Advanced Development of User Applications for Mobile Devices

Android Studio and Application Development

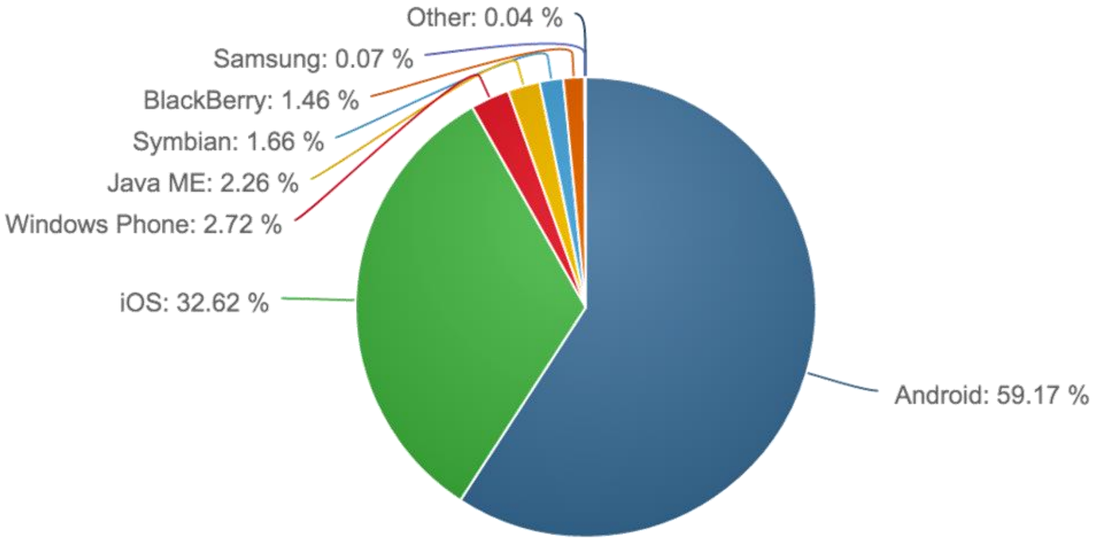
Agenda

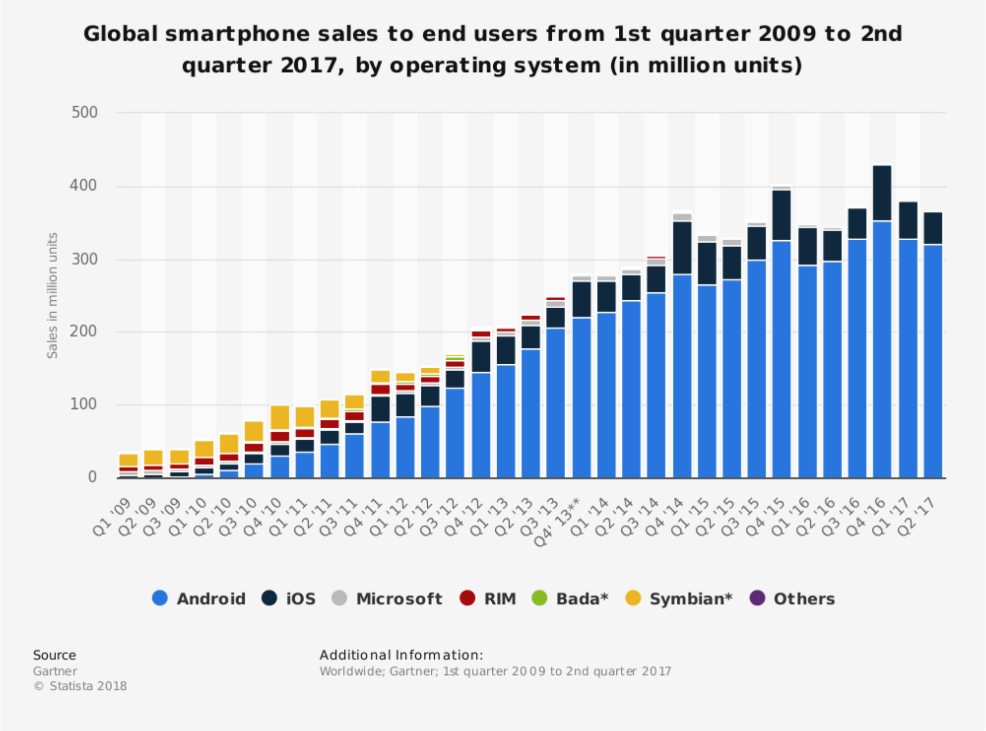
- Architecture and components
- Development environment
- Resources
- Styles and themes

Architecture and components

Android platform

- Open Handset Alliance, 2007.
  - 84 companies (Google, Samsung, Dell, Intell, LG, Nvidia...)
- First device T-Mobile G1, 2008.

