Мидл Java-разработчик

Пояснения к словам иностранного происхождения

Здесь собраны пояснения к словам иностранного происхождения, которые используются в программе курса, но пока не включены в словарную норму русского языка.

6 месяцев

продолжительность курса

3 проекта

в портфолио

1 ЧАС

00

Бесплатный модуль

119 ЧАСОВ

01

Разработка приложения при помощи Spring Framework и Spring Boot

- Погружение в современную Java
- Погружение в современные средства разработки и паттерны
- Spring Framework
- Spring Boot

120 ЧАСОВ

02

Разработка классического и реактивного приложения при помощи Spring Data и Spring Web

- Разработка классического приложения при помощи Spring Data и Spring Web: Servlet Stack
- Разработка реактивного приложения при помощи Spring Data и Spring Web: Reactive Stack
- Создание RESTful сервисов с использованием Spring Rest, OpenAPI и Redis
- Spring Security

120 ЧАСОВ

03

Разработка микросервисного приложения

Базовый тариф	Разработка приложений с использованием микросервисной архитектуры
Расширенный тариф	Apache Kafka Мониторинги и логи с помощью ELK-стека
Максимальный тариф	Развёртывание приложения с использованием CI/CD, Kubernetes, Helm
Общий спринт	Завершающий модуль

10 ЧАСОВ / 24 ЧАСА / 5 ЧАСОВ

Дополнительные модули

- Контейнеризация и управление многоконтейнерными Java-приложениями при помощи Docker
- Базовый SQL для разработки
- Аргументация в рабочей коммуникации

Бесплатный вводный модуль 00

1 час

- Познакомитесь с программой: поймёте, почему важно знать не только паттерны, но и инструменты, зачем тренироваться на реальных кейсах, как устроена практика. Также уточните, как будут проходить спринты и кто будет сопровождать вас на пути.
- Пройдёте входной тест. Он будет полезен и вам, и нам. Вы сможете убедиться в том, что курс будет оптимален по сложности. Мы — будем уверены, что наши участники курса обладают достаточными навыками для его прохождения.
- Всё это поможет понять, подходит ли вам курс.

Содержание модуля 00

Подробнее о курсе

Входной тест

Разработка приложения при помощи Spring Framework и Spring Boot 01

8 недель
4 спринта
3 воркшопа
2 работы с проверкой ревьюером

- Освоите главные нововведения в Java за последние несколько лет, которые появились в LTS версии Java 21
- Изучите функциональный подход к программированию, поработаете со Stream API, узнаете про легковесные потоки, современные возможности создания многопоточных приложений, а также уменьшение boilerplate-кода за счёт Records и Pattern Matching
- Научитесь тестировать свои приложения с помощью фреймворков JUnit 5 и Mockito
- Изучите стандарт при написании корпоративных приложений — фреймворка Spring Framework, его логичного развития — Spring Boot, а также возможностей их тестирования через TestContext Framework, SpringBoot Test и кеширование контекстов

Содержание модуля 01

Спринт 1

Погружение в современную Java

Чему научитесь

- Java 8 как отправная точка
- На пути к Java 21 - изменения в языке
- На пути к Java 21 - изменения в API
- Современная многопоточность в Java
- Тестирование на Java
- Архитектура JUnit 5
- Assertions JUnit 5 и Mock-объекты
- Типы тестов JUnit 5
- Миграция с JUnit 4

Практика

Перепишете тесты с JUnit 4 на JUnit 5

Лайв-кодинг

Лайв-кодинг по практическому освоению работы с виртуальными потоками и интерфейсами Sequenced Collections

---

Спринт 2

Погружение в современные средства разработки и паттерны

Чему научитесь

- Принципы разработки YAGNI, DRY, KISS, DBU и бритва Оккама
- Принципы проектирования и организации кода SOLID
- Архитектурные паттерны разработки корпоративных приложений
- Базовые паттерны разработки корпоративных приложений
- Система контроля версий Git и парадигмы разработки
- Продвинутая работа с удалёнными репозиториями, тегами
- Продвинутая работа с коммитами
- Системы сборки Maven и Gradle
- Управление зависимостями
- Жизненный цикл сборки проекта и его профили
- Использование плагинов

Практика

Напишите свой собственный плагин для Gradle

Лайв-кодинг

Лайв-кодинг по примитивам синхронизации и потокобезопасным структурам данных

Спринт 3

Spring Framework

Чему научитесь

- Spring как IoC-контейнер
- Инфраструктурные бины Spring
- Создание бинов через Java-аннотации
- Создание бинов через Java-конфигурации
- Основные возможности Spring
- Аспектно-ориентированное программирование в Spring
- Практика по разработке на Spring Framework
- Тестирование Spring-приложений
- TestContext Framework
- Практика по тестированию Spring-приложений

Практика

Создадите собственное MVC-приложение блог на Spring Framework и протестируете Web и DAO слои приложения

