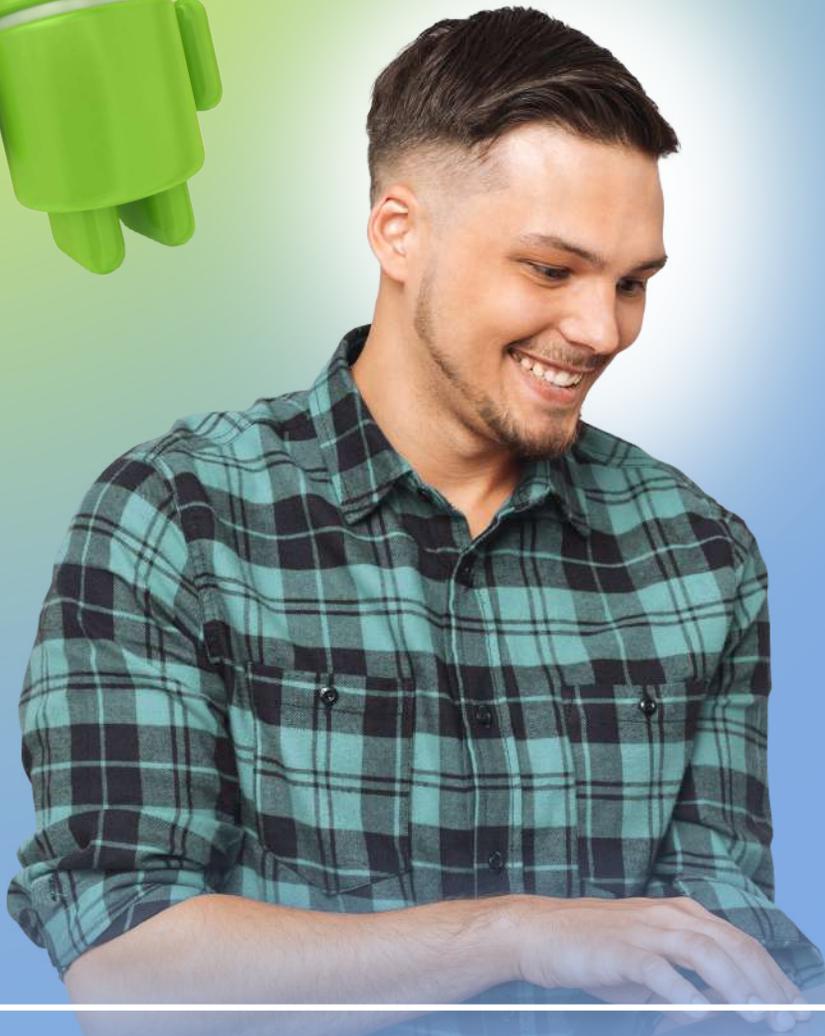


top

КОМПЬЮТЕРНАЯ
АКАДЕМИЯ



учёба с нуля
и бесплатно



диплом о проф.
переподготовке



трудоустройство
во время обучения



живая помощь
от преподавателей



платформа
с упражнениями



обновленная
программа 2025

Разработка приложений под

ANDROID

736 академических часов

1 год длительность курса

368 часов аудиторных занятий

368 часов самостоятельной работы

Наши партнёры:

Yandex Cloud

hh карьерный маркетплейс

Программа обучения

Введение в язык программирования Java и Android

- Интро в Java.
- Сравнительный анализ языка Java с другими языками программирования.
- Виртуальная машина.
- Байт-код.
- Android.
- Java под Android.
- Android SDK.
- Алгоритм.
- Понятие блок-схемы.
- Программная среда Android Studio, Android SDK.

Переменные, типы данных, операторы

- Типы данных.
- Переменная.
- Константы и литералы.
- Операторы.
- Построение программ с использованием блок-схем.

Логические операторы, операторы ветвлений, побитовые операторы

- Преобразование типов данных.
- Логические операторы.
- Таблица приоритетов операторов.
- Конструкции логического выбора. Операторы ветвлений.
- Побитовые операторы.

Циклы

- Циклы.
- Работа с интегрированным отладчиком в Eclipse.

Строки, массивы одномерные, многомерные

- Работа со строками.
- Массивы.
- Алгоритмы суммирования.
- Алгоритмы поиска.
- Алгоритмы сортировки.
- Понятие сложности алгоритма.
- Многомерные массивы.

Методы (на примере статических методов)

- Методы.
- Область видимости.
- Рекурсия.

Объектно-ориентированное программирование

- Введение в объектно-ориентированное программирование.
- Понятие класса.
- Понятие объекта.
- Понятие члена класса, поля класса, метода класса.
- Спецификаторы доступа.
- Конструкторы объекта.
- Ключевое слово this.
- Перегрузка методов и конструкторов.
- Статические методы классов.
- Передача объектов в метод.
- Область видимости в методах классов.
- Наследование.

- Понятие интерфейса.
- Вложенные классы.
- Ключевое слово final.
- Сборка мусора.
- Пакеты.
- Шаблоны (Generics).

