

**Учебная программа курса “Java FullStack Developer. Разработчик на языке программирования Java”**

Учебный центр MYEDU, принадлежащий Myedu OÜ Утверждено 19.09.2024	
Название учебной программы	<b>Java FullStack Developer. Разработчик на языке программирования Java</b>
Группа учебных программ	153135 Разработка и анализ программного обеспечения и приложений / Tarkvara ja rakenduste arendus ning analüüs
Цели обучения	К концу обучения учащийся соответствует уровню Junior Java Developer и знает основы программирования на Java и может разрабатывать простые приложения
Знания и навыки, приобретенные при успешном завершении учебной программы. Достижимые результаты обучения	В конце обучения учащийся: - создает программное обеспечение, используя современные стандарты написания кода и возможности языка Java - применяет принципы объектно-ориентированного программирования - имеет знания и навыки работы с Java Core, Collections - применяет юнит-тесты в разработке приложений - знает инструмент сборки maven - знает систему контроля версий: git - понимает принципы CI / CD
Для кого этот курс. Целевая группа	Специалисты, которые работают или планируют работать в сфере ИТ, и хотят получить базовые знания и навыки, необходимые для работы в качестве Java Software Developer в Эстонии и за ее пределами
Требования на начало обучения, предъявляемые к обучаемому лицу, если они являются предпосылкой достижения результатов обучения	Для участия в обучении требуются базовые знания работы с компьютером. Учащийся должен иметь компьютер или ноутбук с оперативной памятью не менее 4 Гб с доступом в Интернет
Язык обучения	Русский язык
Общий объем обучения, в том числе доля аудиторной, практической и самостоятельной работы	400 академических часов
Самостоятельная работа	240 академических часов
Работа в аудитории	160 академических часов
Продолжительность обучения	12 месяцев
Содержание обучения: - Структура и объем обучения - Содержание обучения	Программа обучения создана на основе профессионального стандарта Noorem tarkvaraarendaja, tase 4 (Junior Software Developer, EstQF Level 4, Младший разработчик программного обеспечения, уровень 4) Теоретический материал и практическая работа под руководством преподавателя  <b>Модуль 1. Java Syntax</b> <i>16 вебинаров, 32 академических часа аудиторной работы и 48 академических часов самостоятельной работы</i> Основы синтаксиса Java и первая программа. Работа с переменными: типы данных (int, String), их объявление, инициализация, ввод с клавиатуры (Scanner). Элементарные математические операции, конкатенация строк. Операторы управления (if, циклы while, for), работа с массивами. Обзор основных методов работы со строками, использование IDE (IDEA). Введение в ООП: объекты, конструкторы, наследование

	<p><b>Модуль 2. Java Core</b> <i>16 вебинаров, 32 академических часа аудиторной работы и 48 академических часов самостоятельной работы</i> Основы ООП: инкапсуляция, полиморфизм, интерфейсы, абстрактные классы. Перегрузка и переопределение методов, использование this и super. Введение в Stream API, лямбда-выражения и функциональные интерфейсы. Ассоциации: композиция и агрегация, множественное наследование и дефолтные методы интерфейсов. Приведение типов, instanceof, выражения switch. Методы класса Object (equals, hashCode, toString), immutable объекты, рекурсия. Основы многопоточности: Thread, Runnable, синхронизация, ExecutorService, работа с потоками и блокировками. Вложенные классы, сериализация (JSON/XML/YAML), Reflection API, аннотации</p>
	<p><b>Модуль 3. Java Professional</b> <i>16 вебинаров, 32 академических часа аудиторной работы и 48 академических часов самостоятельной работы</i> Управление памятью и сборка мусора в JVM, типы ссылок (Weak, Soft, Phantom). Паттерны проектирования: поведенческие, порождающие, структурные. Основы Maven: создание проектов, зависимости, плагины. Работа с библиотеками Guava и Apache Commons Collections. Тестирование с JUnit и Mockito: аннотации, моки, проверка вызовов. Логирование: уровни, инструменты (slf4j, log4j). Основы сетей: топология, модель OSI, DNS. Архитектура ПО: клиент-сервер, трехуровневая архитектура. Протоколы HTTP/HTTPS, работа с запросами и сессиями. Введение в сервлеты и контейнеры (Tomcat), MVC и JSP, создание веб-приложений</p>
	<p><b>Модуль 4. Работа с базами данных. Hibernate</b> <i>16 вебинаров, 32 академических часа аудиторной работы и 48 академических часов самостоятельной работы</i> Основы баз данных: реляционные и нереляционные модели, группы SQL команд (DDL, DML). Установка MySQL, создание и управление таблицами (insert, select, update, delete). Работа с данными: выборка с условиями, join, группировка, индексы. Транзакции и уровни изоляции, концепция ACID. Проектирование БД: нормальные формы, ключи, отношения. JDBC: интерфейсы, получение и обновление данных. ORM (Hibernate): основные концепции, аннотации, работа с отношениями (OneToMany, ManyToOne), наследование Entity</p>
	<p><b>Модуль 5. Spring + Spring Boot</b> <i>16 вебинаров, 32 академических часа аудиторной работы и 48 академических часов самостоятельной работы</i> Введение в Spring: принципы IoC и DI, бины, контекст (ApplicationContext), AOP. Основные модули Spring: core, data access, web, web mvc. Проектирование REST API: методы HTTP, коды ответов, операции с коллекциями. Архитектура приложения: контроллер (прием запросов), сервис (бизнес-логика), DAO (хранение данных). Spring ORM: транзакции, SessionFactory. Тестирование в Spring: unit и integration testing, аннотации. Spring Security: аутентификация и авторизация, OAuth2, сессии. Spring Boot: стартеры, автоконфигурация, встроенный Tomcat, Spring Data JPA</p>
	<p><b>Итоговый проект</b> Проект, требования к которому основаны на реальных потребностях компаний-работодателей. Сдача и защита проекта; добавление проекта в портфолио</p>

