

Android 101

Цэл курса

- Стартовый проект
- Соединение с интернетом
- RecyclerView
- Intents
- Жизненный Цикл
- Экран свойств
- Хранение данных в SQLite
- Content providers:
использование и создание
- Background задачи
- Улучшение UI

[goo.gl/KAefqs](https://classroom.udacity.com/courses/ud851)

<https://classroom.udacity.com/courses/ud851>

Sunshine

Today, July 1


 18°
13°

Light Rain

	Tomorrow Moderate Rain	17° 13°
	Sunday Clear	16° 12°
	Monday Scattered Clouds	12° 12°
	Tuesday Broken Clouds	15° 15°
	Wednesday Light Snow	-1° -2°
	Thursday Snow	-1° -2°

Details

Sunday, July 3

 16°
12°

Clear

Humidity	90 %
Pressure	996 hPa
Wind	8 km/h E

Settings

Location
Mountain View, CA 94043

Temperature Units
Metric

Weather Notifications
Enabled

<https://github.com/udacity/ud851-Sunshine>

<https://github.com/udacity/ud851-Exercises>



Android Studio

Version 2.2 RC

-  **Start a new Android Studio project**
-  Open an existing Android Studio project
-  Check out project from Version Control ▾
-  Import project (Eclipse ADT, Gradle, etc.)
-  Import an Android code sample



New Project

Android Studio

Configure your new project

Application name:

Company Domain:

Package name: [Edit](#)

Include C++ Support

Project location: ...





Target Android Devices

Select the form factors your app will run on

Different platforms may require separate SDKs

Phone and Tablet

Minimum SDK

Lower API levels target more devices, but have fewer features available.
By targeting API 15 and later, your app will run on approximately 97.4% of the devices that are active on the Google Play Store.

[Help me choose](#)

