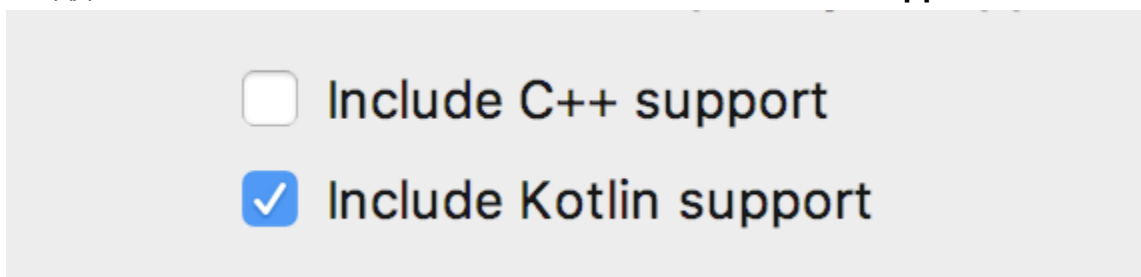


## Создание нового проекта

- 1 Откройте Android Studio
- 2 В диалоге **Welcome to Android Studio** выберите **Start a new Android Studio project**.
- 3 В диалоге **New Project**, дайте имя вашему приложению, например **My First App**.
- 4 Имя домена можете принять по умолчанию.
- 5 Для поддержки языка Котлин отметьте чекбокс **Include Kotlin support**.



Местоположение проекта можете оставить по умолчанию. Обычно проекты сохраняются в папке с именем AndroidStudioProjects в документах пользователя. Здесь могут возникнуть проблемы, если имя пользователя указано русскими буквами, потому что среда разработки не воспринимает кириллицу в пути к файлам. В таком случае вы увидите предупреждение. Сохраните проект в другом месте.

- 1 Нажмите **Next**.
- 2 В диалоге **Target Android Devices** выберите тип платформы **Phone and Tablet** и минимально поддерживаемую версию Android API 17 или выше. Поддержка более старых версий нам пока не нужна, поскольку отражается на доступном функционале инструментов разработки. Нажмите **Next**.
- 3 В диалоге добавления активности нужно выбрать шаблон предустановленных окон приложения и компонентов интерфейса. Выберите **Empty Activity**. Нажмите **Next**.
- 4 В диалоге **Configure Activity** оставьте все по умолчанию. Нажмите **Finish**.

После этих шагов, Android Studio:

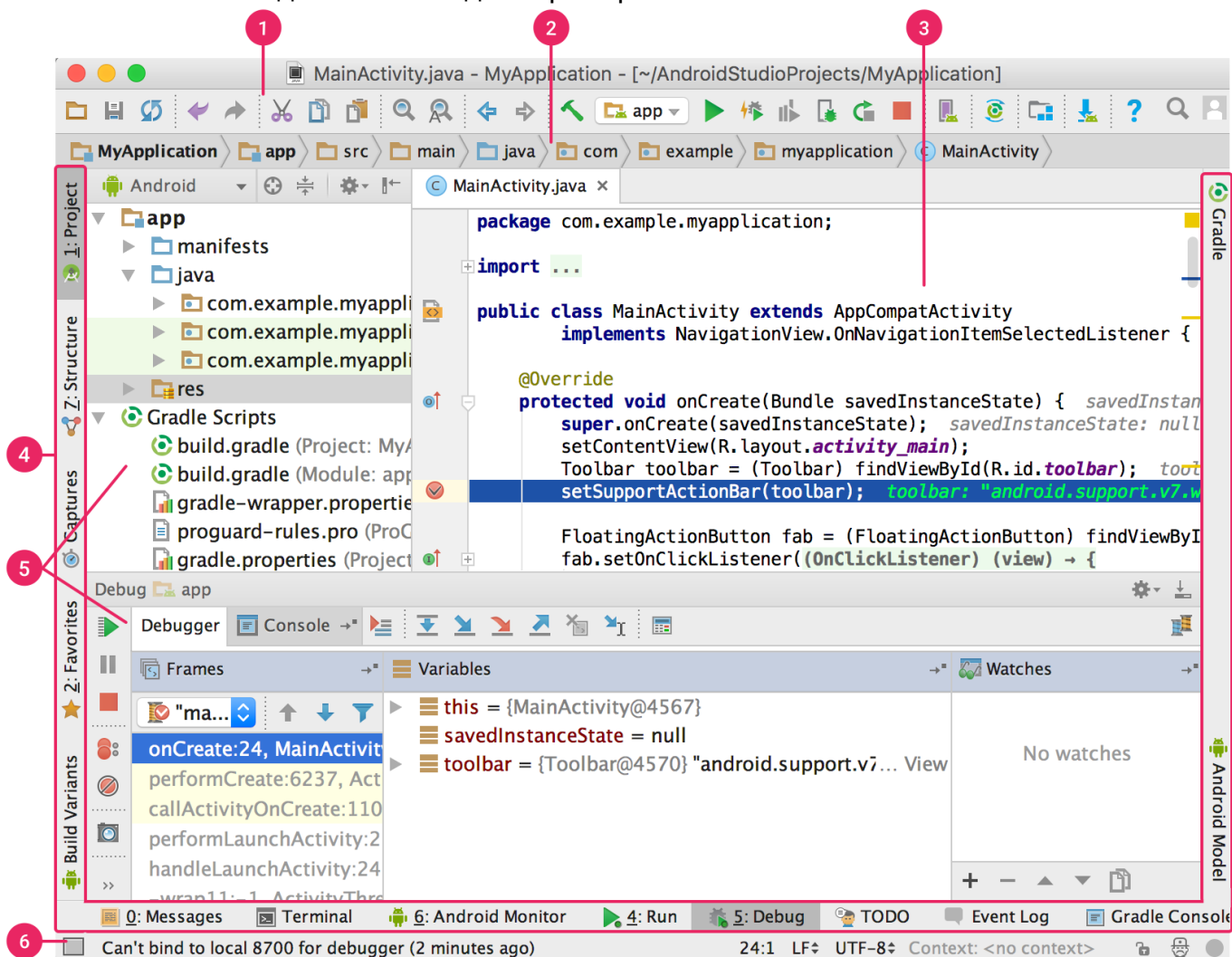
Создает папку для проекта Android Studio на диске, в местоположении, указанном при создании проекта.

