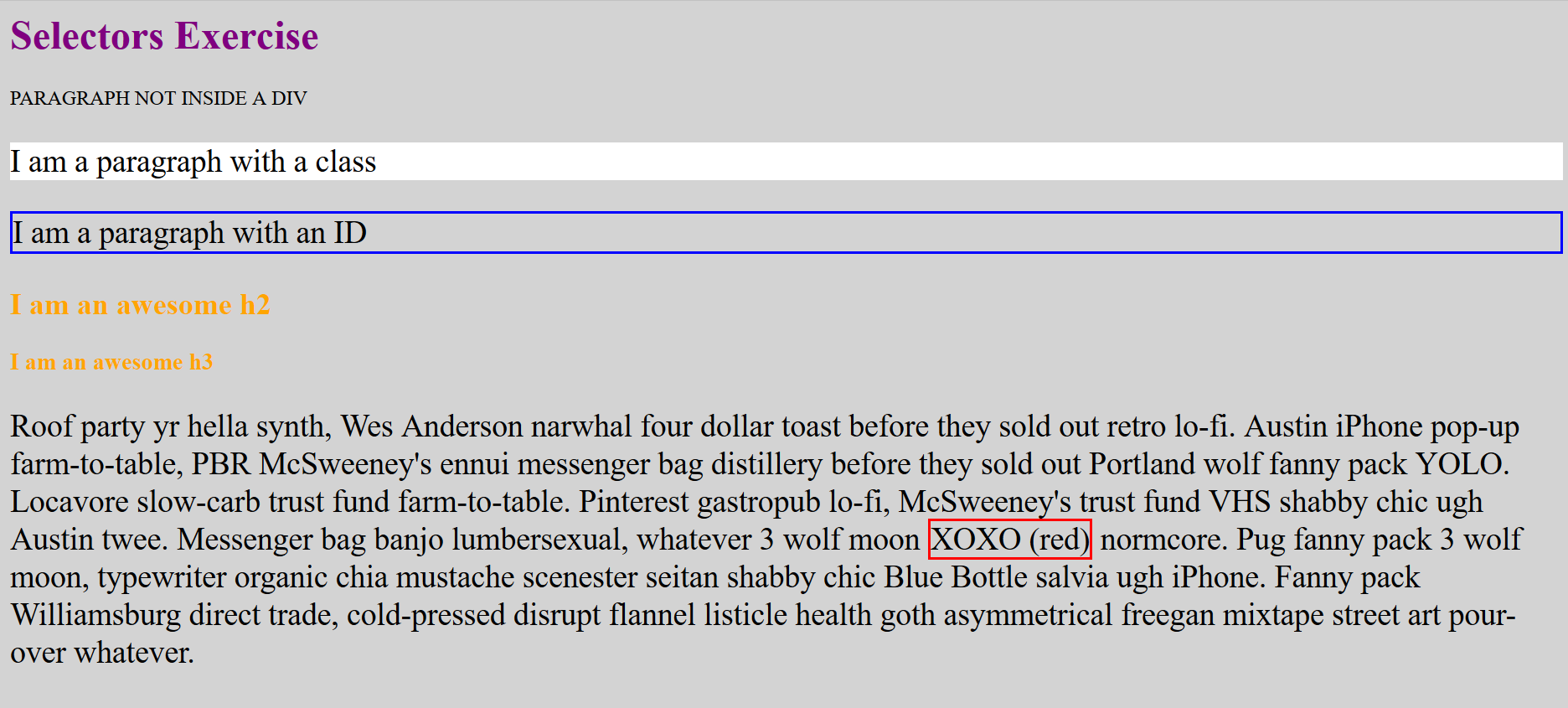
}

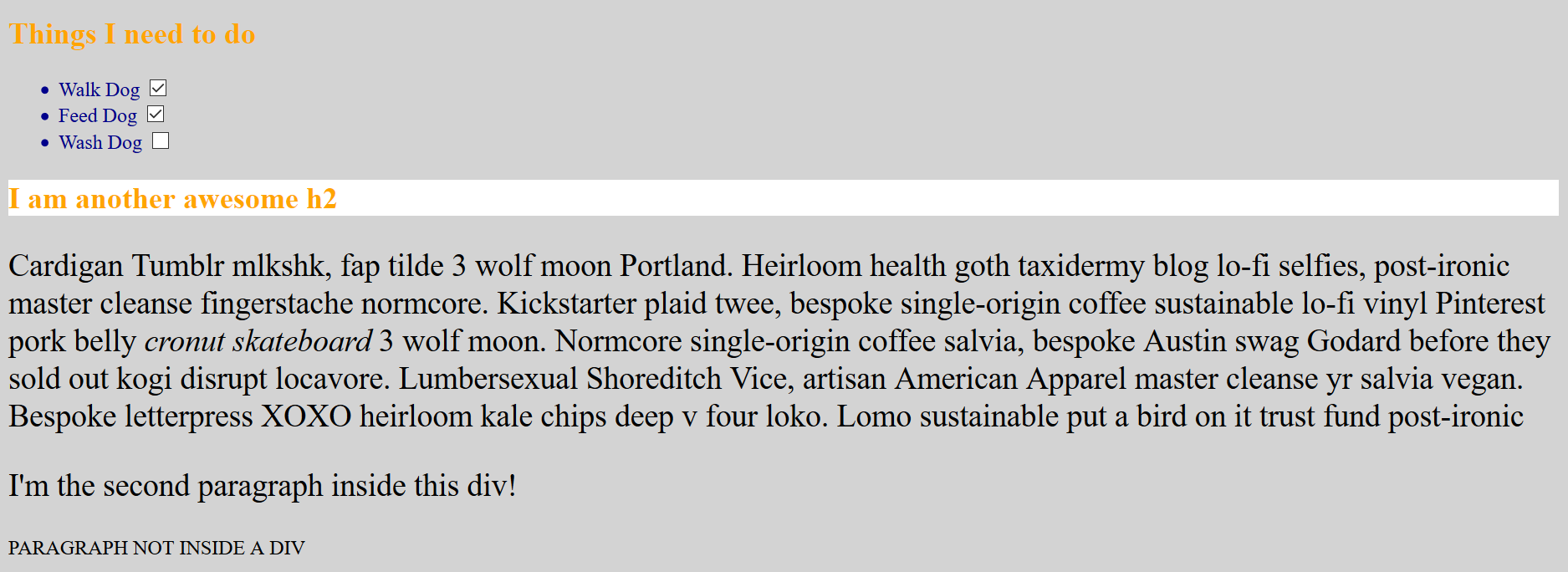
div > ul{

    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;

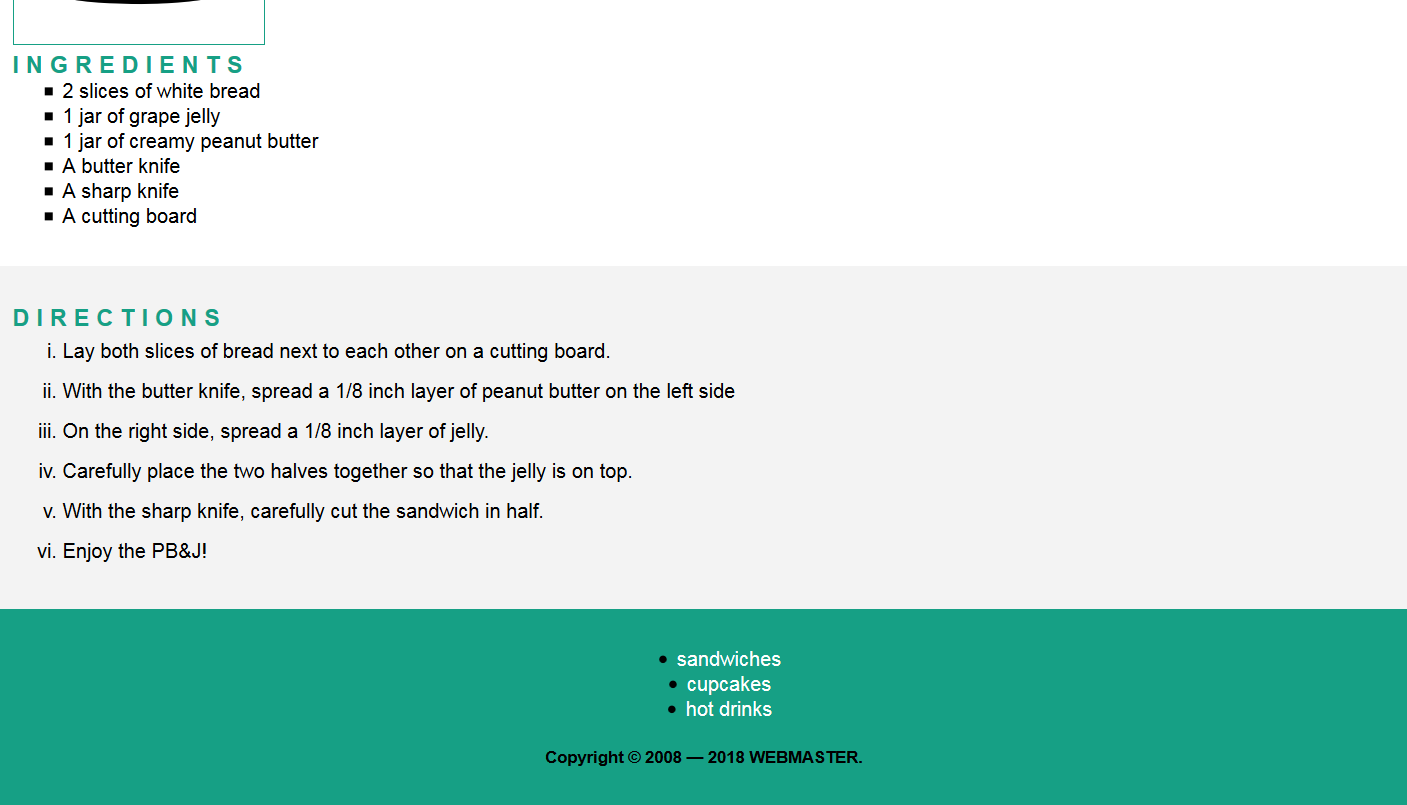
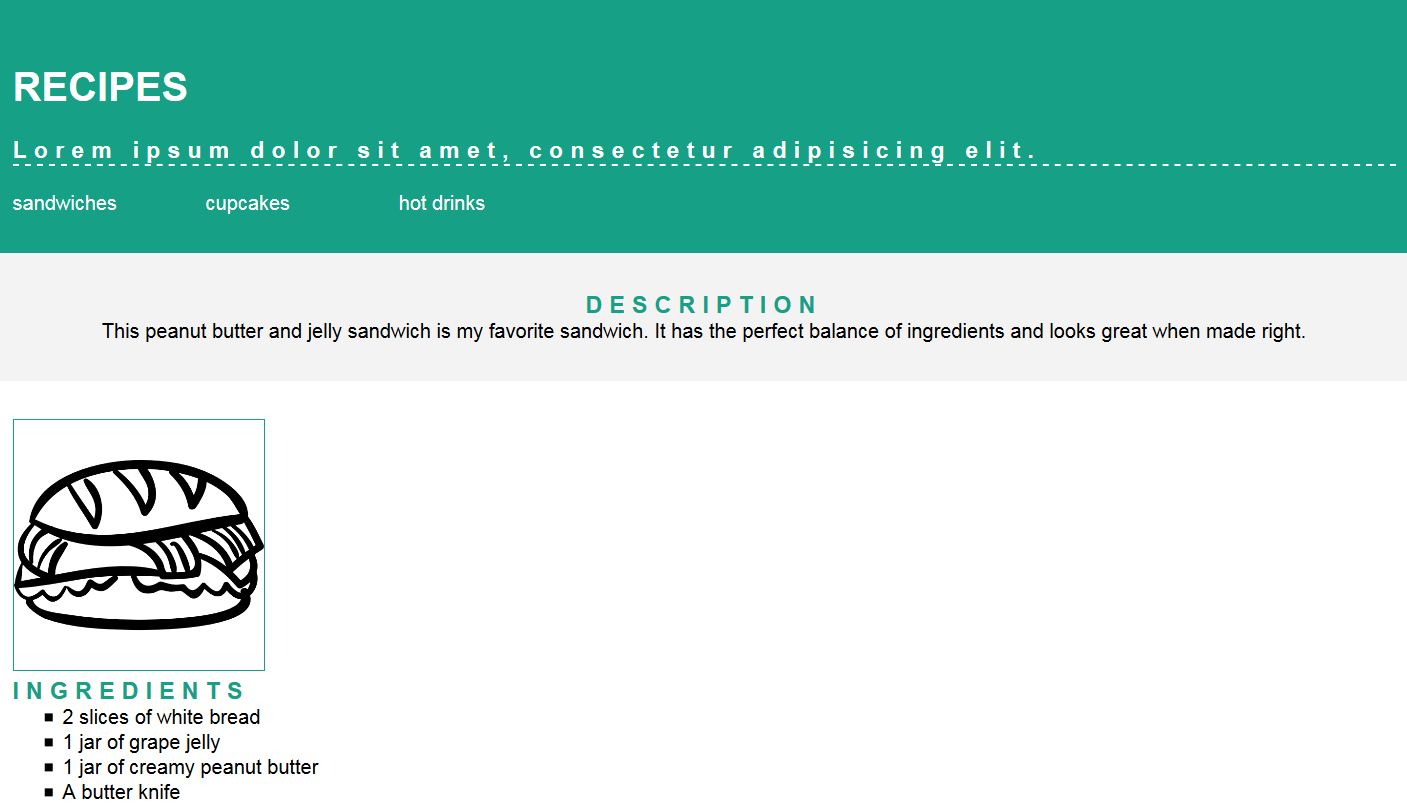
\*/

footer h5{

    padding-top: 20px;

}

Bendras rezultatas naršyklėje turi atrodyti taip:



### 

### Tema. CSS tinklalapio maketo kūrimo technikos.

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

Bendras rezultatas naršyklėje turi atrodyti taip:







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

### Tema. JavaScript kalbos pagrindai.

#### *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);

})();

1. 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;

}

#### 

### Tema. Dinaminis manipuliavimas naudojant DOM API.

#### *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>

1. Script.js

$(".accordion").on("click", ".accordion-header", function() {

$(this).toggleClass("active").next().slideToggle();

});

1. 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>

1. 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");

}

});

1. 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 PRIDĖKITE IR PAŠALINKITE 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>

1. 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();

}

});

});

## Kompetencija. Taikyti programinės įrangos kūrimui naudojamus informatikos principus ir metodus.

### Mokymosi rezultatas. Suprasti skaičiavimo sistemas.

### Tema. Skaičiavimo sistemos.

*1 užduotis.* APRAŠYKITE KAIP NUMERAVO SENOVĖS BABILONIEČIAI, EGIPTIEČIAI, GRAIKAI, SLAVAI.

#### Informaciją galite rasti čia:

1. Wikipedia (2019). Numeral system. [žiūrėta 2019-05-27]. Prieiga per internetą <https://en.wikipedia.org/wiki/Numeral_system>.

2. Vilniaus universitetas (2019). Apie skaičiavimo sistemas. [žiūrėta 2019-05-27]. Prieiga per internetą <https://www.mif.vu.lt/ljmm/rasmenys/skaiciavimo_sistemos.htm>.

*2 užduotis.* SURAŠYKITE KOKIOSE ŽMOGAUS VEIKLOS SRITYSE SKAIČIAVIMO SISTEMOS TURI DIDŽIAUSIĄ REIKŠMĘ.

Atsakydami į šį klausimą galite naudotis ekonominės veiklos rūšių klasifikatoriumi:

Lietuvos statistikos departamentas (2019). *Ekonominės veiklos rūšių klasifikatorius*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://osp.stat.gov.lt/static/evrk2.htm>

*3 užduotis.* SURAŠYKITE VISAS JUMS ŽINOMAS SKAIČIAVIMO SISTEMAS (POZICINĖS).

Dešimtainė, dvejetainė, aštuntainė, šešioliktainė, N-ainė skaičiavimo sistemos. Papildomai galite paskaityti:

Wikipedia (2019). List of numeral systems. [žiūrėta 2019-05-27]. Prieiga per internetą <https://en.wikipedia.org/wiki/List_of_numeral_systems>

*4 užduotis.* UŽRAŠYKITE DEŠIMTAINIUS SKAIČIUS 125, 256, 2019 DVEJETAINIAIS SKAIČIAIS.

12510 = 11111012

25610 = 1000000002

201910 = 111111000112

Apie skaičių keitimą iš vienos sistemos į kitą galite paskaityti čia:

Tutorials Point (2019). Number System Conversion. [žiūrėta 2019-05-27]. Prieiga per internetą <https://www.tutorialspoint.com/computer_logical_organization/number_system_conversion.htm>

*5 užduotis.* UŽRAŠYKITE DEŠIMTAINIAIS SKAIČIAIS ŠIUOS DVEJETAINĖS SISTEMOS SKAIČIUS: 101100110001, 1010101, 001110011.

1011001100012 = 286510

10101012 = 8510

0011100112 = 11510

*6 užduotis.* UŽRAŠYKITE DEŠIMTAINIUS SKAIČIUS 14, 56, 2019 AŠTUNTAINIAIS SKAIČIAIS.

1410 = 168

5610 = 708

201910 = 37438

*7 užduotis.* UŽRAŠYKITE DEŠIMTAINIAIS SKAIČIAIS ŠIUOS AŠTUNTAINĖS SISTEMOS SKAIČIUS: 77, 66, 32.

778 = 6310

668 = 5410

328 = 2610

*8 užduotis.* UŽRAŠYKITE DEŠIMTAINIUS SKAIČIUS 62, 216, 2019 ŠEŠIOLIKTAINIAIS SKAIČIAIS.

6210 = 3e16

21610 = d816

201910 = 7e316

*9 užduotis.* UŽRAŠYKITE DEŠIMTAINIAIS SKAIČIAIS ŠIUOS ŠEŠIOLIKTAINĖS SISTEMOS SKAIČIUS: 1D5F, A9BC, FF.

1d5f16 = 751910

a9bc16 = 4345210

FF16 = 25510

### Tema. Informacijos matavimo vienetai.

*10 užduotis.* UŽRAŠYKITE VISUS JUMS ŽINOMUS INFORMACIJOS KIEKIO MATAVIMO VIENETUS.

Bitas, baitas, kilobaitas, megabaitas gigabaitas, terabaitas, petabaitas, eksabaitas. Papildomai galite paskaityti čia:

Wikipedia (2019). Units of information. [žiūrėta 2019-05-27]. Prieiga per internetą <https://en.wikipedia.org/wiki/Units_of_information>.

*11 užduotis.* TURITE 16GB USB ATMINTUKĄ.

Paskaičiuokite kiek toks atmintukas turi megabaitų (mb), kilobaitų (kb) ir baitų (b).

Paskaičiuokite kiek galite patalpinti į tokią laikmeną bitų informacijos.

16GB = 16 384 MB (megabaitų)

16GB = 16 777 216 KB (kilobaitų)

16GB = 17 179 869 184 B (baitų)

16GB = 137 438 953 472 b (bitų)

Papildomai galite paskaityti čia:

GbmbOrg (2019). Data Units Conversion. [žiūrėta 2019-05-27]. Prieiga per internetą <https://www.gbmb.org.>

*12 užduotis.* JŪSŲ INTERNETO PASLAUGŲ TIEKĖJAS PER MĖNESĮ UŽTIKRINA JUMS 8589934592 BITŲ DUOMENŲ KIEKĮ.

Kiek nuotraukų galite persiųsti kiekvieną mėnesį jei vienos nuotraukos dydis bus 5mb.

Galima persiųsti 204 nuotraukas. 8589934592 bitų tai 1024MB. 1024MB / 5MB = 204,8 nuotraukos.

#### *13 užduotis.* JŪSŲ INTERNETO GREITAVEIKA 100MBPS.

#### Kiek laiko užtruks 100mb failo siuntimas. Priimame, kad interneto greitaveika nesikeičia ir yra pastovi.

100Mbps tai yra 100 megabitų per sekundę arba 12,5 megabaitų (MB) per sekundę. 100MB / 12,5MB/s = 8 sekundės.

Papildomai galite paskaityti čia:

Wikipedia (2019). Data rate units. [žiūrėta 2019-05-27]. Prieiga per internetą <https://en.wikipedia.org/wiki/Data-rate_units.>

#### *14 užduotis.* KOKIA TURI BŪTI JŪSŲ INTERNETO GREITAVEIKA, KAD PER 5 MINUTES GALIMA BŪTŲ PERDUOTI 600MB INFORMACIJOS.

5 minutės tai 300 sekundžių. 600MB / 300 sekundžių = 2MB/s arba 16Mbps.

### Mokymosi rezultatas. Taikyti algoritmų ir logikos mokslo pagrindus programuojant.

### Tema. Logikos mokslo pagrindai.

#### *1* *užduotis.* FORMALIZUOKITE PATARLĘ: „DEGTINEI Į TROBĄ ĮEINANT PROTAS IŠEINA LAUKAN“ TEIGINIŲ LOGIKOS PRIEMONĖMIS.

#### Sakinį sudaro du elementarūs teiginiai:

p – degtinė įeina į trobą,

q – protas išeina laukan.

Sakinio prasmė: “Jeigu degtinė įeina į trobą, tai protas išeina laukan”. Elementarius teiginius sieja implikacijos jungtis. Simboliškai tai užrašoma taip: **p → q**.

#### *2 užduotis*. FORMALIZUOKITE PATARLĘ: „DEGTINĖ IR ALUS GIMDO VARGUS“ TEIGINIŲ LOGIKOS PRIEMONĖMIS.

#### Sakinį sudaro du elementarūs teiginiai:

p – degtinė gimdo vargus;

q – alus gimdo vargus.

Sakinio prasmė: “Alus arba degtinė, arba abu drauge gimdo vargus”. Elementarius teiginius sieja silpnosios disjunkcijos jungtis. Simboliškai tai užrašoma taip: **p∨q**.

#### *3 užduotis.* UŽRAŠYKITE DE MORGANO TAISYKLĖS KONJUNKCIJAI SIMBOLINĘ IŠRAIŠKĄ.

#### De Morgano taisyklės konjunkcijai simbolinė išraiška yra:

**¬ ( p∧q)∼(¬ p∨¬ q)**

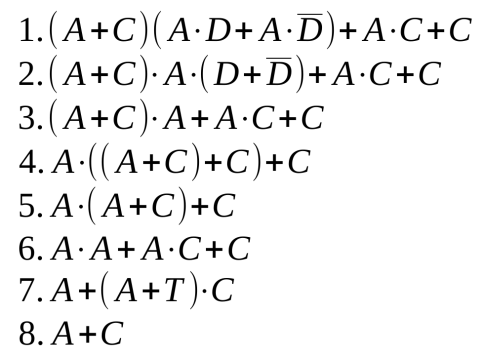
Skaitoma: “Netiesa, kad p ir q ekvivalentiška tam, kad ne-p arba ne-q”. Interpretacija. Pasižymime:

p – asmuo myli žmoną,

q – asmuo myli uošvę.

Tada teiginys „Netiesa, kad asmuo myli ir žmoną, ir uošvę“ekvivalentiškas teiginiui „Asmuo nemyli arba žmonos, arba uošvės, arba abiejų“

*4 užduotis*. UŽRAŠYKITE DE MORGANO TAISYKLĖS KONJUNKCIJAI SIMBOLINĘ IŠRAIŠKĄ.



Papildomai galite paskaityti čia:

Wikipedia (2019). Boolean Algebra. [žiūrėta 2019-05-27]. Prieiga per internetą <https://en.wikipedia.org/wiki/Boolean_algebra.>

### Tema. Algoritmai ir algoritmavimas.

#### *5 užduotis.* PARAŠYKITE PROGRAMĄ BET KOKIA JUMS ŽINOMA PROGRAMAVIMO KALBA IR/ARBA PAVAIZDUOKITE GRAFIŠKAI ALGORITMĄ, KURIS APSKAIČIUOJA SKAIČIŲ MASYVO SUMĄ IR VIDURKĮ.

#### Pvz. Algoritmas, sukurtas Java programavimo kalba atrodo taip:

package javaalgorithms;

public class IterativeAlgorithm {

public static void main(String[] args) {

int[] intArray = {54, 17, 1, 21, 85, 3, 18, 42};

int result = 0;

for(int i = 0; i < intArray.length; i++){

result += intArray[i];

}

System.out.println("Sum of the array elements: " + result);

System.out.println("Average of the array elements: " + (double) result / intArray.length);

}

}

#### *6 užduotis.* PARAŠYKITE PROGRAMĄ BET KOKIA JUMS ŽINOMA PROGRAMAVIMO KALBA IR/ARBA PAVAIZDUOKITE ALGORITMĄ, KURIS SURANDA MAŽIAUSIĄ MASYVO SKAIČIŲ.

#### **Pavyzdys**. Algoritmas Java programavimo kalba:

package javaalgorithms;

public class RecursiveAlgorithm {

public static void main(String[] args) {

int[] intArray = {27, 32, 19, -7, 41, 38, 99, -36, 54, 17, 1, 21, 85, 3, -18, 42};

int arrayLength = intArray.length;

System.out.println(findMinNumber(intArray, arrayLength));

}

private static int findMinNumber(int[] intArray, int arrayLength) {

if (arrayLength == 1) {

return intArray[0];

} else {

return Math.min(intArray[arrayLength - 1],

}

findMinNumber(intArray, arrayLength - 1));

}

}

#### *7 užduotis*. PARAŠYKITE PROGRAMĄ BET KOKIA JUMS ŽINOMA PROGRAMAVIMO KALBA IR/ARBA PAVAIZDUOKITE ALGORITMĄ, KURIS SKAIČIUOJA 120 FIBONAČIO SKAIČIŲ.

#### **Pavyzdys**. Algoritmas Java programavimo kalba:

package javaalgorithms;

import java.math.BigInteger;

public class JavaDynamicProgrammingFibonacci {

public static void main(String[] args) {

int n = 120;

System.out.println(fibonacciNumberCounting(n));

}

private static BigInteger fibonacciNumberCounting(int n) {

BigInteger a = BigInteger.valueOf(0);

BigInteger b = BigInteger.valueOf(1);

BigInteger c = BigInteger.valueOf(1);

for (int j = 2; j <= n; j++) {

c = a.add(b);

a = b;

b = c;

}

return c;

}

}

#### *8 užduotis*. PARAŠYKITE PROGRAMĄ BET KOKIA JUMS ŽINOMA PROGRAMAVIMO KALBA IR/ARBA PAVAIZDUOKITE ALGORITMĄ, KURIS NUSTATO AR RINKINYJE (SET) YRA ELEMENTŲ RINKINYS (SUBSET) LYGUS NURODYTAI SUMAI.

