Полная программа

Groovy Developer

Groovy Developer

Длительность курса: 146 часов

Модуль 1. Синтаксис и встроенные функции языка

Тема 1
Groovy - JVM и Java

Цель занятия
описать текущее состояние рынка IT, нишу Groovy, связь Groovy с Java;
познакомиться с Git;
объяснить, почему Groovy-разработчику важно уметь пользоваться Git.

После занятия вы сможете:
настроить рабочее место для разработки на Groovy;
написать первую программу на Groovy;
настроить Git.

Краткое содержание
Groovy SDK;
установка;
настройка проекта в IDE;
комбинация с Java;
скрипты vs классы;
особенности языка.

Домашние задания

Настройка проекта и оформление репозитория
Цель
Научиться создавать проект Gradle;
Подготовиться к выполнению домашних заданий.

Тема 2
Языковые конструкции. Способы запуска. Операторы

Цель занятия
познакомиться с видами типизации, динамической типизацией;
познакомиться с основными математическими операторами.

После занятия вы сможете:
писать код со статической и динамической типизацией;
эффективно применять математические операторы в Groovy;
запускать Groovy-скрипты из Java-кода.

Краткое содержание
динамическая типизация;
переменные, методы;
GroovyShell, GroovyScriptEngine;
математические операции.

Тема 3
Языковые конструкции. Строки и regex

Цель занятия
познакомиться с особенностями строк в Groovy;
познакомиться с подстановками в строках, регулярными выражениями.

После занятия вы сможете:
использовать подстановки в строках;
эффективно манипулировать строками с использованием регулярных выражений.

Краткое содержание
строки;
подстановки;
регулярные выражения.

Тема 4
Языковые конструкции. Классы, интерфейсы, трейты, объекты, конструкторы, методы

Цель занятия
познакомиться с основами концепции ООП и реализацией ООП в Groovy;
познакомиться с closures и особенностями их применения;
познакомиться с модификаторами доступа.

После занятия вы сможете:
применять концепции ООП в Groovy;
применять модификаторы доступа.

Краткое содержание
ООП в groovy;
модификаторы доступа;
приведение типов;
delegate, this, owner.

Домашние задания

Эмулятор банкомата
Цель
Применить на практике принципы SOLID;
Написать эмулятор ATM (банкомата).

Тема 5
Функциональное программирование

Цель занятия
познакомиться с понятием функциональное программирование;
узнать в чем отличие функционального программирования от ООП;
подробно познакомиться с кложурами, контекстом исполнения, областями видимости;
познакомиться с использованием closures как функций;
познакомиться с каррированием.

После занятия вы сможете:
применять функциональный подход при разработке программы на Groovy.

Краткое содержание
closures;
контекст it;
области видимости;
передача в качестве аргументов методов;
каррирование.

Тема 6
Коллекции

Цель занятия
познакомиться с основами алгоритмов и коллекциями в Groovy;
изучить особенности коллекций в Groovy и отличия от Java;
познакомиться с конвенцией equals и hashCode, реализациями Map и Set;
познакомиться с особенностями итерирования по коллекциям в Groovy.

После занятия вы сможете:
использовать коллекции в Groovy;
итерироваться по коллекциям и манипулировать данными в коллекциях (аггрегация, фильтрация и т. д.).

Краткое содержание
итерирование по объектам (все есть коллекция);
методы groovy - аналоги stream api;
словари.

Домашние задания

Применение коллекций
Цель
Попрактиковать различные аспекты коллекций.
https://github.com/Groovy-Developer/groovy-homeworks/blob/main/hw-3

Тема 7
Concurrency в Groovy

Цель занятия
познакомиться с особенностями многопоточного выполнения кода groovy в JVM
познакомиться с фреймворком GPars
применять фреймворк GPars для разработки concurrency и parallel кода на groovy

Краткое содержание
Parallel Collections;
Parallel Arrays;
GParsPool;
Avoid Side-Effects in Functions;
GParsExecutorsPool;
Memoize;
Map-Reduce;
Combine.

Домашние задания

Обработчик JSON-ов
Цель
Научиться обрабатывать JSON;
Научиться работать с файлами.