Далее происходит построение проекта (это может занять несколько минут).

Android Studio использует Gradle в качестве системы построения. Иногда система сборки выдаёт сообщения и предупреждения, например сейчас нужно доустановить отсутствующий компонент SDK.

Открывает окно редактора кода, отображающее текущий проект.

Окно Android Studio должно выглядеть примерно так:



Скриншот 1. Интерфейс Android Studio

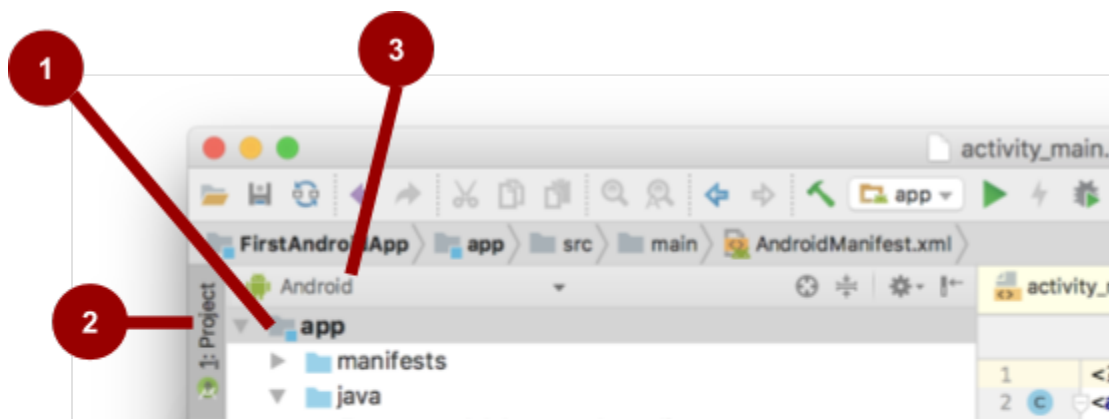
Рассмотрим интерфейс Android Studio

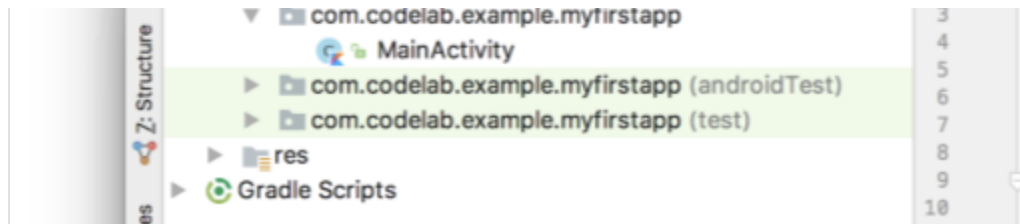
- 1 Панель **Toolbar** предоставляет быстрый доступ к самым востребованным командам, как запуск приложения или открытие SDK менеджера.
- 2 Панель **Navigation bar** помогает перемещаться по проекту и открывать файлы для редактирования. Он обеспечивает более компактный вид структуры, видимой в окне **Project**.
- 3 **Окно редактора** позволяет создавать и редактировать код. В зависимости от типа открытого файла вид редактора может измениться. Например, при просмотре файла макета редактор отображает редактор макета.
- 4 Панель **Tool buttons** расположена по периметру окна IDE и содержит кнопки, позволяющие разворачивать или сворачивать отдельные окна инструментов.
- 5 **Окна инструментов** предоставляют доступ к определенным задачам, таким как управление проектами, поиск, управление версиями и многое другое. Их можно развернуть и свернуть. Далее мы рассмотрим их более подробно.
- 6 **Status bar** отображает состояния проекта и самой IDE, а также различные предупреждения или сообщения.