#### **Pavyzdys**. Algoritmas Java programavimo kalba:

package javaalgorithms;

public class JavaDynamicProgrammingSubsetSum {

public static void main(String[] args) {

int[] set = {3, 34, 4, 12, 5, 2};

int sum = 170;

int setLength = set.length;

if (isSubsetSum(set, setLength, sum)) {

System.out.println("Found a subset with given sum");

} else {

System.out.println("No subset with given sum");

}

}

private static boolean isSubsetSum(int[] set, int n, int sum) {

boolean[][] subset = new boolean[sum + 1][n + 1];

for (int i = 0; i <= n; i++){

subset[0][i] = true;

}

for (int i = 1; i <= sum; i++) {

subset[i][0] = false;

}

for (int i = 1; i <= sum; i++) {

for (int j = 1; j <= n; j++) {

subset[i][j] = subset[i][j - 1];

if (i >= set[j - 1]) {

subset[i][j] = subset[i][j] || subset[i - set[j - 1]][j - 1];

}

}

}

return subset[sum][n];

}

}

#### *9 užduotis*. PARAŠYKITE PROGRAMĄ BET KOKIA JUMS ŽINOMA PROGRAMAVIMO KALBA IR/ARBA PAVAIZDUOKITE ALGORITMĄ, KURIS SURŪŠIUOJA SVEIKŲJŲ SKAIČIŲ MASYVĄ.

**Pavyzdys**. Algoritmas Java programavimo kalba:

package javaalgorithms;

import java.util.Arrays;

public class JavaQuickSort {

private int[] array;

private int length;

public static void main(String[] args) {

JavaQuickSort sorter = new JavaQuickSort();

int[] arrayOne = {24, -2, 45, 20, 56, -75, 2, 56, 99, 53, -12};

System.out.println("Unsorted array: \n" + Arrays.toString(arrayOne));

System.out.println("Sorted array: ");

sorter.sort(arrayOne);

for (int i : arrayOne) {

System.out.print(i);

System.out.print(", ");

}

}

private void sort(int[] inputArr) {

if (inputArr == null || inputArr.length == 0) {

return;

}

this.array = inputArr;

length = inputArr.length;

quickSort(0, length - 1);

}

private void quickSort(int lowerIndex, int higherIndex) {

int i = lowerIndex;

int j = higherIndex;

int pivot = array[lowerIndex + (higherIndex - lowerIndex) / 2];

while (i <= j) {

while (array[i] < pivot) {

i++;

}

while (array[j] > pivot) {

j--;

}

if (i <= j) {

exchangeNumbers(i, j);

i++;

j--;

}

}

if (lowerIndex < j) {

quickSort(lowerIndex, j);

}

if (i < higherIndex) {

quickSort(i, higherIndex);

}

}

private void exchangeNumbers(int i, int j) {

int temp = array[i];

array[i] = array[j];

array[j] = temp;

}

}

#### *10 užduotis.* PARAŠYKITE PROGRAMĄ BET KOKIA JUMS ŽINOMA PROGRAMAVIMO KALBA IR/ARBA PAVAIZDUOKITE ALGORITMĄ, KURIS SURŪŠIUOJA SVEIKŲJŲ SKAIČIŲ MASYVĄ.

#### **Pavyzdys**. Algoritmas Java programavimo kalba:

package javaalgorithms;

public class JavaBubbleSort {

public static void main(String[] args) {

int[] inputNumber = {4, 2, 9, 6, 23, 12, 34, 0, 1};

bubble\_srt(inputNumber);

}

private static void bubble\_srt(int[] array) {

int n = array.length;

int k;

for (int m = n; m >= 0; m--) {

for (int i = 0; i < n - 1; i++) {

k = i + 1;

if (array[i] > array[k]) {

swapNumbers(i, k, array);

}

}

printNumbers(array);

}

}

private static void swapNumbers(int i, int j, int[] array) {

int temp;

temp = array[i];

array[i] = array[j];

array[j] = temp;

}

private static void printNumbers(int[] input) {

for (int inp : input) {

System.out.print(inp + " ");

}

System.out.println("\n");

}

}

#### *11 užduotis*. PARAŠYKITE PROGRAMĄ BET KOKIA JUMS ŽINOMA PROGRAMAVIMO KALBA IR/ARBA PAVAIZDUOKITE ALGORITMĄ, KURIS SIMBOLIŲ EILUTĘ ATSPAUSDINA ATVIRKŠČIAI.

**Pavyzdys**. Algoritmas Java programavimo kalba

package javaalgorithms;

import java.util.Stack;

public class JavaDataStructuresStack {

public static void main(String[] args) {

String str = "Java for all";

reverseWords(str);

}

private static void reverseWords(String str) {

Stack<Character> characterStack = new Stack<>();

for (int i = 0; i < str.length(); ++i) {

if (str.charAt(i) != ' ') {

characterStack.push(str.charAt(i));

} else {

while (!characterStack.empty()) {

System.out.print(characterStack.pop());

System.out.print(" ");

}

}

while (!characterStack.empty()) {

System.out.print(characterStack.pop());

}

}

}

#### *12 užduotis*. PARAŠYKITE PROGRAMĄ BET KOKIA JUMS ŽINOMA PROGRAMAVIMO KALBA IR/ARBA PAVAIZDUOKITE ALGORITMĄ, KURIS IEŠKO MAŽIAUSIĄ SKAIČIŲ AIBĖJE.

**Pavyzdys**. Algoritmas Java programavimo kalba

package javaalgorithms;

class Node {

int data;

Node left, right;

Node(int d) {

data = d;

left = right = null;

}

}

package javaalgorithms;

public class JavaDataStructuresBinaryTree {

public static void main(String[] args) {

JavaDataStructuresBinaryTree binaryTree = new JavaDataStructuresBinaryTree();

Node root = null;

root = binaryTree.insert(root, 4);

binaryTree.insert(root, 23);

binaryTree.insert(root, -12);

binaryTree.insert(root, 3);

binaryTree.insert(root, 65);

binaryTree.insert(root, 54);

System.out.println("Minimum value of binary binaryTree is "

+ binaryTree.minvalue(root));

}

private Node insert(Node node, int data) {

if (node == null) {

return (new Node(data));

} else {

if (data <= node.data) {

node.left = insert(node.left, data);

} else {

node.right = insert(node.right, data);

}

return node;

}

}

private int minvalue(Node node) {

Node current = node;

while (current.left != null) {

current = current.left;

}

return (current.data);

}

}

### 

### Mokymosi rezultatas. Taikyti programinio kodo dizaino modelius programuojant.

### 

### Tema. Dizaino šablonai.

#### *1 užduotis*. UŽPILDYKITE PATEIKTĄ LENTELĘ.

#### Dizaino modeliai (šablonai):

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Dizaino šablono rūšys** | | |
|  | **Creational** | **Structural** | **Behavioral** |
| **Šablonas** | Abstract Factory | Adapter | Chain of Responsibility |
| Builder | Bridge | Command |
| Factory Method | Composite | Interpreter |
| Prototype | Decorator | Iterator |
| Singleton | Facade | Mediator |
|  | Flyweight | Memento |
|  | Proxy | Observer |
|  |  | State |
|  |  | Strategy |
|  |  | Template Method |
|  |  | Visitor |

Papildomai galite paskaityti čia:

Data & Object Factory (2019). .*NET Design Patterns [*žiūrėta 2019-05-27]. Prieiga per internetą <https://www.dofactory.com/net/design-patterns>

#### *2 užduotis.* APRAŠYKITE PASIRINKTINAI TRIS DIZAINO ŠABLONUS.

**Singleton šablonas (Vienintelis objektas):** užtikrina, kad būtų leidžiama naudoti tik vieną klasės egzempliorių. Naudojamas kai reikia vieno klasės egzempliorius ir būtina kontroliuoti prieigą prie vieno objekto. Pavyzdys: objektas reprezentuoja resursą: ekraną, klaviatūrą, garso posistemę ir pan.

**Decorator šablonas (Dekoratorius):** naudojamas dinamiškai keisti ir pridėti funkcionalumo objektui. Objekto atsakomybė ir elgesys turėtų būti keičiami dinamiškai. Konkretūs įgyvendinimai turėtų būti atsieti nuo atsakomybės ir elgesio. Pavyzdys: daugelis įmonių savo pašto sistemas sukūrė naudodamosi dekoratoriais. Kai pranešimai siunčiami iš įmonės į išorinį adresą, pašto serveris papuošia originalų pranešimą su autorių teisių ir konfidencialumo informacija. Kol pranešimas lieka vidinis, informacija nėra pridedama.

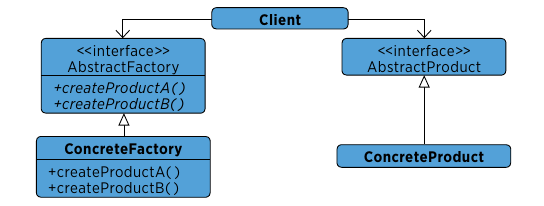
**Strategy šablonas (Strategija):** naudojamas, kuomet bendrą algoritmą reikia varijuoti tam tikru aspektu. Pavyzdžiui apskaitos programoje strategija galėtų būti - mokesčių skaičiavimas skirtingoms šalims. Tai leistų nesunkiai kurti programos versijas skirtingoms valstybėms.

Papildomai galite paskaityti čia:

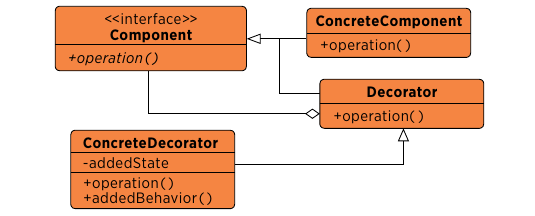
Wikipedia (2019). *Software design pattern*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://en.wikipedia.org/wiki/Software_design_pattern.>

### Tema. Dizaino šablonų taikymas.

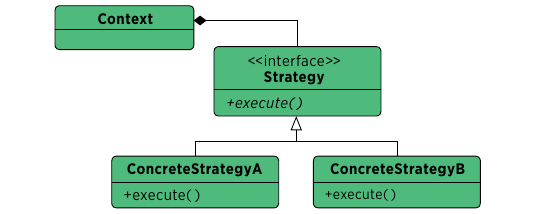
#### *3 užduotis.* NAUDOJANT UML MODELIAVIMO KALBĄ SUKURKITE KLASIŲ DIAGRAMĄ „ABSTRAKČIOS GAMYKLOS“ ŠABLONUI (ABSTRACT FACTORY PATTERN).



#### *4 užduotis.* NAUDOJANT UML MODELIAVIMO KALBĄ SUKURKITE KLASIŲ DIAGRAMĄ „DEKORATORIAUS“ ŠABLONUI (DECORATOR PATTERN).



#### *5 užduotis*. NAUDOJANT UML MODELIAVIMO KALBĄ SUKURKITE KLASIŲ DIAGRAMĄ „STRATEGIJOS“ ŠABLONUI (STRATEGY PATTERN).



Braižant UML klasių diagramas galite naudoti:

JGraph Ltd (2019). draw.io. [žiūrėta 2019-05-27]. Prieiga per internetą <https://www.draw.io.>

#### *6 užduotis.* PARINKITE DIZAINO ŠABLONĄ PATEIKTAM GYVENIMO PAVYZDŽIUI: GAMYKLA GAMINA TRIJŲ MODELIŲ AUTOMOBILIUS.

#### Geriausiai šioje situacijoje naudoti fabrikos šabloną.

Papildomai galite paskaityti:

HowToDoInJava (2019). Java Factory Pattern Explained. [žiūrėta 2019-05-27].Prieiga perinternetą<https://howtodoinjava.com/design-patterns/creational/implementing-factory-design-pattern-in-java.>

#### *7 užduotis.* PARINKITE DIZAINO ŠABLONĄ PATEIKTAM GYVENIMO PAVYZDŽIUI: TURIME PAŠTO SERVERĮ.

#### Geriausiai šioje situacijoje naudoti dekoratoriaus šabloną.

Papildomai galite paskaityti:

HowToDoInJava (2019). *Decorator Design Pattern in Java*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://howtodoinjava.com/design-patterns/structural/decorator-design-pattern.>

#### *8 užduotis.* PARAŠYKITE .NET PROGRAMĄ, KURI REALIZUOTŲ „ADAPTER“ ŠABLONĄ.

#### Pvz. programa, realizuojanti šį šabloną Java programavimo kalba atrodo taip:

package adapterpattern;

public interface Bird {

public void fly();

public void makeSound();

}

package adapterpattern;

public class Sparrow implements Bird {

public void fly() {

System.out.println("Flying");

}

public void makeSound() {

System.out.println("Chirp Chirp");

}

}

package adapterpattern;

public interface ToyDuck {

public void squeak();

}

package adapterpattern;

public class PlasticToyDuck implements ToyDuck {

public void squeak() {

System.out.println("Squeak");

}

}

package adapterpattern;

public class BirdAdapter implements ToyDuck {

Bird bird;

public BirdAdapter(Bird bird) {

this.bird = bird;

}

public void squeak() {

bird.makeSound();

}

}

package adapterpattern;

public class Main {

public static void main(String[] args) {

Sparrow sparrow = new Sparrow();

ToyDuck toyDuck = new PlasticToyDuck();

ToyDuck birdAdapter = new BirdAdapter(sparrow);

System.out.println("Sparrow...");

sparrow.fly();

sparrow.makeSound();

System.out.println("ToyDuck...");

toyDuck.squeak();

System.out.println("BirdAdapter...");

birdAdapter.squeak();

}

}

#### 

#### *9 užduotis.* PARAŠYKITE .NET PROGRAMĄ, KURI REALIZUOTŲ „TEMPLATE METHOD“ ŠABLONĄ.

#### Pvz. programa, realizuojanti šį šabloną Java programavimo kalba atrodo taip:

package templatemethodpattern;

abstract class OrderProcessTemplate {

public boolean isGift;

public abstract void doSelect();

public abstract void doPayment();

public final void giftWrap() {

try {

System.out.println("Gift wrap successfull");

} catch (Exception e) {

System.out.println("Gift wrap unsuccessful");

}

}

public abstract void doDelivery();

public final void processOrder(boolean isGift) {

doSelect();

doPayment();

if (isGift) {

giftWrap();

}

doDelivery();

}

}

package templatemethodpattern;

public class NetOrder extends OrderProcessTemplate {

@Override

public void doSelect() {

System.out.println("Item added to online shopping cart");

System.out.println("Get gift wrap preference");

System.out.println("Get delivery address.");

}

@Override

public void doPayment() {

System.out.println

("Online Payment through Netbanking, card or Paytm");

}

@Override

public void doDelivery() {

System.out.println

("Ship the item through post to delivery address");

}

}

package templatemethodpattern;

public class StoreOrder extends OrderProcessTemplate {

@Override

public void doSelect() {

System.out.println("Customer chooses the item from shelf.");

}

@Override

public void doPayment() {

System.out.println("Pays at counter through cash/POS");

}

@Override

public void doDelivery() {

System.out.println("Item deliverd to in delivery counter.");

}

}

package templatemethodpattern;

public class TemplateMethodPatternClient {

public static void main(String[] args) {

OrderProcessTemplate netOrder = new NetOrder();

netOrder.processOrder(true);

System.out.println();

OrderProcessTemplate storeOrder = new StoreOrder();

storeOrder.processOrder(true);

}

}

### 

### Mokymosi rezultatas. Naudoti informacinių sistemų kūrimo principus ir metodus programinės įrangos projektavime.

### Tema. Daugiasluoksnė programų architektūra ir MVC struktūra.

#### *1 užduotis*. APRAŠYKITE KLIENTO-SERVERIO (CLIENT SERVER) PROGRAMINĖS ĮRANGOS MODELĮ.

#### Aprašant programinės įrangos modelius naudokitės šia medžiaga:

Maskeliūnas, S. (2007). Programų sistemų architektūra ir projektavimas. Mokymo medžiaga parengta vykdant projektą “Programų sistemų magistrantūros įsteigimas”. [žiūrėta 2019-05-27]. Prieiga per internetą <https://klevas.mif.vu.lt/~donatas/PSArchitekturaProjektavimas/Knyga/BPD/PSAPKnyga.pdf>

#### *2 užduotis*. APRAŠYKITE MODELIO-VAIZDO-KONTROLERIO (MODEL VIEW CONTROLLER, MVC) PROGRAMINĖS ĮRANGOS MODELĮ.

#### Aprašant programinės įrangos modelius naudokitės šia medžiaga:

Maskeliūnas, S. (2007). Programų sistemų architektūra ir projektavimas. Mokymo medžiaga parengta vykdant projektą “Programų sistemų magistrantūros įsteigimas”. [žiūrėta 2019-05-27]. Prieiga per internetą <https://klevas.mif.vu.lt/~donatas/PSArchitekturaProjektavimas/Knyga/BPD/PSAPKnyga.pdf>

#### *3 užduotis*. NAUDOJANT MVC ARCHITEKTŪROS MODELĮ, SUKURKITE PAPRASTĄ JAVA DARBUOTOJŲ VALDYMO PROGRAMĄ.

#### Užduoties sprendimui galite panaudoti pateiktą kodą:

package javamvc;

class EmployeeModel {

private String id;

private String name;

public String getId() {

return id;

}

public void setId(String id) {

this.id = id;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

}

package javamvc;

public class EmployeeController {

private EmployeeModel model;

private EmployeeView view;

public EmployeeController(EmployeeModel model, EmployeeView view) {

this.model = model;

this.view = view;

}

public void setEmployeeName(String name) {

model.setName(name);

}

public String getEmployeeName() {

return model.getName();

}

public void setEmployeeId(String rollNo) {

model.setId(rollNo);

}

public String getEmployeeId() {

return model.getId();

}

public void updateView() {

view.printEmployeeDetails(model.getName(), model.getId());

}

}

package javamvc;

class EmployeeView {

public void printEmployeeDetails(String name, String no) {

System.out.println("EmployeeModel: ");

System.out.println("Name: " + name);

System.out.println("ID: " + no);

}

}

package javamvc;

public class Main {

public static void main(String[] args) {

EmployeeModel employee = new EmployeeModel();

EmployeeView view = new EmployeeView();

EmployeeController controller = new EmployeeController(employee, view);

employee.setId("1");

employee.setName("Tom");

controller.updateView();

controller.setEmployeeName("Dave");

controller.updateView();

}

}

Papildomai galite paskaityti:

Wikipedia (2019). Model View Controller. [žiūrėta 2019-05-27]. Prieiga per internetą <https://en.wikipedia.org/wiki/Model-view-controller.>

#### *4 užduotis*. NAUDOJANT SPRING KARKASĄ SUKURKITE PAPRASTĄ WEB VARTOTOJŲ REGISTRACIJOS PROGRAMĄ.

#### Užduoties sprendimui galite naudoti šį šaltinį:

Java Guides (2019). *Spring MVC Tutorials and Articles*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://www.javaguides.net/2018/10/spring-mvc-sign-up-form-handling.html.>

### Tema. Pagrindiniai programavimo principai.

#### *5 užduotis*. APRAŠYKITE SOLID IR DRY PROGRAMAVIMO PRINCIPUS.

#### Užduoties sprendimui naudokite šiuos resursus:

1. Wikipedia (2019). SOLID. [žiūrėta 2019-05-27]. Prieiga per internetą <https://en.wikipedia.org/wiki/SOLID.>

2. Wikipedia (2019). Don’t repeat yourself. [žiūrėta 2019-05-27]. Prieiga per internetą <https://en.wikipedia.org/wiki/Don’t_repeat_yourself.>

#### *6 užduotis*. APRAŠYKITE SEPARATION OF CONCERNS IR CODE REUSE PROGRAMAVIMO PRINCIPUS.

#### Užduoties sprendimui naudokite šiuos resursus:

1. Wikipedia (2019). Separation of Concerns. [žiūrėta 2019-05-27]. Prieiga per internetą <https://en.wikipedia.org/wiki/Separation_of_concerns.>

2. Wikipedia (2019). Code reuse. [žiūrėta 2019-05-27]. Prieiga per internetą <https://en.wikipedia.org/wiki/Code_reuse.>

#### *7 užduotis*. NAUDOJANT FUNKCINIO PROGRAMAVIMO PRINCIPUS REALIZUOKITE GREITO RŪŠIAVIMO ALGORITMĄ.

#### 

package javafunctional;

import java.util.ArrayList;

import java.util.Arrays;

import java.util.List;

import java.util.function.Function;

import java.util.function.Predicate;

import java.util.stream.Collectors;

import java.util.stream.Stream;

public class JavaQuickSortFunctional {

private static Function<Integer, Predicate<Integer>> smallerThan = x -> y -> y < x;

public static void main(String[] args) {

List<Integer> integerList = Arrays.asList(5, 6, 7, 23, 4, 5645, 6, 1223,

44453, 60182, 2836, 23993, 1);

System.out.println(quickSort(integerList));

}

private static List<Integer> quickSort(List<Integer> integerList) {

if (integerList.isEmpty()) return new ArrayList<>();

return Stream.concat(Stream.concat(quickSort(integerList.stream().skip(1)

.filter(smallerThan.apply(integerList.get(0)))

.collect(Collectors.toList())).stream(),

Stream.of(integerList.get(0))), quickSort(integerList.stream().skip(1)

.filter(smallerThan.apply(integerList.get(0)).negate())

.collect(Collectors.toList())).stream()).collect(Collectors.toList());

}

}

Papildomai apie funkcinį programavimą galite paskaityti čia:

Oracle (2019). Lambda Expressions. [žiūrėta 2019-05-27]. Prieiga per internetą <https://docs.oracle.com/javase/tutorial/java/javaOO/lambdaexpressions.html.>

#### *8 užduotis*. NAUDOJANT FUNKCINIO PROGRAMAVIMO PRINCIPUS REALIZUOKITE SĄRAŠO GRUPAVIMO, ELEMENTŲ SKAIČIAVIMO IR RŪŠIAVIMO ALGORITMĄ.

#### 

package javafunctional;

import java.util.Arrays;

import java.util.LinkedHashMap;

import java.util.List;

import java.util.Map;

import java.util.function.Function;

import java.util.stream.Collectors;

public class JavaGroupCountSortFunctional {

public static void main(String[] args) {

List<String> items = Arrays.asList("Tom", "Dave", "John", "Mary",

"Steve", "Tom", "Mary");

Map<String, Long> result = items.stream()

.collect(Collectors.groupingBy(Function

.identity(), Collectors.counting()));

Map<String, Long> finalMap = new LinkedHashMap<>();

result.entrySet().stream().sorted(Map.Entry.<String, Long>comparingByKey())

.forEachOrdered(e -> finalMap.put(e.getKey(), e.getValue()));

System.out.println(finalMap);

}

}

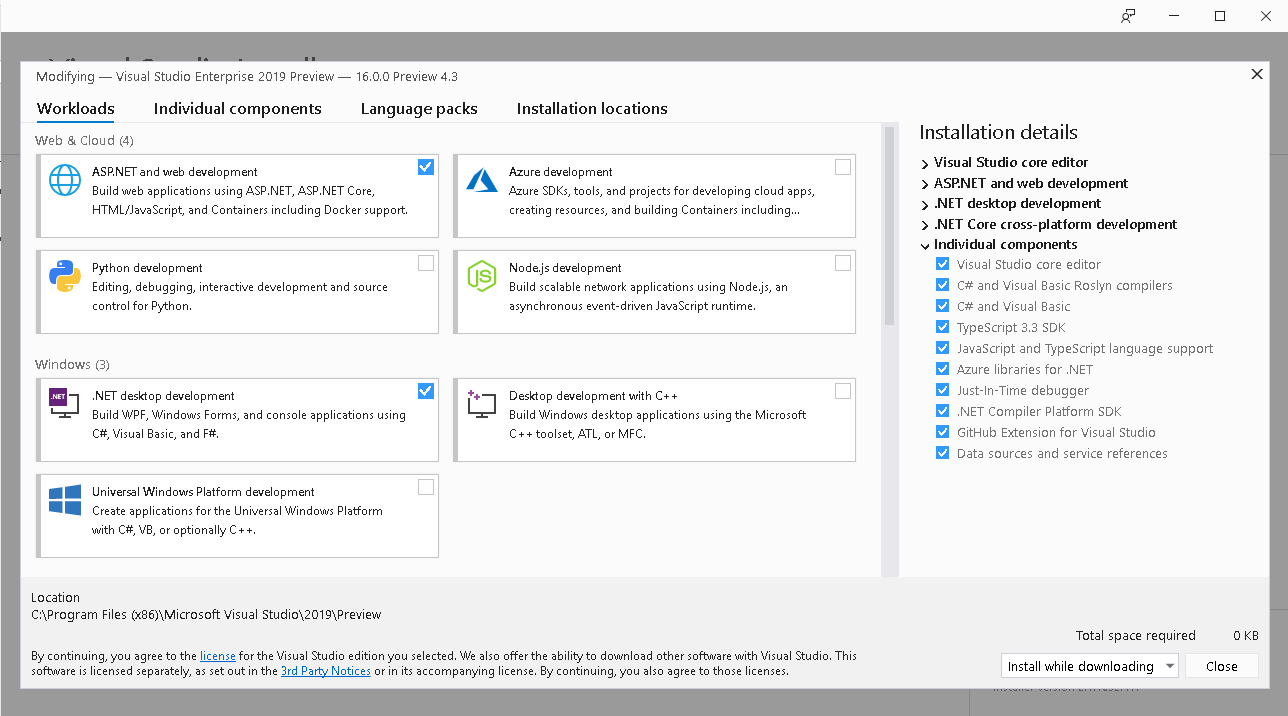
## Kompetencija. Kurti tipinę programinę įrangą.

