otus
онлайн-образование

Полная программа

Разработчик на Spring Framework

Курс о разработке веб-приложений на Spring, о фреймворках и вспомогательных технологиях Spring.

Длительность курса: 166 часов

Модуль 1. Введение

Тема 1
Знакомство с курсом

Цель занятия

познакомиться со структурой курса;
обсудить часто встречающиеся вопросы

Краткое содержание

знакомство;
обсуждение целей и маршрута курса;
ответы на вопросы

Тема 2
Введение в Spring Framework // ДЗ

Цель занятия

ориентироваться в проектах Spring для дальнейшего изучения;
применять принцип IoC при написании классов и тестов;
создавать контекст Spring;
определять Spring Beans в контексте;
организовывать правильный DI.

Краткое содержание

Spring Framework и его Spring Projects;
IoC в общем виде;
IoC-контейнер в Spring (контекст);
XML-контекст со Spring Beans и DI между ними.

Домашние задания

Приложение для проведения тестирования студентов - только вывод вопросов и вариантов ответа

Цель

Цель: закрепление материала прошедших лекций:
проектирование доменных сущностей;
ООП;
паттерн DAO;
создание и конфигурирование приложений с помощью Spring;
XML-based конфигурация контекста;
SOLID;
основы тестирования в Java.

Результат: приложение на чистом Spring с XML-based конфигурацией контекста и юнит-тестом сервиса

Тема 3
ЛикБез - Основы архитектуры ПО

Цель занятия

Разбираться в базовых архитектурных принципах, а так же когда и как их можно применять

Краткое содержание

На примерах, рассмотрим, что такое архитектура приложения, как с ней связаны соглашения об именовании. Обсудим и применим несколько архитектурных принципов: KISS, DRY, YAGNY, SOLID (частично)

Тема 4
Конфигурирование Spring-приложений // ДЗ

Цель занятия

конфигурировать Spring-приложения в современном Java-based стиле (как сейчас и все пишут);
ориентироваться в многослойной архитектуре;
пользоваться Spring Expression Language (SpEL);
задавать параметры приложения с помощью .properties файлов.

Краткое содержание

Java-based конфигурацию контекста Spring;
Annotation-based конфигурацию контекста;
аннотация стереотипов, @Autowired и многие другие;
многослойная архитектура;
SpEL;
параметры конфигурации из .properties-файлов

Домашние задания

Приложение по проведению тестирования студентов (с самим тестированием)

Цель

Цель: закрепление материала прошедших лекций: создание и конфигурирование приложений с помощью Spring;
Java-based и Annotation-based конфигурация контекста;
свойства Spring-приложений;
SpEL;
SOLID;
основы тестирования в Java.

Результат: Приложение на чистом Spring с Java-/Annotation-based конфигурацией контекста и юнит-тестами

Тема 5
AOP, Spring AOP

Цель занятия

использовать аспектно-ориентированное программирование (там где нужно),
видеть в коде ключевую функциональность Spring - Spring AOP,
реализовывать в приложениях crosscutting-функциональность с помощью Spring AOP

Краткое содержание

crosscutting-функциональность;
AOP и как устроены Proxy;
Advices и где их применять;
Join point и Pointcut, Pointcuts в стиле AspectJ;
свои аспекты;
best practices, связанные с AOP

Тема 6
"Чёрная магия" Spring Boot // ДЗ

Цель занятия

ориентироваться в возможностях Spring Boot для различных функциональностей и технологий;
максимально быстро создавать production-grade standalone Spring-приложения с помощью Spring Boot Starters;
писать автоконфигурации и использовать существующие;
писать property в YAML-формате.

Краткое содержание

Spring Boot и зачем он;
проект и научимся собирать артефакт;
pom-файл, стартеры, структура проекта;
пример теста;
разные интересные возможности.

Домашние задания

Перенести приложение для тестирования студентов на Spring Boot

Цель

Цель: Закрепление материала прошедших лекций: создание и конфигурирование приложений с помощью Spring Boot;
локализация;
SOLID.

Результат: Приложение на Spring Boot с юнит-тестами

Тема 7
Продвинутая конфигурация (часть 1) - Scopes, Lifecycle

Цель занятия

разбираться в областях видимости (Scopes) бинов;
использовать возможность жизненного цикла бинов.

