otus
онлайн-образование

Полная программа

Автоматизатор тестирования на Kotlin

Курс по активной прокачке навыков автоматизации тестирования на Kotlin для QA-инженеров и разработчиков на Kotlin

Длительность курса: 132 часа

Модуль 1. Введение в Kotlin

Тема 1

Вводное занятие: обзор курса, обзор языка Kotlin, области применения (серверная часть, Андроид)

Цель занятия

сформировать общее представление о курсе;
познакомиться с историей и особенностями языка Kotlin;
установить и настроить среду разработки.

Краткое содержание

roadmap курса;
история языка Kotlin;
области применения языка;
установка IDE;
kotlin Playground.

Тема 2

Типы данных в языке Kotlin

Цель занятия

сформировать общее представление о типах данных в Kotlin; изучить модель памяти, встроенные типы и объявление переменных; разобраться с null safety и работой со строками.

Краткое содержание

модель памяти Kotlin;
встроенные типы языка;
объявление переменных;
null safety;
работа со строками.

Тема 3

Классы и функции

Цель занятия

закрепление основ языка Kotlin.

Краткое содержание

объявление класса;
методы класса и модификаторы доступа;
data class;
enum;
object;
companion object;
интерфейсы, абстрактные классы.

Тема 4

Контроль выполнения потока

Цель занятия

закрепление основ языка Kotlin.

Краткое содержание

циклы: for, forEach, while, операторы перехода break, continue;
условные конструкции (if-else, when), в т. ч. с объявлением переменных;
тернарный оператор;
исключения в Kotlin.

Тема 5

Обобщенные типы // ДЗ

Цель занятия

закрепление основ языка Kotlin.

Краткое содержание

лямбда-выражения;
обобщенные типы;
функции, использующие обобщенные типы в возвращаемых значениях и аргументах;
in, out.

Домашние задания

Написание обобщенной функции transformCollection

Цель

написать обобщенную функцию transformCollection.

Тема 6

Коллекции // ДЗ

Цель занятия

закрепление основ языка Kotlin.

Краткое содержание

коллекции в Kotlin;
мутабельные и иммутабельные коллекции;
методы для работы с коллекциями: map, flatMap, filter, zip, reduce, groupBy;
функции области видимости: apply, let, run, with, also.

Домашние задания

Коллекции

Цель

применить функции группировки и фильтрации для анализа данных в коллекциях.

Тема 7

Функции-расширения, делегаты, DSL // ДЗ

Цель занятия

научиться создавать расширения к существующим классам;
уметь использовать библиотеки для работы с коллекциями объектов;
научиться создавать делегаты к классам.

Краткое содержание

реализация функций-расширений;
работа с делегатами.

Домашние задания

Функции-расширения, делегаты, DSL

Цель

реализовать функции-расширения.

Тема 8

Корутины

Цель занятия

научиться создавать неблокирующие функции;
сформировать навык ориентироваться в чужом коде с использованием корутин.

Краткое содержание

понятие многопоточности;
корутины vs потоки;
скоупы;
диспатчеры и контекст;
каналы;
средства отладки в IDE.

Тема 9

Консультация по пройденному материалу и ДЗ №1

Цель занятия

получить ответы на вопросы по пройденному материалу.

Краткое содержание

повтор пройденного материала;
ответы на вопросы по теории и ДЗ.

Модуль 2. Работа с внешними источниками

Тема 1

Погружение в Gradle

Цель занятия

научиться настраивать build.gradle.kts для своих нужд;
уметь кастомизировать поведение встроенных джоб;
сформировать понимание, для чего нужны плагины.

Краткое содержание

история;
установка и инициализация проекта;
основные составляющие файлы;
анатомия build.gradle;
кастомизация джоб.

Тема 2

Пара слов о Docker // ДЗ

Цель занятия

научиться создавать собственные образы;
уметь развертывать приложения из образов;
уметь работать с запущенными контейнерами.

Краткое содержание

постановка проблемы: как плохо нам жилось без Docker;
принцип работы Docker;
понятие container и image;
создание своего образа и запуск контейнера;
основные команды для работы с запущенным контейнером;
очистка системы;
dockerfile;
dockerhub.

Домашние задания

Docker

Цель

упаковать простое приложение в Docker-образ и опубликовать его на Docker Hub.