Тема 8
Консультация + смотрим примеры

Цель занятия
ответить на оставшиеся вопросы;
упорядочить знания предыдущих занятий;
посмотреть на примеры решения задач;
получить обратную связь.

Краткое содержание
обратная связь и ответы на вопросы;
список задач с решениями по пройденной теме.

Модуль 2. Продвинутые функции языка. Расширения языка (DSL)

Тема 1
Метапрограммирование

Цель занятия
познакомиться с метаклассами.

После занятия вы сможете:
менять структуру и поведения классов в рантайме с помощью мета-программирования.

Краткое содержание
MetaClass;
MethodMissing, FieldMissing;
ExpandoMetaClass.

Тема 2
Паттерн декоратор и расширения

Цель занятия
познакомиться с методами-расширениями.
использовать мета-классы и методы-расширения.

Краткое содержание
use, Expando;
методы-расширения;
паттерн декоратор.

Тема 3
Паттерн Builder

Цель занятия
познакомиться со встроенными builder-ми в groovy;
научиться создавать свои собственные builder-ы;
понять связь между builder-ми и DSL.

Краткое содержание
JsonBuilder, XmlBuilder;
NodeBuilder;
ObjectGraphBuilder.

Домашние задания

Обработчик документов
Цель
Научиться обрабатывать JSON;
Научиться работать с файлами.

Тема 4
Делегирование и введение в DSL

Цель занятия
познакомиться с понятием DSL;
изучить использование closures для построения DSL.

После занятия вы сможете:
строить простые DSL;
применять closures при построении DSL.

Краткое содержание
использование Closure для построения DSL

Домашние задания

XLSX-builder
Цель
Прочитать документацию по DSL на groovy.org;
Сделать собственный DSL для какого-либо класса.

Тема 5
AST-трансформации

Цель занятия
познакомиться с AST-трансформацией;
сравнить AST-трансформацию с lombok;
познакомиться с основными аннотациями, применяемыми для AST-трансформации.

После занятия вы сможете:
применять AST-трансформацию при построении DSL.

Краткое содержание
AST transformations vs Lombok;
Interface ASTTransformation;
local и global;
@Bindable, @Vetoable, @Immutable;
@Memoized.

Тема 6
GDSL Практика построения DSL + смотрим примеры

Цель занятия
познакомиться с GDSL;
изучить особенности GDSL, интеграцию с intellij idea.

После занятия вы сможете:
применять GDSL на практике.

Краткое содержание
GDSL in IntelliJ Idea;
Delegating Closure Calls;
global context;
Extending GroovyDSL.

Домашние задания

DSL для конфигурации веб-сервера
Цель
Прочитать документацию по DSL на groovy.org;
Сделать собственный DSL для какого-либо класса.

Модуль 3. Системы сборки проекта

Тема 1
Обзор систем сборки

Цель занятия
узнать о существующих системах сборки пректов (ant, maven, grape, gradle);
научиться использовать grape;
научиться настраивать maven для сборки Groovy;
научиться генерировать ant- и maven-скрипты сборки.

Краткое содержание
Grape и @Grab-аннотация;
Maven и Groovy;
Ant и Groovy, AntBuilder.

Тема 2
Gradle DSL (часть 1): обзор

Цель занятия
углубить знания о Gradle;
изучить управление зависимостями в Gradle;
изучить настройку многомодульного проекта в Gradle;
изучить написание своих методов-расширений для Gradle;
изучить написание методов для настройки шагов сборки Gradle;
изучить создание своих плагинов для Gradle.

После занятия вы сможете:
создавать многомодульные проекты с Gradle;
кастомизировать шаги сборки Gradle;
создавать проекты с Gradle;
пользоваться инфраструктурой Gradle, включая таски, конфигурации и пр.;
генерировать "толстые" jar-файлы и запускать их;
генерировать Docker-образы.

Краткое содержание
обзор сборщика Gradle;
многомодульность;
управление зависимостями;
написание своих методов расширений для Gradle;
методы для настройки шагов сборки Gradle;
введение в Gradle DSL.