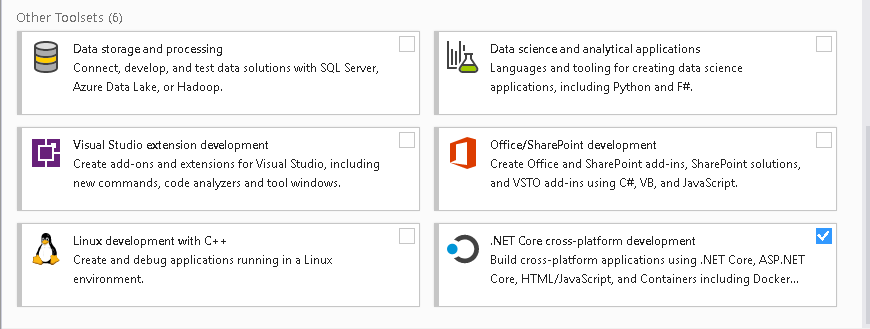
### Mokymosi rezultatas. Naudoti C# programavimo kalbos įrankius ir sintaksę.

### Tema. Visual Studio aplinka.

*1 užduotis.* ĮDIEGTI VISUAL STUDIO INTEGRUOTĄ PROGRAMAVIMO APLINKĄ SU “.NET CORE CROSS-PLATFORM DEVELOPMENT” IR “ASP.NET AND WEB DEVELOPMENT” MODULIAIS.

Diegimo metu pasirinkti moduliai:





*2 užduotis.*  PATIKRINKITE AR TEISINGAI ĮDIEGTAS PROGRAMAVIMO SU .NET CORE KARKASU PRIEMONĖS, GAUNANT INFORMACIJĄ APIE KARKASĄ KOMANDINĖJE EILUTĖJE.

Naudoti programą komandinėje eilutėje dotnet –info

*3 užduotis.* NAUDOJANT VISUAL STUDIO IDE SUKURKITE .NET CORE PROGRAMĄ KONSOLEI, KURI IŠVEDA PASISVEIKINIMĄ SU PROGRAMOS AUTORIUMI IR JO VARDĄ.

Programinis kodas:

|  |
| --- |
| using System;  namespace ConsoleApp  {  class Program  {  static void Main(string[] args)  {  Console.WriteLine("Sveiki, Autoriau!");  }  }  } |

*4 užduotis.*  PROGRAMĄ IŠ *3 užduoties* SUKOMPILIUOKITE IR PALEISKITE NAUDOJANT KOMANDINĘ EILUTĘ.

Programai sukompiliuoti naudoti ir paleisti naudoti programas komandinėje eilutėje:

dotnet run – programos kompiliavimas ir paleidimas

dotnet build – programos kompiliavimas

dotnet <MyApp>.dll – sukompiliuotos programos paleidimas, kur MyApp yra sukompiliuoto projekto pavadinimas

### Tema. C# programavimo pagrindai ir sintaksė.

*5 užduotis #5.*  NAUDOJANT VISUAL STUDIO IDE SUKURKITE.NET CORE PROGRAMĄ KONSOLEI, KURI TURI TURĖTI MAIN METODĄ IR METODĄ PAVADINIMU PRINTTEXT, KURIS ATSPAUSDINA TEKSTĄ, PERDUODAMĄ PARAMETRU.

Programinis kodas:

|  |
| --- |
| using System;  namespace U1  {  class Program  {  static void Main()  {  PrintText("Metodas kvieciamas");  }  private static void PrintText(string text)  {  Console.WriteLine(text);  }  } |

*6 užduotis.* PAPILDYKITE PROGRAMĄ *„5 užduotis #5“* VIENO IR KELIŲ EILUČIŲ TIPŲ KOMENTARAIS.

Programinis kodas:

|  |
| --- |
| using System;  namespace U1  {  class Program  {  /\* Paskirtis: programa spausdina tekstą  \* Sukūrimo metai: 2019  \*/  static void Main()  {  PrintText("Metodas kvieciamas"); // teksto spausdinimas  }  private static void PrintText(string text)  {  Console.WriteLine(text);  }  }  } |

*7 užduotis.* SUKURKITE PROGRAMĄ, KURIOJE YRA SĄLYGOS SAKINYS, KURIS ATSPAUSDINTŲ PRANEŠIMĄ JEIGU KINTAMOJO REIKŠMĖ DIDESNĖ UŽ 100, JEIGU NE – TADA ATSPAUSDINTŲ KITĄ PRANEŠIMĄ.

Programinis kodas:

|  |
| --- |
| if (num > 100)  {  Console.WriteLine("Daugiau nei 100");  }  else  {  Console.WriteLine("Mažiau už 100");  } |

*8 užduotis.* SUKURKITE PROGRAMĄ, KURIOJE YRA SĄLYGOS SAKINYS, KURIS PATIKRINTŲ IR ATSPAUSDINTŲ PRANEŠIMĄ „SKAIČIUS YRA 0...100 TARPE“ JEIGU KINTAMOJO REIKŠMĖ YRA ATITINKAMAME RĖŽYJE.

Programinis kodas:

|  |
| --- |
| if (num < 100 && num > 0)  {  Console.WriteLine("Skaičius yra 0...100 rėžyje");  }  else  {  Console.WriteLine("Skaičiaus rėžyje nėra");  } |

*9 užduotis.* SUKURTI DU INT TIPO KINTAMUOSIUS (PVZ. A IR B) IR PRISKIRTI JIEMS REIKŠMES.

Programinis kodas:

|  |
| --- |
| using System;  namespace ConsoleApp  {  class Program  {  static void Main(string[] args)  {  var a = 13;  var b = 2;  // aritmetine dalis  var sum = a + b;  var multi = a \* b;  var subtract = a - b;  var division = a / b;  var mod = a % b;  Console.WriteLine($"A:{a} ir B:{b} suma={sum}");  Console.WriteLine($"A:{a} ir B:{b} sandauga={multi}");  Console.WriteLine($"A:{a} ir B:{b} skirtumas={subtract}");  Console.WriteLine($"A:{a} ir B:{b} dalmuo={division}");  Console.WriteLine($"A:{a} ir B:{b} liekana={mod}");  // loginiai operatoriai  var areEqual = a == b;  var aIsGreater = a > b;  var bIsGreater = a < b;  var areDifferent = a != b;  Console.WriteLine($"A:{a} ir B:{b} yra lygus {areEqual}");  Console.WriteLine($"A:{a} yra didesnis už B:{b} {aIsGreater}");  Console.WriteLine($"A:{a} yra mažesnis už B:{b} {bIsGreater}");  Console.WriteLine($"A:{a} ir B:{b} yra nelygus {areDifferent}");  // inkremento, dekremento dalis  a++;  Console.WriteLine(a);  b++;  Console.WriteLine(b);  //jungtiniai priskyrimo  a += 5;  Console.WriteLine(a);  b -= 20;  Console.WriteLine(b);  a \*= 3;  Console.WriteLine(a);  b /= 3;  Console.WriteLine(b);  }  }  } |

*10 užduotis.* SUKURKITE PROGRAMĄ, KURI PAPRAŠO ĮVESTI VARDĄ IR PATEIKIA IŠVESTĮ „LABAS, [VARDAS]“, PAGAL ĮVESTĄ VARDĄ.

Programinis kodas:

|  |
| --- |
| using System;  namespace ConsoleApp  {  class Program  {  static void Main(string[] args)  {  Console.WriteLine("Įveskite savo vardą.");  var input = Console.ReadLine();  Console.WriteLine($"Labas, {input}");  }  }  } |

*11 užduotis.* SUKURKITE PROGRAMĄ SU METODU, KURIS PRIIMA SKAIČIŲ IR PANAUDOJA BOOL IŠRAIŠKĄ KODE, KURI NURODO AR SKAIČIUS YRA TEIGIAMAS.

Programinis kodas:

|  |
| --- |
| bool IsPositive(int num)  {  return num > 0;  } |

*12 užduotis.* SUKURKITE PROGRAMĄ SU METODU, KURIS GRĄŽINA BOOL DUOMENŲ TIPĄ, PAVADINIMU NUMBERISNEGATIVE.

Programinis kodas:

|  |
| --- |
| bool NumberIsNegative(int num)  {  if (num > 0)  {  return true;  }  else  {  return false;  }  } |

*13 užduotis.* SUKURKITE PROGRAMĄ SU METODU, KURIS PATIKRINA AR PADUOTAS SKAIČIUS YRA TRIŽENKLIS.

Programinis kodas:

|  |
| --- |
| bool NumberIs3Digit(int num)  {  if (num < 1000 && num > 100)  {  return true;  }  else if (num > -1000 && num < -100)  {  return true;  }  else  {  return false;  }  } |

*14 užduotis.* SUKURKITE PROGRAMĄ SU METODU, KURIS PRIIMA TRIS INT TIPO PARAMETRUS (SK – SKAIČIUS PATIKRINIMUI, r1, r2 – SKAIČIAI NUSAKANTYS RĖŽIUS) IR GRĄŽINA BOOL DUOMENŲ TIPĄ.

Programinis kodas:

|  |
| --- |
| static bool IsInRange1(int sk, int r1, int r2)  {  if (r1 <= 0 || r2 <= 0)  {  if (r1 < r2)  {  return false;  }  if (sk <= r1 && sk >= r2)  {  return true;  }  else  {  return false;  }  }  else  {  if (r1 > r2)  {  return false;  }  if (sk >= r1 && sk <= r2)  {  return true;  }  else  {  return false;  }  }  } |

*15 užduotis.* SUKURKITE PROGRAMĄ SU METODU, KURIS NAUDOJANT CIKLĄ ATSPAUSDINA Į KONSOLĘ NELYGINIUS SKAIČIUS NUO 1 IKI 10.

Programinis kodas:

|  |
| --- |
| for (int i = 1; i < 10; i += 2)  {  Console.WriteLine(i);  } |

*16 užduotis.* PARAŠYKITE FAKTORIALO METODĄ NAUDOJANT REKURSIJĄ.

Programinis kodas:

|  |
| --- |
| int Factorial(int number)  {  if (number == 1)  {  return 1;  }  else  {  return number \* Factorial(number - 1);  }  } |

*17 užduotis.* PARAŠYKITE FIBINAČIO SEKOS METODĄ NAUDOJANT REKURSIJĄ *.*

Programinis kodas:

|  |
| --- |
| int Fibonacci(int number)  {  if (number == 0)  {  return 0;  }  else if (number == 1)  {  return 1;  }  else  {  return Fibonacci(number - 2) + Fibonacci(number - 1);  }  } |

*18 užduotis*. SUKURKITE PROGRAMĄ, KURIOJE SUKURIAMI ŠIO TIPO KINTAMIEJI IR ATSPAUSDINAMI KONSOLĖS LANGE: BOOL, BYTE , CHAR, INT, FLOAT, DOUBLE, LONG, STRING.

Programinis kodas:

|  |
| --- |
| using System;  namespace ConsoleApp  {  class Program  {  static void Main(string[] args)  {  bool @bool = true;  byte @byte = 1;  char @char = 'g';  int @int = 12222;  float @float = 23.5f;  double @double = 13.44d;  long @long = 1111111111111111L;  string @string = "tekstas123";  var @bool1 = true;  var @byte1 = 1;  var @char1 = 'g';  var @int1 = 12222;  var @float1 = 23.5f;  var @double1 = 13.44d;  var @long1 = 1111111111111111L;  var @string1 = "tekstas123";  }  }  } |

*19 užduotis.* MASYVO, KURIS TURI REIKŠMES {1,66,2,8,33,5,2}:

Programinis kodas:

|  |
| --- |
| var arr = new[] { 1, 66, 2, 8, 33, 5, 2 };  //1a  foreach (var item in arr)  {  Console.WriteLine(item);  }  //1b  var sum = 0;  foreach (var item in arr)  {  sum += item;  }  Console.WriteLine($"Masyvo suma yra {sum}");  //1c  var avg = (double)sum / arr.Length;  Console.WriteLine($"Masyvo vidurkis yra {avg}");  //1d  var min = arr[0];  foreach (var item in arr)  {  if (item < min)  {  min = item;  }  }  Console.WriteLine($"Masyvo mažiausias narys yra {min}");  //1e  var max = arr[0];  foreach (var item in arr)  {  if (item > min)  {  min = item;  }  }  Console.WriteLine($"Masyvo didžiausias narys yra {min}"); |

*20 užduotis.* SUKURKITE PROGRAMĄ, KURI IŠ PRADŽIŲ PAPRAŠYTŲ ĮVESTI TIEK SKAIČIŲ KIEK REIKĖS ĮVESTI (Į KOKIO DYDŽIO MASYVĄ) IR TADA IŠ KOMANDINĖS EILUTĖS NUSKAITYTŲ VISUS SKAIČIUS.

Programinis kodas:

|  |
| --- |
| Console.WriteLine("Įveskite narių skaičių");  var totalItemsCount = int.Parse(Console.ReadLine());  var userInputArray1 = new int[totalItemsCount];  for (int i = 0; i < userInputArray1.Length; i++)  {  Console.WriteLine($"Įveskite {i + 1} skaičių");  userInputArray1[i] = int.Parse(Console.ReadLine());  }  foreach (var item in userInputArray1)  {  Console.WriteLine(item);  }  var sum1 = 0;  foreach (var item in userInputArray1)  {  sum1 += item;  }  Console.WriteLine($"Masyvo suma yra {sum}");  var avg1 = (double)sum1 / userInputArray1.Length;  Console.WriteLine($"Masyvo vidurkis yra {avg1}");  var mult = 1;  foreach (var item in userInputArray1)  {  mult \*= item;  }  Console.WriteLine($"Masyvo sandauga yra {mult}"); |

*21 užduotis.* SUKURKITE PAIEŠKOS MASYVE METODĄ, KURIS PRIIMTŲ SVEIKŲ SKAIČIŲ MASYVĄ IR SVEIKĄ SKAIČIŲ KAIP PARAMETRUS.

Programinis kodas:

|  |
| --- |
| int FindIndex(int num, int[] arr)  {  for (int i = 0; i < arr.Length; i++)  {  if (arr[i] == num)  {  return i;  }  }  return -1;  } |

*22 užduotis.*  SUKURKITE PROGRAMĄ SU METODU , KURIS PRIIMA DU SKAIČIUS IR GRĄŽINA JŲ SUMĄ.

Perkrauti metodą, kuris priimtų tris skaičius ir paskaičiuotų jų sumą.

Programinis kodas:

|  |
| --- |
| int Sum(int a, int b)  {  return a + b;  }  int Sum(int a, int b, int c)  {  return a + b + c;  } |

### Mokymosi rezultatas. Kurti nesudėtingą programinį kodą C# programavimo kalba.

### Tema. Darbas su duomenimis.

*1 užduotis #1.* SUKURKITE PROGRAMĄ, KURI TURI KLASĘ, APRAŠANČIĄ DARBUOTOJĄ, KURIO SAVYBĖS YRA VARDAS, GIMIMO DATA, PAREIGOS IR AMŽIUS (TIK SKAITOMA SAVYBĖ, GAUNAMA IŠ DABARTINĖS DATOS ATĖMUS GIMIMO DATĄ).

Programinis kodas:

|  |
| --- |
| using System;  using System.Linq;  class Linq  {  static void Main(string[] args)  {  var e = new Employee() { BirthDate = new DateTime(1999, 12, 31) };  Console.WriteLine(e.Age);  }  }  class Employee  {  public string Name { get; set; }  public DateTime BirthDate { get; set; }  public string Position { get; set; }  public int Age  {  get  {  TimeSpan diff = DateTime.Now - BirthDate;  return (new DateTime(1, 1, 1) + diff).Year - 1;  }  }  } |

*2 užduotis #2*. SUKURKITE PROGRAMĄ, KURI PAPRAŠO ĮVESTI DU SKAIČIUS IR ATSPAUSDINA JŲ SUMĄ.

Panaudoti parse metodą, kuris paverstų vartotojo įvestį į skaičių.

Programinis kodas:

|  |
| --- |
| using System;  namespace ConsoleApp  {  class Program  {  static void Main(string[] args)  {  Console.WriteLine("Įveskite savo pirmą skaičių.");  var firstNumberInput = Console.ReadLine();  var firstNum = int.Parse(firstNumberInput);  Console.WriteLine("Įveskite savo antrą skaičių.");  var secondNumberInput = Console.ReadLine();  var secondNum = int.Parse(secondNumberInput);  var sum = firstNum + secondNum;  Console.WriteLine($"Suma yra {sum}");  }  }  } |

*3 užduotis #3.* SUKURKITE PROGRAMĄ, KURI PAPRAŠYTŲ VARTOTOJO ĮVESTIES IR SU TRYPARSE PATIKRINKITE AR ĮVESTIS TEISINGA, JEI NETEISINGA, PROGRAMĄ UŽBAIGTI PANAUDOJUS RETURN RAKTAŽODĮ.

Parašyti switch sakinį, kuris tikrintų įvesties sąlygas 1,2,3.

Programinis kodas:

|  |
| --- |
| Console.WriteLine("Iveskite skaičių");  var input1 = Console.ReadLine();  int result1;  if (!int.TryParse(input1, out result1))  {  Console.WriteLine("Neteisinga įvestis");  return;  }  switch (result1)  {  case 1:  Console.WriteLine("Skaičius 1");  break;  case 2:  Console.WriteLine("Skaičius 2");  break;  case 3:  Console.WriteLine("Skaičius 3");  break;  default:  Console.WriteLine("Įvestas nežinomas skaičius");  break;  } |

*4 užduotis #4.* SUKURKITE PROGRAMĄ SU MENIU, KURI ATSPAUSDINTŲ MENIU PUNKTUS SKAIČIŲ ARITMETINĖMS FUNKCIJOMS: SKAIČIŲ SUDĖTIS, DAUGYBA, KĖLIMAS KVADRATU.

Programinis kodas:

|  |
| --- |
| public void Main(string[] args)  {  Console.WriteLine("1. Skaičių sudėtis");  Console.WriteLine("2. Skaičių daugyba");  Console.WriteLine("3. Skaičiaus kvadratas");  var input = Console.ReadLine();  int result;  if (!int.TryParse(input, out result))  {  Console.WriteLine("Neteisinga įvestis");  return;  }  switch (result)  {  case 1:  NumberSum();  break;  case 2:  NumberMultiplication();  break;  case 3:  NumberPow();  break;  default:  Console.WriteLine("Tokio pasirinkimo nėra");  break;  }  }  private void NumberPow()  {  Console.WriteLine("Iveskite skaičių");  var input = Console.ReadLine();  int result;  if (!int.TryParse(input, out result))  {  Console.WriteLine("Neteisinga įvestis");  return;  }  var pow = Math.Pow(result, 2);  Console.WriteLine($"Skaičius kvadratu {pow}");  }  private void NumberMultiplication()  {  Console.WriteLine("Iveskite pirmą skaičių");  var input = Console.ReadLine();  int result;  if (!int.TryParse(input, out result))  {  Console.WriteLine("Neteisinga įvestis");  return;  }  Console.WriteLine("Iveskite antrą skaičių");  var input1 = Console.ReadLine();  int result1;  if (!int.TryParse(input1, out result1))  {  Console.WriteLine("Neteisinga įvestis");  return;  }  var mult = result \* result1;  Console.WriteLine($"Skaičių sandauga {mult}");  }  private void NumberSum()  {  Console.WriteLine("Iveskite pirmą skaičių");  var input = Console.ReadLine();  int result;  if (!int.TryParse(input, out result))  {  Console.WriteLine("Neteisinga įvestis");  return;  }  Console.WriteLine("Iveskite antrą skaičių");  var input1 = Console.ReadLine();  int result1;  if (!int.TryParse(input1, out result1))  {  Console.WriteLine("Neteisinga įvestis");  return;  }  var sum = result + result1;  Console.WriteLine($"Skaičių suma {sum}");  } |

*5 užduotis #5.* PARAŠYKITE PROGRAMĄ, KURI SUKURIA FAILĄ SU TEKSTU, JĮ NUSKAITO IR TURINĮ IŠVEDA Į EKRANĄ.

Programinis kodas:

|  |
| --- |
| using System;  using System.IO;  using System.Text;  class Failai  {  public static void Main()  {  string fileName = "mytest.txt";  using (StreamWriter fileStr = File.CreateText(fileName))  {  fileStr.WriteLine(" Hello, files");  }  using (StreamReader sr = File.OpenText(fileName))  {  string s = "";  while ((s = sr.ReadLine()) != null)  {  Console.WriteLine(s);  }  }  }  } |

*6 užduotis #6.* PARAŠYKITE PROGRAMĄ, KURI PRIE EGZISTUOJANČIO FAILO PRIDEDA PAPILDOMO TEKSTO, O NE JĮ PERRAŠO.

Programinis kodas:

|  |
| --- |
| using System;  using System.IO;  using System.Text;  class Failai  {  public static void Main()  {  string fileName = @"mytest.txt";  using (StreamWriter fileStr = File.CreateText(fileName))  {  fileStr.WriteLine("Naujas turinys");  }  using (StreamReader sr = File.OpenText(fileName))  {  string s = "";  Console.WriteLine(" Failo turinys:");  while ((s = sr.ReadLine()) != null)  {  Console.WriteLine(s);  }  Console.WriteLine("");  }  using (StreamWriter file = new StreamWriter(@"mytest.txt", true))  {  file.WriteLine("Papildomas turinys");  }  using (StreamReader sr = File.OpenText(fileName))  {  string s = "";  Console.WriteLine(" Naujas failo turinys: ");  while ((s = sr.ReadLine()) != null)  {  Console.WriteLine(s);  }  Console.WriteLine("");  }  Console.Read();  }  } |

*7 užduotis #7.* PARAŠYKITE PROGRAMĄ, KURI PATIKRINA AR FAILAS EGZISTUOJA IR JĮ IŠTRINA.

Programinis kodas:

|  |
| --- |
| using System;  using System.IO;  using System.Text;  class Failai  {  public static void Main()  {  string fileName = @"mytest.txt";  if (File.Exists(fileName))  {  File.Delete(fileName);  Console.WriteLine($" Failai { fileName } ištrintas");  }  else  {  Console.WriteLine("Failas neegzistuoja");  Console.ReadKey();  }  }  } |

*8 užduotis #8.* PARAŠYKITE PROGRAMĄ, KURI SUKURIA FAILĄ SU NURODYTU KIEKIU EILUČIŲ (KIEKVIENOJE EILUTĖJE SAUGOMAS EILUTĖS INDEKSAS IR VARTOTOJO ĮVESTAS TEKSTAS).

Programinis kodas:

|  |
| --- |
| using System;  using System.IO;  class Failai  {  static void Main()  {  string fileName = @"mytest.txt";  string[] lines;  if (File.Exists(fileName))  {  File.Delete(fileName);  }  Console.WriteLine("Ivesti eiluciu skaiciu:");  int n = Convert.ToInt32(Console.ReadLine());  lines = new string[n];  for (int i = 0; i < n; i++)  {  Console.WriteLine($" Iveskite eilutes {i+1} turini : ");  lines[i] = Console.ReadLine();  }  File.WriteAllLines(fileName, lines);  Console.WriteLine("");  using (StreamReader sr = File.OpenText(fileName))  {  string s = "";  Console.WriteLine(" Failo turinys: ");  while ((s = sr.ReadLine()) != null)  {  Console.WriteLine(s);  }  Console.WriteLine("");  }  if (File.Exists(fileName))  {  string[] content = File.ReadAllLines(fileName);  Console.WriteLine(" Paskutines failo eilutes turinys: ");  Console.WriteLine($" {lines[content.Length - 1]}");  }  }  } |

### Tema. LINQ, .NET Generics ir kolekcijos.

*9 užduotis #9*. SUKURKITE PROGRAMĄ, KURI TURI SWAP METODĄ, LEIDŽIANTĮ SUKEISTI REIKŠMES VIETOMIS, T. Y. KVIEČIAME SWAP(A,B), KUR A=2, B=3, O METODAS PADARO B=3 IR A=2. SUKURTI GENERIC METODO VERSIJĄ.

Programinis kodas:

|  |
| --- |
| void Swap<T>(ref T lhs, ref T rhs)  {  T temp;  temp = lhs;  lhs = rhs;  rhs = temp;  } |

*10 užduotis #10.* SUKURKITE SVEIKŲ SKAIČIŲ MASYVĄ IŠ VARTOTOJO ĮVESTIES IR ATSPAUSDINTI JO NARIŲ SUMĄ.

Masyvo dydį nurodo vartotojas.

Programinis kodas:

|  |
| --- |
| using System;  public class Masyvai  {  public static void Main()  {  int sum = 0;  Console.Write("Iveskite masyvo dydi:");  int size = Convert.ToInt32(Console.ReadLine());  int[] arr = new int[size];  for (int i = 0; i < arr.Length; i++)  {  Console.Write("elementas {0}: ", i);  arr[i] = Convert.ToInt32(Console.ReadLine());  }  for (int i = 0; i < arr.Length; i++)  {  sum += arr[i];  }  Console.Write($"Masyvo elementu suma: {sum}");  }  } |

*11 užduotis #11.* SUKURKITE PROGRAMĄ, KURI PRAŠO VARTOTOJO VESTI SVEIKUOSIUS SKAIČIUS TOL, KOL ŠIS NEĮVES 0, TUOMET IŠVESKITE KIEK SKAIČIŲ VARTOTOJAS ĮVEDĖ, JŲ SUMĄ IR ARITMETINĮ VIDURKĮ.

Nulis į skaičių sąrašą nepridedamas.

Programinis kodas:

|  |
| --- |
| using System;  using System.Collections.Generic;  using System.IO;  class Pratimai  {  static void Main()  {  List<int> myList = new List<int>();  Console.WriteLine("Ivestkite skaicius. Nutraukti ivedima galima ivedus 0");  string line;  int number;  int sum = 0;  while ((line = Console.ReadLine()) != "0")  {  number = int.Parse(line);  myList.Add(number);  sum += number;  }  Console.WriteLine($"Ivestu skaiciu vidurkis: {sum / myList.Count}, suma: {sum}");  }  } |

*12 užduotis #12.* SUKURTI PROGRAMĄ, KURI TELEFONO NUMERIŲ IR VARDŲ SAUGOJIMUI NAUDOJA DICTIONARY KLASĘ, KUR TELEFONO NUMERIS YRA DUOMENŲ POROS RAKTAS, O VARDAS POROS REIKŠMĖ.

Programinis kodas:

|  |
| --- |
| using System;  using System.Collections.Generic;  public class Knyga  {  public static void Main()  {  Dictionary<int, string> book = new Dictionary<int, string>();  int phoneNumber;  Console.Write("Iveskite tel. nr.:");  while((phoneNumber = int.Parse(Console.ReadLine())) !=0)  {  Console.Write("Iveskite varda:");  string name = Console.ReadLine();  book.Add(phoneNumber, name);  Console.Write("Iveskite tel. nr.:"); // naujas ciklas  }  Console.WriteLine(" Telefonu knyga:");  foreach (var item in book)  {  Console.WriteLine($"{item.Value} {item.Key}");  }  }  } |

*13 užduotis #13.* SUKURKITE PROGRAMĄ, KURI SUKURIA SVEIKŲ SKAIČIŲ NUO 1 IKI 10 DUOMENŲ MASYVĄ IR NAUDOJANT LINQ UŽKLAUSĄ IŠVEDAMI TIK LYGINIAI SKAIČIAI.