Краткое содержание

Scopes (области видимости);
Lifecycle (жизненный цикл) бинов;
Также будут рассмотрены дополнительные возможности для конфигурирования приложений;

Тема 8
Продвинутая конфигурация (часть 2) - Conditionals, Events, тестирование // ДЗ

Цель занятия

использовать Best Practices для конфигурирования Spring Boot-приложений;
использовать события (Events) для передачи управления;
эффективно тестировать Spring Boot-приложения.
писать приложения с использованием Spring Shell.

Краткое содержание

профили;
@Conditional-аннотации;
@SpringBootTest;
приложение на Spring Shell.

Домашние задания

Переписать интерфейс приложения для тестирования студентов с использованием Spring Shell

Цель

Цель: закрепление материала прошедших лекций: Spring Shell;
тестирование с контекстом Spring Boot;
органичение тестового контекста.

Результат: Приложение на Spring Shell с тестами на @SpringBootTest

Модуль 2. Работа с базами данных

Тема 1
DAO на Spring JDBC // ДЗ

Цель занятия

эффективно использовать JDBC вместе со Spring JDBC для разработки приложений с мощью чистого SQL;
правильно применять паттерн DAO для подключения к БД;
пользоваться embedded БД для написания тестов и при разработке простых приложений.

Краткое содержание

паттерн DAO;
необходимость кода для чистого JDBC;
простое DAO на Spring JDBC с минимумом boilerplate кода.

Домашние задания

Приложение, позволяющее вести каталог книг в библиотеке

Цель

Цель: закрепление материала прошедших лекций: проектирование доменных сущностей;
ООП;
паттерн DAO/Repository;
встроенная БД H2;
работа с БД через Spring Jdbc;
решение типовых проблем при работе с БД;
интеграционное тестрование репозиториев с помощью @JdbcTest

Результат: Приложение на Spring Jdbc с интеграционными тестами

Тема 2
Основы ORM, JPA, Hibernate как провайдер JPA

Цель занятия

применять JPA для описания маппинга классов-entities на таблицы реляционной БД;
использовать Hibernate в качестве JPA Vendor.

Краткое содержание

JPA, как прослойку между приложением и ORM-фреймворком;
JPA mapping сущности (entity) на БД.

Тема 3
JPQL, Spring ORM, DAO на основе Spring ORM + JPA // ДЗ

Цель занятия

писать ORM DAO с помощью Spring ORM + JPA + Hibernate (в качестве JPA Vendor-а) в Spring приложениях;
использовать JPQL (аналог HQL) для построения SQL-подобных запросов.

Краткое содержание

интерфейсы JPA;
JPQL;
функционал Spring ORM для JPA.

Домашние задания

Использовать ORM в качестве способа доступа к данным в приложении для хранения книг в библиотеке

Цель

Цель: закрепление материала прошедших лекций: проектирование доменных сущностей;
ООП;
паттерн DAO/Repository;
встроенная БД H2;
работа с БД через Spring ORM,
JPA,
Hibernate;
решение типовых проблем при работе с БД и ORM;
интеграционное тестрование репозиториев с помощью @DataJpaTest

Результат: Приложение на Spring ORM, JPA, Hibernate с интеграционными тестами

Тема 4
Транзакции, Spring Tx

Цель занятия

применять особенности транзакции в реляционных БД для правильной разработки слоя DAO;
использовать декларативное и императивное управление транзакциями в Spring-приложениях с помощью Spring Tx.

Краткое содержание

ACID;
уровни изоляции транзакций;
декларативное управление транзакциями в Spring Tx;
propagation транзакций в Spring Tx;
императивное управление транзакциями в Spring Tx.

Тема 5
"Белая магия" Spring Data: Spring Data JPA // ДЗ

Цель занятия

использовать абстракции Spring Data для реализации собственных репозиториев в общем виде;
использовать "белую магию" Spring Data JPA для создания репозиториев для JPA сущностей.

Краткое содержание

проекты Spring Data;
репозитории на Spring Data JPA;
кастомизация (@Query, custom repositories).

Домашние задания

Использовать Spring Data JPA в качестве способа доступа к данным в приложении для хранения книг в библиотеке

Цель

Цель: закрепление материала прошедших лекций: ООП;
паттерн DAO/Repository;
встроенная БД H2;
работа с БД через Spring Data JPA, JPA, Hibernate; решение типовых проблем при работе с БД и ORM;
интеграционное тестрование репозиториев с помощью @DataJpaTest

Результат: Приложение на Spring Data JPA и Hibernate с интеграционными тестами

Тема 6
SQL и NoSQL базы данных

Цель занятия

использовать особенности нереляционных (NoSQL) БД;
выбирать NoSQL БД для решения задач.

