

**Днепр**  
пр-т. Яворницкого, 101  
(056) 7660788

**Одесса**  
ул. Садовая, 3  
(048) 728 6660

**Винница**  
ул. Соборная, 20  
(0432) 52 0820

**Киев**  
ул. Жилянская, 128/28  
(044) 537 22 60

**Житомир**  
ул. Скороульського, 3  
(067) 303 0533

**Запорожье**  
ул. Рекордная, 20  
(067) 547 7455

**Кривой Рог**  
пр-т Мира, 44А  
(067) 828 8586

**Каменское**  
ул. Медицинская, 2-Е  
(050) 380 1838

**Луцк**  
ул. Евгения Сверстюка, 1  
(097) 114 5250

**Львов**  
ул. Замарстыновская, 83а  
(032) 240 3851

**Мариуполь**  
ул. Архитектора  
Нильсена, 26/21  
(0629) 41 2724

**Николаев**  
ул. Лягина, 4  
(0512) 67 0053

**Полтава**  
ул. Пилипа Орлика, 36а  
(0322) 7 3544

**Ровно**  
ул. Соборная, 16  
(050) 445 2111

**Тернополь**  
ул. Крушельницкой, 18  
(067) 968 9985

**Ивано-Франковск**  
ул. Новгородская, 49  
(050) 488 9951

**Харьков**  
ул. Маломясницкая, 9/11  
(057) 752 0501

**Хмельницкий**  
ул. Подольская, 93  
(068) 300 3009

**Херсон**  
ул. Перекопская, 20  
(067) 652 2762

**Чернигов**  
ул. Пятницкая, 39  
(0462) 91 9120

**Черновцы**  
ул. Кордуби, 25  
(050) 366 9905

**Компьютерная Академия ШАГ**

**18** **55**

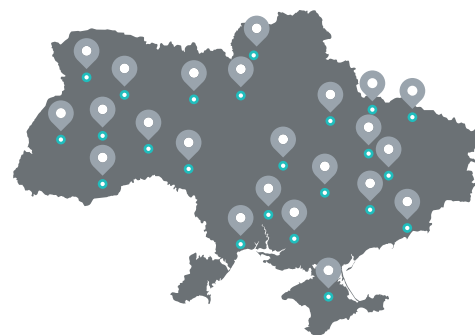
Стран мира

Филиалов

**40000**

Студентов

Компьютерная Академия ШАГ  
**Филиалы**  
**Украина**



**Филиалы**  
**Международные**



itsstep.dp.ua

## Программа обучения

Аудиторные занятия - 368 академ. часа  
Самостоятельная работа - 298 академ. часа  
Soft Skills - 18 академ. часа  
Английский- 138 академ. часа  
Длительность курса - 1 год

## Разработка приложений под Android

# JAVA

### В РАМКАХ КУРСА ВЫ НАУЧИТЕСЬ:

- Понимать и использовать принципы создания программ с помощью Java.
- Создавать, компилировать, выполнять отладку проектов в IDE Android Studio.
- Проектировать и реализовывать сложные алгоритмы с использованием механизмов условий и циклов.
- Использовать принципы ООП. Понимать архитектуру Android.
- Разбираться в принципах жизненного цикла приложений Android.
- Использовать различные виды Activity.
- Взаимодействовать с источниками данных.
- 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.
- Использовать мультимедиа-возможности в рамках создаваемого приложения.
- Использовать картографические возможности.
- Использовать механизмы интеграции с таким веб-сервисами, как Facebook, Twitter.
- Понимать принципы и требования регистрации приложения в Google Play и других магазинах приложений.
- Проектировать классы различной степени сложности и иерархии.
- Использовать механизмы Generics для построения шаблонных классов, классы JCF.
- Сохранять и читать информацию из файлов.
- Использовать механизмы многопоточности Java.
- Использовать систему контроля версий.
- Применять паттерны проектирования.
- Использовать unit testing.

# Программа обучения JAVA

## Введение в язык программирования Java и Android

- Введение в Java.
- Сравнительный анализ Java с другими языками программирования.
- Понятие виртуальная машина.
- Байт-код.
- Android, Java под Android.
- Android SDK.
- Алгоритмы.
- Блок-схемы.
- Программная среда Android Studio, Android SDK.

## Переменные, типы данных, операторы

- Типы данных.
- Переменные.
- 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

- Понятие цикла.
- Работа с интегрированным отладчиком в Eclipse.

## Строки, массивы одномерные, многомерные

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

## Методы

- Методы (на примере статических методов).
- Область видимости.
- Понятие рекурсии.