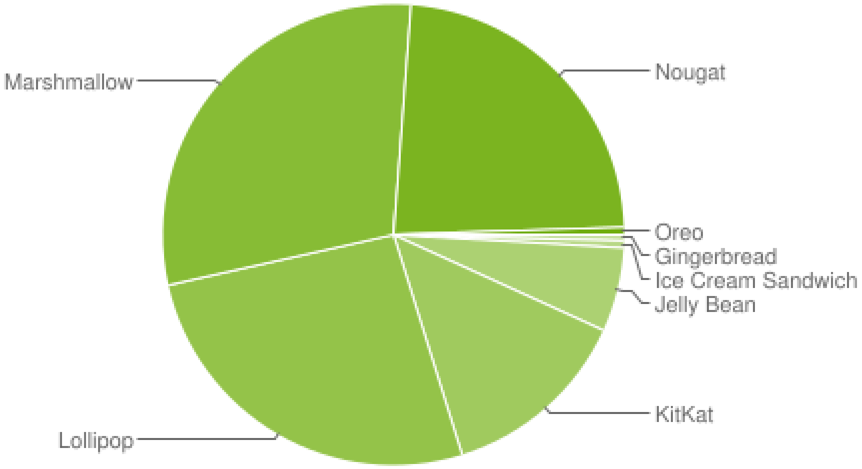



Global smartphone sales to end users from 1st quarter 2009 to 2nd quarter 2017, by operating system (in million units)

Source
Gartner
© Statista 2018

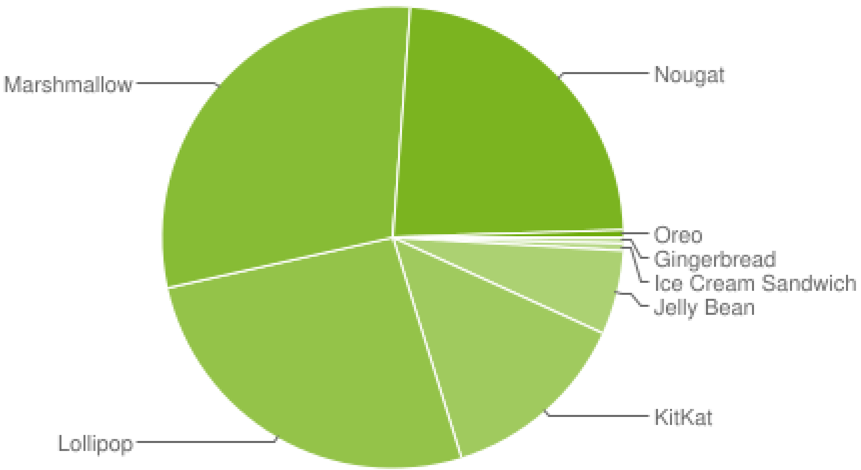
Additional Information:
Worldwide; Gartner; 1st quarter 2009 to 2nd quarter 2017

Versions



Code name	Version	API level
Alpha	1.0	API level 1
Beta	1.1	API level 2
Cupcake	1.5	API level 3
Donut	1.6	API level 4
Eclair	2.0	API level 5
Eclair	2.0.1	API level 6
Eclair	2.1	API level 7
Froyo	2.2.x	API level 8
Gingerbread	2.3 - 2.3.2	API level 9
Gingerbread	2.3.3 - 2.3.7	API level 10
Honeycomb	3.0	API level 11
Honeycomb	3.1	API level 12

Versions



Code name	Version	API level
Ice Cream Sandwich	4.0.1 - 4.0.2	API level 14
Ice Cream Sandwich	4.0.3 - 4.0.4	API level 15
Jelly Bean	4.1.x	API level 16
Jelly Bean	4.2.x	API level 17
Jelly Bean	4.3.x	API level 18
KitKat	4.4 - 4.4.4	API level 19
Lollipop	5.0	API level 21
Lollipop	5.1	API level 22
Marshmallow	6.0	API level 23
Nougat	7.0	API level 24
Nougat	7.1	API level 25
Oreo	8.0	API level 26
Oreo	8.1	API level 27
Pie	9.0	API level 28