Главное окно можно упорядочить, чтобы предоставить больше места на экране, скрывая или перемещая панели инструментов и окна инструментов. Для доступа к большинству функций IDE также можно использовать [сочетания клавиш](#).

В любой момент можно выполнить поиск по исходному коду, базам данных, командам, элементам пользовательского интерфейса и т. д., дважды нажав клавишу Shift или щелкнув увеличительное стекло в верхнем правом углу окна Android Studio. Это может быть очень полезно, если, например, вы пытаетесь найти конкретную команду IDE, которую вы забыли, как запустить.

Окно структуры проекта





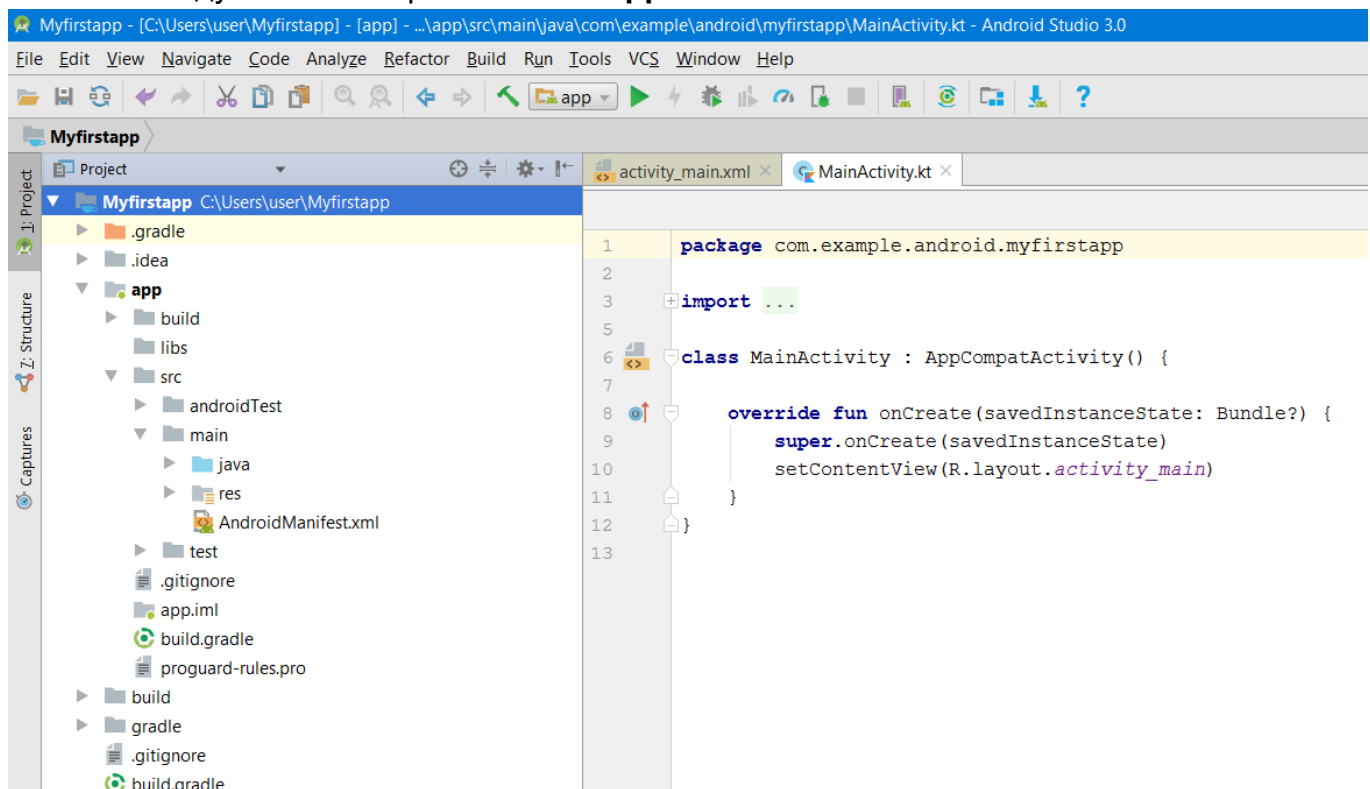
Скриншот 2. Окно структуры проекта Android Studio

## Изучение структуры и макета проекта

Можно посмотреть иерархию файлов приложения несколькими способами.

- 1) Окно структуры проекта (1) с отображением дерева файлов. (См. 1 на Скриншоте 2)
- 2) Кнопка **Project** (2) скрывает и отображает окно структуры проекта. Включите пункт меню **View > Tool Buttons** чтобы увидеть эту кнопку.
- 3) Текущий выбор вида проекта **Project > Android**. Нажмите выпадающий список **Android** (3) для просмотра других доступных видов проекта.