Исключения

- Что такое исключительная ситуация.
- Принципы обработки исключительных ситуаций.
- Понятие checked и unchecked исключений.
- Ключевое слово try, catch, throw, finally.
- Подробности использования исключительных ситуаций.
- Раскрутка стека вызовов.

JavaCollectionFramework

- Классы-обертки.
- Введение в JCF.
- Интерфейсы JCF.
- Создание коллекций с помощью фабричных методов.
- Классы JCF.

Аннотации, Анонимные классы, Lambda выражения

- Аннотации.
- Анонимные классы.
- Lambda выражения.

Stream API

- Stream API.
- Цели, задачи и практическое использование.

Системы контроля версий

- Зачем нужен контроль версий.
- Обзор систем контроля версий.
- Git.
- Использование внешних сервисов (github).

Работа в команде, управление программными проектами

- Какие задачи решаются управлением программными проектами.
- Диаграммы Ганта.
- Важные вопросы по управлению программными проектами.
- Анализ терминов предметной области.
- Характеристики проекта.
- Расходы, связанные с проектом.
- Общий обзор моделей и методологий процесса разработки.
- Подробнее о Scrum.

Использование JUnit и фреймворка Mockito

- Что такое модульное тестирование?
- Цели и задачи модульного тестирования.
- Необходимость модульного тестирования.
- Обзор инструментов для модульного тестирования.
- Инструмент JUnit.
- Фреймворк Mockito.

Паттерны проектирования

- Что такое паттерны проектирования?
- Причины возникновения паттернов проектирования.

- Понятие паттерна проектирования.
- Принципы применения паттернов проектирования.
- Принципы выбора паттернов проектирования.
- Принципы разделения паттернов на категории.
- Введение в UML.
- Использование UML при анализе паттернов проектирования.
- Порождающие паттерны.
- Структурные паттерны.
- Паттерны поведения.

Паттерны MVC, MVP, MVVM

- Что такое паттерн MVC?
- Цели и задачи паттерна Model-View-Controller.
- Model.
- View.
- Controller.
- Примеры использования паттерна MVC.
- Паттерн MVP.
- Паттерн MVVM.
-

Принципы проектирования классов SOLID

- Обзор проблем, встречающихся при проектировании и разработке классов.
- Принципы проектирования классов SOLID.
- Примеры использования принципов SOLID.

ИТОВОЙ ПРОЕКТ

Android. Обзор необходимого ПО

- Платформа Android.
- Уровни архитектуры Android.
- Процесс выполнения кода. Виртуальная машина Dalvik.
- Типы Android приложений.
- Установка необходимого ПО для разработки.
- Настройка эмуляторов.
- Детальный обзор Android Studio для разработки Android приложений.

Структура Android-проекта. Пользовательский интерфейс приложения

- Создание "Hello, World"-проекта.
- Физическая структура проекта.
- Запуск проекта на эмуляторе и на телефоне.
- События.

Жизненный цикл приложения Android, Activity, intent

- Жизненный цикл приложения Android.
- Понятие Activity.
- Жизненный цикл Activity.
- Основные методы, реализующие цикл.
- Намерения intent.

Material Design

- Что такое Material Design?
- Цели и задачи Material Design.
- Принципы Material Design.
- Разработка приложения с Material Design.

Layout, Views и ресурсы

- Введение в XML.
- Цели и задачи разметки (layout).
- Виды разметки.

- Примеры использования различных видов разметки.
- Лучшие практики.
- Виды и виджеты.

Детальный обзор виджетов Android. Основные и полезные методы виджетов, важные аспекты их создания и работы

- Доступ к виджетам.
- Создание виджетов.
- Инициализация виджетов.
- Базовые виджеты.
- Обработка действий пользователя на виджеты.
- Виджеты для скроллинга.

Многопоточность, AsyncTask

- Многопоточность в Java.
 1. Что такое многопоточность?
 2. Класс Thread.
 3. Интерфейс Runnable.
 4. Приоритеты потоков.
 5. Синхронизация потоков.
- Асинхронность.
- Использование ExecutorService.

Сохранение информации в телефоне

- Файлы.
 1. Что такое файл?
 2. Чтение данных из файла.
 3. Запись данных в файл.
 4. Удаление файлов, созданных приложением
- Использование SharedPreferences для сохранения информации.

Расширенные приёмы работы с виджетами

- Адаптеры.
- Расширенные текстовые поля.
- Использование виджетов.
- Практическая работа: приложение «Регистрационная форма», приложение «Туристический блог».

Меню, управляющая и оповещающая информация, диалоги

- Меню приложения. Инициализация и обработка информации.
- Использование меню в приложениях.
- Оповещение Notification. Создание, поддержка, обработка выбора, закрытие.
- Диалоговые окна.
- Практические примеры.

Фрагменты

- Что такое фрагмент?
- Цели и задачи фрагментов.
- Создание фрагмента.
- Жизненный цикл фрагмента.
- Добавление фрагмента.
- Удаление фрагмента.
- Замена фрагмента.
- Примеры создания и использования фрагментов.
- Использование Navigation drawer.