## Объектно-ориентированное программирование

- Введение в объектно-ориентированное программирование.
- Понятия класса и объекта.
- Понятия 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.
- Наследование.
- Понятие интерфейса.
- Вложенные классы.
- Ключевое слово final.
- Сборка мусора.
- Пакеты.
- Шаблоны (Generics).

## Исключения

- Принципы обработки исключительных ситуаций.
- Понятие checked- и unchecked-исключений.
- Ключевые слова try, catch, throw, finally.
- Особенности использования исключительных ситуаций.
- Раскрутка стека вызовов.

## Java Collection Framework

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

## Аннотации, анонимные классы, Lambda-выражения

- Понятие аннотации.
- Анонимные классы.
- Lambda-выражения.

## Stream API

- Понятие Stream API.
- Цели, задачи.
- Практическое использование.

## Системы контроля версий

- Причины использования контроля версий.
- Обзор систем контроля версий.
- Git.
- Использование внешних сервисов.

## Работа в команде, управление программными проектами

- Задачи, которые 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.
- Обзор методологии Scrum.

## Использование JUnit и фреймворка Mockito

- Цели и задачи модульного тестирования.
- Необходимость модульного тестирования.
- Обзор инструментов для модульного тестирования.
- Инструмент JUnit.
- Фреймворк Mockito.

## Паттерны проектирования

- Причины возникновения паттернов проектирования.
- Понятие паттерна проектирования.
- Принципы применения, выбора.
- Разделение паттернов на категории.
- Введение в UML.
- Использование UML при анализе паттернов проектирования.
- Виды паттернов: порождающие, структурные, паттерны поведения.

## Паттерны MVC, MVP, MVVM

- Цели и задачи паттерна Model-View-Controller.
- Понятие элементов Model, View, Controller.
- Примеры использования паттерна MVC.
- Паттерн MVP.
- Паттерн MVVM.

## Принципы проектирования классов SOLID

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

### ПРАКТИЧЕСКАЯ РАБОТА

## 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 приложений под Android.

## Структура Android-проекта. Пользовательский интерфейс приложения

- Создание проекта «Hello, World!».
- Физическая структура проекта.
- Запуск на эмуляторе и телефоне.
- Понятие события.

## Жизненный цикл приложения Android, Activity, Intent

- Жизненный цикл приложения Android.
- Понятие Activity, его жизненный цикл.
- Основные методы, которые реализуют цикл.
- Намерения Intent.

## Material Design

- Цели и задачи Material Design.
- Принципы Material Design.
- Разработка приложения с помощью Material Design.

## Layout, Views и ресурсы

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

## Детальный обзор виджетов Android. Важные аспекты их создания и работы

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

## Многопоточность, AsyncTask

- Многопоточность в Java.
- Понятие многопоточности.
- Класс Thread.
- Интерфейс Runnable.
- Приоритеты потоков.
- Синхронизация потоков.
- Понятие асинхронности.
- Использование ExecutorService.

## Сохранение информации в телефоне

- Файлы. Чтение данных из файла.
- Запись данных в файл.
- Удаление файлов.
- Использование SharedPreferences для сохранения информации.

## Расширенные приемы работы с виджетами

- Понятие адаптера.
- Расширенные текстовые поля.
- Использование виджетов.

### ПРАКТИЧЕСКАЯ РАБОТА

## Меню, управляющая, 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

- Цели и задачи фрагментов.
- Создание фрагмента.
- Жизненный цикл фрагмента.
- Добавление, удаление фрагмента.
- Замена фрагмента.
- Примеры создания и использования фрагментов.
- Использование Navigation Drawer.

## Изображения, графика, анимация

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

## Сервисы, намерения Intent

- Понятие, виды служб. Связь служб и Intent
- Слушатели BroadcastReceiver.

## Основы работы 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.
- Объединения. План выполнения запроса. Оптимизация.

## Использование баз данных в Android-приложениях

- База данных SQLite. Курсор.
- Получение, добавление, обновление, удаление данных в базе данных.
- Контент-провайдер.
- Разработка приложения, которое использует источник данных.

## Адресная книга, календари, события, звонки, sms

- Адресная книга и ее использование.
- Работа со звонками и sms.
- Разработка приложений Список контактов, Будильник.

- Календарь и событие.
- Использование календаря.

## 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 Volley, Retrofit.

## Google Cloud Platform, Firebase

- Понятие Google Cloud Platform.
- Использование Firebase.

## Работа с картами Google Maps API

- Понятие 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.

## Регистрация в Google Play

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

### ИТОГОВЫЙ ПРОЕКТ