Programinis kodas:

|  |
| --- |
| using System;  using System.Linq;  class Linq  {  static void Main()  {  int[] numbers = new int[10] { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 };  var q = from num in numbers  where (num % 2) == 0  select num;  Console.Write("Lyginiai skaiciai:");  foreach (int num in q)  {  Console.Write($"{num} ");  }  }  } |

*14 užduotis #14.* SUKURKITE PROGRAMĄ, KURI SURŪŠIUOJA DUOTĄ MIESTO PAVADINIMŲ SĄRAŠĄ PAGAL ABĖCĖLĘ.

Programinis kodas:

|  |
| --- |
| using System;  using System.Linq;  using System.Collections.Generic;  class Linq  {  static void Main(string[] args)  {  string[] cities =  { "ROME","LONDON","NAIROBI","CALIFORNIA","ZURICH","NEW DELHI","AMSTERDAM","ABU DHABI", "PARIS" };  Console.WriteLine("Surusiuotas sarasas: ");  IEnumerable<string> cityOrder = cities  .OrderBy(str => str);  foreach (string city in cityOrder)  Console.WriteLine(city);  Console.ReadLine();  }  } |

*15 užduotis #15*. SUKURKITE PROGRAMĄ, KURI „VARTOTOJAI ĮVESTAI“ TEKSTO EILUTEI, IŠVEDA ŽODŽIUS NE TRUMPESNIUS, NEI NURODĖ VARTOTOJAS. PVZ.: TEKSTUI „GERA DIENA“ IR „5“ IŠVEDAMAS TIK VIENAS ŽODIS „DIENA“.

Programinis kodas:

|  |
| --- |
| using System;  using System.Linq;  class Linq  {  static void Main(string[] args)  {  Console.WriteLine("Iveskite teksto eilute:");  string text = Console.ReadLine();  Console.Write("Iveskite minimalu zodzio ilgi:");  int len = Convert.ToInt32(Console.ReadLine());  var q = text.Replace(",", "").Split(" ")  .Where(p => p.Length >= len);  Console.WriteLine($"Zodziai netrumpesni nei {len} simboliai:");  foreach (string word in q)  Console.WriteLine(word);  }  } |

*16 užduotis #16.* PARAŠYKITE LAMBDA IŠRAIŠKĄ, KURI DUOTĄ SVEIKĄ SKAIČIŲ PAKELIA KVADRATU IR PADEMONSTRUOTI JOS VEIKIMĄ.

Programinis kodas:

|  |
| --- |
| Func<int, int> kvadratu = x => x \* x;  Console.WriteLine(kvadratu(2)); |

*17 užduotis #17.* PARAŠYKITE LAMBDA IŠRAIŠKĄ, KURI DUOTAM SVEIKAM SKAIČIUI GRĄŽINTŲ TRUE, JEI SKAIČIUS YRA LYGINIS IR FALSE KITU ATVEJU.

Programinis kodas:

|  |
| --- |
| int[] arr = new int[] { 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 };  Func<int, bool> lyginis = x => x % 2 == 0;  foreach (var item in arr.Where(lyginis)) // arba tiesiog lambda funkcija *Where* sąlygoje  {  Console.WriteLine(item);  } |

### Tema. NET karkaso komponentai.

*18 užduotis.* SUKURKITE METŲ MĖNESIŲ REIKŠMIŲ ENUMERACIJĄ , KUR SAUSIS YRA 1, O GRUODIS – 12.

Vartotojui leisti įvesti skaičių ir išvesti skaičių atitinkantį mėnesį iš enumeracijos.

Programinis kodas:

|  |
| --- |
| public static void Main()  {  Console.Write("Iveskite menesi: ");  int month = int.Parse(Console.ReadLine());  Console.WriteLine($"Skaicius {month} atitinka {(Months)month} menesi");  }  enum Months : byte { Sau = 1, Vas, Kov, Bal, Geg, Bir, Lie, Rgp, Rgs, Spa, Lap, Gru }; |

*19 užduotis.* SUKURKITE IR PANAUDOKITE METODĄ, KURIS ESAMAI DATAI GRĄŽINA KELIAS REIKŠMES IŠKART – METUS, MĖNESĮ IR DIENĄ – NAUDOJANT VALUETUPLE KLASĘ.

Programinis kodas:

|  |
| --- |
| public static void Main()  {  var (year, month, day) = GetTodayDetails();  Console.WriteLine($"Siandien yra {year} metai {month} menuo {day} diena");  }  private static (int, int, int) GetTodayDetails()  {  var now = DateTime.Now;  return (now.Year, now.Month, now.Day);  } |

*20 užduotis.* SUKURKITE KLASĘ VARTOTOJO INFORMACIJAI SAUGOTI (VARDUI, PAVARDEI, AMŽIUI, EL. PAŠTO ADRESUI).

Šiai klasei sukurti dekonstrukcijos metodą, kuris būtų naudojamas vartotojo klasės objektui dekonstruoti.

Programinis kodas:

|  |
| --- |
| public static void Main()  {  var user = new User  {  FirstName = "Vardis",  LastName = "Pavardenis",  Email = "vardis@pavardenis.lt",  Age = 23  };  var (firstName, lastName) = user;  Console.WriteLine($"Vardas yra: {firstName} {lastName}");  }  public class User  {  public string FirstName { get; set; }  public string LastName { get; set; }  public int Age { get; set; }  public string Email { get; set; }  public void Deconstruct(out string firstName, out string lastName)  {  firstName = FirstName;  lastName = LastName;  }  } |

### Tema. Kodavimo standartai.

*21 užduotis.* SUKURKITE SU 2 METODAIS INTERFEISĄ, ATITINKANTĮ MICROSOFT PAVADINIMŲ GAIRES.

Programinis kodas:

|  |
| --- |
| interface IEmployee  {  void GoToWork();  int GetSalary(int days);  } |

*22 užduotis.* SUKURKITE KLASĘ SU 3 SAVYBĖMIS IR 2 METODAIS, ATITINKANČIĄ MICROSOFT PAVADINIMŲ GAIRES.

Programinis kodas:

|  |
| --- |
| class Employee  {  public int Id { get; set; }  public string Name { get; set; }  public DateTime BirthDate { get; set; }  public int GetSalary(int days)  {  return 1000;  }  public void GoToWork()  {  Console.WriteLine("Going to work...");  }  } |

*23 užduotis.* ĮGYVENDINTINKITE SINGLETON PROJEKTAVIMO ŠABLONĄ PROGRAMOS KONFIGŪRACIJAI LAIKYTI: DUOMENŲ BAZĖS PRISIJUNGIMO INFORMACIJAI, MAKSIMALIAM KLIENTŲ .

Programinis kodas:

|  |
| --- |
| public class Configuration  {  public string DbConnectionString { get; set; }  public int MaxClients { get; set; }  private static Configuration instance;  private Configuration() { }  public static Configuration Instance  {  get  {  if (instance == null)  {  instance = new Configuration();  }  return instance;  }  }  } |

*24 užduotis.* SUKURKITE PROGRAMĄ SU METODU, KURIS GRĄŽINA DVIEJŲ SVEIKŲ SKAIČIŲ SUMĄ IR JAM PRIDĖTI METODO SANTRAUKĄ.

Programinis kodas:

|  |
| --- |
| /// <summary>  /// Metodas grąžiną skaičių sumą  /// </summary>  /// <param name="n1">Pirmasis skaičius</param>  /// <param name="n2">Antrasis skaičius</param>  private static void PrintText(int n1, int n2)  {  return n1+n2;  } |

### Tema. MVC (RAZOR engine) technologijos panaudojimas vartotojo sąsajai programuoti.

*25 užduotis.* APIBRĖŽKITE MVC ŠABLONĄ, ESMINIUS JO KOMPONENTUS IR JŲ ATSAKOMYBES.

Užduoties sprendimui naudokite šiuos resursus:

* <https://en.wikipedia.org/wiki/Model%E2%80%93view%E2%80%93controller>
* <https://www.martinfowler.com/eaaDev/uiArchs.html#ModelViewController>

*26 užduotis.* APIBŪDINKITE RAZOR VARIKLIO PASKIRTĮ*.*

Užduoties sprendimui naudokite šiuos resursus:

* <https://docs.microsoft.com/en-us/aspnet/web-pages/overview/getting-started/introducing-razor-syntax-c>
* <https://en.wikipedia.org/wiki/ASP.NET_Razor>
* <https://docs.microsoft.com/en-us/aspnet/core/mvc/views/razor>

*27 užduotis.* APIBRĖŽKITE KAS YRA RAZOR PAGES.

Užduoties sprendimui naudokite šiuos resursus:

* <https://docs.microsoft.com/en-us/aspnet/core/razor-pages/>
* <https://docs.microsoft.com/en-us/aspnet/core/tutorials/razor-pages/razor-pages-start>
* <https://www.learnrazorpages.com/>

*28 užduotis.* APIBRĖŽKITE KAS YRA RESTFUL SERVISAS IR KAS UŽTIKRINA, KAD SERVISĄ GALIMA LAIKYTI RESTFUL.

Užduoties sprendimui naudokite šiuos resursus:

* <https://restfulapi.net/>
* <https://www.restapitutorial.com/>
* <https://en.wikipedia.org/wiki/Representational_state_transfer>

*29 užduotis.* APIBRĖŽKITE RICHARDSON MATURITY MODEL IR JO LYGIUS.

Užduoties sprendimui naudokite šiuos resursus:

* <https://martinfowler.com/articles/richardsonMaturityModel.html>
* <https://restfulapi.net/richardson-maturity-model/>
* <http://restcookbook.com/Miscellaneous/richardsonmaturitymodel/>

*30 užduotis.* APIBRĖŽKITE, KAS YRA TARPINĖ PROGRAMINĖ ĮRANGA (ANGL. MIDDLEWARE) ASP.NET CORE KARKASO KONTEKSTE.

Užduoties sprendimui naudokite šiuos resursus:

* <https://docs.microsoft.com/en-us/aspnet/core/fundamentals/middleware/>
* <https://docs.microsoft.com/en-us/aspnet/core/fundamentals/middleware/write>
* <https://www.tutorialspoint.com/asp.net_core/asp.net_core_middleware.htm>

*31 užduotis.* APIBRĖŽKITE KAM ASP.NET CORE KARKASE YRA NAUDOJAMA STARTUP KLASĖ, JOS ATSAKOMYBES.

Užduoties sprendimui naudokite šiuos resursus:

* <https://docs.microsoft.com/en-us/aspnet/core/fundamentals/startup>
* <https://www.c-sharpcorner.com/article/understanding-startup-class-in-asp-net-core/>
* <https://www.tutorialsteacher.com/core/aspnet-core-startup>

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

### Tema. Objektinis programavimas.

*1 užduotis.* SUKURKITE KLASĘ KOORDINATĖS TAŠKUI POINT.

Programinis kodas:

|  |
| --- |
| public class Point  {  public int X { get; set; }  public int Y { get; set; }  public string Color { get; set; }  public Point()  {  }  public Point(string color)  {  Color = color;  }  public Point(int x, int y)  {  X = x;  Y = y;  }  public Point(int x, int y, string color)  {  X = x;  Y = y;  Color = color;  }  } |

*2 užduotis.* SUKURKITE NAUJĄ PROJEKTĄ MATHEMATICS.

Programinis kodas:

|  |
| --- |
| public static class Calculator  {  public static int Add(int a, int b)  {  return a + b;  }  public static int Subtract(int a, int b)  {  return a - b;  }  public static int Multiply(int a, int b)  {  return a \* b;  }  public static int Divide(int a, int b)  {  return a / b;  }  }  static void Main(string[] args)  {  Calculator.Add(1, 2);  Calculator.Subtract(1, 2);  Calculator.Divide(4, 2);  Calculator.Multiply(3, 2);  } |

*3 užduotis.* SUKURKITE KLASĘ AUTOMOBILIO INFORMACIJAI SAUGOTI.

Programinis kodas:

|  |
| --- |
| using System;  using System.Collections.Generic;  using System.Text;  namespace ConsoleApp  {  public class Car  {  public int MaxFuelAmount { get; }  public Car(string name)  {  this.MaxFuelAmount = 50;  this.Name = name;  }  public int Milage { get; private set; }  public int FuealAmount { get; private set; }  public string Name { get; }  public void Drive()  {  if (this.FuealAmount < 10)  {  Console.WriteLine("Neužtenka kuro važiuoti");  return;  }  this.Milage += 10;  }  public void Refuel(int amount)  {  if (this.FuealAmount + amount > MaxFuelAmount)  {  Console.WriteLine("Per didelis kuro kiekis");  return;  }  this.FuealAmount += amount;  }  }  } |

*4 užduotis.*  SUKURKITE KLASĘ GYVŪNAS SU ATRIBUTAIS VARDAS IR ŠEIMA IR METODĄ PASISVEIKINTI, KURIS ATSPAUSDINTŲ TEKSTĄ PAGAL ESAMUS ATRIBUTUS: „LABAS AŠ Y IŠ X GIMINĖS”.

Programinis kodas:

|  |
| --- |
| class Animal  {  public Animal(string name, string family)  {  Name = name;  Family = family;  }  public string Name { get; set; }  public string Family { get; set; }  public void Greet()  {  Console.WriteLine($"Labas aš {Name} iš {Family} giminės");  }  }  class Cat : Animal  {  private int meowCount;  public Cat(string name) : base(name, "Feline")  {  }  public void Meaow()  {  Console.WriteLine("Meaow!");  meowCount++;  }  }  class Dog : Animal  {  private int barkCount;  public Dog(string name) : base(name, "Canine")  {  }  public void Bark()  {  Console.WriteLine("Woof");  barkCount++;  }  }  class Lizard : Animal  {  private int coughtFlyCount;  public Lizard(string name) : base(name, "Reptile")  {  }  public void CatchFly()  {  Console.WriteLine("Fly cought");  coughtFlyCount++;  }  } |

*5 užduotis.* PADEMONSTRUOKITE POLIMORFIZMO GEBĖJIMUS SUKURIANT PROGRAMĄ SU ATITINKAMOMIS KLASĖMIS.

Programinis kodas:

|  |
| --- |
| public class Shape  {  private static int ShapeCount;  public Shape()  {  ShapeCount++;  }  public static int GetShapeCount()  {  return ShapeCount;  }  }  public class Cilinder : Circle  {  public Cilinder(double radius, double height) : base(radius)  {  this.Height = height;  }  public double Height { get; }  public override double Area()  {  return this.Perimeter() \* (this.Radius + this.Height);  }  public double AreaSide()  {  return base.Perimeter() \* this.Height;  }  public override double Perimeter()  {  return (4 \* this.Radius) + (2 \* this.Height);  }  public double Volume()  {  return base.Area() \* this.Height;  }  }  public class Circle : Shape  {  public Circle(double radius)  {  this.Radius = radius;  }  public double Radius { get; }  public virtual double Area()  {  return Math.PI \* Math.Pow(this.Radius, 2);  }  public virtual double Perimeter()  {  return Math.PI \* 2 \* this.Radius;  }  }  public class Cube : Square  {  public Cube(double side) : base(side)  {  }  public override double Area()  {  return base.Area() \* 6;  }  public override double Perimeter()  {  return base.Perimeter() \* 6;  }  public double Volume()  {  return Math.Pow(this.Side, 3);  }  }  public class Square : Shape  {  public Square(double side)  {  Side = side;  }  public double Side { get; }  public virtual double Perimeter()  {  return 4 \* this.Side;  }  public virtual double Area()  {  return Math.Pow(this.Side, 2);  }  } |

*6 užduotis.* SUKURKITE ABSTRAKČIĄ KLASĘ DARBUOTOJAS, KURI TURĖTŲ ATRIBUTĄ VARDAS IR KONSTRUKTORIŲ, KURIS INICIJUOJA ATRIBUTŲ REIKŠMES IR ABSTRAKTŲ VOID TIPO METODĄ DOWORK().

Sukurti klases programuotojas , vadybininkas, kurios paveldėtų klasę darbuotojas.

Programinis kodas:

|  |
| --- |
| public abstract class Employee  {  protected Employee(string name)  {  this.Name = name;  }  public string Name { get; }  public abstract void DoWork();  }  public class Accountant : Employee  {  public Accountant(string name) : base(name)  {  }  public override void DoWork()  {  Console.WriteLine("I deal with numbers all day");  }  }  public class Manager : Employee  {  public Manager(string name) : base(name)  {  }  public override void DoWork()  {  Console.WriteLine("I do managment stuff");  }  }  public class Programmer : Employee  {  public Programmer(string name) : base(name)  {  }  public override void DoWork()  {  Console.WriteLine("I am coding");  }  } |

*7 užduotis.* ATLIKTĄ *5 užduotį* „POLIMORFIZMO“ TEMOJE MODIFIKUOKITE: KLASĘ FIGŪRA PADARANT ABSTRAKČIAI IR JOJE SUKURTI METODUS PLOTAS IR PERIMETRAS, KURIE BŪTŲ ABSTRAKTŪS.

Programinis kodas:

|  |
| --- |
| public abstract class Shape  {  private static int ShapeCount;  protected Shape()  {  ShapeCount++;  }  public static int GetShapeCount()  {  return ShapeCount;  }  public abstract double Area();  public abstract double Perimeter();  }  public class Square : Shape  {  public Square(double side)  {  Side = side;  }  public double Side { get; }  public override double Perimeter()  {  return 4 \* this.Side;  }  public override double Area()  {  return Math.Pow(this.Side, 2);  }  }  public class Circle : Shape  {  public Circle(double radius)  {  this.Radius = radius;  }  public double Radius { get; }  public override double Area()  {  return Math.PI \* Math.Pow(this.Radius, 2);  }  public override double Perimeter()  {  return Math.PI \* 2 \* this.Radius;  }  }  public class Cube : Square  {  public Cube(double side) : base(side)  {  }  public override double Area()  {  return base.Area() \* 6;  }  public override double Perimeter()  {  return base.Perimeter() \* 6;  }  public double Volume()  {  return Math.Pow(this.Side, 3);  }  }  public class Cilinder : Circle  {  public Cilinder(double radius, double height) : base(radius)  {  this.Height = height;  }  public double Height { get; }  public override double Area()  {  return this.Perimeter() \* (this.Radius + this.Height);  }  public double AreaSide()  {  return base.Perimeter() \* this.Height;  }  public override double Perimeter()  {  return (4 \* this.Radius) + (2 \* this.Height);  }  public double Volume()  {  return base.Area() \* this.Height;  }  } |

*8 užduotis.* SUKURKITE INTERFEISĄ GEOMETRINEI FIGŪRAI SU PLOTO IR PERIMETRO SKAIČIAVIMO METODAIS.

Sukurti klasę, įgyvendinančią sukurtą interfeisą.

Programinis kodas:

|  |
| --- |
| interface IIShape  {  double CalculateArea();  double CalculatePerimeter();  }  class Shape : IIShape  {  public double CalculateArea()  {  throw new NotImplementedException();  }  public double CalculatePerimeter()  {  throw new NotImplementedException();  }  } |

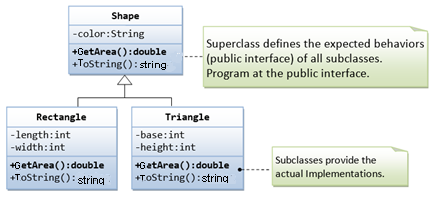
### Tema. UML klasių diagramos.

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

Programinis kodas:

|  |
| --- |
| 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 this.name;  }  public double GetPrice()  {  return this.price;  }  public double GetPriceWithVat()  {  return GetPrice() \* GetVat();  }  public abstract double GetVat();  }  public class Product : AbstractProduct, ShoppingCartItem  {  private static double VAT = 1.21;  public Product(string name, double price) : base(name, price) { }  public override double GetVat()  {  return VAT;  }  }  public class Periodical : AbstractProduct, ShoppingCartItem  {  private static double PERIODICAL\_VAT = 1.09;  public Periodical(string name, double price) : base(name, price) { }  public override double GetVat()  {  return PERIODICAL\_VAT;  }  }  public class Medicine : AbstractProduct, ShoppingCartItem  {  private static double MEDICINE\_VAT = 1.05;  public Medicine(string name, double price) : base(name, price) { }  public override double GetVat()  {  return MEDICINE\_VAT;  }  }  public interface ShoppingCartItem  {  double GetPrice();  double GetPriceWithVat();  }  public class ShoppingCart  {  private List<ShoppingCartItem> items = new List<ShoppingCartItem>();  public void Add(ShoppingCartItem item)  {  items.Add(item);  }  public double GetTotalPrice()  {  double totalPrice = 0;  foreach (ShoppingCartItem item in items)  {  totalPrice = totalPrice + item.GetPrice();  }  return totalPrice;  }  public double GetTotalPriceWithVat()  {  double totalPriceWithVat = 0;  foreach (ShoppingCartItem item in items)  {  totalPriceWithVat += item.GetPriceWithVat();  }  return totalPriceWithVat;  }  }  public class Uzd  {  public static void Main(string[] args)  {  Medicine panadol = new Medicine("Panadol", 12.45);  Periodical newspaper = new Periodical("Lietuvos rytas", 3);  ShoppingCart myCart = new ShoppingCart();  myCart.Add(panadol);  myCart.Add(newspaper);  myCart.GetTotalPrice();  }  } |

*10 užduotis.* PAGAL PATEIKTAS C# KLASES NUBRAIŽYKITE UML KLASIŲ DIAGRAMĄ.



### 3.3.3. Tema. Objektinio programavimo gerųjų praktikų (SOLID principų) taikymas .NET.

*11 užduotis.* APRAŠYKITE SOLID (SRP, OCP, LSP, ISP, DIP) PROGRAMAVIMO PRINCIPUS.

Užduoties sprendimui naudokite šiuos resursus:

* <https://en.wikipedia.org/wiki/SOLID>
* <http://www.blackwasp.co.uk/SOLIDPrinciples.aspx>

*12 užduotis.* APRAŠYKITE KAM SKIRTI INVERSION OF CONTROL (IOC) KONTEINERIAI IR KOKIE YRA.

Užduoties sprendimui naudokite šiuos resursus:

* <https://www.martinfowler.com/articles/injection.html>
* <https://en.wikipedia.org/wiki/Dependency_injection>

*13 užduotis.*  NAUDOJANTIS PATEIKTA UML KLASIŲ DIAGRAMA SUPROGRAMUOKITE SOLID PRINCIPUS ATITINKANTĮ KAVOS APARATO KODĄ IR PROGRAMĄ NAUDOJANČIĄ ŠIAS KLASES IR DEPENDENCY CONSTRUCTOR INJECTION.

Paaiškinimus ir pagalbinį papildomą kodą galima rasti <https://lestard.eu/2014/easy-di/>

Programinis kodas:

|  |
| --- |
| using System;  namespace Playground  {  public class Program  {  static void Main(string[] args)  {  Mill mill = new Mill();  WaterTank waterTank = new WaterTank();  BeanContainer beanContainer = new BeanContainer();  CoffeePowderProvider powderProvider = new CoffeePowderProvider(mill, beanContainer);  CoffeeMachine coffeeMachine = new CoffeeMachine(powderProvider, waterTank);  coffeeMachine.MakeCoffee();  Console.ReadKey();  }  }  public class Mill  {  public void Grind()  {  Console.WriteLine($"{nameof(Mill)}: mala");  }  }  public class BeanContainer  {  public void GetBeans(int amount)  {  Console.WriteLine($"{nameof(BeanContainer)}: tiekiamos {amount} kavos pupelės");  }  }  public class WaterTank  {  public void GetWater()  {  Console.WriteLine($"{nameof(WaterTank)}: gaunamas vanduo");  }  }  public class CoffeePowderProvider  {  private Mill mill;  private BeanContainer container;  public CoffeePowderProvider(Mill mill, BeanContainer container)  {  this.mill = mill;  this.container = container;  }  public void GetPowder()  {  Console.WriteLine($"{nameof(CoffeePowderProvider)}: kava malama");  container.GetBeans(10);  mill.Grind();  Console.WriteLine($"{nameof(CoffeePowderProvider)}: kava sumalta");  }  }  public class CoffeeMachine  {  private CoffeePowderProvider coffeePowderProvider;  private WaterTank waterTank;  public CoffeeMachine(CoffeePowderProvider coffeePowderProvider, WaterTank waterTank)  {  this.coffeePowderProvider = coffeePowderProvider;  this.waterTank = waterTank;  }  public void MakeCoffee()  {  Console.WriteLine($"{nameof(CoffeeMachine)}: kava pradedama ruošti");  waterTank.GetWater();  coffeePowderProvider.GetPowder();  Console.WriteLine($"{nameof(CoffeeMachine)}: kava paruošta");  }  }  } |

### Mokymosi rezultatas. Naudoti daugiagijo programavimo .NET karkase priemones.

### Tema. Daugiagijis (Multithreaded) programavimas.

*1 užduotis.* SUKURKITE METODĄ, KURIS NURODYTĄ KIEKĮ KARTŲ (PVZ. 100) Į EKRANĄ IŠVEDA ESAMOS GIJOS ID.

Programinis kodas:

|  |
| --- |
| static async Task Main(string[] args)  {  Thread t = new Thread(Write);  t.Start(100);  Write(100);  }  static void Write(object times)  {  for (int i = 0; i < (int)times; i++)  {  Console.Write(Thread.CurrentThread.ManagedThreadId);  }  } |

*2 užduotis.* PANAUDOKITE METODĄ IŠ „THREAD DARBAS SU GIJOMIS*“ 1 užduoties.*

Programinis kodas:

|  |
| --- |
| static async Task Main(string[] args)  {  Task t2 = Task.Factory.StartNew(()=>Write(10));  //t2.Wait();  Task t3 = Task.Run(() => Write(10));  //t3.Wait();  }  static void Write(int times)  {  for (int i = 0; i < times; i++)  {  Console.Write(Thread.CurrentThread.ManagedThreadId);  }  } |

*3 užduotis*. SUKURKITE KONSOLINĘ PROGRAMĄ SU METODU, KURIS 100 KARTŲ PADIDINTŲ BENDRAI PASIEKIAMĄ INT TIPO KINTAMĄJĮ VIENETU.