Краткое содержание

принципы на которых строятся нереляционные (NoSQL) базы данных;
анализ некоторых из них;
запросы к ним.

Тема 7
Spring Data для подключения к нереляционным БД // ДЗ

Цель занятия

разрабатывать репозитории для хранения данных в NoSQL БД;
использовать другие проекты Spring Data.

Краткое содержание

Spring Data Key-Value для подключения к Key-Value базам данных (и даже к SynchronizedMap);
MondoDB;
MongoDB с помощью Spring Data MongoDB.

Домашние задания

В приложении для хранения книг в библиотеке, использовать MongoDB как источник хранения данных и Spring Data MongoDB в качестве способа доступа к ним

Цель

Цель: закрепление материала прошедших лекций: ООП;
паттерн DAO/Repository; MongoDB/Embedded MongoDB;
работа с БД через Spring Data MongoDB;
организация данных в MongoDB и решение типовых проблем;
интеграционное тестрование репозиториев с помощью @DataMongoTest

Результат: Приложение на Spring Data MongoDB с интеграционными тестами

Модуль 3. Разработка Web-приложений

Тема 1
Введение в Spring MVC, Spring MVC View на Spring Boot // ДЗ

Цель занятия

различать архитектуры MVC и Spring MVC;
создавать простые классические Web-приложения на основе Spring MVC и Spring Boot;
создавать REST-сервисы на основе Spring MVC и Spring Boot.

Краткое содержание

архитектуры MVC и Spring MVC;
Web-слой на Spring MVC;
простой REST-сервис (без View).

Домашние задания

CRUD приложение с Web UI и хранением данных в БД

Цель

Цель: разрабатывать полноценные классические Web-приложения
Результат: Web-приложение полностью на стеке Spring

Тема 2
Современные приложения на Spring MVC // ДЗ

Цель занятия

понимать особенности и ограничения архитектурного стиля REST
создавать сложные классические приложения с использованием RestController;
понимать современные приложения, как основанные на AJAX архитектуре и jQuery, так и SPA-приложения (React).

Краткое содержание

архитектурный стиль REST;
инструменты для осуществления AJAX-запросов (XMLHttpRequest, JQuery, axios, fetch);
современные приложения на React;
варианты написания приложений с использованием React и Angular.

Домашние задания

Переписать приложение с использованием AJAX и REST-контроллеров

Цель

Цель: использовать Spring MVC для разработки современных AJAX/SPA приложений с помощью Spring MVC
Результат: современное приложение на стеке Spring

Тема 3
Reactor. Введение в реактивное программирование

Цель занятия

применять принципы, на которых построено Reactive программирование;
пользоваться библиотекой RxJava для построения реактивных приложений;
применять различные Rx-операторы для необходимых ситуаций.

Краткое содержание

принципы построений реактивных систем;
RxJava;
различные операторы RxJava.

Тема 4
Spring WebFlux // ДЗ

Цель занятия

создавать с помощью Spring WebFlux современные реактивные Web-приложения.

Краткое содержание

Reactive-стек Spring;
Spring WebFlux;
приложение на полностью реактивном стеке Spring с использованием Spring WebFlux + Reactive Spring Data Repositories.

Домашние задания

Использовать WebFlux

Цель

Цель: разрабатывать Responsive и Resilent приложения на реактивном стеке Spring с помощью Spring Web Flux и Reactive Spring Data Repositories
Результат: приложение на реактивном стеке Spring

Модуль 4. Spring Security

Тема 1
Spring Security: Архитектура

Цель занятия

различать аутентификацию и авторизацию;
различать задачи безопасности в Enterprise приложениях;
разбираться в архитектуре Spring Security.

Краткое содержание

задачи безопасности в Enterprise приложениях;
система фильтров Spring Security;
внутреннее устройство Spring Security и его основных классов.

Тема 2
Spring Security: Механизмы аутентификации // ДЗ

Цель занятия

различать различные механизмы аутентификации;
внедрять HTTP Basic аутентификацию для защиты Web-приложения;
внедрять Form-based аутентификацию для защиты Web-приложения;
использовать вторичные виды аутентификации (такие как Remember Me и Anonymous).