В виде отображения **Project > Project** вы увидите полную структуру папок проекта, как она выглядит на диске. Здесь много файлов и папок и новичку здесь можно запутаться. Главное, на что следует обратить внимание: проект имеет модульную структуру и основной модуль нашего приложения – **app**.





Скриншот 3. Окно структуры проекта Android Studio в виде Project > Project

Если раскрыть папку **app**, мы увидим такие папки:

**build** — здесь файлы, создаваемые системой в процессе компиляции, лучше ничего там не менять

**libs** — папка для сторонних библиотек, подключаемых в проект

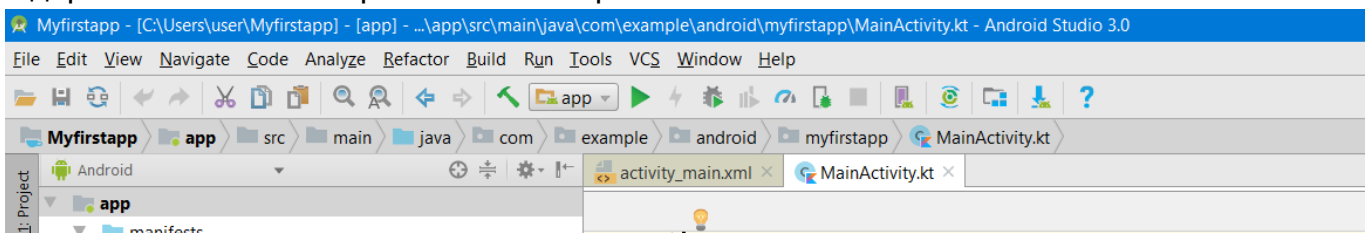
**src** — папка для исходного кода и ресурсов

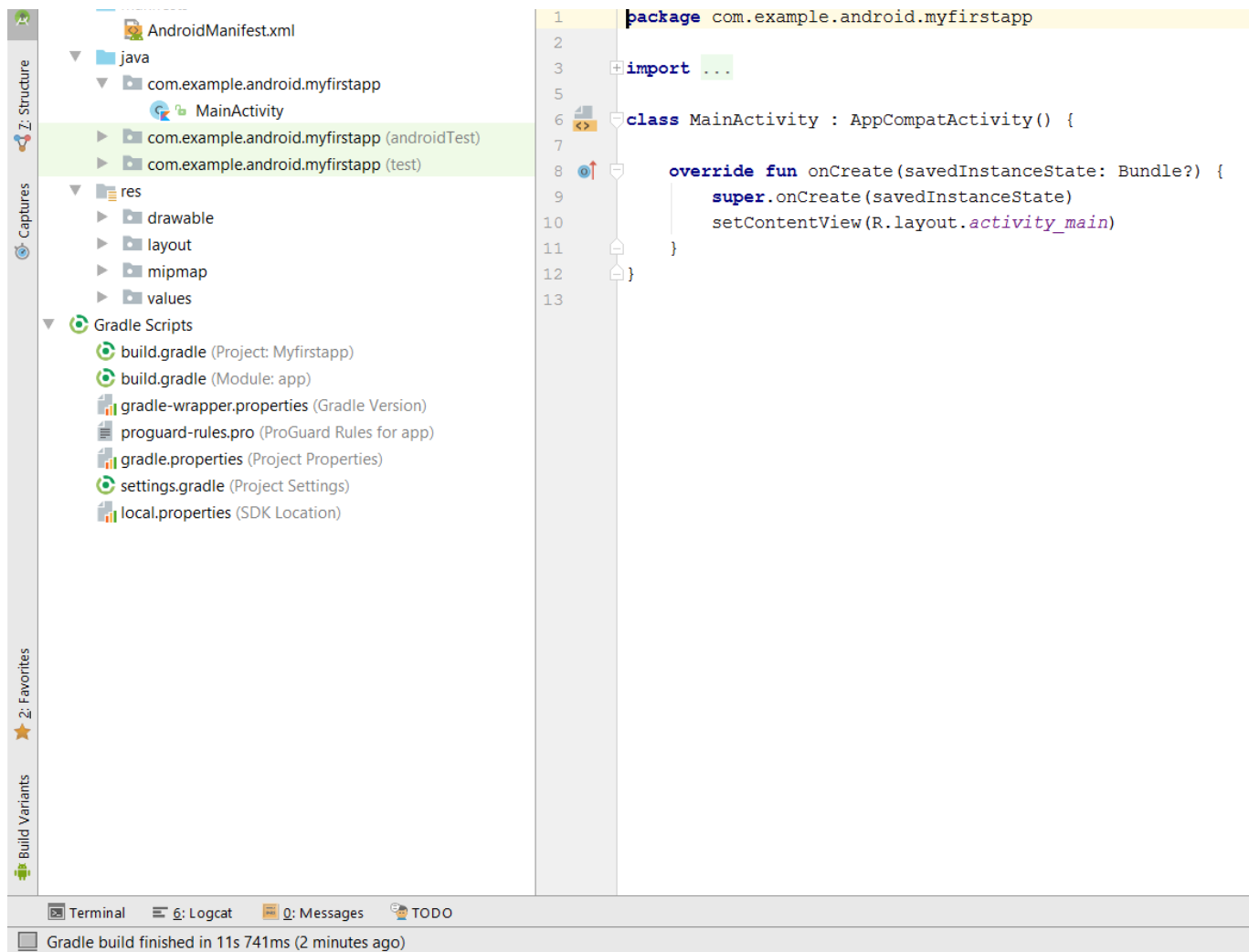
внутри **src** находится папка **main** — это основная рабочая папка, с которой мы будем иметь дело