Wear

Minimum SDK

TV

Minimum SDK

Android Auto

Glass

Minimum SDK

Cancel

Previous

Next

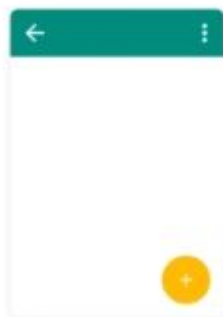
Finish



Add an Activity to Mobile



Add No Activity



Basic Activity



Empty Activity



Fullscreen Activity



Google AdMob Ads Activity



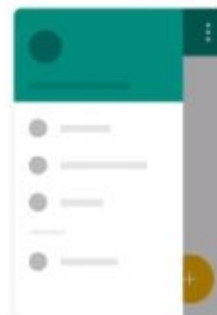
Google Maps Activity



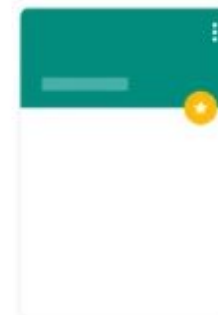
Login Activity



Master/Detail Flow



Navigation Drawer Activity



Scrolling Activity

Cancel

Previous

Next

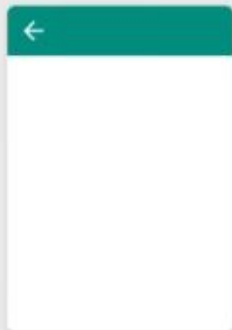
Finish



Customize the Activity



Creates a new empty activity



Empty Activity

Activity Name: MainActivity

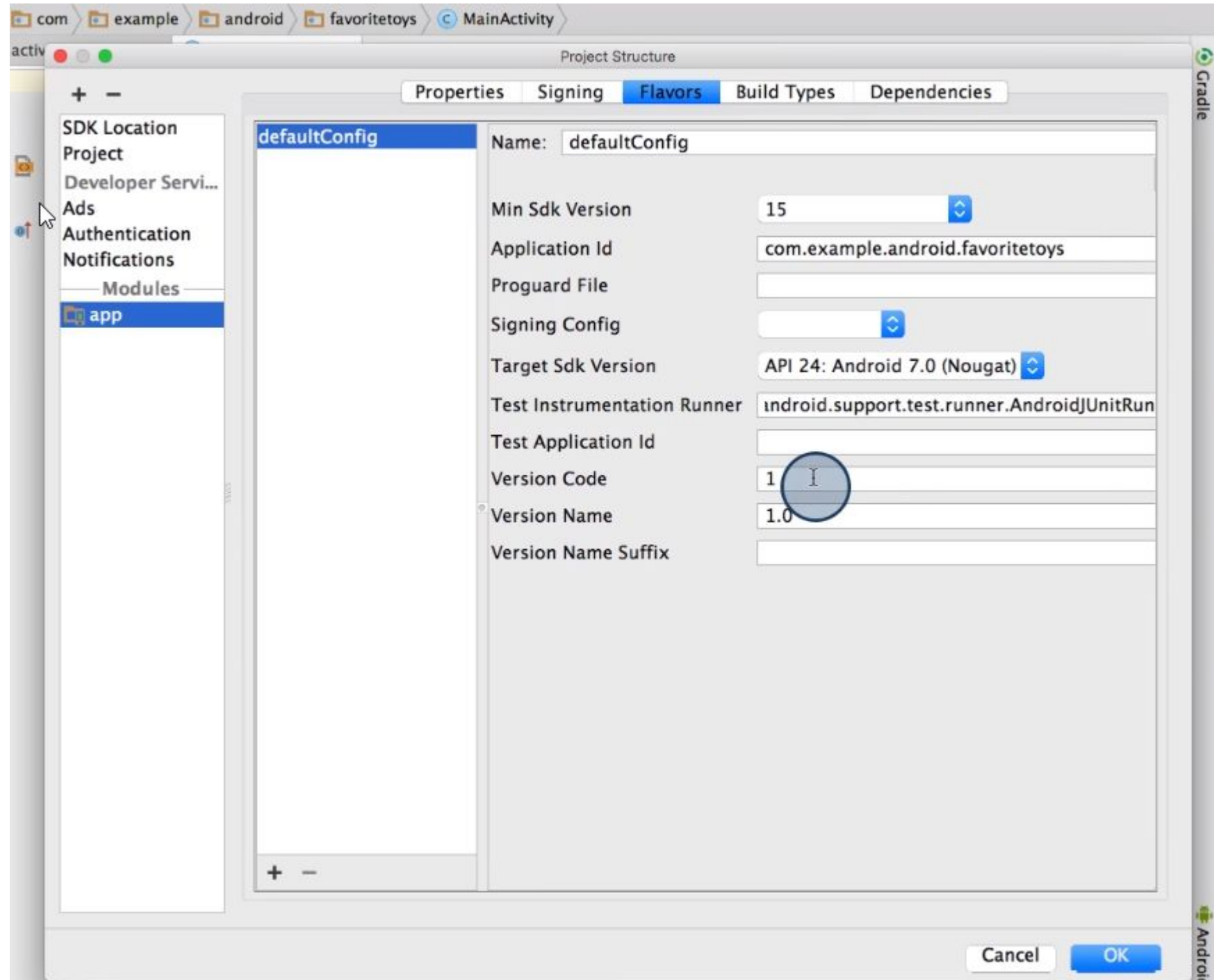
Generate Layout File

Layout Name: activity_main

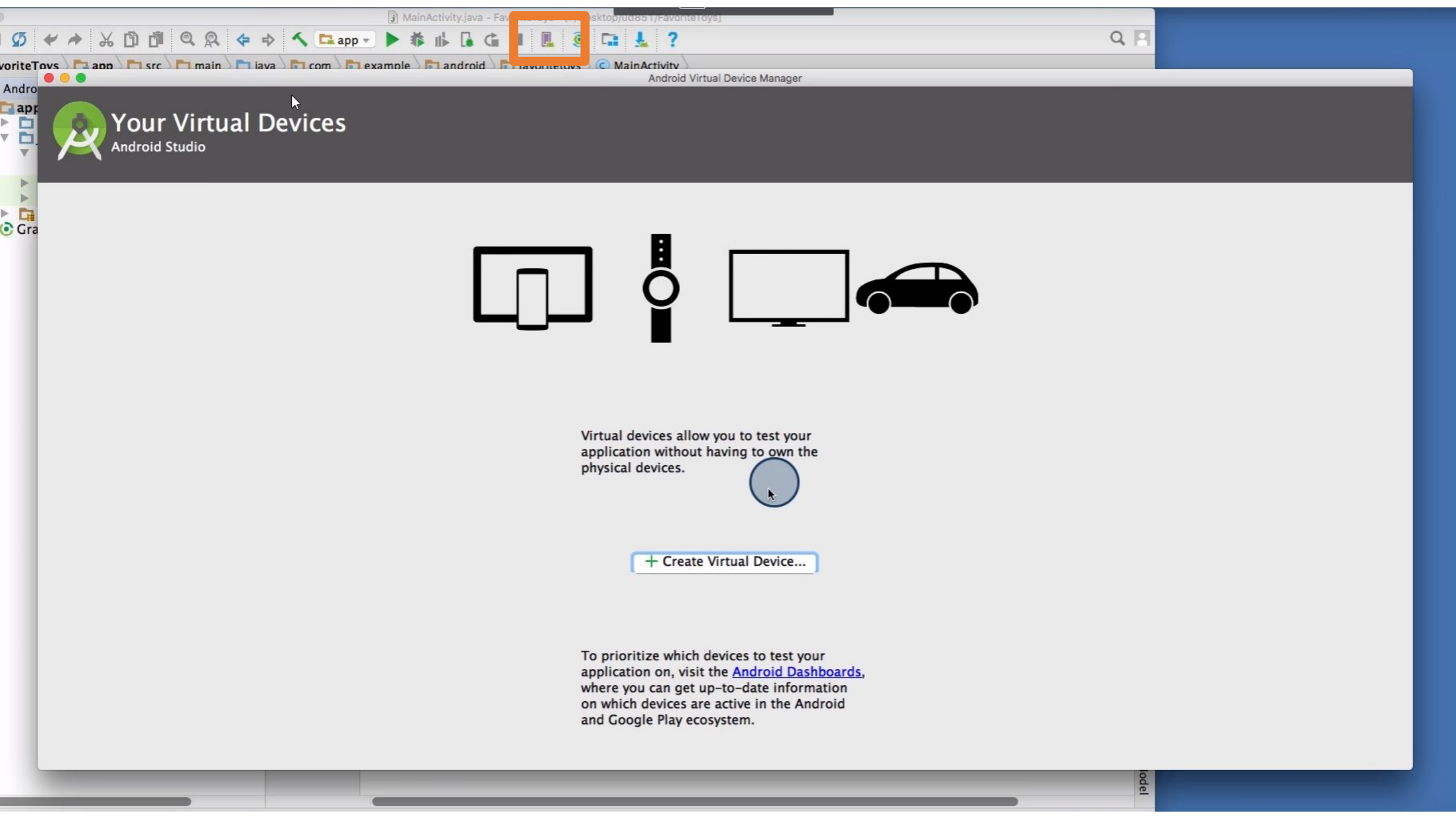
Backwards Compatibility (AppCompat)

Настройки

- Project Structure
 - App
 - Flavours



- minSDK версия – самая маленькая, которую поддерживаем
 - На устройствах с версией ниже это приложение не будет видно в Gplay
- targetSDK защищает наоборот от более новых версий



Your Virtual Devices

Android Studio



Virtual devices allow you to test your application without having to own the physical devices.



[+ Create Virtual Device...](#)

To prioritize which devices to test your application on, visit the [Android Dashboards](#), where you can get up-to-date information on which devices are active in the Android and Google Play ecosystem.



Select Hardware

Android Studio

Choose a device definition

Category	Name	Size	Resolution	Density
TV	Nexus S	4.0"	480x800	hdpi