Тема 3

Работа с сетью: http клиент ktor

Цель занятия

освежить знания о протоколе HTTP;
научиться использовать ktor для отправки get-post-put-delete запросов.

Краткое содержание

основы HTTP;
пара слов о других популярных клиентах (OkHttp, Retrofit);
подключение ktor клиента;
отправка GET запроса без тела;
отправка POST запроса с телом;
десериализация ответа в Kotlin Class;
неблокирующий подход;
базовые основы архитектуры: место HTTP слоя в приложении.

Тема 4

Взаимодействие с реляционной СУБД на примере PostgreSQL

Цель занятия

освежить знания об SQL;
научиться писать запросы к БД из кода приложения;
познакомиться с бесплатным клиентом DBeaver.

Краткое содержание

основы SQL;
СУБД Postgres: разворачивать в docker;
клиент DBeaver;
голый JDBC vs ORM (на примере Exposed);
базовые основы архитектуры: место слоя БД в приложении.

Тема 5

Взаимодействие с нереляционными СУБД на примере MongoDB // ДЗ

Цель занятия

научиться писать запросы к MongoDB из кода приложения;
понимать разницу между SQL и NoSQL.

Краткое содержание

отличие NoSQL от SQL;
краткий рассказ про Redis;
MongoDB (разворачивать в Docker!) с библиотекой kmongo;
базовые основы архитектуры: место слоя БД в приложении.

Домашние задания

Взаимодействие с нереляционными СУБД на примере MongoDB

Цель

загрузить данные из внешнего API в MongoDB и реализовать поиск по сохраненным записям.

Тема 6

Консультация по пройденному материалу и ДЗ №2

Цель занятия

получить ответы на вопросы по пройденному материалу.

Краткое содержание

повтор пройденного материала;
ответы на вопросы по теории и ДЗ.

Модуль 3. Базовые фреймворки тестирования

Тема 1

Модульные тесты

Цель занятия

научиться создавать модульные (unit) тесты для функций и объектов классов;
уметь использовать утверждения для проверки примитивных и составных типов результата.

Краткое содержание

JUnit5 архитектура;
JUnit5 аннотации;
JUnit5 наборы тестов;
JUnit5 параллелизация тестов;
JUnit5 параметризация тестов;
расширения в JUnit5.

Тема 2

Мокирование

Цель занятия

научиться создавать моки частей приложения для проведения юнит-тестирования.

Краткое содержание

для чего нужны моки;
mockito;
практическая часть: реализация тестов на код из прошлой практики, используя мок внешней API.

Тема 3

BDD подход

Цель занятия

научиться разрабатывать и выполнять тесты на основе спецификаций;
изучить граничные значения, классы эквивалентности;
научиться создавать тест-кейсы и тест-план.

Краткое содержание

спецификации и БДД;
kotest;
ответы на вопросы по ДЗ;
рефлексия.

Тема 4

Репортинг

Цель занятия

использовать плагины для генерации отчетов в HTML / XML;
оценивать тестовое покрытие и подготавливать отчет в HTML по покрытию тестами;
использовать Allure-плагин и Allure TestOps для отчетов.

Краткое содержание

JUnit XML / TestNG Results Rendering;
code coverage report (JaCoCo / Kover);
cucumber reporting;
allure reporting plugin;
рефлексия.

Модуль 4. Специфика тестирования серверной части

Тема 1

Основы серверной разработки на примере Spring Framework

Цель занятия

научиться создавать простые REST API;
сформировать понимание, как устроена серверная часть веб-приложений.

Краткое содержание

Spring Boot;
Spring MVC;
разработка простого REST API с подключением к базе данных;
подключение Swagger плагина.

Тема 2

Работа с очередями сообщений на примере Apache Kafka

Цель занятия

сформировать знания об асинхронных паттернах взаимодействия микросервисов.

Краткое содержание

краткая информация об очередях сообщений и Kafka;
основы Kafka: топики, партиции, консьюмер группы;
продолжение работы с приложением из предыдущего урока;
добавление TopicListener, TopicPublisher.

Тема 3

Автоматизация тестирования API (RestAssured, Postman)

Цель занятия

научиться разрабатывать автоматические тесты для веб-серверов (API).

Краткое содержание

подход №1: использование Junit5 + ktor client для тестирования;
подход №2: использование RestAssured (демонстрация на примере разрабатываемого на прошлых двух занятиях приложения).