внутри **main** находятся папки **java** и **res** — это папки для кода и ресурсов, рассмотрим их позже

Также в папке модуля **app** имеется файл системы сборки **build.gradle**, мы рассмотрим его позже.

Если переключиться на вид проекта **Project > Android**, вы увидите только папку модуля **app** с упрощенной структурой проекта. Это представление удобнее, поскольку здесь отфильтрованы только те файлы, с которыми мы непосредственно будем работать. Но следует помнить, что в таком виде структура папок проекта не соответствует той, что хранится на диске. Например, вы не найдете на диске в папке проекта папку **manifests**. А в дереве папок в таком режиме не отображаются папки **src** и **main**.





Скриншот 4. Окно структуры проекта Android Studio в виде Project>Android

В виде **Project > Android** вы увидите в папке **app** такие папки: **manifests**, **java**, и **res**.

- 1 Раскройте папку **manifests**. Эта папка содержит файл AndroidManifest.xml. Этот файл описывает все компоненты приложения Android и считывается системой среды выполнения Android при запуске приложения.
- 2 Раскройте папку **java**. Все файлы кода на языке Kotlin и Java организованы здесь. Папка **java** содержит три подпапки: **com.example.android.myfirstapp** (или указанное имя домена): Эта папка содержит файлы исходного кода Kotlin и Java для вашего приложения. С этой папкой мы будем работать большую часть времени, поскольку это главный пакет проекта.

**com.example.android.myfirstapp (androidTest)**: эта папка для размещения инструментальных тестов.

**com.example.android.myfirstapp (test):** эта папка для размещения модульных тестов.

Подробнее о тестах вы можете узнать из [Курса по архитектуре андроид-приложений](#).

- 1 Раскройте папку **res**. Эта папка содержит все ресурсы для вашего приложения, включая изображения, файлы макетов экранов, строковые ресурсы, значки и иконки, цвета и стили оформления. Она включает такие вложенные папки:**drawable**: Все изображения для вашего приложения будут сохранены в этой папке.**layout**: Эта папка содержит файлы макета для активности – экранов приложения. В настоящее время ваше приложение имеет одну активность с файлом макета `activity_main.xml`.**mipmap**: Эта папка содержит значки запуска приложения – это те значки, которые отображаются на андроид-устройстве после установки приложения.**values**: Содержит ресурсы, такие как строки и цвета, используемые в приложении.

Также в окне структуры проекта есть папка **Gradle Scripts**. Раскройте ее. В начале списка есть два файла с одинаковыми именами `build.gradle`. Это файлы сборки для системы Gradle, которая используется для компиляции, построения и упаковки приложений и библиотек. В скобках после имени файлов указана принадлежность файла модулю App или всему проекту. Файл сборки уровня проекта содержит настройки для всего проекта, а файл сборки уровня модуля содержит настройки для модуля. Чаще всего мы будем работать именно с файлом сборки уровня модуля. Он содержит такие основные секции, как:

**android{...}**, где указаны версии инструментов разработки, минимальная поддерживаемая версия API, идентификатор и версия приложения для Google Play, и другие параметры

**dependencies{...}**, которая содержит список библиотек, подключаемых к проекту. Могут быть подключены как локальные (помещенные в папку **libs**), так и хранящиеся удаленно библиотеки.

## Запуск android-приложения

Наиболее очевидный способ проверить работу приложения в процессе разработки – установить и запустить его на андроид-устройстве.

Для запуска приложения из среды разработки Android Studio мы можем использовать 2 способа:

подключенный к ПК смартфон или планшет под управлением системы Android

эмулятор Android

Оба этих способа имеют свои преимущества и недостатки, и какой использовать, выбирать вам. Для начинающего разработчика подойдет любой способ.

Что такое андроид-смартфон, вы наверняка уже знаете, раз заинтересовались темой разработки приложений. А что такое эмулятор Android? Это запуск системы Android на ПК в отдельной виртуальной машине.

Существует много эмуляторов Android, например **BlueStacks**, **Droid4X**, **Nox APP Player**, **Genymotion**, и другие. Большинство из них разработаны для игр и не очень подходят для тестирования приложений, кроме **Genymotion**.