Programinis kodas:

|  |
| --- |
| static int c = 0;  static object o = new object();  static async Task Main(string[] args)  {  //var t1 = new Thread(Add); var t2 = new Thread(Add); var t3 = new Thread(Add);  //var t4 = new Thread(Add); var t5 = new Thread(Add);  //t1.Start(); t2.Start(); t3.Start(); t4.Start(); t5.Start();  //t1.Join(); t2.Join(); t3.Join(); t4.Join(); t5.Join();  List<Task> tasks = new List<Task>(5);  tasks.Add(Task.Run(() => Add()));  tasks.Add(Task.Run(() => Add()));  tasks.Add(Task.Run(() => Add()));  tasks.Add(Task.Run(() => Add()));  tasks.Add(Task.Run(() => Add()));  await Task.WhenAll(tasks);  Console.WriteLine(c);  Console.ReadKey();  }  static void Add()  {  for (int i = 0; i < 100; i++)  {  Thread.Sleep(10);  //lock (o)  {  c++;  Console.WriteLine($"T{Thread.CurrentThread.ManagedThreadId}: {c}");  }  }  } |

### Tema. Asinchroninis programavimas naudojant .Net aplinką.

### 

*4 užduotis.* SUKURKITE METODUS ASINCHRONINIAM ILGAI TRUNKANČIŲ UŽDUOČIŲ VYKDYMUI IMITUOTI, KURIAME PANAUDOTI AWAIT TASK.DELAY() ASYNC METODAI.

Programinis kodas:

|  |
| --- |
| static async Task Main(string[] args)  {  Stopwatch watch = new Stopwatch();  watch.Start();  await AggregateWork();  watch.Stop();  Console.WriteLine(watch.ElapsedMilliseconds);  Console.ReadKey();  }  public static async Task AggregateWork()  {  var t1 = DoWorkAsync(2 \* 1000);  var t2 = DoWorkAsync(3 \* 1000);  var t3 = DoWorkAsync(5 \* 1000);  await t1;  await t2;  await t3;  }  public static async Task DoWorkAsync(int ms)  {  await Task.Delay(ms);  } |

*5 užduotis.* SUKURKITE ASYNC METODĄ, KURIS SUKELTŲ KLAIDĄ.

Programinis kodas:

|  |
| --- |
| static async Task Main(string[] args)  {  try  {  await GetException();  }  catch (Exception exc)  {  Console.WriteLine("Caught: " + exc.Message);  }  Console.ReadKey();  }  static async Task GetException()  {  await Task.Delay(1);  throw new Exception("Catch me");  } |

*6 užduotis.* MODIFIKUOKITE 4-OJE ŠIOS TEMOS UŽDUOTYJE PARAŠYTĄ INKAPSULIUOJANTĮ METODĄ TAIP, KAD REIKĖTŲ LAUKTI NE KIEKVIENOS UŽDUOTIS ATSKIRAI, O BENDRAI VISŲ TRIJŲ UŽDUOČIŲ PABAIGOS.

Programinis kodas:

|  |
| --- |
| class Program  {  static async Task Main(string[] args)  {  Stopwatch watch = new Stopwatch();  watch.Start();  await AggregateWork();  watch.Stop();  Console.WriteLine(watch.ElapsedMilliseconds);  Console.ReadKey();  }  public static async Task AggregateWork()  {  Task[] tasks = new Task[3];  tasks[0] = DoWorkAsync(2 \* 1000);  tasks[1] = DoWorkAsync(3 \* 1000);  tasks[2] = DoWorkAsync(5 \* 1000);  await Task.WhenAll(tasks);  }  public static async Task DoWorkAsync(int ms)  {  await Task.Delay(ms);  }  } |

### Mokymosi rezultatas. Testuoti programinę įrangą naudojant su .NET karkasu suderinamus testavimo įrankius ir metodus.

### Tema. Programų testavimas.

*1 užduotis.* SUKURKITE KLASĘ SU VIENU TEKSTINIU LAUKU, KURIS PADUODAMAS PER KONSTRUKTORIŲ IR METODU, KURIS GRĄŽINA TĄ LAUKĄ.

Programinis kodas:

|  |
| --- |
| public class SimpleText  {  string myText;  public SimpleText(string text)  {  myText = text;  }  public string GetText()  {  return myText;  }  }  [TestClass]  public class SimpleTextTest  {  [TestMethod]  public void SimpleText\_GetText()  {  var text = "myText";  var obj = new SimpleText(text);  var actualText = obj.GetText();  Assert.AreEqual(text, actualText);  }  } |

*2 užduotis*. SU MS TEST TESTAVIMO KARKASU SUKURKITE MODULIŲ TESTŲ PROJEKTĄ.

Programinis kodas:

|  |
| --- |
| using Microsoft.VisualStudio.TestTools.UnitTesting;  using System;  namespace UnitTestProject1  {  [TestClass]  public class UnitTest1  {  [TestMethod]  public void ShouldReturnCorrectSum()  {  // arrange  var expected = 4;  // act  var sum = 2 + 2;  // assert  Assert.AreEqual(4, sum);  }  [TestMethod]  [ExpectedException(typeof(Exception))]  public void ShoulThrowException()  {  throw new Exception();  }  [TestMethod]  public void ShouldFail()  {  // arrange  var expected = 4;  // act  var sum = 2 + 3;  // assert  Assert.AreEqual(4, sum);  }  }  } |

*3 užduotis.* SUKURKITE KLASĘ TODOITEM, KURIS TURĖTŲ ATRIBUTUS: PAVADINIMAS, STATUSAS. PRADINIS STATUSAS TURI BŪTI „NOTSTARTED*“.*

Programinis kodas:

|  |
| --- |
| using System;  namespace ConsoleApp  {  public class ToDoItem  {  public ToDoItem(string name)  {  Name = name;  }  public string Name { get; }  public string Status { get; private set; } = "NotStarted";  public void Start()  {  if (Status != "NotStarted")  {  throw new Exception("Invalid action");  }  Status = "Doing";  }  public void Done()  {  if (Status != "Doing")  {  throw new Exception("Invalid action");  }  Status = "Done";  }  public void Archive()  {  if (Status != "Done")  {  throw new Exception("Invalid action");  }  Status = "Archived";  }  }  } |

|  |
| --- |
| using ConsoleApp;  using Microsoft.VisualStudio.TestTools.UnitTesting;  using System;  namespace UnitTestProject1  {  [TestClass]  public class ToDoItemTests  {  [TestMethod]  public void Start\_Test()  {  // Arrange  var todoItem = new ToDoItem("");  // Act  todoItem.Start();  // Assert  Assert.AreEqual("Doing", todoItem.Status);  }  [TestMethod]  [ExpectedException(typeof(Exception))]  public void Start\_Test\_ShouldThrow()  {  // Arrange  var todoItem = new ToDoItem("");  // Act  todoItem.Start();  todoItem.Done();  todoItem.Start();  }  [TestMethod]  public void Done\_Test()  {  // Arrange  var todoItem = new ToDoItem("");  // Act  todoItem.Start();  todoItem.Done();  // Assert  Assert.AreEqual("Done", todoItem.Status);  }  [TestMethod]  [ExpectedException(typeof(Exception))]  public void Done\_Test\_ShouldThrow()  {  // Arrange  var todoItem = new ToDoItem("");  // Act  todoItem.Done();  // Assert  Assert.Fail();  }  [TestMethod]  public void Archive\_Test()  {  // Arrange  var todoItem = new ToDoItem("");  // Act  todoItem.Start();  todoItem.Done();  todoItem.Archive();  // Assert  Assert.AreEqual("Archived", todoItem.Status);  }  [TestMethod]  [ExpectedException(typeof(Exception))]  public void Archive\_Test\_ShouldThrow()  {  // Arrange  var todoItem = new ToDoItem("");  // Act  todoItem.Archive();  }  }  } |

*4 užduotis.* SUKURKITE METODĄ, KURIS VARTOTOJO PAPRAŠYTŲ ĮVESTI NORIMO MASYVO DYDĮ IR VISUS JO NARIUS.

Programinis kodas:

|  |
| --- |
| static void Main(string[] args)  {  try  {  Console.WriteLine("Įveskite norimo masyvo dydį");  int arrSize = int.Parse(Console.ReadLine());  var arr = new int[arrSize];  for (int i = 0; i < arrSize; i++)  {  Console.WriteLine($"Įveskite {i + 1} masyvo narį");  arr[i] = int.Parse(Console.ReadLine());  }  Console.WriteLine("Kuri nari atspausdinti");  var input = int.Parse(Console.ReadLine());  var item = arr[input - 1];  Console.WriteLine(item);  Console.Read();  }  catch (OverflowException)  {  Console.WriteLine("Nurodytas netinkamas masyvo dydis");  }  catch (IndexOutOfRangeException)  {  Console.WriteLine("Nurodytas neegzistuojantis masyvo indeksas");  }  catch (FormatException)  {  Console.WriteLine("Neteisinga įvestis");  }  } |

*5 užduotis.* SUKURKITE METODĄ „LOGIN“, KURIS PAPRAŠYTŲ VARTOTOJĄ ĮVESTI PRISIJUNGIMĄ IR SLAPTAŽODĮ.

Programinis kodas:

|  |
| --- |
| static void Main(string[] args)  {  try  {  Login();  }  catch (Exception e)  {  Console.WriteLine(e.Message);  }  }  private static void Login()  {  Console.WriteLine("Įveskite vartotojo vardą");  var userName = Console.ReadLine();  Console.WriteLine("Įveskite slaptaždį");  var pass = Console.ReadLine();  if (userName == "admin" && pass == "admin")  {  throw new Exception("Neteisingi duomenys");  }  Console.WriteLine("Sveiki prisijungę");  } |

*6 užduotis.*  SUKURKITE KLASĘ BEAR. KURI TURI ATRIBUTUS : RŪŠIS, AR MIEGA.

Programinis kodas:

|  |
| --- |
| public class Bear  {  public string Type { get; set; }  public bool IsSleeping { get; set; }  public void GiveHoney(Honey honey)  {  if (honey.HoneyType == "Buckwheat" ||  honey.HoneyType == "Mustard" ||  !honey.IsFresh)  {  throw new BadHoneyException(honey);  }  if (IsSleeping)  {  throw new BearIsSleepingException(100);  }  Console.WriteLine($"The {Type} bear is enjoying the honey.");  }  }  public class Honey  {  public string HoneyType { get; set; }  public bool IsFresh { get; set; }  public override string ToString()  {  return $"Honey:{nameof(HoneyType)} - {HoneyType} \n {nameof(IsFresh)} - {IsFresh}";  }  }  public class BadHoneyException : Exception  {  public BadHoneyException(Honey honey) : base()  {  Honey = honey;  }  public BadHoneyException(Honey honey, string message) : base(message)  {  Honey = honey;  }  public BadHoneyException(Honey honey, string message, Exception innerException) : base(message, innerException)  {  Honey = honey;  }  public Honey Honey { get; }  public override string Message  {  get  {  return $"Bear doesn't like { Honey.ToString() }";  }  }  }  public class BearIsSleepingException : Exception  {  public BearIsSleepingException(int roarDecibals) : base()  {  RoarDecibals = roarDecibals;  }  public BearIsSleepingException(int roarDecibals, string message) : base(message)  {  RoarDecibals = roarDecibals;  }  public BearIsSleepingException(int roarDecibals, string message, Exception innerException) : base(message, innerException)  {  RoarDecibals = roarDecibals;  }  public int RoarDecibals { get; }  public override string Message  {  get  {  return $"Bear is roaring at {RoarDecibals} decibals. It is very angry. {base.Message}";  }  }  } |

### Tema. Programų derinimas.

*7 užduotis.* APIBRĖŽKITE, KAS YRA PROGRAMINĖS ĮRANGOS ŽURNALAI.

Užduoties sprendimui naudokite šiuos resursus:

* <https://en.wikipedia.org/wiki/Log_file>

*8 užduotis.* ĮVARDINKITE, KOKIOS ŽURNALŲ PRIEMONĖS YRA SKIRTOS .NET PLATFORMAI.

Užduoties sprendimui naudokite šiuos resursus:

* <https://en.wikipedia.org/wiki/List_of_.NET_libraries_and_frameworks#Logging_frameworks>

*9 užduotis.* KONSOLINĖS APLIKACIJOS PROJEKTE, NAUDOJANT SERILOG ŽURNALO ĮRANKĮ, IŠVESKITE PAPRASTĄ INFORMACINĮ ŽURNALO PRANEŠIMĄ Į KONSOLĘ.

Package Manager konsolėje įvykdyti:

|  |
| --- |
| PM> Install-Package Serilog  PM> Install-Package Serilog.Sinks.Console |

Programinis kodas:

|  |
| --- |
| using Serilog;  namespace Playground  {  public class Program  {  static void Main(string[] args)  {  var log = new LoggerConfiguration()  .WriteTo  .Console()  .CreateLogger();  log.Information("Logging exercise");  }  }  } |

*10 užduotis.* APIBRĖŽKITE, KAS YRA KODO DERINIMO PROCESAS.

Užduoties sprendimui naudokite šiuos resursus:

* <https://en.wikipedia.org/wiki/Debugging>

*11 užduotis.* ĮVARDINKITE, KOKIOS KODO DERINIMO PRIEMONES PATEIKIAMOS VISUAL STUDIO APLINKOJE.

Užduoties sprendimui naudokite šiuos resursus:

* <https://docs.microsoft.com/en-us/visualstudio/get-started/csharp/tutorial-debugger>

*12 užduotis.* APIBRĖŽKITE, KAS YRA STATINĖ KODO ANALIZĖ.

Užduoties sprendimui naudokite šiuos resursus:

* <https://en.wikipedia.org/wiki/Static_program_analysis>

*13 užduotis.* IŠVARDINKITE KELIS STATINĖS KODO ANALIZĖS ĮRANKIUS, SKIRTUS .NET PLATFORMAI.

Užduoties sprendimui naudokite šiuos resursus:

* <https://en.wikipedia.org/wiki/List_of_tools_for_static_code_analysis#.NET>

1. **Kompetencija. Analizuoti skirtingų tipų reikalavimus, apibūdinančius kompiuterinę programą.**

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

### Tema. Reikalavimų programinei įrangai formatai.

*1 užduotis.* UŽRAŠYKITE VARTOTOJO PASAKOJIMO (USER STORY) FORMATĄ.

Vartotojo pasakojimo formatas:

**Kaip** <rolė>

**aš noriu** <veiksmas>

**tam, kad** <vertė>.

**Rolė** - aktyvus sistemos dalyvis - žmogus arba automatinė sistema.

**Veiksmas** - veiksmas kuris turi įvykti sistemoje.

**Vertė** - atspindi vertę verslui. Kam to reikia?

**Pavyzdžiai:**

1. Kaip personalo vadovas aš noriu turėti atrankos klausimyną tam, kad galėčiau suprasti, ar reikia perduoti kandidatą funkciniam vadovui.

2. Kaip vadovas aš noriu turėti galimybę peržiūrėti jau sukurtus klausimus tam, kad prisiminčiau, kokius klausimus jau turiu ir nuspręsti, ar pozicijai, kuriai šiuo metu vykdau atranką, galiu juos vėl panaudoti ar reikia atnaujinti esamą klausimyną.

3. Kaip sistemos vartotojas aš noriu pažymėti failus kurių nereikia kopijuoti, kad mano atsarginės kopijos erdvė neužsipildytų dalykais, kurių aš nenoriu išsaugoti.

Papildomai galite paskaityti:

* <https://en.wikipedia.org/wiki/User_story>

*2 užduotis.* YRA PATEIKTI FUNKCINIAI PROGRAMINĖS ĮRANGOS REIKALAVIMAI. PABANDYKITE SUFORMUOTI TRIS VARTOTOJO PASAKOJIMUS (USER STORIES*).*

Vartotojo pasakojimo pavyzdžiai pagal funkcinius reikalavimus:

* Aš kaip administratorius noriu registruoti naujus vartotojus sistemoje.
* Aš kaip administratorius norius keisti vartotojų slaptažodžius.
* Aš kaip vartotojas noriu prisijungti prie sistemos su savo vartotojo vardu ir slaptažodžiu

**Vartojimo atvejų formatas**

*3 užduotis.* APRAŠYKITE GERŲ VARTOTOJO PASAKOJIMO FORMATĄ (INVEST).

INVEST - vartotojo pasakojimai turintys šias savybes:

* Nepriklausomi (independent) - pasakojimai turėtų būti nepriklausomi vienas nuo kito.
* Aptariami (negotiable) - pasakojimai aptariami ir diskutuojami.
* Vertingi (valuable) - turi būti vertingi klientui arba pirkėjui.
* Įvertinami (estimable) - turi būti įmanoma pasakojimus įvertinti.
* Pakankamai maži (small enough) - rekomenduojamas dydis 1-5 dienos.
* Patikrinami (testable) - kiekvieną pasakojimą turi būti įmanoma patikrinti (ištestuoti).

### Tema. Reikalavimų peržiūros procesas.

*4 užduotis.* APRAŠYKITE REIKALAVIMŲ PERŽIŪROS ŽINGSNIUS.

Reikalavimų peržiūra (*Requirements Review*):

* Vykdoma sprinto eigoje ruošiant vartotojo pasakojimus kitam sprintui.
* Peržiūros tikslas, kad kiekvienas peržiūrėtas pasakojimas proceso gale atitiktų reikalavimų paruošimo kriterijus (angl. *Definition of Ready*).
* Siekiama kad būtų peržiūrėta tiek pasakojimų, kad komanda turėtų darbo visą ateinantį sprintą.
* Esant poreikiui, vartotojo pasakojimai detalizuojami ir papildomi.
* Komanda įvertina ar įmanoma pasakojimą realizuoti bei įvertina reikalaujamo darbo apimtį (angl. *estimate*).
* Per dideli pasakojimai padalinami į mažesnius.
* Komanda užduoda klausimus ir teikia pasiūlymus kaip aiškiau ir tiksliau užrašyti vartotojo pasakojimus.

*5 užduotis.* APRAŠYKITE REIKALAVIMŲ PARUOŠIMO SPRINTUI KRITERIJUS (DEFINITION OF READY).

Definition of Ready tai sąlygos kurias privalo atitikti kokybiškai parašytas vartotojo pasakojimas ir jo priėmimo kriterijai. Pavyzdys**:**

* Vartotojo pasakojimas aprašyta pagal sutartą formatą (nurodomas konkretus sutartas formatas).
* Vartotojo pasakojimas peržiūrėtas ir reikalaujamo darbo apimtis įvertinta komandos.
* Visi su vartotojo pasakojimu susiję klausimai atsakyti ir pastebėti neatitikimai ištaisyti.
* Vartotojo pasakojimo apimties įvertinimas neviršija 8 pasakojimo vienetų.

*6 užduotis.* APRAŠYKITE REIKALAVIMŲ BAIGTUMO APIBRĖŽIMĄ.

*Definition of Done* tai sąlygos kada vartotojo pasakojimas gali būti laikomas užbaigtu. Vartotojo pasakojimas: „Kaip naudotojas aš noriu sukurti užduotį, tam kad nepamirščiau svarbių darbų.”. Priėmimo kriterijai (*Definition of Done*):

* Galiu sukurti užduotį tik tada, kai nurodau užduoties aprašymą.
* Užduoties aprašymas gali būti ne ilgesnis negu 300 simbolių.
* Galiu taip pat nurodyti užduoties baigimo datą.
* Datos formatas yyyy-mm-dd.
* Mano sukurtas priminimas matomas tik man.

*7 užduotis.* APRAŠYKITE PAGRINDINIUS SCRUM PROCESO ĮVYKIUS.

*Scrum* proceso įvykiai:

1. Išleidimo planavimo susitikimas (*Release Planning Meeting*) arba produkto darbų tvarkymo susitikimas (*Backlog Grooming Meeting*).

2. Sprinto planavimo susitikimas (*Sprint Planning Meeting*).

3. Kasdienis scrum susitkimas (*Daily Scrum*).

4. Sprinto peržiūra (*Sprint Review*).

5. Sprinto retrospektyva (*Sprint Retrospective*).

Aprašant kiekvieną įvykį naudokite šį šaltinį:

* <https://www.scrumguides.org/scrum-guide.html#events>

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

### Tema. Funkciniai reikalavimai.

*1 užduotis.* APRAŠYKITE KAS YRA FUNKCINIAI REIKALAVIMAI.

Funkciniai reikalavimai tai programinės įrangos ar sistemos atliekamų paslaugų aprašymas, sistemos reakcijos į tam tikrus įvedimo ar išvedimo duomenis aprašymas, sistemos elgesys tam tikroje situacijoje, programos reakcijos į tam tikrus vartotojo veiksmus. Pavyzdys: sistemoje galima pateikti įvairių tipų dokumentus.

Sistemos vartotojai gali pateikti dokumentus. Kiekvienas pateikiamas dokumentas yra tam tikro tipo. Dokumentų tipus kuria administratorius. Administratorius nustato, kokių tipų dokumentus vartotojų grupėms leidžiama kurti. Grupė gali turėti teises kuri neribotą kiekį skirtingų tipų dokumentų. Dokumentai pateikiami nurodant (ar sistemai generuojant) atitinkamas savybes. Vartotojas gali sukurti tam tikro tipo dokumentą, jei priklauso bent vienai grupei, kuriai leidžiama sukurti to tipo dokumentą. Pavyzdžiui, organizacijoje yra trys grupės: darbuotojai, konsultantai, administracija. Darbuotojams leidžiama pateikti atostogų prašymus, konsultantams leidžiama pateikti prašymus padengti komandiruotės išlaidas, administracija gali tvirtinti abiejų tipų dokumentus. Vartotojas priklausantis tiek darbuotojų, tiek konsultantų grupėms gali pateikti atostogų prašymus ir prašymus padengti komandiruotės išlaidas. Direktorius priklausantis visoms trims grupėms, gali teikti abiejų tipų dokumentus ir juos tvirtinti.

*2 užduotis.* PATEIKTAM PAVYZDŽIUI („FUNKCINIAI KOMPIUTERINĖS PROGRAMOS REIKALAVIMAI“ POTEMĖS UŽDUOČIAI) PARAŠYKITE, KAIP BUS NUSTATOMA AR FUNKCINIS REIKALAVIMAS BUVO ĮVYKDYTAS AR NE.

Aprašant reikalavimų įvykdymo būseną naudokite šiuos šaltinius:

* <https://en.wikipedia.org/wiki/Software_verification_and_validation>
* <https://en.wikipedia.org/wiki/Software_testing>

### Tema. Nefunkciniai reikalavimai.

*3 užduotis.* APRAŠYKITE KAS YRA NEFUNKCINIAI REIKALAVIMAI.

Nefunkciniai reikalavimai tai apribojimai sistemos atliekamoms paslaugoms ar funkcijoms, dažniausiai laiko apribojimai, programavimo proceso apribojimai, standartai ir pan.. Pavyzdys: visi veiksmai sistemoje turi atsakyti greičiau nei per 1 sekundę.

Reikalavimas taikomas laikantis prielaidos, kad sistemoje yra iki 10 tūkstančių vartotojų, kurių kiekvienas yra vidutiniškai pateikęs 20 dokumentų. Sistema yra pasiekiama internetu klientui turint ne platesnį nei 10Mbps ryšio kanalą bei vidutiniškai 50 ms atsako laiką iš serverio.

*4 užduotis.* PATEIKTAM PAVYZDŽIUI („NEFUNKCINIAI (TECHNINIAI, SAUGOS) KOMPIUTERINĖS PROGRAMOS REIKALAVIMAI „ POTEMĖS“ UŽDUOČIAI) PARAŠYKITE KAIP BUS NUSTATOMA AR NEFUNKCINIS REIKALAVIMAS BUVO ĮVYKDYTAS AR NE.

Aprašant reikalavimų įvykdymo būseną naudokite šiuos šaltinius:

* <https://en.wikipedia.org/wiki/Software_verification_and_validation>
* <https://en.wikipedia.org/wiki/Software_testing>

# Modulis „Nesudėtingų duomenų bazių projektavimas ir kūrimas (.NET)“

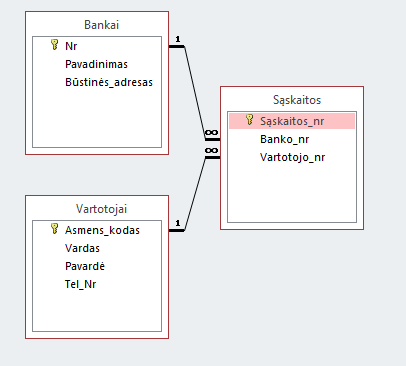
## Kompetencija. Projektuoti tipines reliacines ir objektines duomenų bazes.