Краткое содержание

различные механизмы первичной аутентификации - HTTP Basic и Form-based и как их реализовывать с помощью Spring Security;
вторичные механизмы аутентификации и как они поддерживаются Spring Security.

Домашние задания

В CRUD Web-приложение добавить механизм аутентификации

Цель

Цель: защитить Web-приложение аутентификацией и простой авторизацией
Результат: приложение с использованием Spring Security

Тема 3
Spring Security: Авторизация

Цель занятия

внедрять в приложение различные механизмы авторизации - на основе URL и методов сервисов, для глубокой защиты приложений.

Краткое содержание

классы Spring Security, участвующие в авторизации;
авторизация на основе URL с помощью встроенного DSL;
механизмы для авторизации на методах сервисов.

Тема 4
Spring Security: ACL // ДЗ

Цель занятия

внедрять в приложение безопасность на основе доменных сущностей - ACLs.

Краткое содержание

авторизация на основе доменных сущностей;
структура ACL на примере реализации в БД;
основные классы Spring Security, участвующие в авторизации на основе доменных сущностей;
пример приложения с ACLs.

Домашние задания

Ввести авторизацию на основе URL и/или доменных сущностей

Цель

Цель: научиться защищать приложение с помощью полноценной авторизации и разграничением прав доступа
Результат: полноценное приложение с безопасностью на основе Spring Security

Модуль 5. "Около" и "Дзен"

Тема 1
Spring Batch // ДЗ

Цель занятия

использовать Spring Batch для организации пакетной обработки данных в приложении;
применять Spring Batch не только в больших проектах.

Краткое содержание

абстракции, предоставляемые Spring Batch;
основной принцип абстракций;
различные абстракции и их реализации в Spring Batch.

Домашние задания

На основе Spring Batch разработать процедуру миграции данных из реляционного хранилища в NoSQL или наоборот

Цель

Цель: мигрировать данные с помощью Spring Batch
Результат: приложение для пакетных обработок данных на Spring Batch

Тема 2
Монолиты vs. Microservices Round 1, Messaging, Enterprise Integration Patterns (EIP)

Цель занятия

различать два похода к разработке Enterprise-приложений - монолиты и микросервисы;
видеть проблемы, возникающие при создании монолитов;
использовать Best Practices при разработке монолитов.

Краткое содержание

микросервисная и монолитная архитектуры;
преимущества и недостатки обеих архитектур;
различные виды интеграций;
EIP (Enterprise Intergration Patterns) и Spring Integration.

Тема 3
Spring Integration: Messages и Channels

Цель занятия

использовать различное многообразие сообщение для работы со Spring Integration;
использовать различные семантики каналов там, где необходима нужная семантика интеграции;
пользоваться встроенный DSL для настройки связей в Spring Integration;
различать базовые Endpoints и Flow Components.

Краткое содержание

паттерн Message;
реализация Message в SPring Integration;
классификация каналов с точки зрения семантики, буферизации и т.д.;
реализация разных каналов в Spring Integration.

Тема 4
Spring Integration: Endpoints и Flow Components // ДЗ

Цель занятия

использовать Endpoints и Flow Components для разработки сложных Enterprise-приложений с почти любой интеграцией.

Краткое содержание

различные паттерны Endpoint и их реализации в Spring Integration;
на примере узнаем, как с помощью Flow Commponents управлять потоками интеграции.

Домашние задания

Реализовать обработку доменной сущности через каналы Spring Integration

Цель

Участники смогут осуществлять "интеграцию" частей приложения с помощью EIP
Результат: приложение с применением EIP на Spring Integration

Тема 5
Монолиты vs. Microservices (Round 2), Spring Boot Actuator - must have в микросервисах // ДЗ

Цель занятия

использовать Best Practices для разработки приложений на микросервисной архитектуре;
использовать изобилие возможностей Spring Boot Actuator для создания production-grade приложений и микросервисов;
применять HATEOAS подход для разработки REST-сервисов.

Краткое содержание

микросервисная архитектура и Best Practices;
приложение Spring Boot Actuator;
HATEOAS подход;
Spring Data REST, как самый простой способ создания CRUD микросервисов.