В Android Studio есть собственный эмулятор, который может запускать образы виртуальных устройств с различными характеристиками, таких как смартфоны Nexus и Pixel, а также планшеты наиболее распространенных типов.

Создание виртуального android-устройства (эмулятора)

В Android Studio есть утилита **Android Virtual Device (AVD) manager** для создания виртуального устройства (также известного как эмулятор), моделирующего конфигурацию определенного типа андроид-устройства.

Первым шагом является создание конфигурации, описывающей виртуальное устройство.

- 1 В главном меню Android Studio, выберите **Tools > Android > AVD Manager**, или нажмите иконку AVD Manager в тулбаре



- 2 Нажмите кнопку **+Create Virtual Device**. (Если виртуальное устройство создано ранее, в окне отображаются все существующие устройства и кнопка **+Create Virtual Device** находится внизу.) В диалоговом окне **Select Hardware** отображается список предварительно настроенных типов аппаратных устройств.
- 3 Выберите устройство, например Nexus 5, и нажмите **Next**.



- 4 В окне **System Image**, на вкладке **Recommended**, выберите нужную версию.
- 5 Если ссылка **Download** отображается рядом с версией, значит она еще не установлена и требуется загрузить ее. При необходимости щелкните ссылку, чтобы начать загрузку, и нажмите кнопку **Next** по окончании загрузки.
- 6 В следующем диалоговом окне примите значения по умолчанию и нажмите кнопку **Finish**.
- 7 Если окно **Your Virtual Devices** AVD Manager еще открыто, закройте его.

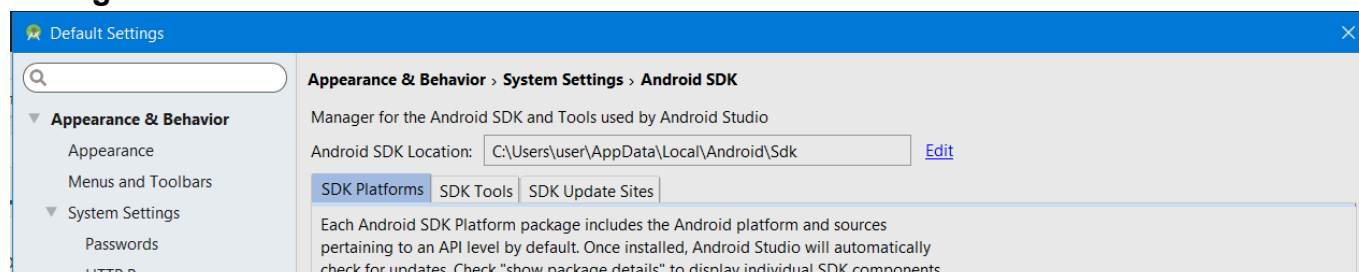
## Возможные проблемы

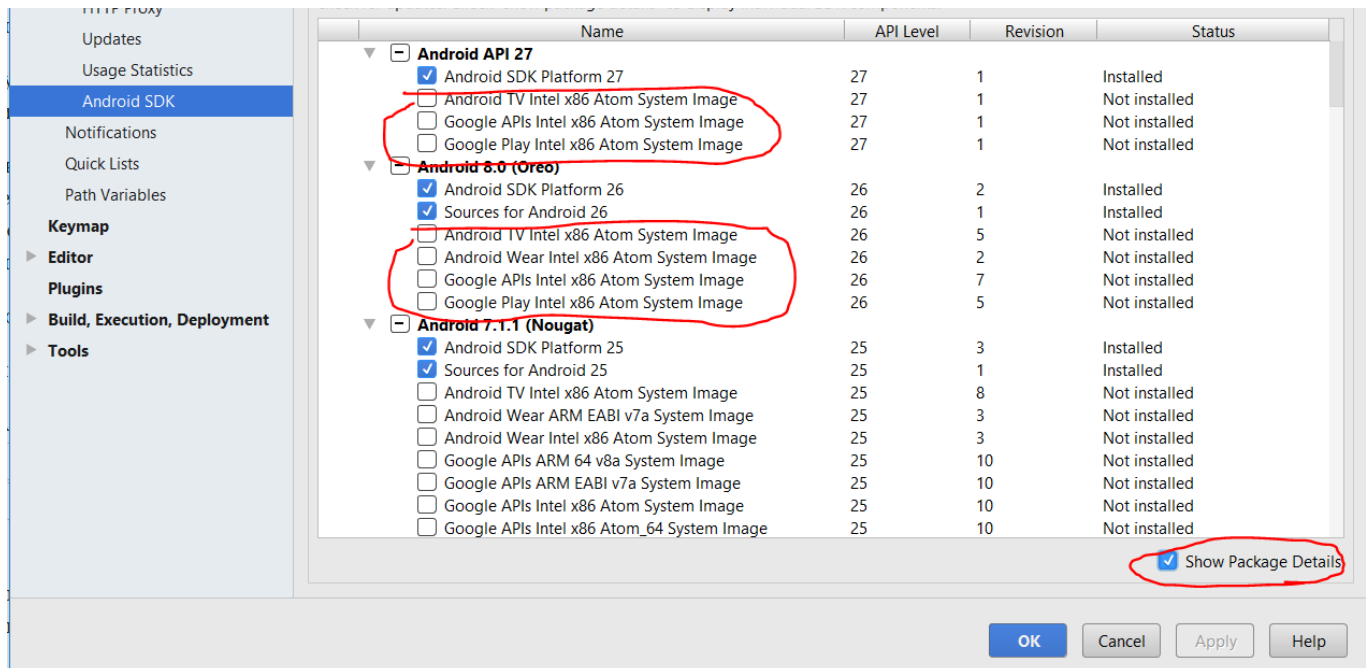
Если у вас на компьютере установлена операционная система Windows, а процессор AMD, то скорее всего вы увидите такое предупреждение в окне выбора оборудования для виртуального устройства. Дело в том, что образы виртуальных устройств с типом **x86** не работают на связке Windows + AMD. Причем на связке Linux + AMD таких проблем нет. Процессоры Intel поддерживают образы **x86** на любой операционной системе. Для Windows + AMD вы можете использовать образы типа **arm**, но они гораздо более медленные. Выходом будет использование реального устройства – андроид-смартфона или планшета, подключенного к компьютеру.