Wear	Nexus One	3.7"	480x800	hdpi
Phone	Nexus 6P	5.7"	1440x2...	560dpi
Tablet	Nexus 6	5.96"	1440x2...	560dpi
	Nexus 5X	5.2"	1080x1...	420dpi
	Nexus 5	4.95"	1080x1...	xxhdpi
	Nexus 4	4.7"	768x1280	xhdpi
	Galaxy Nexus	4.65"	720x1280	xhdpi
	5.4" FWVGA	5.4"	480x854	mdpi
	5.1" WVGA	5.1"	480x800	mdpi
	4.7" WXGA	4.7"	720x1280	xhdpi
	4.65" 720p (Galaxy Nex...	4.65"	720x1280	xhdpi

Nexus 5

Size: normal
Ratio: long
Density: 420dpi

Buttons: New Hardware Profile, Import Hardware Profiles, Clone Device...



System Image

Android Studio

Select a system image

Recommended **x86 Images** Other Images

Release Name	API Level	ABI	Target
Nougat	24	x86_64	Android 7.0
Nougat Download	24	x86	Android 7.0
Marshmallow Download	23	x86_64	Android 6.0
Marshmallow Download	23	x86	Android 6.0
Lollipop Download	22	x86_64	Android 5.1
Lollipop Download	22	x86	Android 5.1
Lollipop Download	21	x86_64	Android 5.0 (with Google
Lollipop Download	21	x86	Android 5.0 (with Google
Lollipop Download	21	x86_64	Android 5.0
Lollipop Download	21	x86	Android 5.0
KitKat Download	19	x86	Android 4.4 (with Google
KitKat Download	19	x86	Android 4.4
Jelly Bean Download	18	x86	Android 4.3 (with Google
Jelly Bean Download	18	x86	Android 4.3
Jelly Bean Download	17	x86	Android 4.2 (with Google
Jelly Bean Download	17	x86	Android 4.2
Jelly Bean Download	16	x86	Android 4.1 (with Google

Nougat



API Level

24

Android

7.0

**Android Open
Source Project**

System Image

x86_64

Recommendation

Consider using a system image with Google APIs to enable testing with Google Play Services.

Questions on API level?