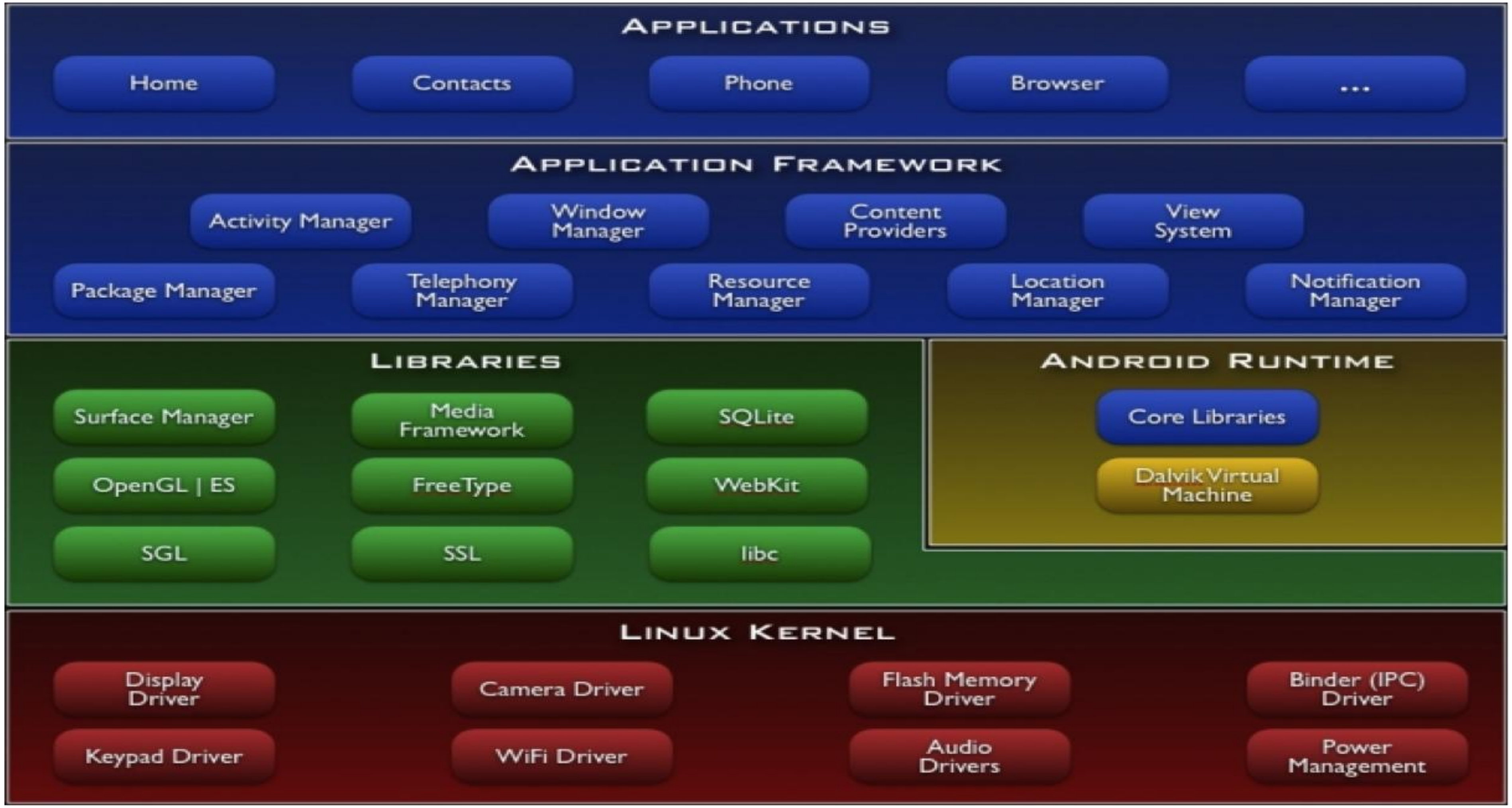
Open licenses

- GNU General Public License

- Apache License

The Apache Software Foundation

Android architecture



Applications

Home

Contacts

Phone

Browser

...