Домашние задания

Использовать метрики, healthchecks и logfile

Цель

Цель: реализовать production-grade мониторинг и прозрачность в приложении
Результат: приложение с применением Spring Boot Actuator

Тема 6
REST-клиенты, SOAP, Spring WebServices и клиенты к ним

Цель занятия

писать REST-клиенты с различными дополнениями для построения полноценных микросервисных систем;
писать на Spring WebServices SOAP-сервисы и клиенты к ним для разработки систем на SOA и SOAP.

Краткое содержание

основной класс для написания REST-клиентов - RestTemplate и напишем клиента с помощью него;
Spring Cache и Spring Retry;
SOA, SOAP, XML, XSD, WSDL;
Code-first и Contract-first разработку;
SOAP-сервис и SOAP-клиент к нему с помощью Spring Web Services.

Тема 7
Docker, оркестрация // ДЗ

Цель занятия

использовать Docker для запуска приложений в повседневной жизни, собирать собственные образы Docker с помощью Dockerfile для запуска собственных приложений, использовать docker-compose для запуска сложных систем.

Краткое содержание

"Matrix from the Hell";
уровень виртуализации Docker;
приложение с помощью Docker;
свой image с помощью Dockerfile.

Домашние задания

Обернуть приложение в docker-контейнер

Цель

Цель: деплоить приложение в современном DevOps-стеке
Результат: обёртка приложения в Docker

Тема 8
Знакомство с Kubernetes

Цель занятия

узнать, что такое Kubernetes и чем он полезен java-разработчику.

Краткое содержание

что такое Kubernetes;
основные части (абстракции);
требования к приложению;
демонстрация Minikube;
демонстрация managed Kubernetes.

Тема 9
Spring Cloud. Конфигурация и взаимодействие

Цель занятия

организовать взаимодействие микросервисов с Config Server с помощью Spring Cloud;
изучить возможности обновления данных микросервисов "на лету" с помощью Spring Cloud Bus;
научиться строить огромные системы на Cloud Service Discovery с использованием Eureka, Ribbon и Feign.

Краткое содержание

взаимодействие участников микросервисной архитектуры в облаках с помощью Eureka; конфигурирование сервиса с помощью Spring Cloud Config.

Тема 10
Spring Cloud. Отказоустойчивость и мониторинг // ДЗ

Цель занятия

строить огромные системы с помощью Zuul, Hystrix, Sleuth, Zipkin.

Краткое содержание

множество технологий для построения огромных систем;
Hystrix как решение для изоляции внешних вызовов;
логистика и мониторинг в Spring Cloud.

Домашние задания

Обернуть внешние вызовы в Hystrix

Цель

Цель: сделать внешние вызовы приложения устойчивыми к ошибкам
Результат: приложение с изолированными с помощью Hystrix внешними вызовами

Тема 11
Применение RabbitMQ

Цель занятия

получить ответы на вопросы по проекту, ДЗ и по курсу

Краткое содержание

вопросы по улучшению и оптимизации работы над проектом;
затруднения при выполнении ДЗ;
вопросы по программе.

Тема 12
Применение Kafka

Цель занятия

познакомиться с платформой Kafka и возможностями применения в приложениях на java.

Краткое содержание

что такое Kafka и для чего это нужно;
как пользовать Kafka в программах на java.

Тема 13
Комплексный пример использования WebFlux и Kafka

Цель занятия

ориентироваться в реактивных фреймворках в стеке Spring;
использовать Reactive-версию Spring Data репозиториев.

Краткое содержание

различные реактивные API и фреймворки;
реактивный стек Spring и принципы его работы для построения реактивных приложений;
особенности работы с БД в реактивном стеке;
Reactive Spring Data Repositories.

Модуль 6. Проектная работа

Тема 1
Выбор темы и организация проектной работы

Цель занятия

выбрать и обсудить тему проектной работы,
спланировать работу над проектом,
ознакомиться с регламентом работы над проектом

Краткое содержание

правила работы над проектом и специфика проведения итоговой защиты;
требования к результату проекта и итоговой документации.

Домашние задания

Проектная работа

Цель

Цель: реализовать собственный проект с применением Spring

Тема 2
Защита проектных работ №1

Цель занятия

защитить проект и получить рекомендации экспертов

Краткое содержание

презентация проектов перед комиссией;
вопросы и комментарии по проектам.

Тема 3
Защита проектных работ №2

Цель занятия

защитить проект и получить рекомендации экспертов.

Краткое содержание

презентация проектов перед комиссией;
вопросы и комментарии по проектам.