See the [API level distribution chart](#)



Cancel

Previous

Next

Finish



Your Virtual Devices

Android Studio

Type	Name	Resolution	API	Target	CPU/ABI	Size on Disk	Actions
	Nexus 5X API 24	1080 x 1920: 420dpi	24	Android 7.0	x86...	650 MB	

[+ Create Virtual Device...](#)



5554:Nexus_5X_API_24

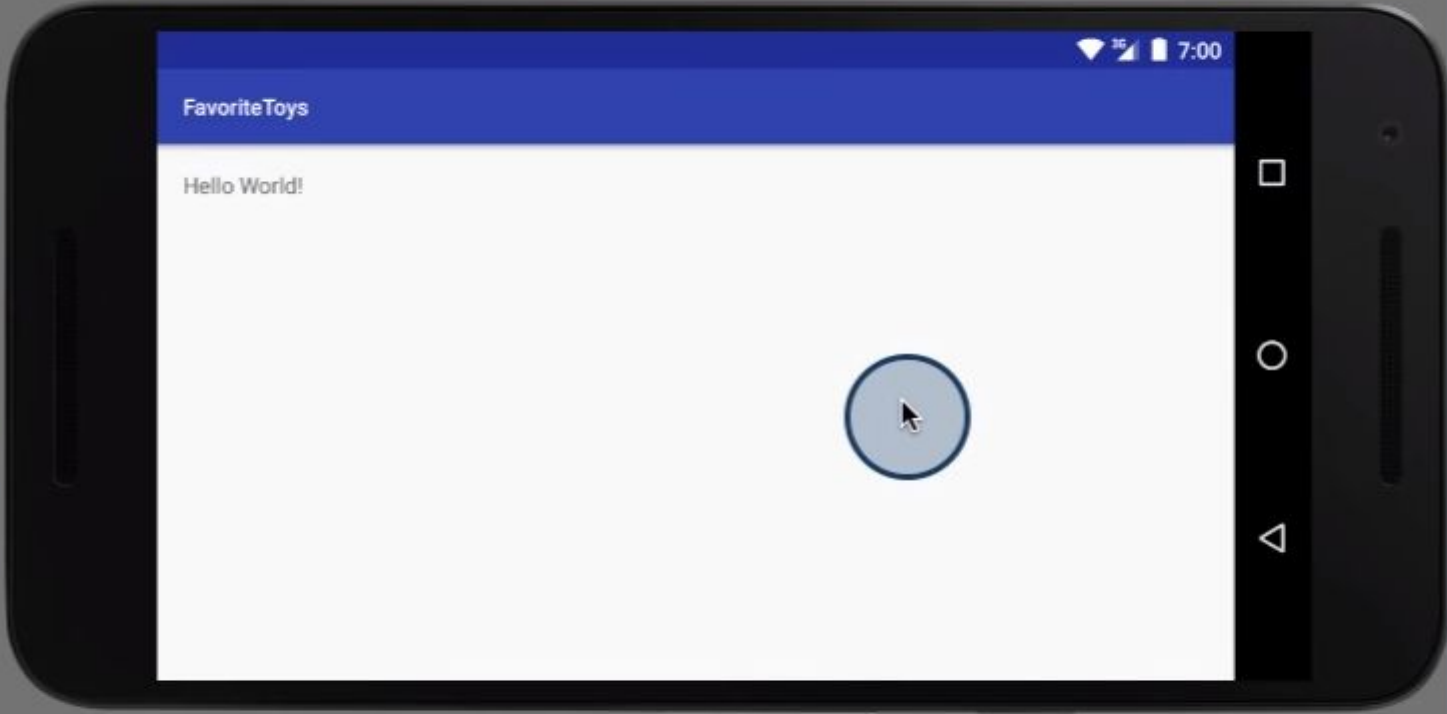


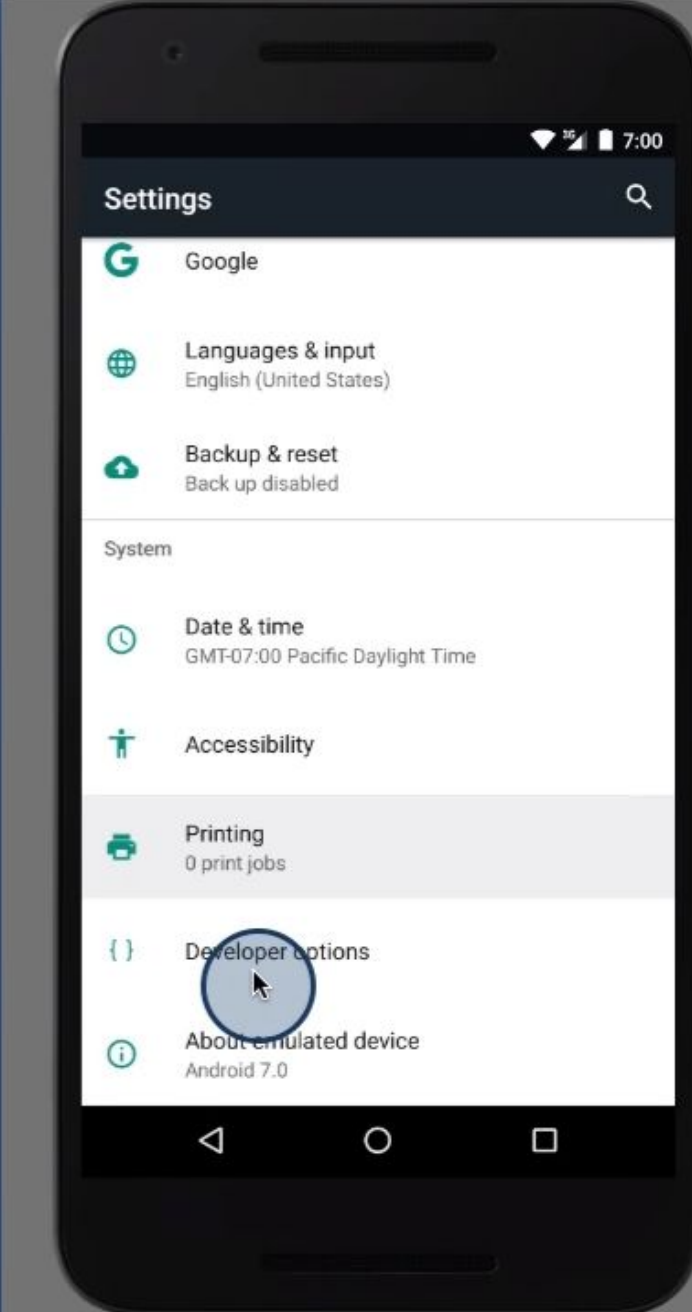
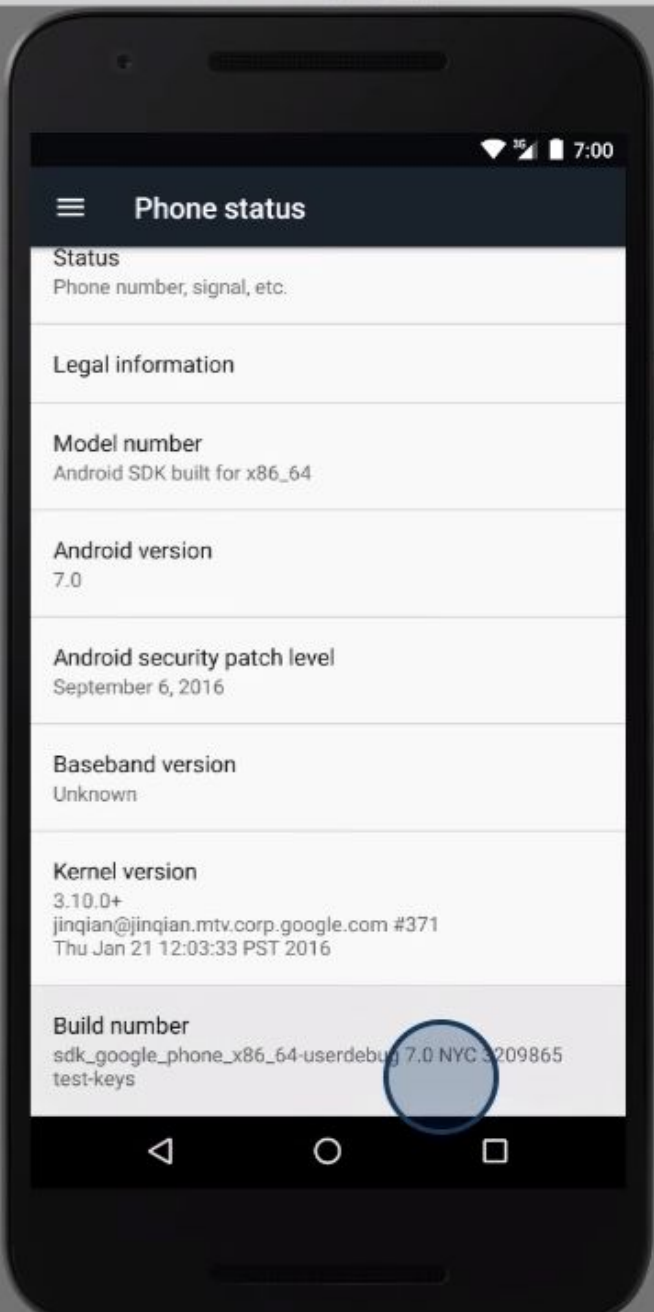
te toys > MainActivity >