<p>Методы обучения</p>	<p><b>Аудиторная (онлайн) работа:</b> теоретический материал (лекции и обсуждение практических примеров)</p> <p><b>Практическая работа (онлайн и самостоятельно):</b> выполнение заданий, разработка программного обеспечения, оформление документации. Все практические задания выполняются в среде IntelliJ IDEA, github и проверяются преподавателем. Дается детальная обратная связь в письменном виде. Преподаватель объясняет новый теоретический материал, отвечает на вопросы студентов по новой теме и предыдущему домашнему заданию.</p> <p>Занятия проходят в Zoom и Discord. Каждый урок записывается, запись доступна закрытой группе в Slack.</p> <p>Между лекциями проходят квесты, задач, тесты, дополнительные видео уроки.</p> <p>Преподаватель доступен для связи каждый день. Если студент пропустил лекцию, посмотрел её в записи и появились вопросы по задачам/теории – он может написать преподавателю свои вопросы и получить ответ в любой день</p> <p><b>Практические упражнения:</b> задания на разработку алгоритмов, применение типов данных и структур, освоение различных инструментов сборки и запуска приложений</p> <p><b>Самостоятельная работа:</b> прочтение дополнительных материалов по теме и просмотр обучающих видео</p> <p>Общение с одногруппниками и преподавателями в закрытом чате в <b>Slack</b></p>
<p>Описание учебной среды и учебных помещений, предметов обстановки и оборудования, необходимых для достижения целей, поставленных в учебном плане, а также приобретения навыков и их соответствия законодательным требованиям охраны здоровья, если таковые установлены</p>	<p>Занятия с преподавателем проходят в форме онлайн видео конференций на платформе Zoom.</p> <p>В учебной среде проводятся теоретические и практические занятия, рассылаются домашние задания.</p> <p>Учащийся задает свои вопросы и получает обратную связь от учителя.</p> <p>Количество студентов в одной группе: до 10 человек.</p> <p>Один академический час длится 45 минут.</p> <p>Каждая встреча длится 2 академических часа.</p> <p>Занятия проводятся два раза в неделю по будням вечером.</p>
<p>Перечень учебных материалов</p>	<p>Учебный материал предоставляется учащимся в электронном виде.</p> <ul style="list-style-type: none"> <li>- конспекты лекций</li> <li>- шаблоны проектов и документации</li> <li>- учебные тренажеры</li> </ul>
<p>Требования к завершению обучения, включая методы оценки и критерии оценки</p>	<p>Посещение онлайн-занятий; выполнение домашних заданий</p> <p>Для оценки результатов обучения используется итоговое тестирование</p>

<p>Условия окончания обучения и выдаваемые документы</p>	<p>Для успешного завершения обучения требуется принять участие по крайней мере в 80% академических и практических занятий и выполнить все домашние задания. Достижение результатов обучения оценивается через практическую работу и итоговое тестирование.</p> <p><b>Сертификат</b> выдается учащемуся, который посетил не менее 80% уроков, выполнил все практические задания и успешно сдал итоговый тест.</p> <p><b>Свидетельство</b> об участии в обучении или об окончании обучения выдается ученику в том случае, если результаты обучения не были достигнуты, но ученик принимал участие в обучении.</p> <p><b>Справка</b> выдается в соответствии с количеством посещенных часов обучения, если учащийся участвовал по крайней мере в половине занятий.</p>
<p>Описание квалификации, опыта учёбы или работы проводящего обучение лица, которые необходимы для прохождения обучения</p>	<p>Java Software Engineer с опытом в разработке и менторингу по Java и карьерным вопросам</p> <p>Эксперт по backend-разработке и работе с базами данных</p> <p>Java Developer с опытом в сфере гейминга и аутсорс-компаниях</p> <p>Software Engineer, эксперт в области автоматизации тестирования на Java</p> <p>Software Engineer, Data Scientist</p> <p>IT Service Engineer с опытом работы в системных интеграторах, телеком-сфере, ритейл-компаниях</p> <p>Backend Developer с опытом работы в банковской сфере</p>