Тема 3
Gradle DSL (часть 2): написание плагина

Цель занятия
изучить разные способы создания плагина (как часть билд-процедуры, buildSrc folder, отдельное приложение);
изучить создание новых задач (Task) в плагине;
изучить конфигурирование плагина;
изучить применение аннотаций в плагине и их ограничения в новых и старых версиях Gradle;
научиться настраивать плагины в билд-процедуре, используя DSL.

После занятия вы сможете:
создавать свои плагины для Gradle;
кастомизировать сборки Gradle с помощью плагинов;
использовать Gradle DSL для написания плагинов.

Краткое содержание
создание плагины (часть build-процедуры, buildSrc folder, отдельное приложение);
задачи (Tasks) в плагине;
конфигурирование плагина;
аннотации в плагине;
настройка плагина в build-процедуре и Gradle DSL.

Домашние задания

Написание плагина для Gradle
Цель
Научиться настраивать и кастомизировать сборку Gradle;
На практике создать кастомные шаги для сборки Gradle.

Тема 4
CI/CD. Jenkins (часть 1): Pipelines и Jenkins DSL

Цель занятия
познакомиться с основами развертывания приложения;
изучить настройку декларативных и императивных пайплайнов в Jenkins/Gitlab;
использование Groovy скриптов для кастомизации пайпланов и кастомизации Jenkins.
создавать и настраивать пайплайны в Jenkins;
гибко настраивать пайплайны с помощью расширения Groovy-скриптами.

Краткое содержание
императивные и декларативные пайплайны;
настройка пайплайнов для сборки Groovy-проектов;
расширение функциональности Jenkins.

Тема 5
CI/CD. Jenkins (часть 2): Plugins API и интеграции

Цель занятия
познакомиться с основами развертывания приложения;
изучить настройку декларативных и императивных пайплайнов в Jenkins/Gitlab;
использование Groovy скриптов для кастомизации пайпланов и кастомизации Jenkins.
создавать и настраивать пайплайны в Jenkins;
гибко настраивать пайплайны с помощью расширения Groovy-скриптами.

Краткое содержание
императивные и декларативные пайплайны;
настройка пайплайнов для сборки Groovy-проектов;
расширение функциональности Jenkins.

Домашние задания

Настройка пайплайна для сборки проекта
Цель
Научиться строить пайплайны для сборки backend-проектов на Java/Groovy;
На практике научиться добавлять свои собственные реализации шагов сборки.

Тема 6
Управление инфраструктурой

Цель занятия
познакомиться с инструментом infrastructor;
познакомиться с подходом Infrastructure as Code;
научиться развертывать VMs с помощью infrastructor.

Краткое содержание
infrastructor DSL;
создание и развертывания инфраструктуры с VM.

Тема 7
Практика по сборке проектов

Цель занятия
получить обратную связь по освоению материала;
ответить на возникающие вопросы.

Краткое содержание
обсуждение вопросов студентов по материалам модуля.

Модуль 4. Разработка бэкенда

Тема 1
Веб-уровень: грувлеты и шаблоны, Rest

Цель занятия
понимать, что такое REST API, зачем его используют;
отличать методы, принципы и т. д.

Краткое содержание
HTTP;
REST.

Тема 2
Веб-уровень: ratpack

Цель занятия
научиться строить REST API на основе веб-фреймворка ratpack;
познакомиться с процессом создания простого веб-сервера.

Краткое содержание
практика разработки веб-сервиса с rest api на основе ratpack.

Тема 3
Веб-фреймворки: Grails, Micronaut

Цель занятия
познакомиться с RAD подходом к разработке;
научиться создавать шаблоны проектов grails, micronaut;
разобраться в типовой структуре веб-проектов.

Краткое содержание
Micronaut framework;
создание проекта и интеграция с groovy-фреймворками;
gRPC;
Grails. Веб-уровень.

Домашние задания

API и web UI для todo-list
Цель
Научиться создавать серверный и пользовательский http-интерфейсы.

Тема 4
Работа с БД. Часть 1: Sql

Цель занятия
познакомиться с транзакцией в реляционной СУБД;
оптимизировать SQL-запросы;
изучить встроенные в Groovy инструменты для работы с БД.