Application Framework

Activity Manager

Window Manager

Content Providers

View System

Package Manager

Telephony Manager

Resource Manager

Location Manager

Notification Manager

Libraries

Surface Manager

Media Framework

SQLite

OpenGL | ES

FreeType

WebKit

SGL

SSL

libc

Android Runtime

Core Libraries

Dalvik Virtual Machine

Linux Kernel

Display Driver

Camera Driver

Flash Memory Driver

Binder (IPC) Driver

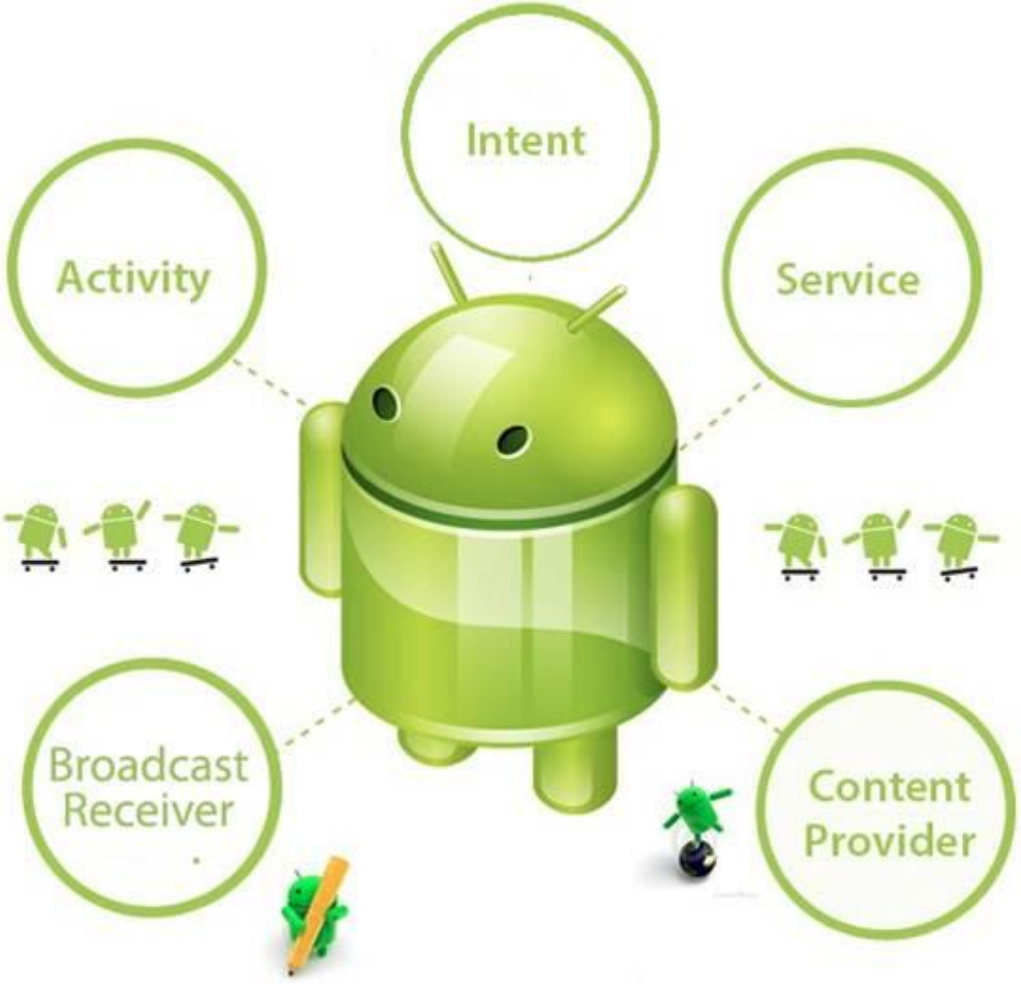
Keypad Driver

WiFi Driver

Audio Drivers

Power Management

Application components



Application components

Activities

Broadcast receivers

Services