### Mokymosi rezultatas. Pateikti reliacinės duomenų bazės schemą.

### 1.1.1. Tema. Duomenų bazių projektavimas (CREATE TABLE sakinys, duomenų normalizavimas).

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

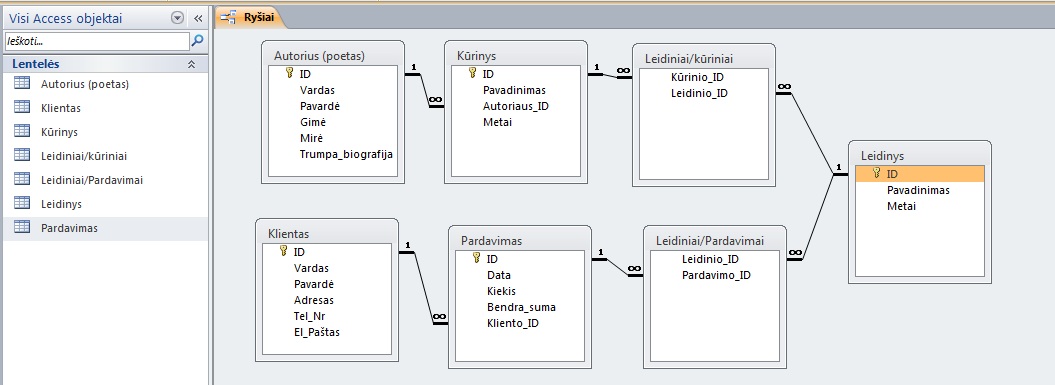
Schema:



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

#### 

Schema:



### Mokymosi rezultatas. Pateikti nereliacinės (NoSQL) duomenų bazės schemą.

### 1.2.1. Tema. NoSQL duomenų bazių tipai ir jų savybės.

#### *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 SCENARIJUI.

#### 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;

### Tema. NoSQL duomenų bazių valdymas.

#### *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

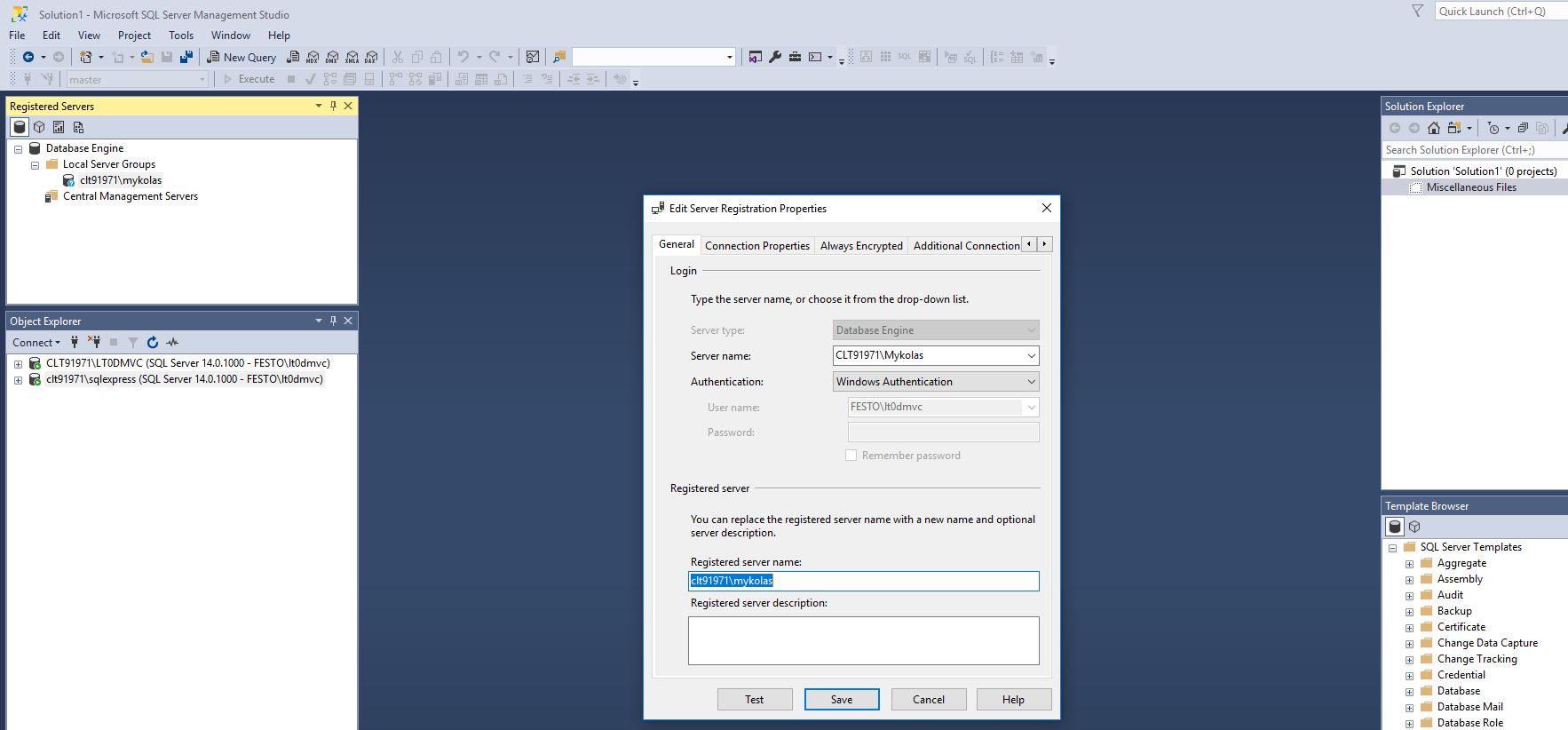
## 2. Kompetencija. Programiškai įgyvendinti ir administruoti duomenų bazes.

### Mokymosi rezultatas. Diegti ir valdyti duomenų bazių valdymo sistemą.

### Tema. DBVS diegimas.

*1 užduotis.* ĮDIEKITE IR SUKONFIGŪRUOKITE MS SQL SERVER EXPRESS EDITION IR SQL SERVER MANAGEMENT STUDIO (SSMS*).*

MS SQL Server ir SSMS yra instaliuoti, patikrinamas užduoties sąlygų įvykdymas: atidarius SQL Server Management Studio ir paspaudus „View registred services” pasirinkus jūsų aplinką dešinys pelės mygtukas ir “Properties” turi matytis Windows autentifikacija ir mokinio vardas DBVS pavadinime.



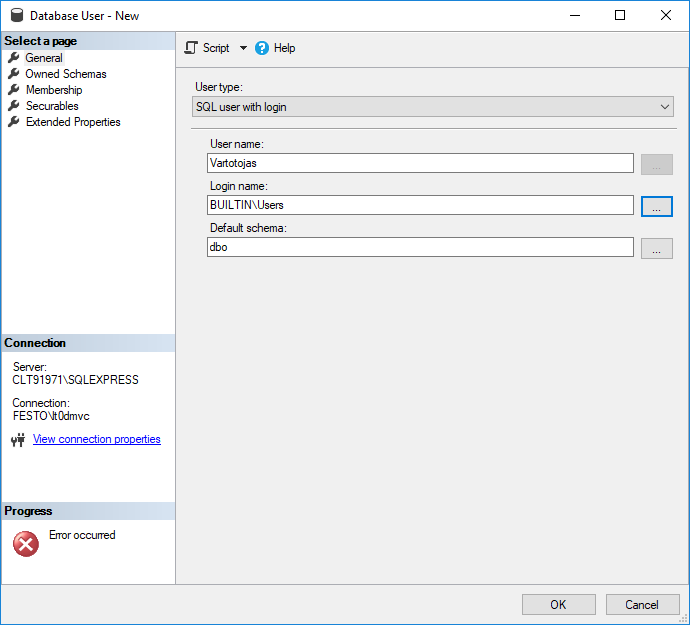
### Tema. DBVS administravimas.

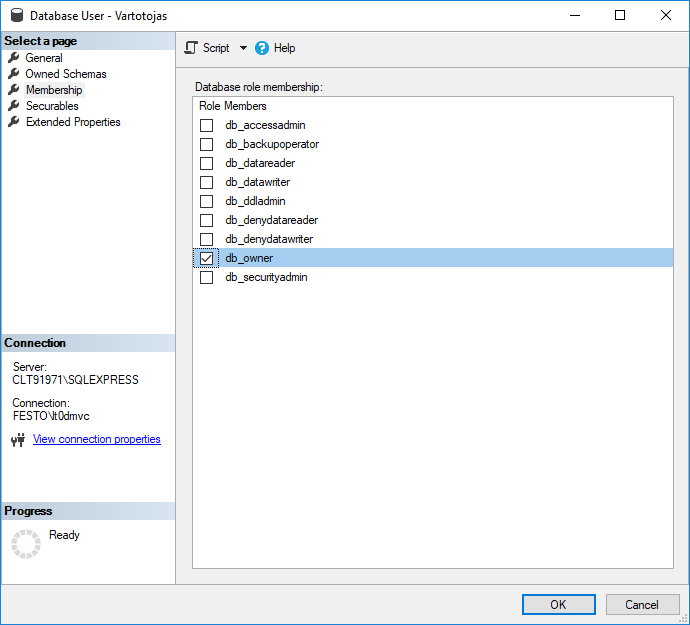
*2 užduotis.* SUKURKITE NAUJĄ SQL VARTOTOJĄ NAUDOJANT SQL SKRIPTĄ ARBA GRAFINĘ SSMS SĄSAJĄ IR SUTEIKITE JAM ADMINISTRATORIAUS TEISES.

SQL kodas:

|  |
| --- |
| CREATE LOGIN [computer\_name\Mary]  FROM WINDOWS  WITH DEFAULT\_DATABASE = [TestData];  GO  -----------------------------------------  USE [TestData];  GO  CREATE USER [Mary] FOR LOGIN [computer\_name\Mary];  GO |

Arba naudojant SSMS:





### 

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

### Tema. Duomenų išrinkimas naudojant SQL select sakinį ir pagrindinius select elementus.

#### *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’;

### Tema. Duomenų išrinkimas naudojant sąryšius (SQL select su join).

#### *2 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;

### Mokymosi rezultatas. Kurti duomenis duomenų bazėje valdančią programinę įrangą.

### 2.3.1. Tema. Duomenų bazių naudojimas programų sistemose naudojant ADO.NET sąsają.

*1 užduotis.* APIBŪDINKITE, KAS YRA ADO.NET IR KOKIAS PROGRAMAVIMO PRIEMONES JIS TEIKIA .NET KARKASE.

Užduoties sprendimui naudokite šiuos resursus:

* <https://docs.microsoft.com/en-us/dotnet/framework/data/adonet/ado-net-overview>

*2 užduotis.* SUKURKITE KONSOLINĘ PROGRAMĄ, KURI DUOMENŲ BAZĖS LENTELĖS DUOMENŲ IŠVEDIMUI NAUDOJA DATAREADER.

Programinis kodas:

|  |
| --- |
| using System.Data.SqlClient;  static void Main(string[] args)  {  SqlConnection connection = new SqlConnection(@"Data Source=(LocalDB)\MSSQLLocalDB;Initial Catalog=Test;Integrated Security=True;");  using (connection)  {  SqlCommand command = new SqlCommand(  "SELECT CategoryID, CategoryName FROM Categories;",  connection);  connection.Open();  SqlDataReader reader = command.ExecuteReader();  if (reader.HasRows)  {  while (reader.Read())  {  Console.WriteLine("{0}\t{1}", reader.GetInt32(0), reader.GetString(1));  }  }  else  {  Console.WriteLine("No rows found.");  }  reader.Close();  }  } |

*3 užduotis.* SUKURKITE KONSOLINĘ PROGRAMĄ, KURI ĮRAŠO DUOMENŲ EILUTĘ Į DUOMENŲ BAZĖS LENTELĘ IR GRĄŽINA ĮRAŠYTOS EILUTĖS NUMERĮ.

Db lentelė turi turėti auto-increment.

Programinis kodas:

|  |
| --- |
| using System.Data;  using System.Data.SqlClient;  static void Main(string[] args)  {  int newProdID = 0;  string sql = "INSERT INTO Production.ProductCategory (Name) VALUES (@Name); SELECT CAST(scope\_identity() AS int)";  using (SqlConnection conn = new SqlConnection(@"Data Source=(LocalDB)\MSSQLLocalDB;Initial Catalog=Test;Integrated Security=True;"))  {  SqlCommand cmd = new SqlCommand(sql, conn);  cmd.Parameters.Add("@Name", SqlDbType.VarChar);  cmd.Parameters["@name"].Value = "testas";  try  {  conn.Open();  newProdID = (int)cmd.ExecuteScalar();  }  catch (Exception ex) { Console.WriteLine(ex.Message); }  }  Console.Write(String.Format("Grąžinta reikšmė: {0}", (int)newProdID));  } |

*4 užduotis.* SUKURKITE KONSOLINĘ PROGRAMĄ, KURI DB LENTELĖS SUKŪRIMUI, PIRMŲ DUOMENŲ ĮVEDIMUI IR TŲ DUOMENŲ IŠVEDIMUI NAUDOJA EF6 KARKASĄ IR CODE FIRST METODĄ.

Programinis kodas:

|  |
| --- |
| public class Program  {  public static void Main()  {  using (var ctx = new SchoolContext())  {  var stud = new Student() { StudentName = "Bill" };  ctx.Students.Add(stud);  ctx.SaveChanges();  Console.Write("Student saved successfully!");  ctx.Students.ToList().ForEach(s =>  Console.WriteLine($"Vardas:{s.StudentName} Id: {s.StudentID}")  );  }  }  }  public class SchoolContext : DbContext  {  public SchoolContext() : base(@"Data Source = .\SQLEXPRESS; Initial Catalog = Test; Integrated Security = SSPI;")  {  }  public DbSet<Student> Students { get; set; }  public DbSet<Grade> Grades { get; set; }  }  public class Student  {  public int StudentID { get; set; }  public string StudentName { get; set; }  public DateTime? DateOfBirth { get; set; }  public byte[] Photo { get; set; }  public decimal Height { get; set; }  public float Weight { get; set; }  public Grade Grade { get; set; }  }  public class Grade  {  public int GradeId { get; set; }  public string GradeName { get; set; }  public string Section { get; set; }  public ICollection<Student> Students { get; set; }  } |

*5 užduotis.* SUKURKITE MIGRACIJĄ ANKSTESNĖS UŽDUOTIES SUKURTAI DUOMENŲ BAZEI: PRIDĖTI LENTELEI LAUKĄ (SUGALVOTI SAVARANKIŠKAI) IR ĮVYKDYTI SĖKMINGĄ MIGRACIJĄ.

Package Manager konsolėje įvykdyti:

|  |
| --- |
| PM> Enable-Migrations |

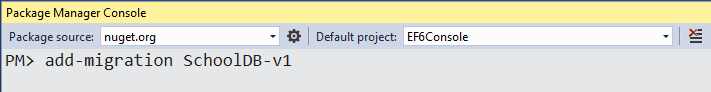
Programinis kodas:

Pataisyti *SchoolContext* metodą pridedant eilutę:

Database.SetInitializer(new MigrateDatabaseToLatestVersion<SchoolContext, ConsoleDB4\_4\_8.Migrations.Configuration>());

|  |
| --- |
| public class SchoolContext : DbContext  {  public SchoolContext() : base(@"Data Source = .\SQLEXPRESS; Initial Catalog = Test; Integrated Security = SSPI;")  {  Database.SetInitializer(new MigrateDatabaseToLatestVersion<SchoolContext, ConsoleDB4\_4\_8.Migrations.Configuration>());  }  public DbSet<Student> Students { get; set; }  public DbSet<Grade> Grades { get; set; }  } |

|  |
| --- |
| PM> Add-Migration |



Sukurti naują tuščia DB ir pataisyti *connection string*. Atnaujinti DB:

|  |
| --- |
| PM> Update-Database |

Pridedam naują lauką į Student entity:

public string StudentSurname { get; set; }

|  |
| --- |
| PM> Add-Migration SchoolDBv2 |

Atnaujinti DB:

|  |
| --- |
| PM> Update-Database |

# Modulis „Programavimo aplinkos ir kūrimo proceso valdymas“

## Kompetencija. Naudoti tarnybinių stočių operacines sistemas.

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

## 1.1.2. Tema. Serverio operacinės sistemos diegimas.

*1 užduotis.* WINDOWS SERVER 2016 DESKTOP EXPERIENCE DIEGIMAS IR KONFIGŪRAVIMAS.

Privalomi žingsniai:

* 1. Veiksmų eiga (žingsniai) - planavimas
  2. Tinklo nustatymai ir konfigūravimas
  3. Windows ugniasienės konfigūravimas
  4. Windows atnaujinimai
  5. Serverio vardo nustatymai
  6. IIS web serverio diegimas ir konfigūracija

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

## 1.2.1. Tema. Pagrindinės serverio administravimo funkcijos ir komandos.

1. *užduotis.* WINDOWS SERVER 2016 SERVER CORE DIEGIMAS IR KONFIGŪRACIJA (POWERSHELL KOMANDINĖS EILUTĖS PAGALBA).

Susipažinome su Windows server operacine sistema ir ją įdiegėme grafinio naudotojo sąsajos (GUI) pagalba. Sekantis žingsnis siekiant optimizuoti serverio resursus – įdiegti Windows Server Core leidimą. Server Core – minimalistinė versija, diegiama komandinės eilutės pagalba. Diegimo procesas toks pat kaip ir prieš tai diegto Standard/Datacenter leidimo su „desktop experience“ pasirinkimu. Visgi šis leidimas administravimo ir konfigūracijos prasme neturi grafinės naudotojo sąsajos. Konfigūracija turi būti atliekama CMD ir Powershell komandinių eilučių pagalba.

Veiksmų eiga (žingsniai), kaip turėtų būti siekiama rezultato:

* 1. Operacinės sistemos diegimas
  2. Administratoriaus paskyros slaptažodžio priskyrimas
  3. Sconfig pasirinkimų meniu
  4. Tinklo konfigūravimas powershell komandinės eilutės pagalba
  5. Tinklo konfigūravimas CMD komandinės eilutės pagalba
  6. Konfigūravimas: kompiuterio vardo keitimas powershell ar CMD komandinės eilutės pagalba
  7. Konfigūravimas: OS kintamieji powershell
  8. Procesų paleidimas

Toliau - IIS web serverio diegimas ir konfigūracija (žr. 1.4)

## Mokymosi rezultatas. Valdyti programinius paketus.

## 1.3.1. Tema. Programiniai paketai.

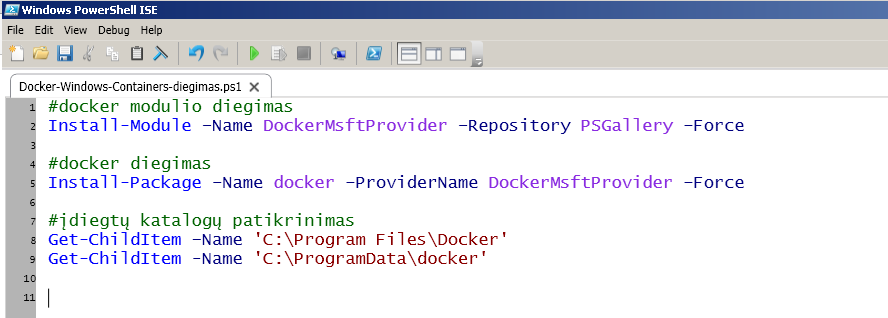
*1 užduotis.* ĮDIEGTI WINDOWS SERVER 2016 HYPER-V ROLĘ.

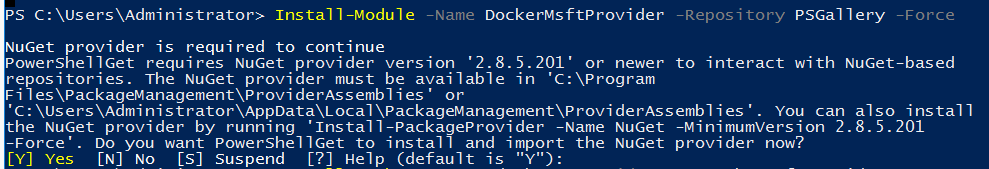
Naudojamos priemonės (aplinkos): Diegimo aplinkai patogiausia naudoti Windows operacinę sistemą su virtualizacijos platforma kaip, kad vmware workstation, vmware player ar virtual box Platformoje sudiegtas ir paruoštas Windows Server 2016 su hyper-v funkcionalumu. Naudojamas hyper-v manager.

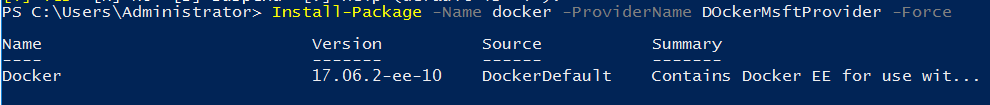
*2 užduotis.* WINDOWS CONTAINERS / DOCKER DIEGIMAS IR KONFIGŪRACIJA.

Diegimo aplinkai patogiausia naudoti Windows operacinę sistemą su virtualizacijos platforma. Tam tinka vmware workstation, vmware player ar virtual box Platformoje sudiegtas ir paruoštas Windows Server 2016 su hyper-v funkcionalumu.

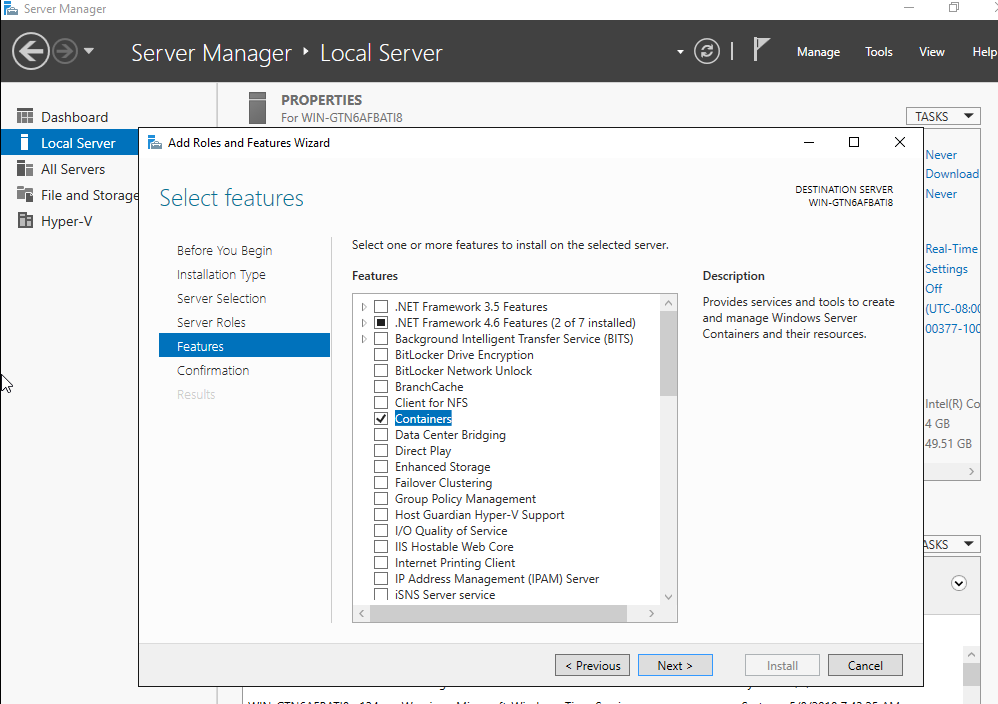
* 1. Windows Containers / Docker diegimas powershell pagalba:



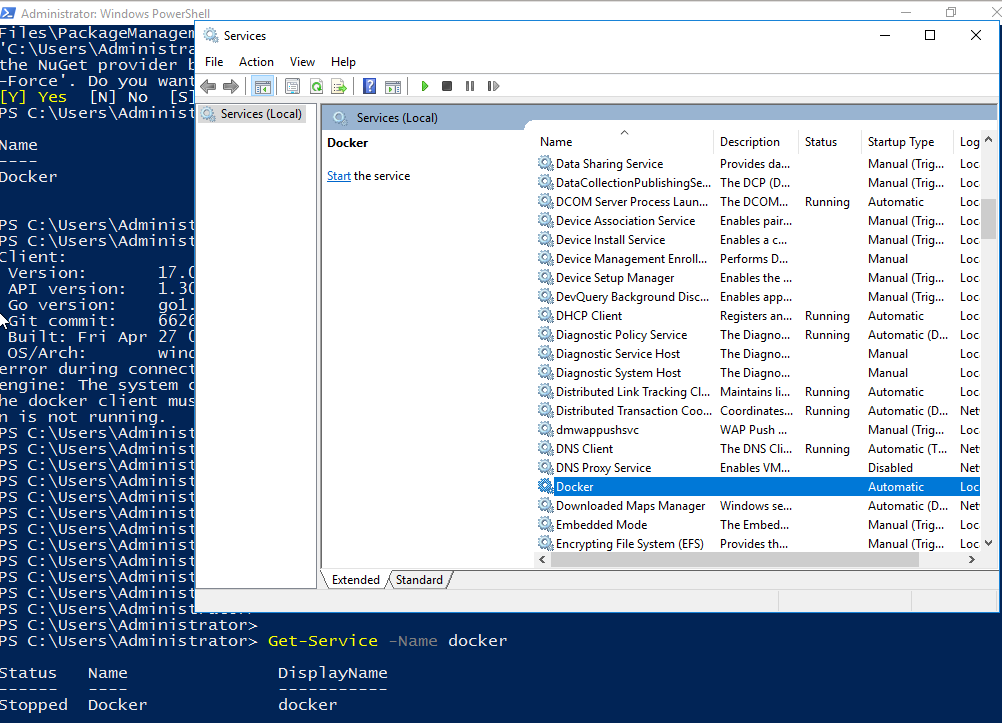




* 1. Alternatyva - Windows Containers / Docker diegimas per GUI. „Server Manager“ meniu pagalba įdiekime naują funkciją : Containers



* 1. Po diegimo serverį reikia perkrauti



Rezultatas: Galima paleisti, sustabdyti ir valdyti Windows konteinerius.

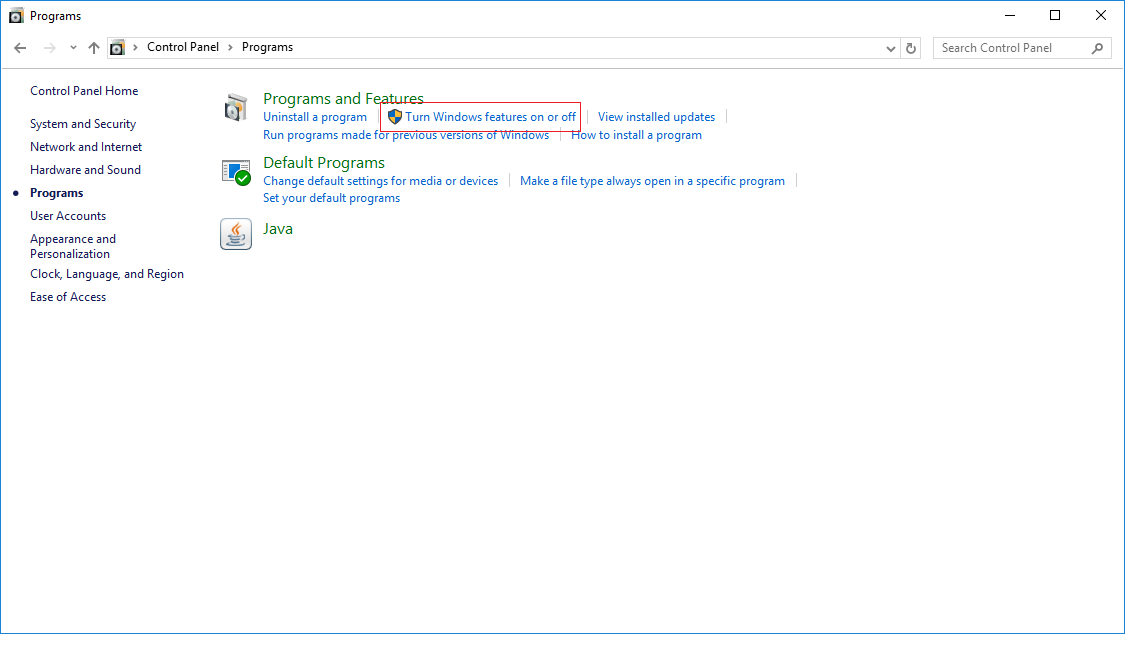
## Mokymosi rezultatas. Naudoti Web serverio programinę įrangą HTTP bylų viešinimui.