Краткое содержание
соединение с БД;
пулы соединений;
DDL, DML, DataSet.

Домашние задания

Самодельный ORM
Цель
Научиться работать с jdbc;
На практике освоить многоуровневую архитектуру приложения.

Тема 5
Groovy Config + Spring IoC

Цель занятия
объяснить преимущества и недостатки монолитной архитектуры, фреймворка Spring, особенностей разработки со Spring в Groovy;
изучить Groovy-конфигурацию для Spring-проекта;
сравнить SpEL с Groovy, посмотреть на конфигурирование spring security с помощью Groovy.

Краткое содержание
Groovy Configuration;
Groovy Object Customization;
Groovy Script Compiler Customization;
Spring Boot Groovy CLI.

Тема 6
Работа с БД. Часть 2: ORM: GORM

Цель занятия
познакомиться с GORM.

После занятия вы сможете:
настраивать использование базы данных и выполнять запросы с использованием GORM;
применять кэширование;
использовать события и метки времени TTL.

Краткое содержание
уровень данных;
GORM.

Домашние задания

Backend с БД для todo-list
Цель
Научиться создавать CRUD-приложения на Spring Boot.

Тема 7
Интеграция groovy с BPM инструментами

Цель занятия
обсудить OpenAPI и особенности его применения;
создать спецификацию REST-интерфейса с использованием OpenAPI;
взаимодействовать с Open API из Groovy.

Краткое содержание
Groovy generator (Generates a Groovy API client);
Accessing The OPEN API From Groovy.

Тема 8
Практика по работе с фреймворками

Цель занятия
получить обратную связь по освоению материала;
ответить на возникающие вопросы.

Краткое содержание
обсуждение вопросов студентов по материалам модуля.

Модуль 5. Тестирование

Тема 1
Модульные тесты

Цель занятия
познакомиться с методами тестирования и подходом TDD;
затронуть некоторые интересные keywords в Groovy;
изучить инструменты expando, cobertura;
познакомиться с junit и mockito;
объяснить на примере, что такое "тестируемое приложение".

Краткое содержание
expando;
покрытие кода тестами с Cobertura;
GroovyLogTestCase;
TDD-подход.

Тема 2
Spoke Framework + автотесты

Цель занятия
познакомиться с инструментом Spoke для интеграционного тестирования;
познакомиться с инструментами Selenium и Geb для создания автотестов.

Краткое содержание
интеграционные тесты;
Geb для автотестов.

Домашние задания

Создание автотестов с использование Geb Spock
Цель
Научиться писать тест-кейсы для интеграционного тестирования Groovy-приложений;
Научиться создавать автотесты для пользовательского интерфейса с использованием фреймворка Geb.

Тема 3
Нагрузочное тестирование

Цель занятия
познакомиться с инструментом Jmeter;
рассмотреть основы нагрузочного тестирования;
определить цели и критерии успешности нагрузочного тестирования;
разработать и использовать скрипты для проведения нагрузочного тестирования.

Краткое содержание
JMeter, скрипты с JSR223.

Домашние задания

Создание плана нагрузочного тестирования с использованием Groovy-скриптов
Цель
На практике настроить план нагрузочного тестирования;
Расширить стандартную функциональность инструментов JMeter с помощью Groovy-скриптов.

Тема 4
Практика по тестированию

Цель занятия
получить обратную связь по освоению материала;
ответить на возникающие вопросы.

Краткое содержание
обсуждение вопросов студентов по материалам модуля.

Модуль 6. Проектный модуль

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
В проектной работе вы:
- закрепите и продемонстрируете полученных знаний и навыков
- отработаете навыки командной работы (можно в командах по 2-3 человека)
- подготовите портфолио для работодателя

Тема 2
Консультация по проектам и домашним заданиям

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 3
Technical Interview

Цель занятия
разобрать и пройти на примере все основные этапы technical интервью.

Краткое содержание
микрособеседование на 10-15 минут каждого студента
разбор вопросов, вызвавших затруднения.

Тема 4
Защита проектных работ

Цель занятия
защитить проект и получить рекомендации экспертов.

Краткое содержание
презентация проектов перед комиссией;
вопросы и комментарии по проектам.