Code Academy

A: Saulėtekio al. 15, Vilnius
T: +370 636 55 502

Front-end + Java programa

Front-end + Java programa Kurso TEMOS	VAL.	APRAŠYMAS
1. HTML/CSS pradmenys	100	**HTML-1:** Admininistratori prisistatymas apie program. Introduction to Lecturer; Introduction to Program; Introduction to Deliverables; Introduction to Teaching Methodology; Classroom introductions. **Online:** Introduction to Online Lecturer; Introduction to Program; Introduction to Teaching Methodology; How does a computer work? What are programming languages/how they work? **HTML-2:** Introduction to different programming languages; Front-end vs Back-end; Detailed review of HTML/CSS/JS differences;

A: Saulėtekio al. 15, Vilnius
T: +370 636 55 502

KURSO TEMOS	VAL.	APRAŠYMAS
		Demo of HTML/CSS/JS in action, with student repeating all but JS; Dvelve into differences between HTML/CSS; Introduction to HTML; HTML structure (doctype to head to body); Introduction to HTML syntax (incl. self-closing tags) and basic elements: h1-h6, p, img. HTML/CSS comments; **HTML-3:** Recap of HTML/CSS differences; Recap of HTML structures; HTML elements: block vs inline; Core HTML elements: h1-h6, div, p, span, img, a. Inline Internal External CSS; CSS Syntax. Simple CSS properties: color, background. **HTML-4:** HTML: nesting elements; HTML: ul/ol/li; CSS: ID/Class; Basic CSS selectors; HTML: br/hr/strong (inline tipografija); HTML: Box Model; CSS: height/width, margin/padding/border. **HTML-5:** HTML: div header/main/footer/section/nav; CSS: px/em/rem, hex/rgb; Text Enconding CSS: font-size/weight; CSS selectors: spaces, comma sep... Reset/Normalize;

A: Saulėtekio al. 15, Vilnius
T: +370 636 55 502

KURSO TEMOS	VAL.	APRAŠYMAS
		HTML-6: HTML tables; CSS selectors: first-child/last-child... **HTML-7:** TML forms and validation. **HTML-8:** Git + GitHub **HTML-9:** Flexbox Fundamentals **HTML-10:** Flexbox Advanced **HTML-11:** Flexbox Advanced (2) **HTML-12:** Grid CSS **HTML-13:** Responsive Websites (Viewport, Media Queries) **HTML-14:** Mobile-first Design **HTML-15:** Responsive Practice **HTML-16:** CSS Frameworks **HTML-17:** CSS Preprocessors (spec. SCSS)

A: Saulėtekio al. 15, Vilnius
T: +370 636 55 502

KURSO TEMOS	VAL.	APRAŠYMAS
2. JavaScript pradmenys	**140**	**HTML-18:** Front-End Test **HTML-19:** Front-End Test **JS-1** Introduction to Javascript and its possibilities; Brief history of Javascript (and what is Ecmascript); Variables; Data Types; Comments; Math Operators **JS-2** Advanced Math Operators; if-else conditionals; alert() **JS-3** Switch; Ternary Operator; Loops (For) **JS-4** Loops (While...Do); Loops (While); What is DOM? Basic DOM manipulation (textContent, innerHTML); Selectors **JS-5** Functions addEventListener **JS-6:** Collecting data from user inputs (form submit event) **JS-7:** Anon Functions; Exc: Recap of getting and setting data with JS

A: Saulėtekio al. 15, Vilnius
T: +370 636 55 502

KURSO TEMOS	VAL.	APRAŠYMAS
		JS-8: Exc: Recap of getting and setting data with JS **JS-9:** Manipulating CSS with JS (object.style). **JS-10:** Recap of Data Types, more detail into Arrays, Objects. Viewing DOM as an object - properties/methods. Basic DOM properties: e.g. location, document.body, document.forms. **JS-11:** Creating DOM elements with createElement, append/prepend. **JS-12:** JS String, Number, Boolean methods **JS-13:** Review of what is callback? JS callback array methods: forEach, map, filter, find, some, every **JS-14:** JS callback array methods: sort, reduce. **JS-15:** Recap of data type methods, practice. **JS-16:** Basics of creating objects, writing custom methods and calling them. **JS-17:** OOP pagrindai