Лайв-кодинг

Лайв-кодинг по примитивам синхронизации и потокобезопасным структурам данных

Спринт 4

Spring Boot

Чему научитесь

- SpringBoot как развитие Spring Framework
- Сборка SpringBoot-приложений в Maven
- Сборка SpringBoot-приложений в Gradle
- Запуск SpringBoot-приложения
- Настройка приложения
- Автоконфигурации и стартеры
- Мониторинг приложения с использованием SpringBoot Actuator
- Практика по разработке SpringBoot-приложения
- SpringBootTest для тестирования SpringBoot-приложений
- Практика по тестированию SpringBoot-приложения

Практика

Перепишете ранее созданное приложение блог с использованием SpringBoot и протестируете Web и DAO слои приложения

Лайв-кодинг

Spring Boot — Best Practices: работа с различными типами контекстов, правильное использование события спринга, разделение бинов на бизнесовые и инфраструктурные, тесты с кешированием контекстов.

Разработка классического и реактивного приложения при помощи Spring Data и Spring Web 02

8 недель
4 спринта
2 воркшопа
4 работы с проверкой ревьюером

- Узнаете о возможностях написания классических web-приложений в парадигме Model-View-Controller с походами за данными в реляционную базу данных с использованием связки для объектно-реляционного маппинга Spring Data + Hibernate
- Изучите достоинства и недостатки реактивного подхода в написании web-ориентированных приложений, попробуете Reactive Stack и R2DBS для доступа в базу данных, а также научитесь писать тесты с использованием TestContainers и Spring MVC Test
- Изучите возможности написания RESTful-сервисов и использования OpenAPI-спецификации для генерации RestController, а также обеспечение аутентификации и авторизации пользователей с использованием фреймворка Spring Security

Содержание модуля 02

Спринт 5

Разработка классического приложения при помощи Spring Data и Spring Web: Servlet Stack

Чему научитесь

- Доступ к данным в Spring Data JDBC
- Доступ к данным в Spring Data JPA
- Миграция данных с помощью Liquibase
- Тестирование Spring Data
- Фреймворк WebMVC в Spring
- Контроллеры в WebMVC

Практика

Создадите витрину магазина на Spring Boot и протестируете его при помощи Spring Data JPA

Спринт 6

Разработка реактивного приложения при помощи Spring Data и Spring Web: Reactive Stack

Чему научитесь

- Реактивное программирование на Java
- Проекты для создания реактивных приложений
- Реактивный доступ к данным в Spring Data R2DBC
- Концепция сущностей для работы с данными
- Репозитории в Spring Data R2DBC
- Тестирование Spring Data R2DBC
- Фреймворк WebFlux в Spring
- Обработка запросов в функциональном стиле в WebFlux
- Конфигурирование и тестирование WebFlux

Практика

Создадите витрину магазина на реактивном движке Spring Boot и протестируете его при помощи Spring Data R2DBC и Spring WebFlux

Воркшоп

Реактивное программирование в Spring: отличия классического и реактивного подхода, плюсы и минусы реактивного подхода, как не выстрелить себе в ногу, используя реактивный стек

Содержание модуля 02

Спринт 7

Создание RESTful-сервисов с использованием Spring Rest, OpenAPI и Redis

Чему научитесь

- Построение RESTful-сервисов
- Поддержка RESTful-сервисов в Spring
- Клиенты для выполнения REST-запросов
- Спецификация OpenAPI для описания RESTful-сервисов
- Использование Spring Data Redis в качестве кеша

Практика

Доработаете витрину магазина на реактивном движке Spring Boot — добавите RESTful-сервис с использованием OpenAPI и Redis

Спринт 8

Spring Security

Чему научитесь

- Spring Security на стеке сервлетов
- Аутентификация на стеке сервлетов
- Хранение данных аутентификации, X509 и разлогинивание
- Авторизация на стеке сервлетов
- Авторизация OAuth 2.0
- Интеграция Spring Security с Spring MVC на стеке сервлетов
- Аутентификация и авторизации в Spring Security на реактивном стеке
- Авторизация OAuth 2.0 на реактивном стеке
- Интеграция Spring Security с Spring WebFlux на реактивном стеке

Практика

Доработаете витрину магазина на реактивном движке Spring Boot — добавите Spring Security

Воркшоп

Spring Security: использование различных типов аутентификации и авторизации в Spring

Разработка микросервисного приложения 03

8 недель
4 спринта
2 воркшопа
4 работы с проверкой ревьюером

- Узнаете об основных плюсах использования микросервисного подхода к написанию приложений: Saga, Circuit Breaker, API Gateway и Service Discovery
- Изучите основные паттерны и поддержку со стороны фреймворка Spring Cloud в лице Modulith, Consul и Zookeeper
- Научитесь обеспечивать безопасность при обмене данными между микросервисами
- Сможете развёртывать приложения с использованием CI/CD, популярного в настоящее время Kubernetes и Helm
- Узнаете о возможностях стриминговой платформы Apache Kafka и её использовании, чтобы доставлять данные в ELK-стек для обеспечения мониторингов и поставки логов

Практика

- В этот модуль входят спринты разных тарифов: базового, расширенного и максимального.
- В каждом спринте вас ждёт отдельный проект с законченным результатом. При этом проекты связаны между собой: в каждом новом спринте вы можете взять за основу предыдущий проект и продолжить работу с ним уже над новой задачей.