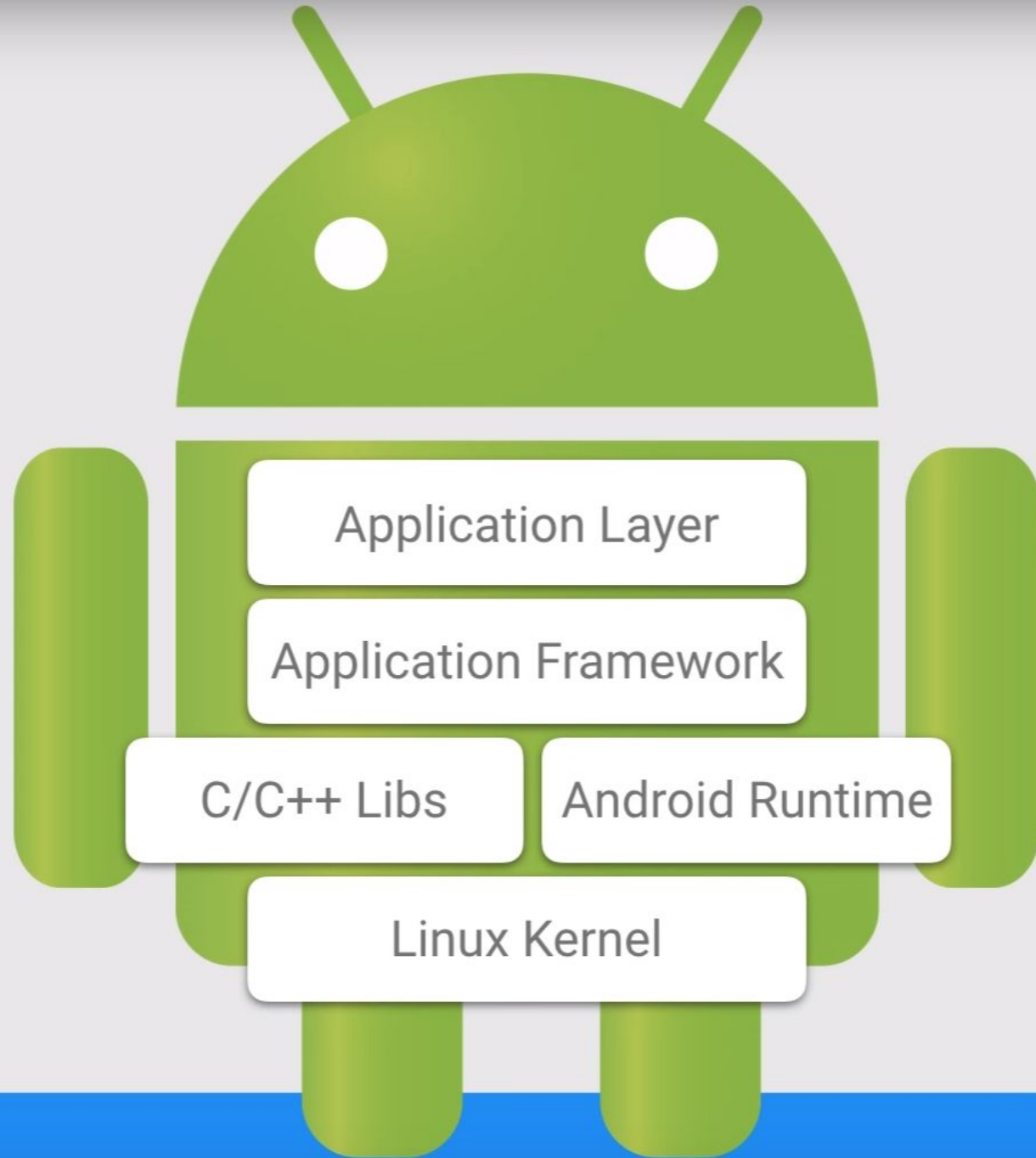
exa

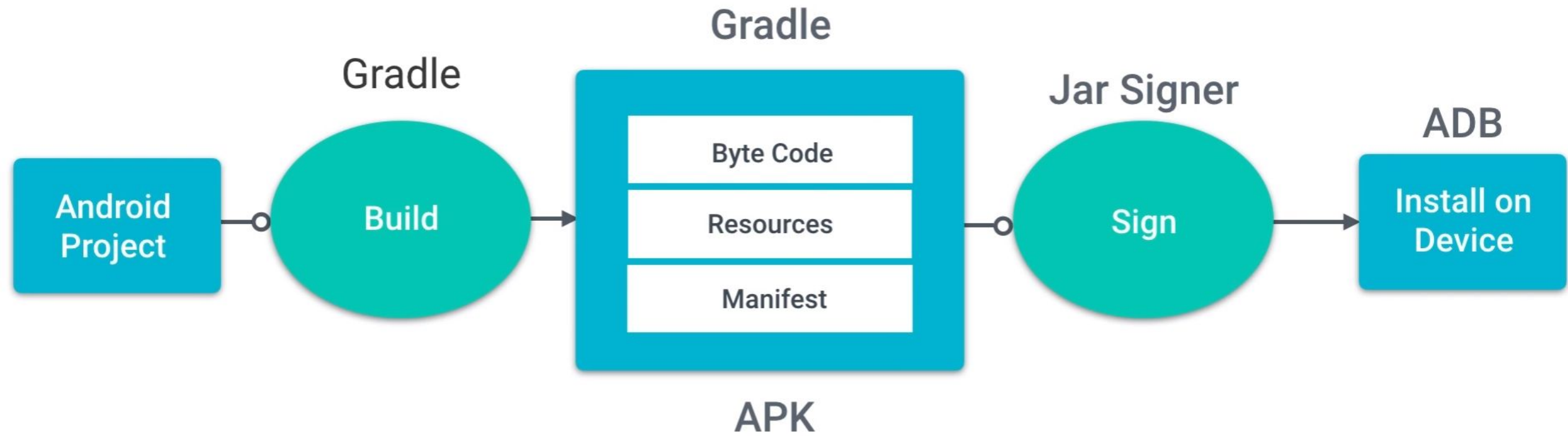
5554:Nexus_5X_API_24

Ma
le
ed v
r.c
ont









11/14 08:29:16: Launching app

```
$ adb push /Users/clamps/Downloads/ud851-Exercises-student/Lesson01-Favorite-Toys/T01.03-Solution-AddScrolling/app/build/outputs/apk/app-debug.apk /data/local/tmp/
```

```
$ adb shell pm install -r "/data/local/tmp/com.android.example.favoritetoys"
```

Success

```
$ adb shell am start -n "com.android.example.favoritetoys/com.example.android.favoritetoys.MainActivity" -a android.intent.action.MAIN -c android.intent.category.LAUNCHER
```

Client not ready yet..Waiting for process to come online

Connected to process 3427 on device Nexus_5X_API_25 [emulator-5554]

W/System: ClassLoader referenced unknown path: /data/app/com.android.example.favoritetoys-1/lib/x86_64

W/art: Before Android 4.1, method android.graphics.PorterDuffColorFilter android.support.graphics.drawable.VectorDrawableCompat.updateTintFilter(android.graphics.PorterDuffColorFilter, android.content.res.ColorStateList, android.graphics.PorterDuffColorFilter)

I/OpenGLRenderer: Initialized EGL, version 1.4

D/OpenGLRenderer: Swap behavior 1

Apps are...

Collections of
Connected
Components

What
is an
Activity?

Activity



Service



Content Provider



Broadcast Receiver



Activity



**Single
focused**
thing that the
user can do.

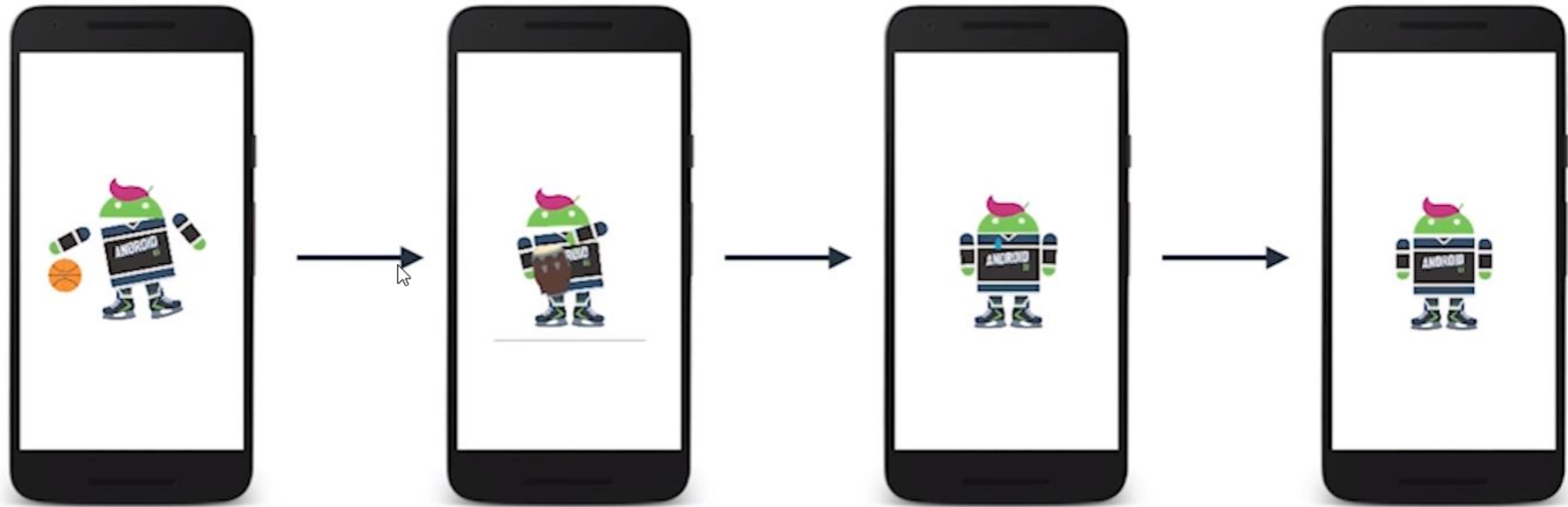


Activity

Responsible for creating the window that your application uses to draw and receive events from the system

Single
focused

Doing what the
user can do.