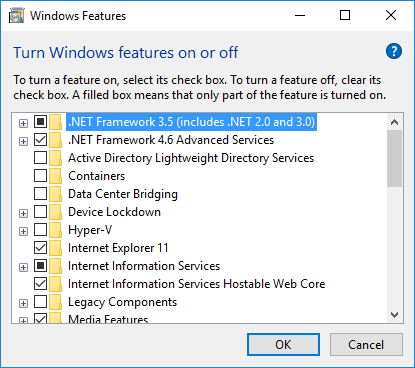
## 1.4.1. Tema. Web serverio aplinka operacinėje sistemoje.

*1 užduotis.* ASMENINIAME KOMPIUTERYJE INSTALIUOKITE IR SUKONFIGŪRUOKITE INTERNET INFORMATION SERVICES (IIS) WEB PROGRAMINĘ ĮRANGĄ.

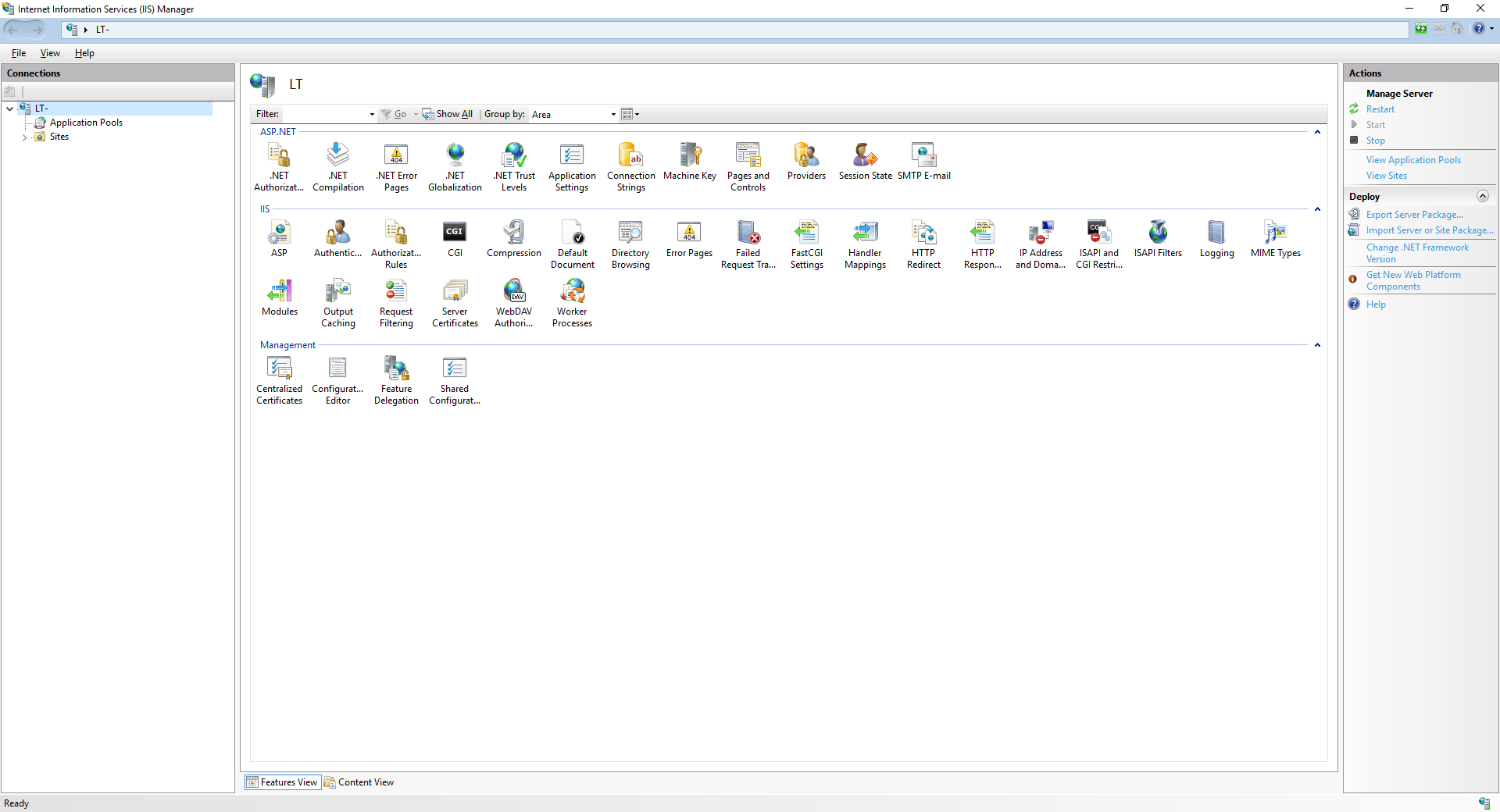
Sprendimas:

1. Iš „Control Panel“ spausti „Programs“, tada spausti „Turn Windows features on or off“



1. Parinkti „Internet Information Services (IIS)“
2. 
3. Spausti Ok
4. Patikrinimui ar veikia paspausti Windows klavišą ir paieškoje įvesti „IIS“. Jei randamas rezultatas, vadinasi serveris buvo sėkmingai instaliuotas.
5. Paspaudus ant paieškos rezultato, turi atsidaryti IIS programinė įranga.

Pavyzdys:



*2 užduotis.* IIS APLIKACIJOS ERDVIŲ KONFIGŪRAVIMAS.

IIS programinės įrangos pagalba sukurti aplikacijos erdves:

1. Application pool
   1. Suteikti savo pavadinimą, be .Net karkaso versijos ir „Classic“ pipeline režimą

Užduotis atlikta teisingai, jei IIS ‚Application Pools“ skiltyje matyti sukurtas Application Pool.

1. Tinklapis (angl. Site)
   1. C:\inetpub direktorijoje sukurti katalogą „aplikacija“. Kataloge patalpinti index.html failą su taisyklingu html turiniu, kuris bus paviešintas tinklapio pagalba.
   2. Sukurti naują tinklapį, priskirti savo sukurtą application pool ir nurodyti kelią į naujai sukurtą katalogą.
   3. Per naršyklę naviguoti iki tinklapio adreso, kuris buvo priskirtas.

Užduotis atlikta teisingai, jei nurodytu adresu per naršyklę matyti index.html failo turinys.

1. Virtuali direktorija (angl. Virtual Directory)
   1. Susikurti naują katalogą „images“ c:\intepub kataloge.
   2. Sukurti virtualią direktoriją ir paviešinti sukurtą katalogą.
   3. Sukurtame kataloge patalpinti paveiksliukus.
   4. Per naršyklę pasitikrinti ar paveiksliukai pasiekiami per priskirtą adresą.

Užduotis atlikta teisingai, jei per IIS yra matoma sukurta virtuali direktorija ir per naršyklę yra matomas paveiksliukas.

1. Aplikacija (angl. Application)
   1. C:\ inetpub direktorijoje susikurti naują katalogą ir jame patalpinti index.html bylą su html turiniu.
   2. Sukurti aplikaciją savo tinklapyje ir nurodyti sukurto katalogo direktoriją.
   3. Per naršyklę įjungti aplikaciją.
   4. Pakoreguoti html bylą taip, kad joje būtų bent vienas paveiksliukas iš anksčiau sukurtos virtualios direktorijos. Per naršyklę pasitikrinti ar paveiksliukai pasiekiami per priskirtą adresą.

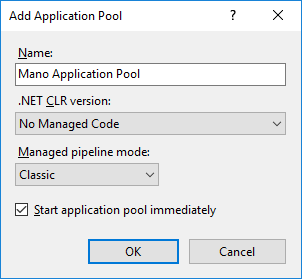
Užduotis atlikta teisingai, jei per naršyklę yra pasiekiama aplikacija ir matomas bent vienas paveiksliukas iš virtualios direktorijos.

1. Programavimo aplinka Visual Studio ir IIS
   1. Sukurti naują .Net karkaso web projektą ir pakeisti nustatymus, kad projektas veiktų per lokalų IIS serverį.
   2. Sukurti virtualią direktoriją per Visual Studio

Užduotis atlikta teisingai, jei web projektas paspaudus “Run”pasileidžia per IIS ne per IIS Express. Ir taip pat, jei tinklapyje “ Default Web Site” yra sukurta aplikacija.

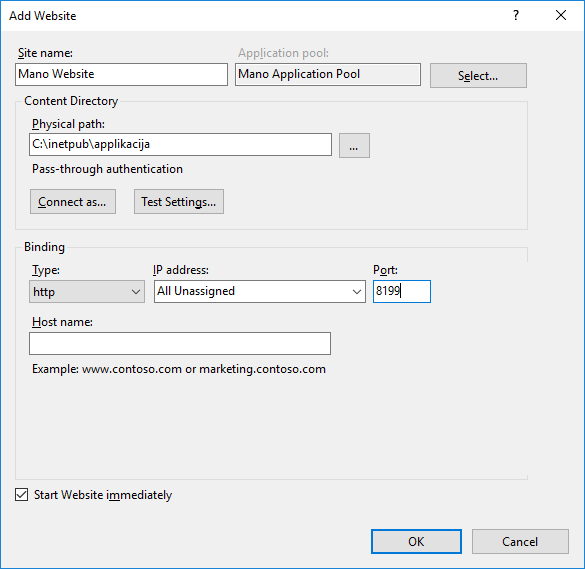
Veiksmų eiga:

1. Įsijungti IIS programinę įrangą. „Application Pools“ skiltyje paspausti „Add application Pool...“. Užpildyti atsidariusį langą su duomenimis (Pav.1) ir spausti Ok.

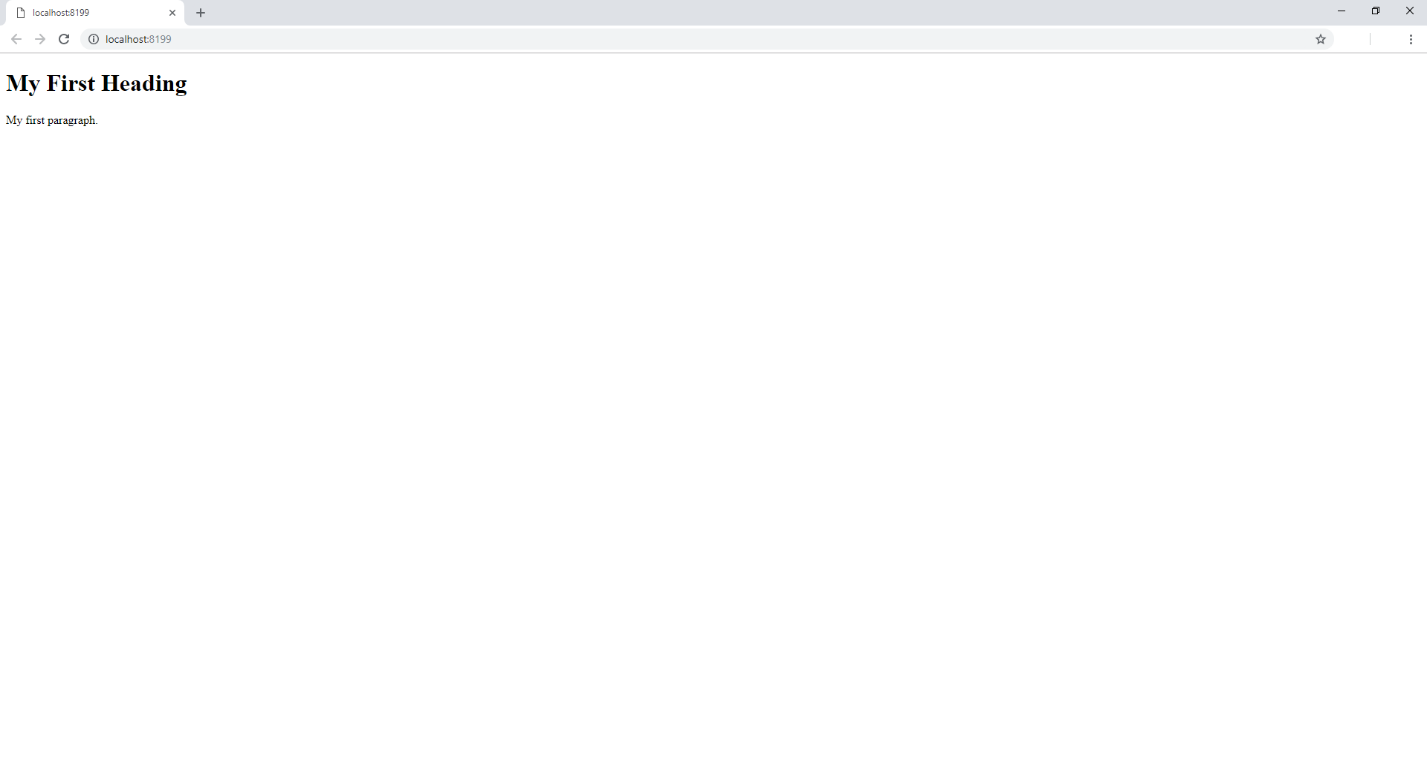


**Pav. 1**

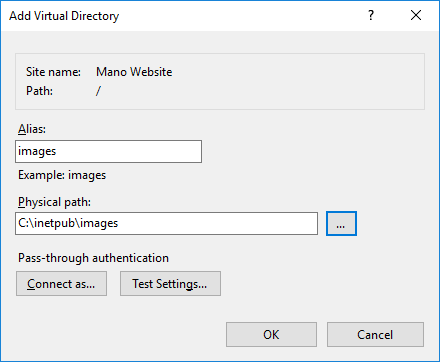
Sukurtas rezultatas turi atsirasti „Application Pools“ skiltyje.

1. Veiksmai:
   1. Sukurti katalogą „aplikacija“ C:\inetpub direktorijoje ir joje patalpinti, bet kokią html bylą, kurios pavadinimas būtų index.html.
   2. IIS skiltyje sites spausti „Add Website...“ ir užpildyti langą su žemiau pateiktu pav.

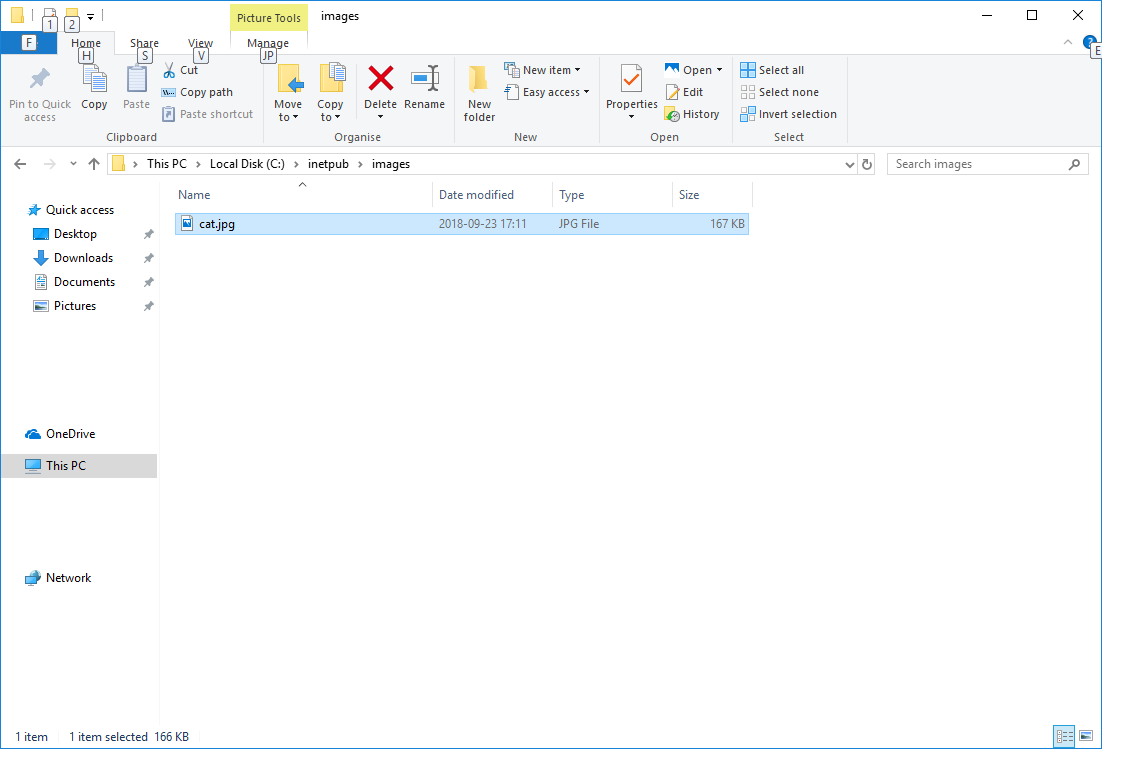
**Pav. 2**

* 1. Per naršyklę patikrinti ar byla paviešinta (Pav. 3). Pagal užpildytus duomenis bylos adresas yra htttp://localhost:8199/aplikacija

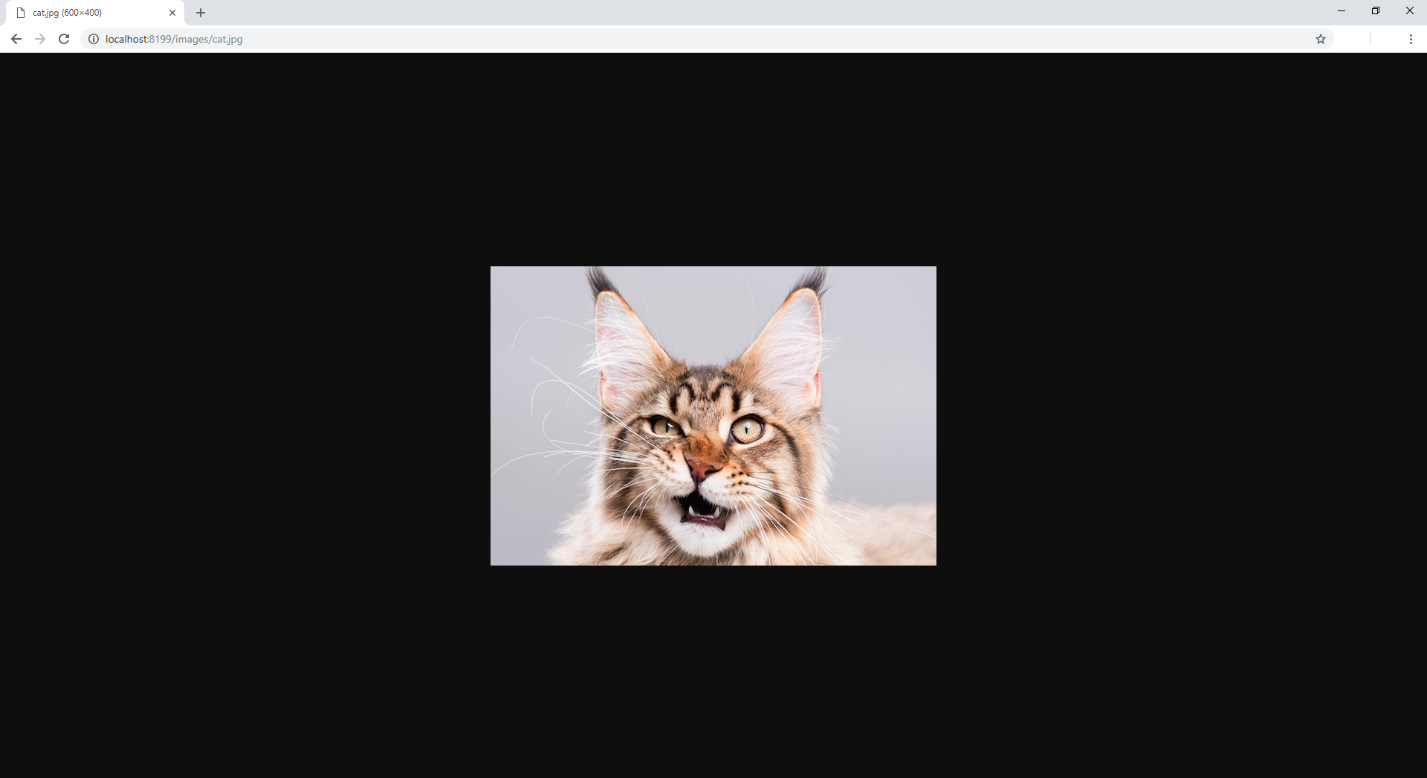
**Pav. 3**

1. Veiksmai:
   1. Susikurti naują katalogą „images“ c:\intepub kataloge.
   2. Anksčiau sukurtame tinklapyje, sukurti virtualią direktoriją pagal pavyzdį (Pav. 4). Paspausti dešinį pelės klavišą ant tinklapio ir pasirinkti „Add Virtual Directory“ 

**Pav. 4**

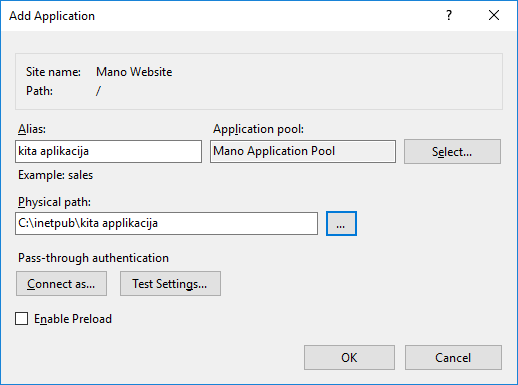
* 1. Į katalogą įkelti paveikslėlius (Pav. 5).

**Pav. 5**

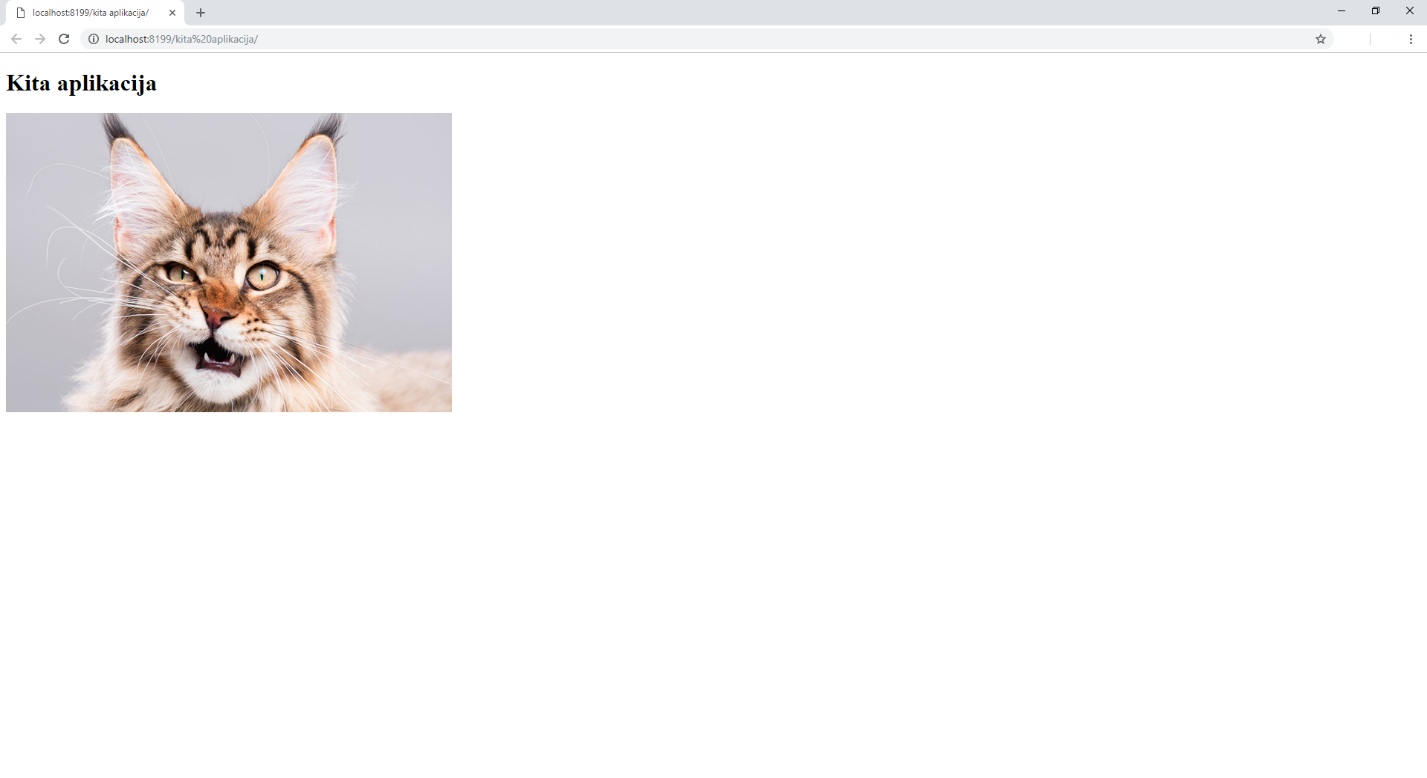
* 1. Jeigu viskas teisingai sukonfigūruota, tai paveikslėliai bus pasiekiami per virtualios direktorijos adresą ir pridėjus resurso direktoriją. Šiuo atveju paveikslėlio pavadinimas (Pav. 6). htttp://localhost:8199/images/cat.jpg

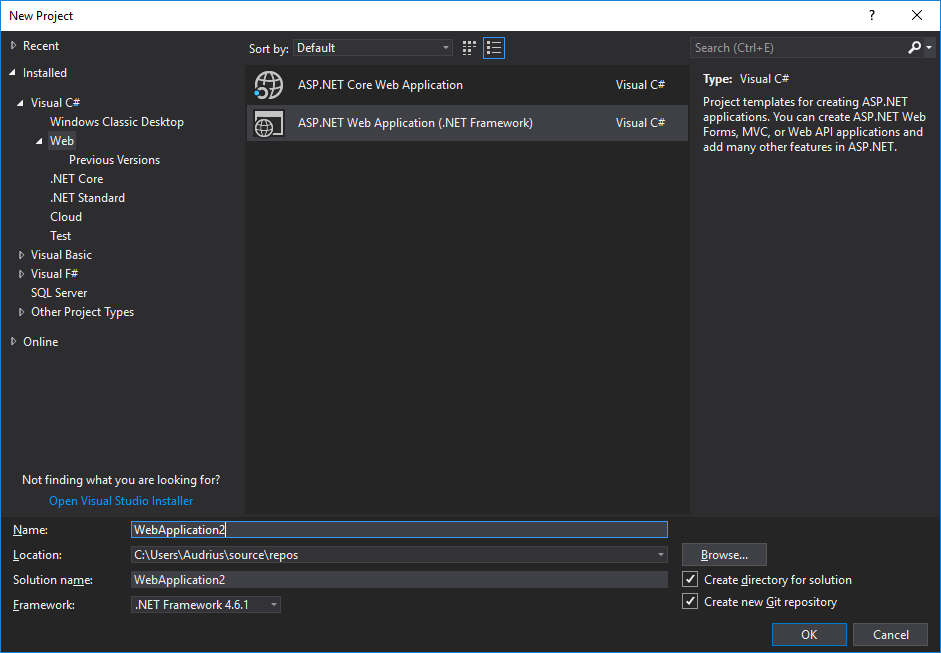
**Pav. 6**

1. Veiksmai:
   1. C:\ inetpub direktorijoje sukurti naują katalogą pvz. „kita aplikacija“. Šiame kataloge patalpinti html turinį. Taip pat privalomas pradinis failas index.html
   2. Anksčiau sukurtame tinklapyje sukurti naują aplikaciją. Paspausti dešinį pelės klavišą ant tinklapio ir pasirinkti „Add Application“. Užpildyti atsidariusį langą pagal pavyzdį (Pav. 7).