Содержание модуля 03

Спринт 9

Разработка приложений с использованием микросервисной архитектуры

Чему научитесь

- Паттерны микросервисной архитектуры
- Паттерны межсервисного взаимодействия
- Паттерны обеспечения безопасности, состояния, тестирования и UI сервисов
- Стек технологий Spring Cloud
- Безопасность и тестирование в Spring Cloud
- Интеграции в Spring Cloud

Практика

Создадите микросервисное приложение — инфраструктуру банка с использованием микросервисной архитектуры

Старт расширенного тарифа

Спринт 10

Apache Kafka

Чему научитесь

- Стриминговая платформа Apache Kafka
- Возможности Apache Kafka
- Интеграция Apache Kafka со Spring
- Возможности Spring Kafka
- Поддержка потоковой обработки данных и тестирование в Spring Kafka

Практика

Можете продолжить работу над микросервисным приложением банка — настроите взаимодействие микросервисов при помощи Apache Kafka

Спринт 11

Мониторинги и логи с помощью ELK-стека

Чему научитесь

- Трейсинг с помощью Spring Cloud Sleuth и Zipkin
- Мониторинги и алерты с помощью Prometheus и Grafana
- Сбор логов с использованием Apache Kafka и Logstash
- Использование Elasticsearch для индексации, хранения и поиска логов
- Использование системы визуализации данных Kibana

Практика

Можете продолжить работу над микросервисным приложением банка — добавите логирование и мониторинг при помощи ELK-стека

Воркшоп

Воркшоп по мониторингам и логированию: какими мониторингами стоит обвесить систему; какие мониторинги поместить на фронт и бэк

Старт максимального тарифа

Спринт 12

Развёртывание приложения с использованием CI/CD, Kubernetes, Helm

Чему научитесь

- Оркестрация контейнеризованных приложений
- Внутреннее устройство Kubernetes
- Возможности Kubernetes
- Пакетный менеджер Helm
- Тестирование, обеспечение безопасности и лучшие практики в Helm
- Непрерывная интеграция и доставка микросервисов
- Декларативные пайплайны в Jenkins

Практика

Можете продолжить работу над микросервисным приложением банка — настроите его развёртывание при помощи CI/CD, Kubernetes, Helm

Воркшоп

Воркшоп по Kubernetes и CI/CD: отличия от docker-compose, swarm; когда использовать Kubernetes, а когда можно обойтись без него; как раскатывать микросервисное приложение, обеспечивая обратную совместимость

Завершающий модуль

1 час

- Подведёте итоги курса

Контейнеризация и управление многоконтейнерными Java-приложениями при помощи Docker

Дополнительный модуль

10 часов

- **Dockerfile**
- **Docker compose**
- **Docker контейнер**
- **Docker**

В первой теме — «Основы Docker» — вы установите Docker и изучите основные команды для работы с ним. Также вы научитесь писать Dockerfile для сборки образов своих приложений. А ещё изучите практики по работе с образами и контейнерами.

Во второй теме — «Управление многоконтейнерными приложениями» — вы настроите взаимодействие между контейнерами, научитесь поднимать контейнер базы данных и приложения с помощью Docker Compose.

Его основное назначение — упростить и автоматизировать процесс развёртывания и управления связанными сервисами и контейнерами.

Базовый SQL для разработки

Дополнительный модуль

24 часа

- **Функции в SQL**
- **Базовые запросы SQL**
- **Связанные таблицы**
- **Одиночные таблицы**
- **PostgreSQL**

В первой теме вы научитесь подключаться к PostgreSQL через терминал и графический интерфейс, а также создавать, удалять и наполнять базы данных с помощью pgAdmin. Во второй теме вы научитесь создавать базы данных, схемы и таблицы по описанию, применять ограничения разных видов, создавать первичный ключ, а также изменять и удалять БД, схемы и таблицы.

Затем вы научитесь писать простые запросы на создание, чтение, удаление, редактирование данных внутри таблицы, сортировать данные, формировать выборки с помощью логических операторов и операторов сравнения, а также преобразовывать типы данных. Вы разберётесь в видах соединений таблиц, научитесь создавать взаимосвязи между таблицами, читать данные из связанных таблиц, пользоваться каскадным удалением, а также обновлять и удалять строки с помощью фильтров по связанным таблицам.

В завершающей теме курса вы узнаете, как группировать и агрегировать данные, научитесь применять функции для работы со строками, временем и другие базовые функции для работы.

Аргументация в рабочей коммуникации

Дополнительный модуль

5 часов

- Аргументация
- Переговоры
- Контраргумент
- Делиберативный диалог
- Убеждение

Вы узнаете, в каких ситуациях уместно прибегать к аргументации, как лучше структурировать выступление и чем хорошие аргументы отличаются от плохих. Разберёте, как не поддаваться на логические уловки, как и зачем подбирать экспертное мнение.