Android Manifest



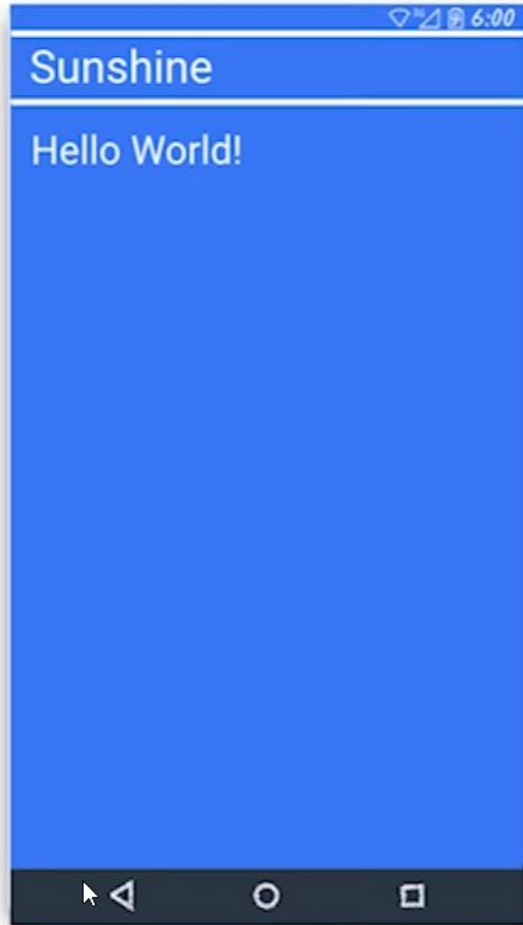
```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.android.favoritetoys">

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name="com.example.android.favoritetoys.MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN"/>
                <category android:name="android.intent.category.LAUNCHER"/>
            </intent-filter>
        </activity>
    </application>

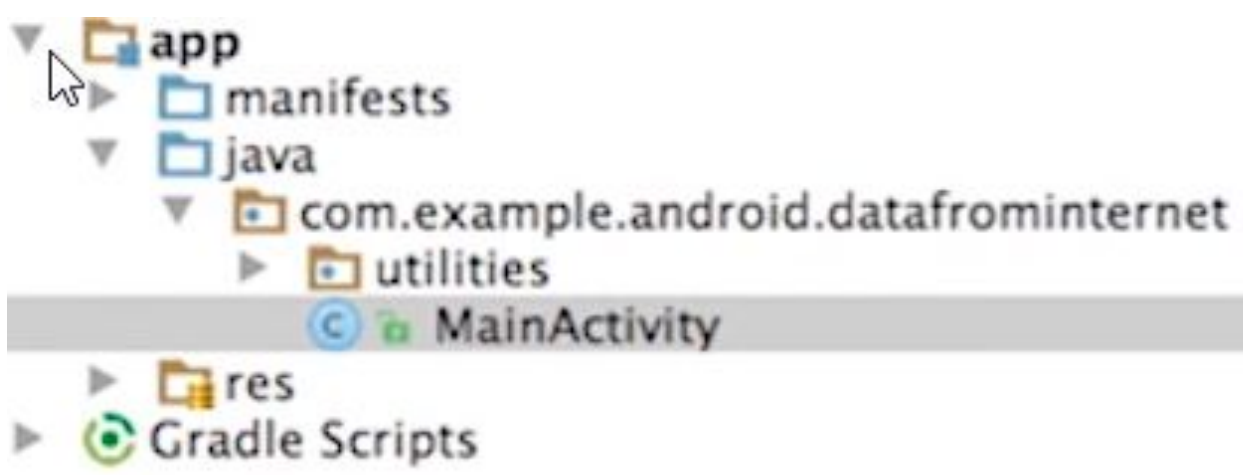
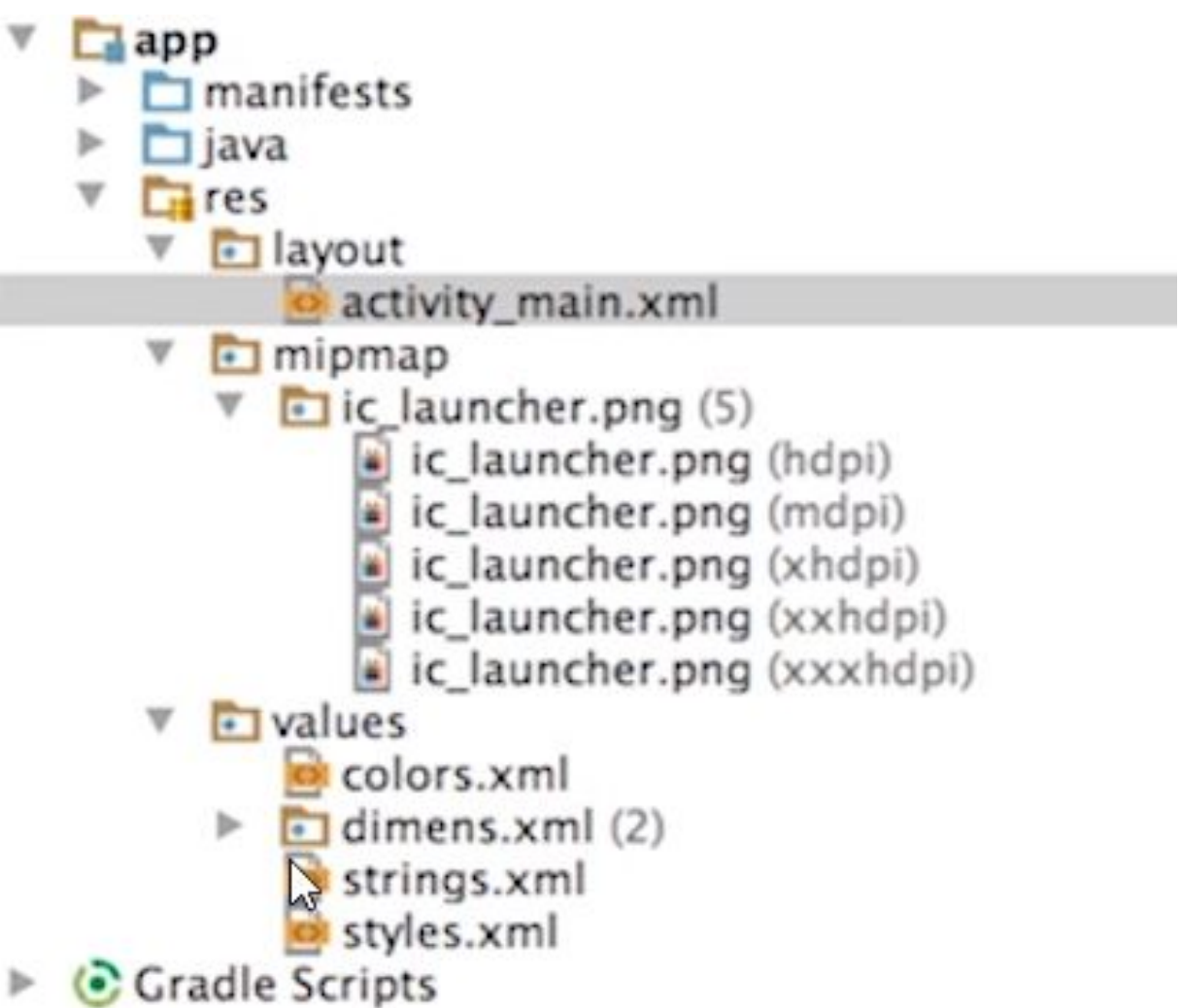
</manifest>
```