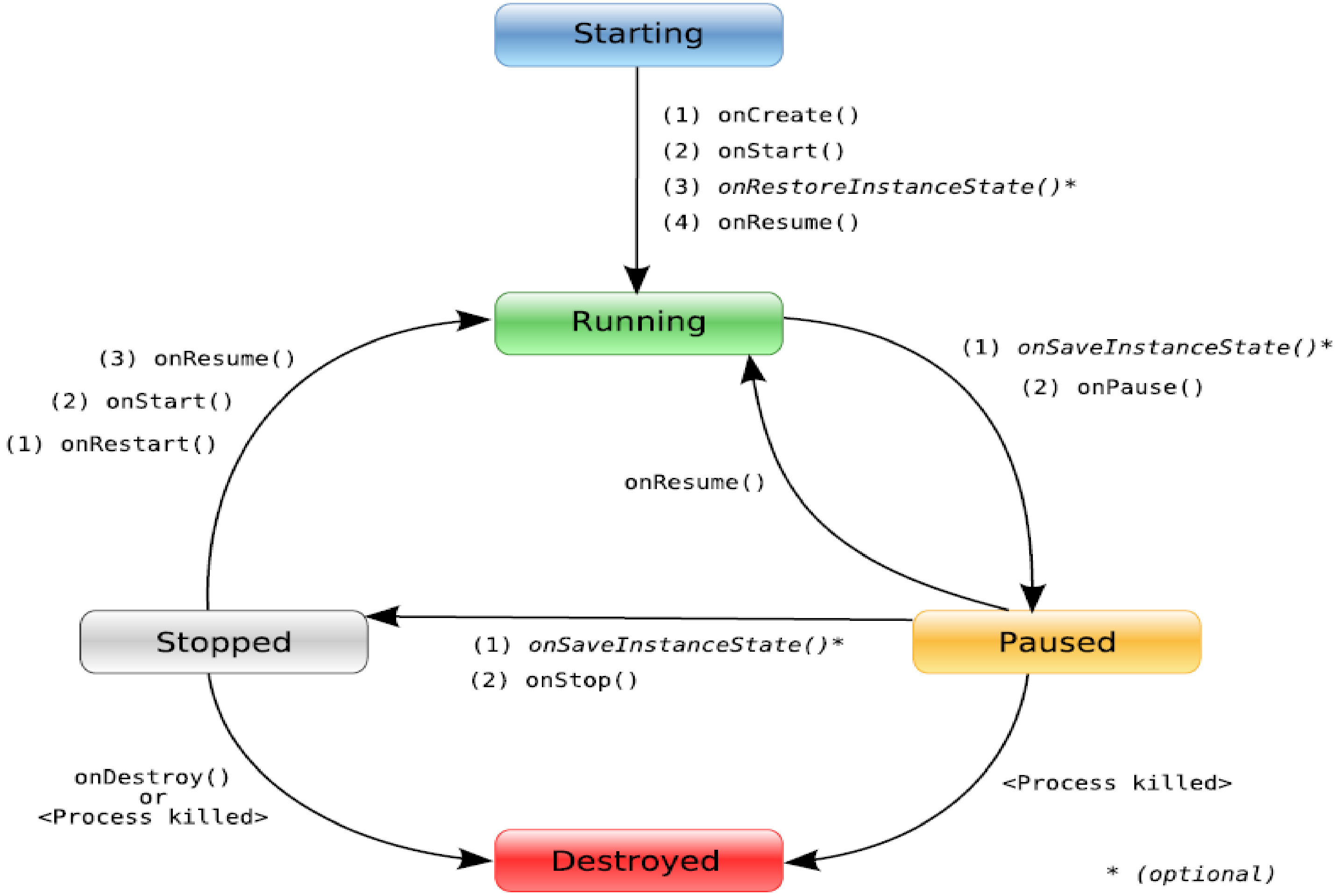
Intents

Content providers

Notifications

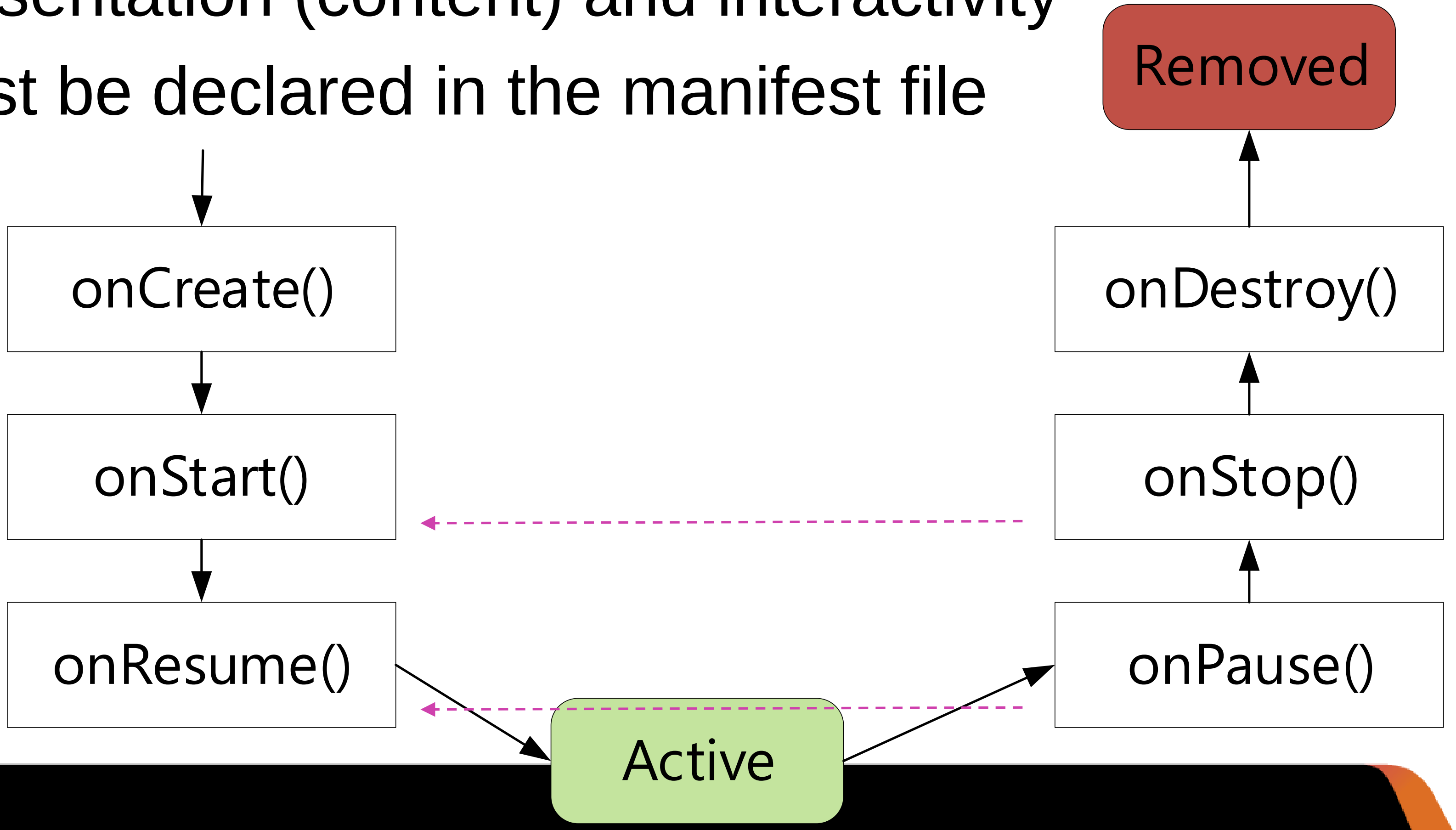
Widgets

Activity lifecycle

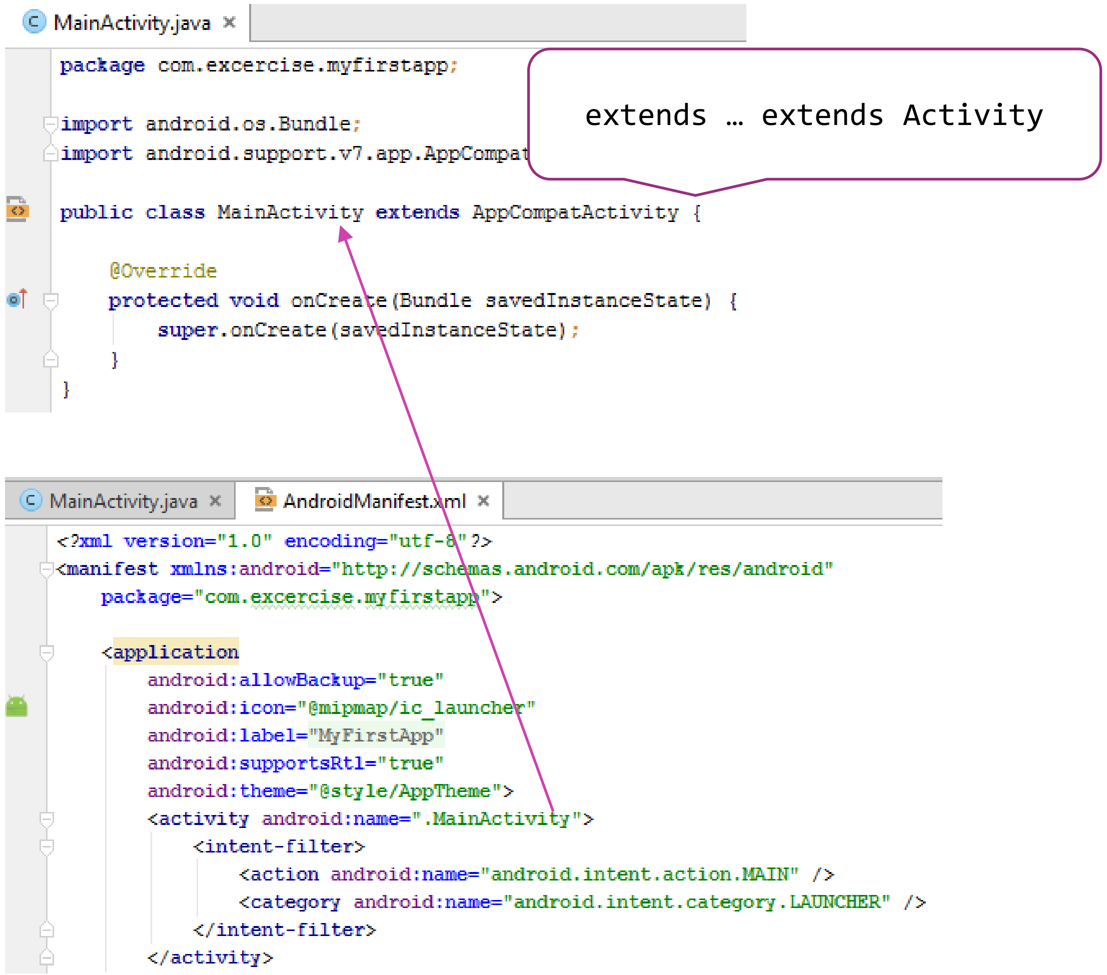


Activity

- *A window* that is displayed to users
- Presentation (content) and interactivity
- Must be declared in the manifest file