Тема 4

Консультация по пройденному материалу и ДЗ №3

Цель занятия

получить ответы на вопросы по пройденному материалу.

Краткое содержание

повтор пройденного материала;
ответы на вопросы по теории и ДЗ.

Модуль 5. Специфика тестирования Android приложений

Тема 1

Основы Android-приложений

Краткое содержание

android Studio: обзорная экскурсия (эмуляторы, работа с логами, работа с БД, профилирование);
анатомия приложения: первый hello-world.

Тема 2

Нативные средства автоматизации

Цель занятия

научиться определять селекторы для элементов на экране;
уметь разрабатывать автоматические тесты для мобильных приложений на Android;
научиться использовать механизмы синхронизации для корректной обработки асинхронных операций и анимаций;
уметь выполнять тестирование многоэкранных приложений с внешними вызовами (интентами);
научиться создавать скриншоты при выполнении теста.

Краткое содержание

espresso (основная библиотека и расширения IdlingResources и Intent);
kakao;
hamcrest;
barista;
знакомство с мобильным приложением «Мои привычки»;
обсуждение сценариев тестирования и возможных затруднений.

Тема 3

Тестовые фреймворки Kakao, Kaspresso, расширение Hamcrest

Цель занятия

научиться определять селекторы для элементов на экране;
научиться разрабатывать автоматические тесты для мобильных приложений на Android;
научиться использовать механизмы синхронизации для корректной обработки асинхронных операций и анимаций;
уметь выполнять тестирование многоэкранных приложения с внешними вызовами (интентами);
уметь создавать скриншоты при выполнении теста.

Краткое содержание

модель Espresso для взаимодействия с элементами (Matcher, ViewAction, ViewAssertion);
расширения Espresso для тестирования интентов, использования механизмов IdlingResource, сложных селекторов для списков;
создание собственных Matcher с использованием Hamcrest;
библиотеки Kakao и Kaspresso для применения DSL для создания тестов;
создание скриншотов отдельных View, целого экрана.

Тема 4

Тестирование Accessibility

Цель занятия

научиться использовать инструменты для проверки соответствия приложения требованиям WCAG 2.

Краткое содержание

основные понятия WCAG и настройки Accessibility в Android;
библиотеки для проверки соответствия требованиям доступности (Accessibility Test Framework for Android).

Тема 5

Тестирование сетевого слоя // ДЗ

Цель занятия

уметь создавать mock API для тестирования сетевых мобильных приложений;
уметь запускать тесты с ограничением скорости сети, имитировать неустойчивую связь.

Краткое содержание

основы сетевого взаимодействия для Android/iOS;
ktor client;
okHttp;
okHttp MockWebServer;
retrofit;
network Link Conditioner (MacOS);
tc (Linux);
clumsy (Windows);
обсуждение использования имитации API для реализации тестов.

Домашние задания

Тестирование сетевого слоя

Цель

выполнить интеграционное тестирование сетевого и локального слоёв приложения с использованием mock-объектов и сформировать отчёт Allure.

Тема 6

Кроссплатформенное тестирование интерфейса методом "черного ящика" // ДЗ

Цель занятия

научиться выбирать тестовый движок TestRunner;
уметь настраивать автоматическое выполнение тестов (как в модели «белого ящика» при наличии доступа к коду, так и в модели «чёрного ящика»);
уметь создать автоматические тесты для существующих приложений без доступа к коду.

Краткое содержание

UI Automator;
appium;

calabash;
RoboElectric TestRunner;
Firebase TestLab;
обсуждение тестирования «чёрного ящика» для мобильного приложения «Мои привычки»;
тестирование произвольных приложений.

Домашние задания

Кроссплатформенное тестирование интерфейса методом «черного ящика»

Цель

проверить функциональность двух экранов и навигацию между ними с помощью инструментальных тестов.

Тема 7

Практика по тестированию реального приложения // ДЗ

Цель занятия

научиться создавать тесты для мобильного приложения "uHabits".

Краткое содержание

подготовка тестовых двойников;
создание DI-контекста для использования тестовых двойников;
разработка юнит-тестов для Repository-слоя приложения;
разработка интеграционных тестов (Kakao + Hamcrest);
настройка отчетов в Allure.

Домашние задания

Практика по тестированию реального приложения

Цель

автоматизировать проверку пользовательских сценариев, используя паттерн Page Object и фреймворки Kakao/Kaspresso.