Android Framework

Text View ▶



— User Interface



`@Override`

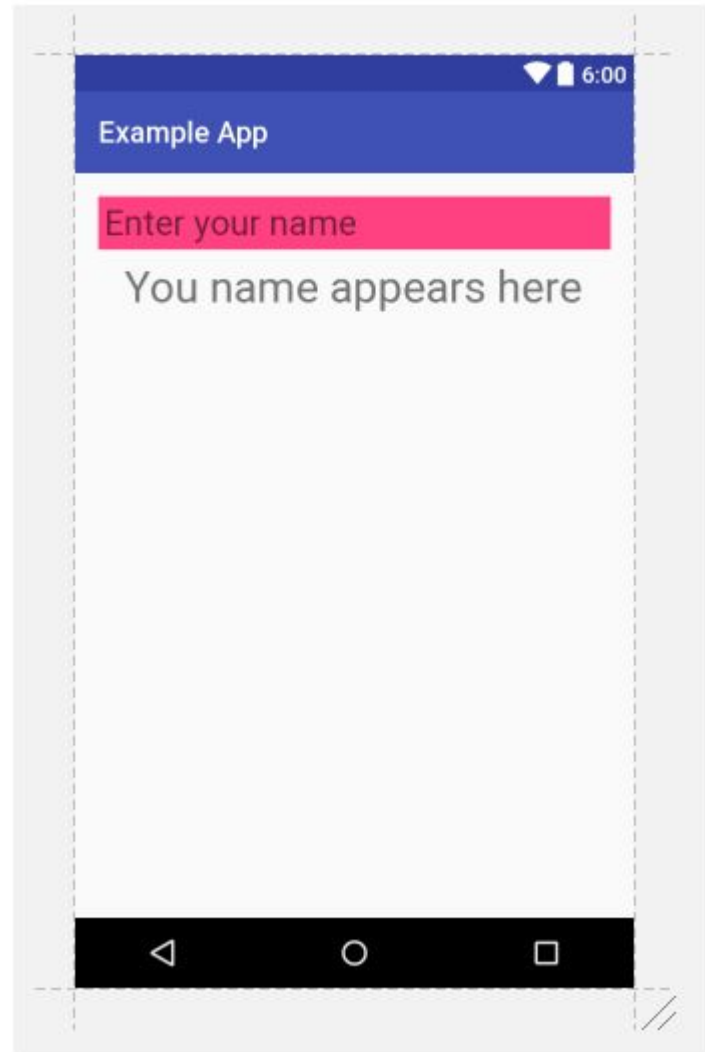
```
protected void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    setContentView(R.layout.activity_main);  
  
    // TODO (29) Use findViewById to get a reference to  
    mSearchBoxEditText = (EditText) findViewById(R.id.  
  
    // TODO (30) Use findViewById to get a reference to  
    mUrlDisplayTextView = (TextView) findViewById(R.id.  
    // TODO (31) Use findViewById to get a reference to  
    mSearchResults = (TextView) findViewById(R.id.tv_g  
}
```

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="16dp"
    tools:context="com.example.android.exampleapp.MainActivity">

    <EditText
        android:id="@+id/edit_text_name_input"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="@color/colorAccent"
        android:hint="Enter your name"
        android:padding="4dp"
        android:textSize="24sp" />

    <TextView
        android:id="@+id/text_view_name_display"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:layout_marginTop="8dp"
        android:text="Your name appears here"
        android:textSize="30sp" />

</LinearLayout>
```



UI Components

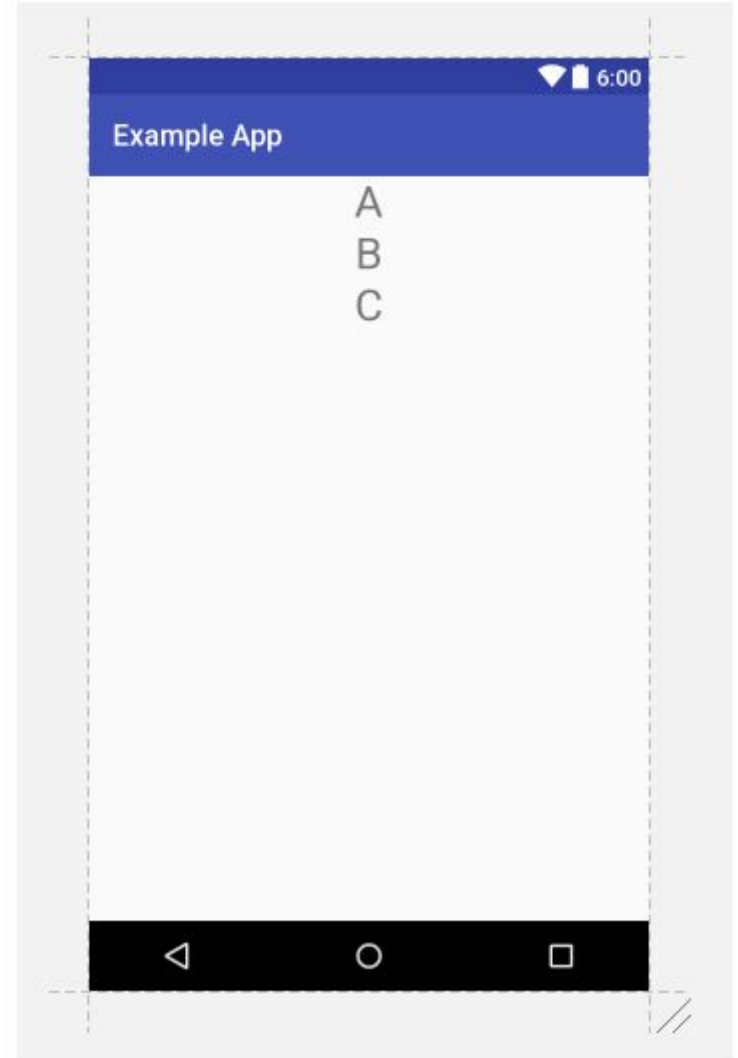
Class Name	Description
TextView	Creates text on the screen; generally non interactive text.
EditText	Creates a text input on the screen
ImageView	Creates an image on the screen
Button	Creates a button on the screen
Chronometer	Create a simple timer on screen

The [android.widget](#) package contains a list of *most* of the UI view classes available to you.

Container view

- Ещё называют Layout-компонентами
- Наследуются от класса ViewGroup
- Нужны, чтобы в них засовывать разные элементы

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context="com.example.android.exampleapp.MainActivity">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:text="A"
        android:textSize="30sp" />
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:text="B"
        android:textSize="30sp" />
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:text="C"
        android:textSize="30sp" />
</LinearLayout>
```



↳ A few examples of common container views are:

Class Name	Description
LinearLayout	Displays views in a single column or row.
RelativeLayout	Displays views positioned relative to each other and this view.
FrameLayout	A ViewGroup meant to contain a single child view.
ScrollView	A FrameLayout that is designed to let the user scroll through the content in the view.
ConstraintLayout	This is a newer viewgroup; it positions views in a flexible way. We'll be exploring constraint layout later in the lesson.

Note that layout views can be nested in one another, so you can nest a `LinearLayout` inside of a `LinearLayout` if you so choose.