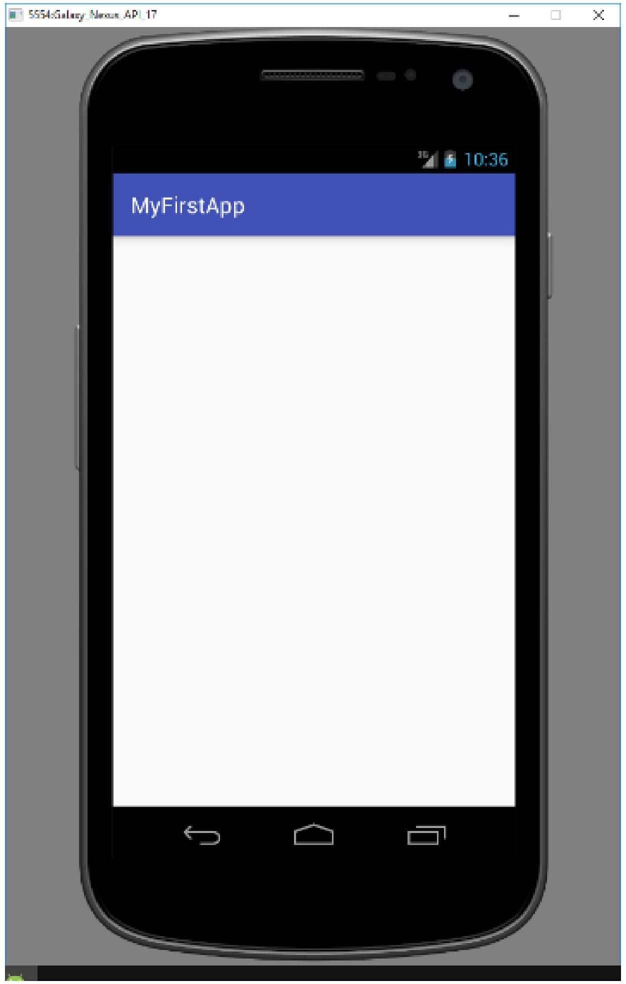
Basics



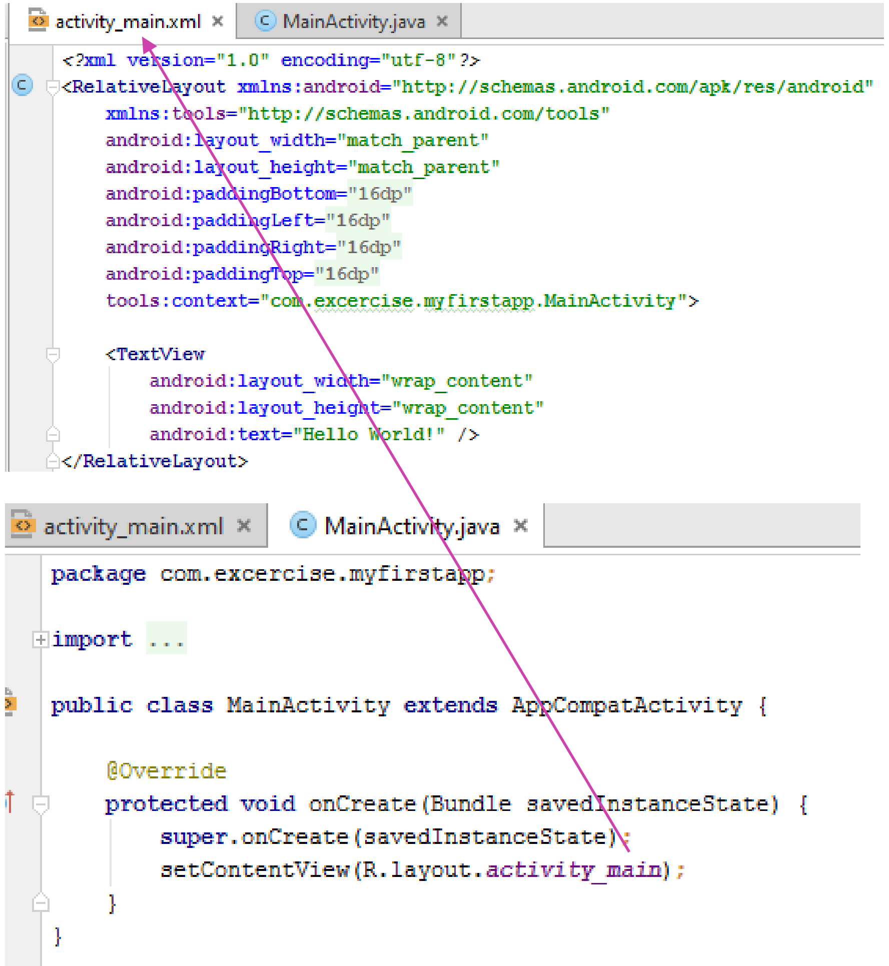
extends … extends Activity

```
package com.excercise.myfirstapp;

import android.os.Bundle;
import android.support.v7.app.AppCompat