Другой распространенной проблемой является отключенная виртуализация процессора. Включить виртуализацию можно в биосе, в интернете можно найти много инструкций, как это делается.

Также нужно в SDK Manager на вкладке SDK Tools установить Intel x86 Emulator Accelerator для лучшей производительности и скорости работы. Только для процессоров Intel.

Если в окне выбора системного образа нет активных образов для установки, загрузите образ необходимой версии Android в SDK менеджере. Откройте окно **SDK Manager** и на вкладке **SDK Platforms** справа внизу установите флаг **Show Package Details** для отображения всех компонентов платформы (см. Скриншот 5). Отметьте для загрузки нужный файл **System Image**. После загрузки выбранный образ будет доступен в окне **AVD Manager**.





Скриншот 5. Установите флаг **Show Package Details** для отображения всех компонентов платформы

Запуск приложения на эмуляторе

- 1 В Android Studio, выберите в главном меню команду **Run > Run app** или нажмите иконку **Run** в тулбаре.

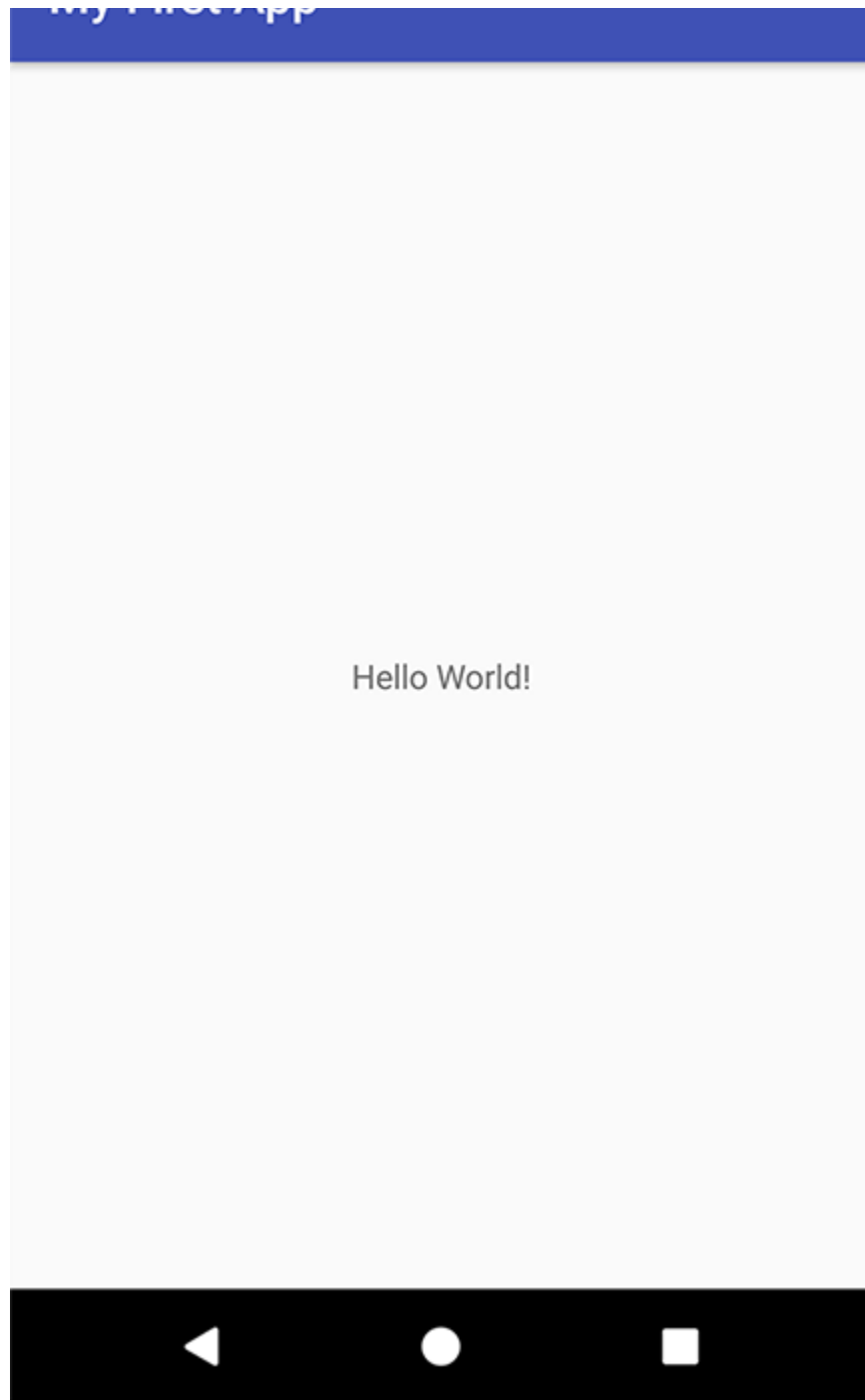


- 2 В окне **Select Deployment Target**, под **Available Virtual Devices**, выберите только что настроенное виртуальное устройство и щелкните **OK**.

Эмулятор запускается и загружается как физическое устройство. В зависимости от скорости компьютера это может занять некоторое время. Можно посмотреть в маленькую горизонтальную строку состояния в самом низу окна Android Studio, чтобы увидеть ход выполнения.

После того как приложение будет построено и эмулятор будет готов, среда Android Studio установит приложение в эмулятор и запустит его. Вы увидите ваше приложение, как показано на следующем снимке экрана.





**Замечание:** рекомендуется запускать эмулятор в начале сеанса. Не закрывайте эмулятор до тех пор, пока не будет выполнена проверка приложения, чтобы не дожидаться повторной загрузки эмулятора. Кроме того, не следует одновременно запускать несколько эмуляторов, чтобы уменьшить использование памяти.

## Запуск приложения на реальном андроид-устройстве

Что вам нужно:

Устройство на Android, например, смартфон или планшет.

Кабель USB для подключения устройства Android к компьютеру.

## Запуск приложения на устройстве

- 1 Подключите ваш смартфон к компьютеру с помощью кабеля USB. Если вы разрабатываете на Windows, вам может потребоваться установить соответствующий драйвер USB для вашего устройства. Для получения справки по установке драйверов см. [Драйвера OEM USB](#).