A: Saulėtekio al. 15, Vilnius
T: +370 636 55 502

KURSO TEMOS	VAL.	APRAŠYMAS
		JS-18: Recap of Arrays, Objects. Practice (exercises) of creating objects, arrays of objects. **JS-19:** Cookies; LocalStorage **JS-20:** Sync vs Async; Promises **JS-21:** REST API principles Fetch GET **JS-22:** Fetch GET with Arrays and Objects **JS-23:** Fetch POST **JS-24:** Fetch GET and POST practice **JS-25:** Fetch DELETE and other REST methods. **JS-26:** Fetch Practice **JS-27:** JS Fetch methods with Postman **JS-28:** JS modules **JS-29:** TEST

A: Saulėtekio al. 15, Vilnius
T: +370 636 55 502

KURSO TEMOS	VAL.	APRAŠYMAS
3. Java objektinis programavimas	**120**	**JS-30:** TEST **Core** Data types and Operators. Control Flow state-ments. OOP in Java. Packages, Folders and naming conventions. Collection Framework in Java. Java Scanner. **Reflection** Extensibility Features. Class Brows-ers. Exposure of Internals. **Generics** Generics vs method overloading. Generics + reflection. Generics for Collections. Generics for class. Generics for methods. **Input-Output in Java** What is a stream? Overview of Streams Bytes vs. Characters Overview of the entire Java IO API **Threads in Java** Non-Threaded Applications vs. Threaded Applications. Process-based multitasking vs. Thread-based multitasking. Thread API in Java. Creating Threads, States of a Thread. Synchroni-zation for threads; static and non-static synchro-nized methods; blocks; the concept of object and class locks. Coordination between threads – wait, notify and notify. All methods for inter-thread communication. Semaphores, Mutex .

A: Saulėtekio al. 15, Vilnius
T: +370 636 55 502

KURSO TEMOS	VAL.	APRAŠYMAS
4. Duomenų bazės	**60**	**Serialization** Object Serialization. Serializable Interface. Serialization API. ObjectInputStream and ObjectOutput. Transient Fields. Objects Mapping (using automapper) **JSON JSON (serialize Object to JSON)** JSON Syntax. JSON in Java, tools, libraries. JSON serializer (bson) example. Some of the important Java 8 features forEach () method in Iterable interface. Default and static methods in Interfaces. Functional Interfaces and Lambda Expressions. Java Stream API for Bulk Data Operations on Collections. **Database development** Intro and Overview; Popular RBDS; MySQL preparation: creating DB, basic SQL queries, creating and managing data SQL alternatives; OBD; MongoDB intro. **JDBC** JDBC Drivers, Features, Writing JDBC code to connect to DB CRUD Operations with JDBC; Statement types in JDBC Types of Rowset, ResultSet in JDBC.

A: Saulėtekio al. 15, Vilnius
T: +370 636 55 502

KURSO TEMOS	VAL.	APRAŠYMAS
5. Spring framework	**40**	**ORM** What is ORM; Popular ORMs JPA implementations; Compare few of them; CRUD Operations with MyBatis/EclipseLink. **Hibernate** Hibernate Introduction; Mapping And Configuration Files In Hibernate; Main Advantage And Disadvantages Of Hibernates; Simple Hibernate Application. **What is Spring** Spring Context: For Dependency Injection (DI); Spring DAO: For database operation using DAO paern. Spring JDBC: For JDBC and Datasource support. Spring ORM: For ORM tool support such as Hibernate. Spring AOP: For Aspect Oriented Programming. Spring Web module: For creating a web application. Spring MVC in Eclipse **DI, IoC** Bean meaning; Bean's lifecycle; What is Depend-ency Injection? Singleton Prototype; XML based configuratio; Annotation based configuration

A: Saulėtekio al. 15, Vilnius
T: +370 636 55 502

KURSO TEMOS	VAL.	APRAŠYMAS
		Spring MVC MVC pattern; Spring Controller; App back-end structure; REST basics; Arguments and return type mapping; File upload using Spring MVC **Spring security** Introduction; Simple examples; Spring Security Custom Login Form XML/Annotation; Examples; Spring Security password hashing example; Overriding the default configure(HttpSecurity) method **Spring Boot** Create stand-alone Spring applications; Embed Tomcat, Jey or Undertow directly (no need to deploy WAR files); Provide opinionated 'starter' POMs to simplify your Maven configuration; Automatically configure Spring whenever possible; Provide production-ready features such as metrics, health checks and externalized configurationAbsolutely no code generation and no require-ment forconfiguration
6. Baigiamasis projektas	**20**	Laikas skirtas baigiamojo projekto įgyvendinimui.
Iš viso:	**480**	