Модуль 6. Дополнительный модуль

Тема 1

Обзор нагрузочного тестирования (Gatling)

Цель занятия

понять роль нагрузочного тестирования и его значение для оценки устойчивости системы при высокой нагрузке;
разобрать процесс интеграции нагрузочного тестирования в CI/CD и его применение на стадии staging;
научиться выявлять узкие места в архитектуре системы и анализировать отклонения роста нагрузки от ожидаемых линейных показателей.

Краткое содержание

введение в нагрузочное тестирование;
lincheck;
JMeter;
Gatling;
K6 + Kotlin;
рефлексия.

Тема 2

Gitlab CI

Цель занятия

использовать систему сборки (в т. ч. для тестирования);
объяснить в чем разница между подходами CI/CD;
объяснить особенности различных инструментов CI/CD.;
настроить тестовый pipeline в GitLab CI.

Краткое содержание

общие идеи CI/CD;
конфигурация как код;
Gitlab CI;
рефлексия.

Тема 3

Интеграция тестов в Gitlab

Цель занятия

научиться настраивать конвейер сборки (с запуском тестов) в Gitlab;
уметь отслеживать изменение метрик покрытия, результатов тестов.

Краткое содержание

тестовые сценарии в контейнерах и конвейере Gitlab CI;
запуск тестовых сервисов;
артефакты сборки.

Тема 4

Тестирование производительности и профилирование

Цель занятия

научиться использовать measure-функции;
уметь разрабатывать правила для smoke-тестов и бенчмарков;
уметь собирать и анализировать информацию от JVM.

Краткое содержание

kotlinx.benchmark;
java Flight Recorder.

Тема 5

Статический анализ кода и уязвимостей

Цель занятия

научиться определять правила тестирования стиля кода;
уметь обнаруживать отклонения от стиля;
уметь находить потенциальные уязвимости в коде.

Краткое содержание

detekt;
horusec;
insider;
выполнение аудита безопасности;
контроль стиля кода через detekt.

Тема 6

Практика

Цель занятия

применять полученные знания по разработке и автоматизации тестов для всестороннего тестирования проекта.

Краткое содержание

сборка приложения для интернет-магазина;
клиент на Android + Backend.

Тема 7

ИИ в работе тестировщика

Цель занятия

использовать современные ИИ-инструменты в тестировании.

Краткое содержание

обзор современных ИИ-инструментов для тестировщика.

Модуль 7. Подготовка к поиску работы

Тема 1

Как правильно оформить резюме чтобы найти работу? // ДЗ

Цель занятия

научиться создавать продающее резюме;
понять структуру резюме и особенности наполнения каждого раздела;
определить особенности резюме для разных карьерных ситуаций: первое место работы, смена проф. области, долгий перерыв в карьере;
узнать, на что обращают внимание рекрутеры и нанимающие менеджеры при рассмотрении резюме.

Краткое содержание

определение резюме;
виды резюме;
структура резюме;
лайфхаки работы с резюме на разных площадках;
ошибки в резюме;
сопроводительное письмо и thank you letter.

Домашние задания

Составление резюме

Цель

отработать навык написания резюме.

Тема 2

Как подготовиться к собеседованию и техническому интервью?

Цель занятия

понять какие бывают собеседования и интервью;
научиться готовиться к HR-собеседованию;
научиться готовиться к техническому интервью;
узнать основные вопросы на собеседованиях;
понять основные причины отказов после собеседований;
инсайты для качественной подготовки к собеседованию.

Краткое содержание

виды собеседований;
основные вопросы на собеседовании;
причины отказа после собеседования;
прохождение HR-собеседования;
прохождение технического интервью;
инсайты эффективного прохождения собеседований.

Модуль 8. Итоговый проект

Тема 1

Консультация по проекту 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 курса.

Домашние задания

Проектная работа

Цель

Юнит-тесты, инструментальные тесты для Open Source мобильного приложения (из предложенного списка или созданного в организации по месту работы), автоматические сценарные тесты на Appium, тесты API на Ktor Client.

Тема 2

Защита проектных работ

Цель занятия

защитить проект и получить рекомендации экспертов.

Краткое содержание

презентация проектов перед комиссией;
вопросы и комментарии по проектам.

В защите могут участвовать и студенты, не выполняющие собственного проекта, но желающие принять участие в обсуждении проектов своих коллег.