Изображения, графика, анимация

- Работа с изображениями как с ресурсами.

- Работа с изображениями как с внешними файлами.
- Отрисовка графических примитивов.
- Использование шрифтов.
- Анимация.

Сервисы, намерения intent

- Службы.
- Intent и службы.
- Слушатели BroadcastReceiver.

Введение в работу с базами данных

- Запросы SELECT, INSERT, UPDATE, DELETE.
- Многотабличные базы данных.
- Нормализация.
- Многотабличные запросы.
- Функции агрегирования.
- Понятие группировки. Ключевое слово GROUP BY.
- Ключевое слово HAVING. Сравнительный анализ HAVING и WHERE.
- Подзапросы.
- Операторы для использования в подзапросах, объединения.
- План выполнения запроса.
- Оптимизация запросов.

Использование баз данных в Android-приложениях

- База данных SQLite.
- Курсор.
- Получение данных из базы данных.
- Добавление данных в базу данных.
- Обновление данных в базе данных.
- Удаление данных в базе данных.
- Контент-провайдер.
- Разработка приложения, использующего источник данных.

Адресная книга, календари, события, звонки, sms

- Адресная книга и ее использование.
- Работа со звонками и sms.
- Календарь и событие.
- Использование календаря.
- Разработка приложения «Список контактов», «Будильник».

Аудио, видео, камера

- Проигрывание аудиофайлов.
- Запись аудио.
- Проигрывание видеофайлов.
- Запись видео.
- Использование камеры.
- Практические примеры.

Сенсоры

- Компас, акселерометр, гироскоп.
- Определение наличия сенсоров.
- Получение данных акселерометра.
- Получение данных гироскопа.
- Практические примеры.

Сетевые возможности, доступ к веб-сервисам

- Основы сетевого взаимодействия.
- Синхронные и асинхронные запросы.
- Использование запросов GET, POST, PUT, DELETE, OPTIONS, HEAD.

- JSON.
- Интеграция с существующими веб-сервисами.
- Использование сторонних библиотек.
 1. Volley.
 2. Retrofit.

Google Cloud Platform, Firebase

- Google Cloud Platform.
- Firebase.

Работа с картами Google Maps API

- Что такое Google Developer Console?
- Получение SHA-1 отпечатка сертификата хранилища ключей.
- Получение уникального Google API key.
- Создание базового приложения с картой.
- Теория о позиционировании карт, сферическая система координат.
- Понятия: latitude, longitude, zoom, bearing, tilt.
- Динамическое позиционирование по карте.
- Создание компонентов карты: маркеры, кривые линии, полигоны.
- Установка слушателей карты.
- Примеры решения популярных задач при работе с картой.

Регистрация в Google Play

- Категории приложений в Google Play.
- Регистрация приложения в Google Play.
- Обновление существующего в Google Play приложения.
- Подпись и распространение приложений.
- Сравнительный анализ других магазинов приложений (Amazon и т.д.).
- Интеграция рекламных баннеров в приложении.
- Роль и назначение рекламных баннеров на разных стадиях раскрутки приложения.

ФИНАЛЬНЫЙ ИТОГОВЫЙ ПРОЕКТ



ПО ОКОНЧАНИИ КУРСА СЛУШАТЕЛЬ БУДЕТ:

- Понимать и использовать принципы создания программ с использованием Java.
- Уметь создавать, компилировать, и производить отладку проектов в IDE Android Studio.
- Уметь проектировать и реализовывать сложные алгоритмы с использованием механизмов условий и циклов.
- Использовать принципы и ООП.
- Понимать архитектуру Android.
- Разбираться в принципах жизненного цикла приложений Android.
- Использовать различные разновидности Activity.
- Разрабатывать виджеты различной степени сложности.
- Применять механизмы оповещений и диалоговые окна.
- Использовать различные виджеты в зависимости от поставленной задачи.
- Разбираться в механизмах графического вывода и работы с изображениями.
- Взаимодействовать с источниками данных.
- Создавать приложения, взаимодействующие с адресной книгой, календарем, звонками, sms.
- Уметь использовать мультимедиа возможности в рамках создаваемого приложения.
- Использовать картографические возможности.
- Владеть механизмами интеграции с таким веб-сервисами, как Facebook, Twitter.
- Понимать принципы и требования к регистрации приложения в Google Play и других магазинах приложений.
- Уметь проектировать классы различной степени сложности и иерархии классов.
- Использовать механизмы generics для построения шаблонных классов.
- Использовать классы JCF.
- Сохранять и читать информацию из файлов.
- Использовать механизмы многопоточности Java.
- Использовать систему контроля версий.
- Применять паттерны проектирования.
- Использовать юнит-тестирование.



17

лет на рынке



380

филиалов в России



2 800

преподавателей



120 000+

студентов



24 000+

ОТЗЫВОВ



4,8

рейтинг
на Я. Картах



Также доступно
онлайн-обучение

*данные действительны
на январь 2025 года