Атрибуты

- XML-атрибуты описывают свойства компонент
- К примеру:
 - `layout_width` – какая будет ширина
 - `Background` – цвет
 - `Padding` – внутренняя ширина границ
 - итд

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="16dp"
    tools:context="com.example.android.exampleapp.MainActivity">

    <EditText
        android:id="@+id/edit_text_name_input"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="@color/colorAccent"
        android:hint="Enter your name"
        android:padding="4dp"
        android:textSize="24sp" />

    <TextView
        android:id="@+id/text_view_name_display"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:layout_marginTop="8dp"
        android:text="Your name appears here"
        android:textSize="30sp" />
```

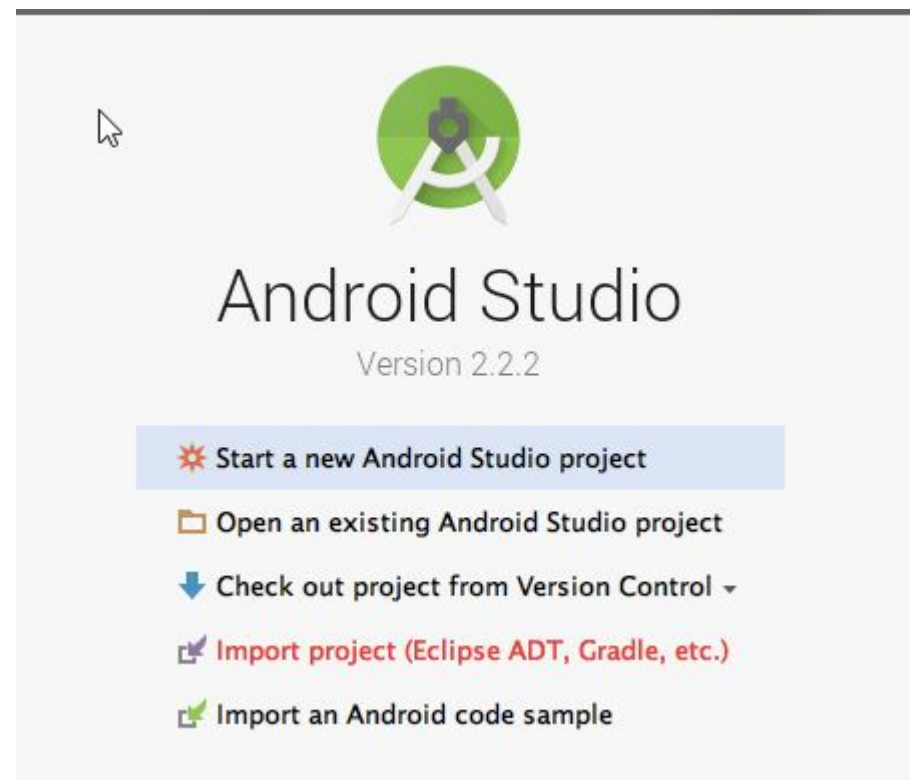
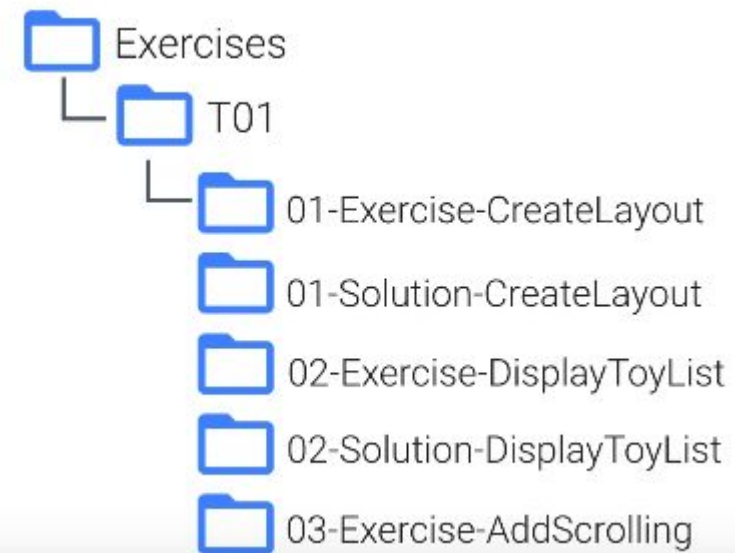
Связь с Java

- У каждой activity есть свой класс
- В методе onCreate – связь с XML с помощью setContentView
- R.* -- сгенерированные Java-репрезентации вашего XML

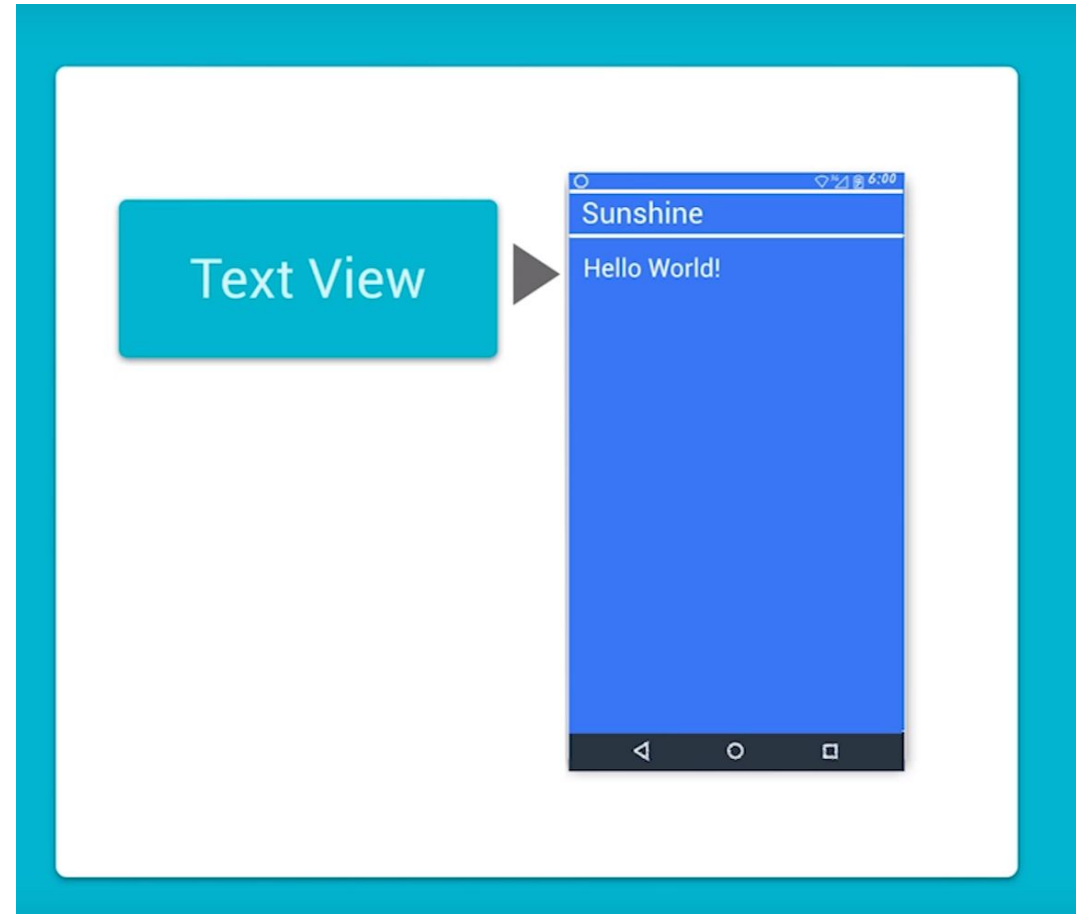
```
public class MainActivity extends AppCompatActivity {  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
        // other code to setup the activity  
    }  
    // other code  
}
```

Лабораторная работа

- Импортируем проект из 01-...
- Смотрим TODO
- Удаляем из build.gradle зависимость к ConstraintLayout



Вторая лабораторная работа



@+id/tv_toy_names

- @ говорит, что это не строка, а Android-ресурс
- + говорит, что мы создаём новый id
- R.id.tv_toy_names – имя скомпилированного TextView