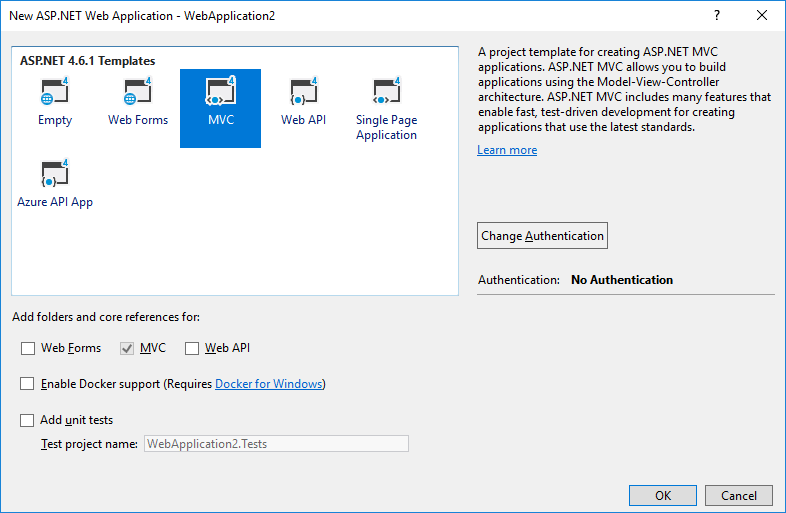


**Pav. 7**

* 1. Aplikacija turi būti paviešinta adresu <http://localhost:8199/kita%20applikacija>
  2. C:\intetpub\kita aplikacija index.html byla, kad galėtų naudoti paveiksliuką iš virtualios direktorijos reikia pridėti paveikslėlio nuorodą <img src=”../images/cat.jpg”> (Pav 8.)
  3. Pav 8.

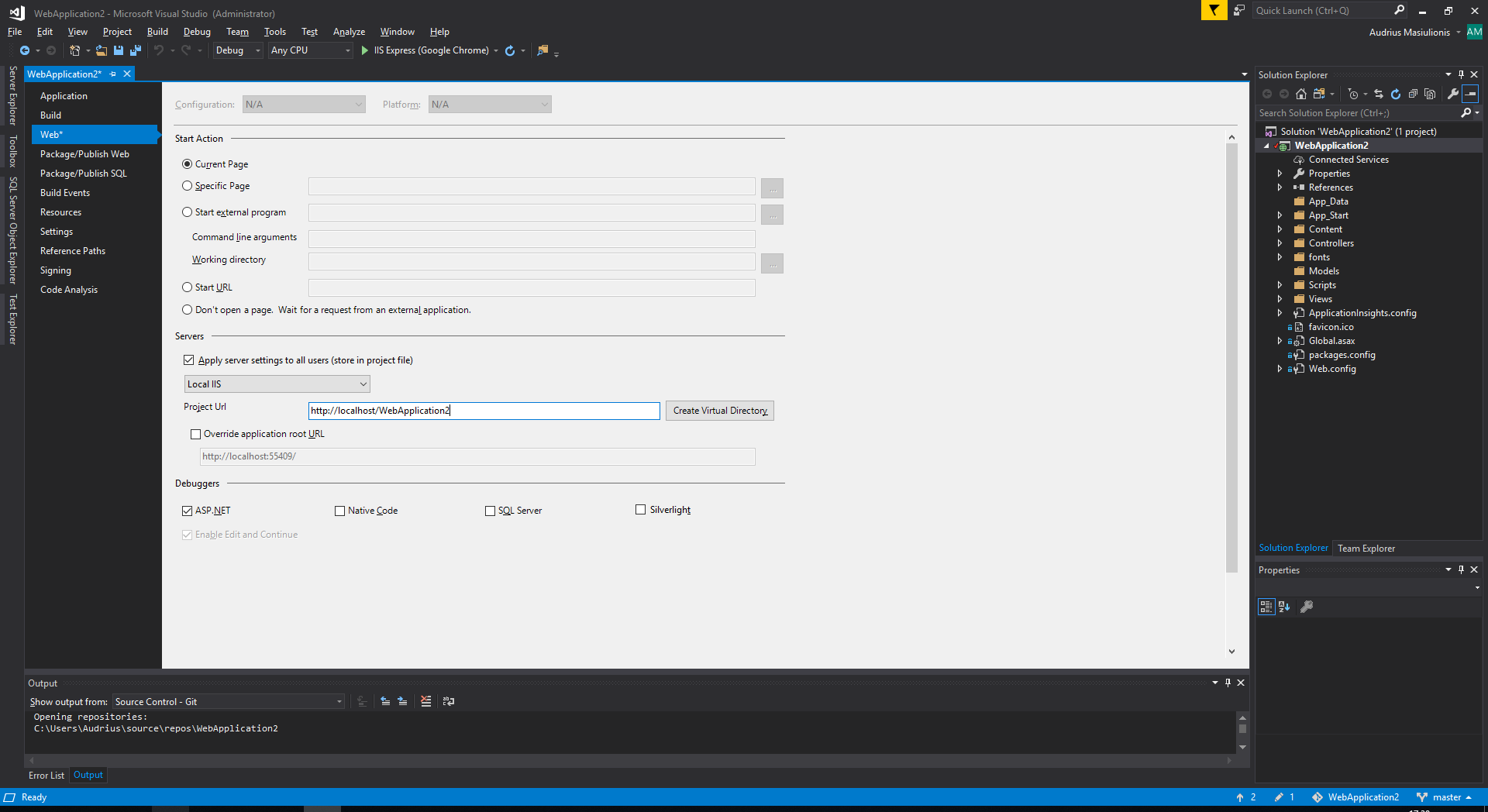
1. Veiksmai:
   1. Pagal pateiktą pavyzdį sukurti web projektą Visual Studio programavimo aplinkoje (Pav. 9 ir pav. 10).

**Pav 9.**

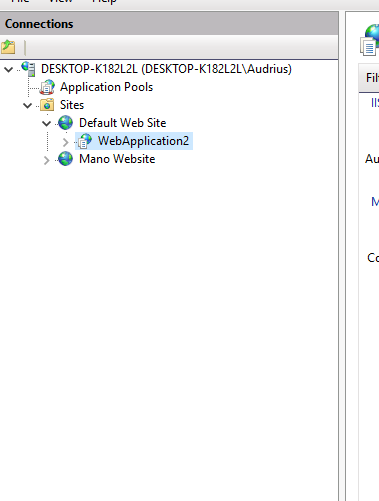


**Pav. 10**

* 1. Įjungti projekto nustatymus „Properties“. Įjungus “Web” skiltį padaryti pakeitimus pagal pavyzdį (Pav. 11). Patikrinti IIS programinėje įrangoje, ar susikūrė aplikacija (Pav 12).



Pav. 11



**Pav. 12**

## Kompetencija. Taikyti aktualias programinės įrangos kūrimo metodikas.

## Mokymosi rezultatas. Suprasti SCRUM proceso dalis ir komandos narių atsakomybes.

## 2.1.1. Tema. Scrum procesas.

*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 pirmajam.

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

(4) 7 plius ar minus 2

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

(3) 15 minučių

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

Komandos, kurios turėtų susiderinti tarpusavyje.

## Mokymosi rezultatas. Analizuoti pateiktus reikalavimus, nustatant programos atitikimą reikalavimams.

## 2.2.1. Tema. Programinės įrangos reikalavimų analizė.

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

(2) Klientas išsako savo lūkesčius projekte

*2 užduotis.* REIKALAVIMŲ ANALIZĖ NEAPIMA JŲ ATSEKAMUMO.

(2) Ne, apima.

(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*. REIKALAVIMŲ ANALIZĖ YRA ITERATYVUS PROCESAS.

* + 1. Taip, yra.

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

(4) Svarbiausi darbai sąrašo pradžioje, mažiausiai svarbūs gale.

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

(2) Gerai suformuluotas. Reiktų paaiškinimo, ką reiškia „greitai ir lengvai“ kad būtų galima testuoti vartotojo sąsają.

## Mokymosi rezultatas. Naudoti projekto eigos valdymo principus.

## 2.3.1. Tema. Projekto eigos valdymas.

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

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

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

1. Kiekvieno sprinto metu.

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

(1) Testuoti produktą

ir

1. 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?

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

*5 užduotis.* KADA BAIGIAMAS SPRINTO VYKDYMAS?

(4) Kai baigiasi sprintui skirtas laikas (time-box expires)

## Kompetencija. Valdyti savo paties ir komandos atliekamą programinio kodo kūrimą.

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

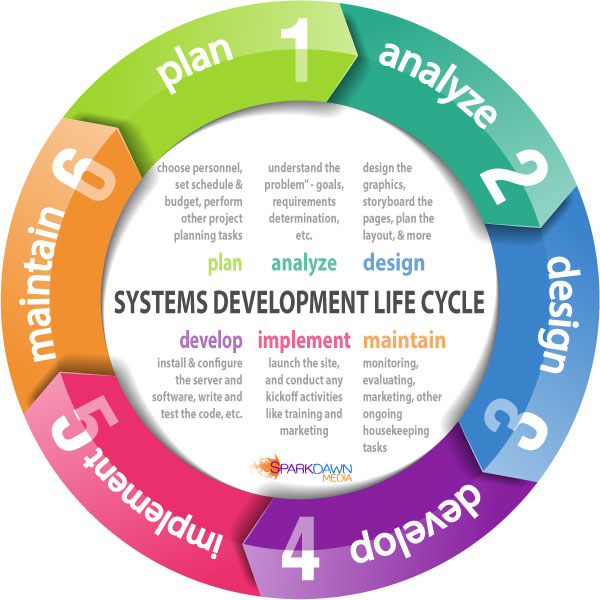
### 3.2.1. Tema. Komandinio darbų planavimo sistemos.

*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:

1. 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.

1. **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.
2. **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.
3. **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.
4. **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:

Atlassian (2019). *Atlassian Documentation.* [žiūrėta 2019-05-27]. Prieiga per internetą <https://confluence.atlassian.com/alldoc/atlassian-documentation-32243719.html.>

### Tema. Programavimo darbų vykdymo sekimas.

*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.>

### 3.3. Mokymosi rezultatas. Vykdyti programinio kodo versijavimą naudojant programinio kodo versijavimo įrankius, tinkamus C# programavimo kalbai.

### 3.3.1. Tema. Išeities kodo saugyklos.

#### *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:** jus 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.>

### 3.3.2. Tema. Programinio kodo versijavimo vykdymas.

#### *9 užduotis.* NUKOPIJUOKITE KODO PAVYZDĮ IŠ NUOTOLINĖS GITHUB KODO SAUGYKLOS*:* https://github.com/PacktPublishing/.NET-Design-Patterns.

#### Kodo kopijavimas iš nuotolinės saugyklos:

1. git clone https://github.com/PacktPublishing/.NET-Design-Patterns

Redaguojame failą README.md (projekto aprašymas). Įrašome į šį failą eilutė: „This is my forked project.“. Pakeitimus įrašome į kodo saugyklą:

2. git add README.

3. 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 projektą su atitinkama klase Calculator.java. Parašome skaičiuotuvo aritmetines operacijas. Kiekvieną parašytą aritmetinį veiksmą įrašome į kodo saugyklą.

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. b

2. c

3. c

4. b

5. a

6. b

7. a

8. a, c, d, f

9. b

10. a

11. d

12. b

13. b

14. c

15. b

16. d

17. b

18. a

19. a

20. a

21. c

22. c

23. d

24. b

25. a

26. b

27. a

28. c

29. a

# Literatūros sąrašas

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 internetąhttps://www.atlassian.com/software/jira/guides/getting-started/best-practices

3. Atlassian (2019). *Atlassian Documentation.* [žiūrėta 2019-05-27].Prieiga per internetąhttps://confluence.atlassian.com/alldoc/atlassian-documentation-32243719.html

4. Atlassian (2019). *Create a project.* [žiūrėta 2019-05-27].Prieiga per internetąhttps://confluence.atlassian.com/get-started-with-jira-core/create-a-project-917965385.html

5. Atlassian (2019). *Creating issues and sub-tasks.* [žiūrėta 2019-05-27].Prieiga per internetąhttps://confluence.atlassian.com/jiracoreserver073/creating-issues-and-sub-tasks-861257329.html

6. Atlassian (2019). *Working with boards.* [žiūrėta 2019-05-27].Prieiga per internetąhttps://confluence.atlassian.com/jiracorecloud/working-with-boards-800712866.html

7. Atlassian (2019). *Working with issues.* [žiūrėta 2019-05-27]. Prieiga per internetąhttps://confluence.atlassian.com/jirasoftwarecloud/working-with-issues-764478424.html

8. Atlassian (2019). *Working with issues in Jira Software*. [žiūrėta 2019-05-27]. Prieiga per internetą https://www.atlassian.com/agile/tutorials/issues

9. Atlassian (2019). *Issue types.* [žiūrėta 2019-05-27]. Prieiga per internetą <https://confluence.atlassian.com/adminjiracloud/issue-types-844500742.html>

10. 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

11. Atlassian (2019). *Deploying a release.* [žiūrėta 2019-05-27]. Prieiga per internetą https://confluence.atlassian.com/jirasoftwarecloud/deploying-a-release-764478183.html

12.Atlassian (2019). *Basic searching*. [žiūrėta 2019-05-27]. Prieiga per internetą https://confluence.atlassian.com/jirasoftwarecloud/basic-searching-764478306.htm

13. Atlassian (2019). *Searching for issues*. [žiūrėta 2019-05-27]. Prieiga per internetą https://confluence.atlassian.com/jirasoftwarecloud/searching-for-issues-764478280.html

14. Atlassian (2019). *Advanced searching*. [žiūrėta 2019-05-27]. Prieiga per internetą https://confluence.atlassian.com/jirasoftwareserver073/advanced-searching-861256227.html

15. Albahari, J. *C# 7.0 in a Nutshell: The Definitive Reference.* Prieiga per internetą https://www.amazon.com/gp/product/1491987650

16. Blackwasp. Prieiga per internetą <http://www.blackwasp.co.uk/SOLIDPrinciples.aspx>

17. Chan, J. *C#: Learn C# in One Day and Learn It Well. C# for Beginners with Hands-on Project.* Prieiga per internetą https://www.amazon.com/Beginners-Hands-Project-Coding-Project-ebook/dp/B016Z18MLG/

18. Cleary, S. *Concurrency in C# Cookbook: Asynchronous, Parallel, and Multithreaded Programming* Prieiga per internetą https://www.amazon.com/gp/product/1449367569

19. Data & Object Factory(2019)*.* .*NET Design Patterns.* [žiūrėta 2019-05-27]. Prieiga per internetą <https://www.dofactory.com/net/design-patterns>

20. Esposito, D. *Microsoft .NET - Architecting Applications for the Enterprise* . Prieiga per internetą <https://www.amazon.com/Microsoft-NET-Architecting-Applications-Enterprise/dp/0735685355>

21. GbmbOrg (2019). *Data Units Conversion.* [žiūrėta 2019-05-27]. Prieiga per internetą https://www.gbmb.org.

22. Github .Prieiga per internetą <https://github.com/facebook/react>.

23. Github. Prieiga per internetą <https://github.com/PacktPublishing/.NET-Design-Patterns>.

24. Git (2019). *Downloads.* [žiūrėta 2019-05-27]. Prieiga per internetą https://git-scm.com/downloads

25. Git (2019). *Getting Started - Installing Git*. [žiūrėta 2019-05-27]. Prieiga per internetą https://git-scm.com/book/en/v2/Getting-Started-Installing-Git

26. 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

27. Git (2019). *Customizing Git - Git Configuration*. [žiūrėta 2019-05-27].

Prieiga per internetą https://git-scm.com/book/en/v2/Customizing-Git-Git-Configuration

28. 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

29. 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.

30. Git (2019). *Git format patch*. [žiūrėta 2019-05-27]. Prieiga per internetą https://git- scm.com/docs/git-format-patch

31. Google / Amazon. *Web services laboratorijos* Prieiga per internetą https://qwiklabs.com/

32. HowToDoInJava (2019). *Java Factory Pattern Explained.* [žiūrėta 2019-05-27]. Prieiga per internetą https://howtodoinjava.com/design-patterns/creational/implementing-factory-design-pattern-in-java

33. HowToDoInJava (2019). *Decorator Design Pattern in Java*. [žiūrėta 2019-05-27]. Prieiga per internetą https://howtodoinjava.com/design-patterns/structural/decorator-design-pattern

34. 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

35. Japikse, P. *C# 6.0 and the .NET 4.6 Framework 7th Edition by Andrew Troelsen*. Prieiga per internetą https://www.amazon.com/gp/product/1484213335

36. Java Guides (2019). *Spring MVC Tutorials and Articles*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://www.javaguides.net/2018/10/spring-mvc-sign-up-form-handling.html.>

37. JGraph Ltd (2019). *draw.io*. [žiūrėta 2019-05-27]. Prieiga per internetą https://www.draw.io.

Lietuvos statistikos departamentas (2019). *Ekonominės veiklos rūšių klasifikatorius*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://osp.stat.gov.lt/static/evrk2.htm>

38. Katacoda. *Docker kursas.* Prieiga per internetą https://www.katacoda.com/courses/docker

39.Kubernetes.*Susipažinimas su Kubernetes.* Prieiga per internetą

https://kubernetes.io/docs/tutorials/kubernetes-basics/create-cluster/cluster-interactive/

40. Lestard. Prieiga per internetą <https://lestard.eu/2014/easy-di/>

41. Learnrazorpages. Prieiga per internetą <https://www.learnrazorpages.com/>

42. Lietuvos statistikos departamentas (2019). *Ekonominės veiklos rūšių klasifikatorius*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://osp.stat.gov.lt/static/evrk2.htm>

43. Martin, C. R., ir Martin, M. ( 2006). *Agile Principles, Patterns, and Practices in C#.* Prieiga per internetą <https://www.amazon.co.uk/dp/0131857258>

44. Martinfowler. Prieiga per internetą <https://www.martinfowler.com/eaaDev/uiArchs.html#ModelViewController>

45. Martinfowler. Prieiga per internetą https://martinfowler.com/articles/richardsonMaturityModel.html

46. Maskeliūnas, S. (2007). *Programų sistemų architektūra ir projektavimas.* Mokymo medžiaga parengta vykdant projektą „Programų sistemų magistrantūros įsteigimas“. [žiūrėta 2019-05-27]. Prieiga per internetą <https://klevas.mif.vu.lt/~donatas/PSArchitekturaProjektavimas/Knyga/BPD/PSAPKnyga.pdf>

47. McLean Hall, G., *Adaptive Code:* *Agile coding with design patterns and SOLID principles (Developer Best Practices)*. Prieiga per internetą <https://www.amazon.com/Adaptive-Code-principles-Developer-Practices-ebook/dp/B071YC5ML9>

48. McConnell, S. *Code Complete*:*A Practical Handbook of Software Construction.* Prieiga per internetą https://www.amazon.com/gp/product/1449367569

49. Microsoft.*Powershell.* Prieiga per internetą

https://mva.microsoft.com/en-US/training-courses/getting-started-with-microsoft-powershell-8276

50. Microsoft. *Windows Server administravimas.* Prieiga per internetą

https://mva.microsoft.com/en-US/training-courses/windows-server-administration-fundamentals-8477

51. Microsoft.*Windows Server Active Directory*.Prieiga per internetą

https://mva.microsoft.com/en-us/training-courses/understanding-active-directory-8233

52. Microsoft.*Windows Server virtualizacija.* Prieiga per internetą

https://mva.microsoft.com/en-US/training-courses/windows-server-2016-deep-dive-virtualization-14094

53. Microsoft.*Windows server tinklai*. Prieiga per internetą <https://mva.microsoft.com/en-US/training-courses/networking-fundamentals-8249>

54. Microsoft. Prieiga per internetą <https://www.edx.org/xseries/microsoft-windows-server-2016>

55. Microsoft Prieiga per internetą <https://info.microsoft.com/TheUltimateGuideToWindowsServer2016.html?ls=Website>

56. Microsoft. Prieiga per internetą https://docs.microsoft.com/en-us/aspnet/web-pages/overview/getting-started/introducing-razor-syntax-c

57. Microsoft. Prieiga per internetą <https://docs.microsoft.com/en-us/aspnet/core/mvc/views/razor>

58. Microsoft. Prieiga per internetą https://docs.microsoft.com/en-us/aspnet/core/razor-pages/

59. Microsoft Prieiga per internetą.https://docs.microsoft.com/en-us/aspnet/core/tutorials/razor-pages/razor-pages-start

60. Microsoft. Prieiga per internetą https://docs.microsoft.com/en-us/aspnet/core/fundamentals/middleware/

61. Microsoft. Prieiga per internetą https://docs.microsoft.com/en-us/aspnet/core/fundamentals/middleware/write

62. Microsoft Prieiga per internetą.<https://www.tutorialspoint.com/asp.net_core/asp.net_core_middleware.htm>

63. Microsoft. Prieiga per internetą https://docs.microsoft.com/en-us/aspnet/core/fundamentals/startup

Prieiga per internetą https://www.c-sharpcorner.com/article/understanding-startup-class-in-asp-net-core/

64. Microsoft. Prieiga per internetą <https://docs.microsoft.com/en-us/visualstudio/get-started/csharp/tutorial-debugger>

65. Nagel, Ch.*Professional C# 7 and .NET Core 2.0* 7th Edition by Christian Nagel. Prieiga per internetą https://www.amazon.com/Professional-NET-Core-Christian-Nagel/dp/1119449278/

66. Oracle. Prieiga per internetą <https://docs.oracle.com/javase/tutorial/java/javaOO/lambdaexpressions.html.>

67. Osherove, R.*The Art of Unit Testing: with examples in C# .*

https://www.amazon.com/gp/product/1617290890

68. Price, M. J. (2017) *C# 7 and .NET Core: Modern Cross-Platform Development: Create powerful cross-platform applications using C# 7, .NET Core, and Visual Studio 2017 or Visual Studio Code.*

Prieiga per internetą https://www.amazon.com/gp/product/1484213335

69. Restcookbook. Prieiga per internetą <http://restcookbook.com/Miscellaneous/richardsonmaturitymodel/>

70. Restfulapi. Prieiga per internetą https://restfulapi.net/

71.Restapitutorial. Prieiga per internetą https://www.restapitutorial.com/

72. Richter, J. *CLR via C# (Developer Reference).* Prieiga per internetą https://www.amazon.com/CLR-via-4th-Developer-Reference/dp/0735667454

73. Skeet,J. *C# in Depth.* Prieiga per internetą <https://csharpindepth.com/>

74. Srcumguides. Prieiga per internetą <https://www.scrumguides.org/scrum-guide.html#events>

75. Restfulapi. Prieiga per internetą <https://restfulapi.net/richardson-maturity-model/>

76. Tutorials Point (2019). *Number System Conversion*. [žiūrėta 2019-05-27]. Prieiga per internetą

<https://www.tutorialspoint.com/computer_logical_organization/number_system_conversion.htm>

77. Tutorialsteacher. Prieiga per internetą <https://www.tutorialsteacher.com/core/aspnet-core-startup>

78. Vilniaus universitetas (2019). *Apie skaičiavimo sistemas*. [žiūrėta 2019-05-27]. Prieiga per internetą <https://www.mif.vu.lt/ljmm/rasmenys/skaiciavimo_sistemos.htm>.

79. Wagner B. *More Effective C#: 50 Specific Ways to Improve Your C#* Prieiga per internetą <https://www.amazon.com/More-Effective-Content-Update-Program/dp/0672337886>

80. Whitaker,R.B.*The C# Player's Guide.* Prieiga per internetą https://www.amazon.com/gp/product/0985580135

81. Wikipedia. Prieiga per internetą

[https://en.wikipedia.org/wiki/List\_of\_tools\_for\_static\_code\_analysis#.NET](https://en.wikipedia.org/wiki/List_of_tools_for_static_code_analysis#.NET82.Wikipedia. Prieiga per internetą https://en.wikipedia.org/wiki/Static_program_analysis)

[82.Wikipedia. Prieiga per internetą https://en.wikipedia.org/wiki/Static\_program\_analysis](https://en.wikipedia.org/wiki/List_of_tools_for_static_code_analysis#.NET82.Wikipedia. Prieiga per internetą https://en.wikipedia.org/wiki/Static_program_analysis)

83. Wikipedia. Prieiga per internetą https://en.wikipedia.org/wiki/List\_of\_tools\_for\_static\_code\_analysis#.NET

84. Wikipedia. Prieiga per internetą <https://en.wikipedia.org/wiki/User_story>

85. Wikipedia. Prieiga per internetą https://en.wikipedia.org/wiki/Software\_verification\_and\_validation

86. Wikipedia. Prieiga per internetą <https://en.wikipedia.org/wiki/Software_testing>

87. Wikipedia. Prieiga per internetą <https://en.wikipedia.org/wiki/Software_verification_and_validation>