- 2 Включите **Отладка через USB** на вашем устройстве.

На большинстве устройств под управлением Android 3.2 или старше, вы можете найти опцию в меню **Настройки > Приложения > Разработка**.

На Android 4.0 и новее, это находится в **Настройки > Для разработчиков**.

**Примечание:** На Android 4.2 и новее, пункт меню **Для разработчиков** по умолчанию скрыт. Чтобы сделать его доступным, перейдите к **Настройки > О телефоне** и нажмите **Номер сборки** семь раз. Вернитесь к предыдущему экрану, чтобы найти пункт меню **Для разработчиков**.

Чтобы запустить приложение из **Android Studio**:

Откройте один из файлов вашего проекта и нажмите **Run app** на панели инструментов. Или нажмите комбинацию клавиш **Shift+F10**.

Откроется окно выбора подключенного реального устройства или эмулятора. Выберите ваше устройство и нажмите **ОК**.

Android Studio устанавливает приложение на выбранное устройство и запускает его.

Это простое приложение отображает строку «Hello World» на экране виртуального или физического устройства Android.

**Note:** Если на устройстве установлена платформа Android, которая не установлена в среде Android Studio, может быть выведено сообщение с запросом о необходимости установки необходимой платформы. Нажмите **Install and Continue**, затем

кнопку **Finish** по окончании установки.

## Неполадки

В случае неполадок при запуске приложения в нижней части окна Android Studio открывается окно **Logcat**, где можно увидеть текст ошибки и возможное решение проблемы. Если вам непонятно, о чем там написано, попробуйте загуглить часть текста ошибки. Очень вероятно, что вы найдете в результатах поиска решение вашей проблемы на форумах для программистов.

Если Приложение не запускается, закройте среду Android Studio и перезапустите ее.

Если среда разработки Android Studio не распознает устройство, попробуйте выполнить следующие действия.

- ① Отключите устройство от компьютера и снова подключите его.
- ② Перезапустите среду Android Studio.

Если компьютер по-прежнему не находит устройство или объявляет его «неавторизованным»:

- ① Отключите устройство.
- ② На устройстве, откройте в меню «Параметры разработчика»
- ③ Проверьте включение отладки USB.
- ④ Подключите устройство к компьютеру.
- ⑤ При появлении соответствующего запроса на устройстве предоставьте необходимые разрешения.