public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
    }
}
```

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

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

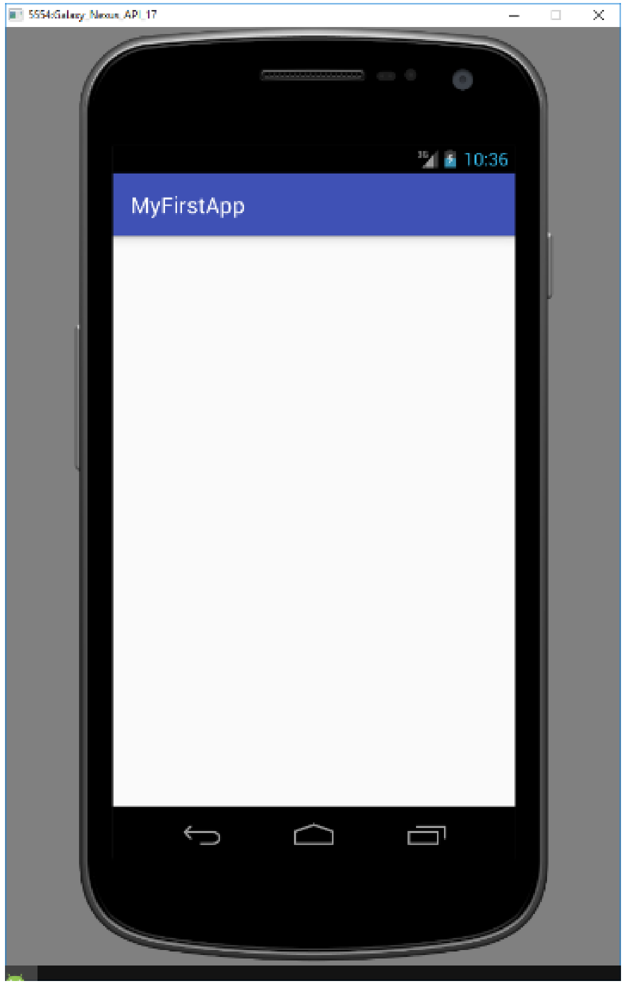


Activity’s user interface

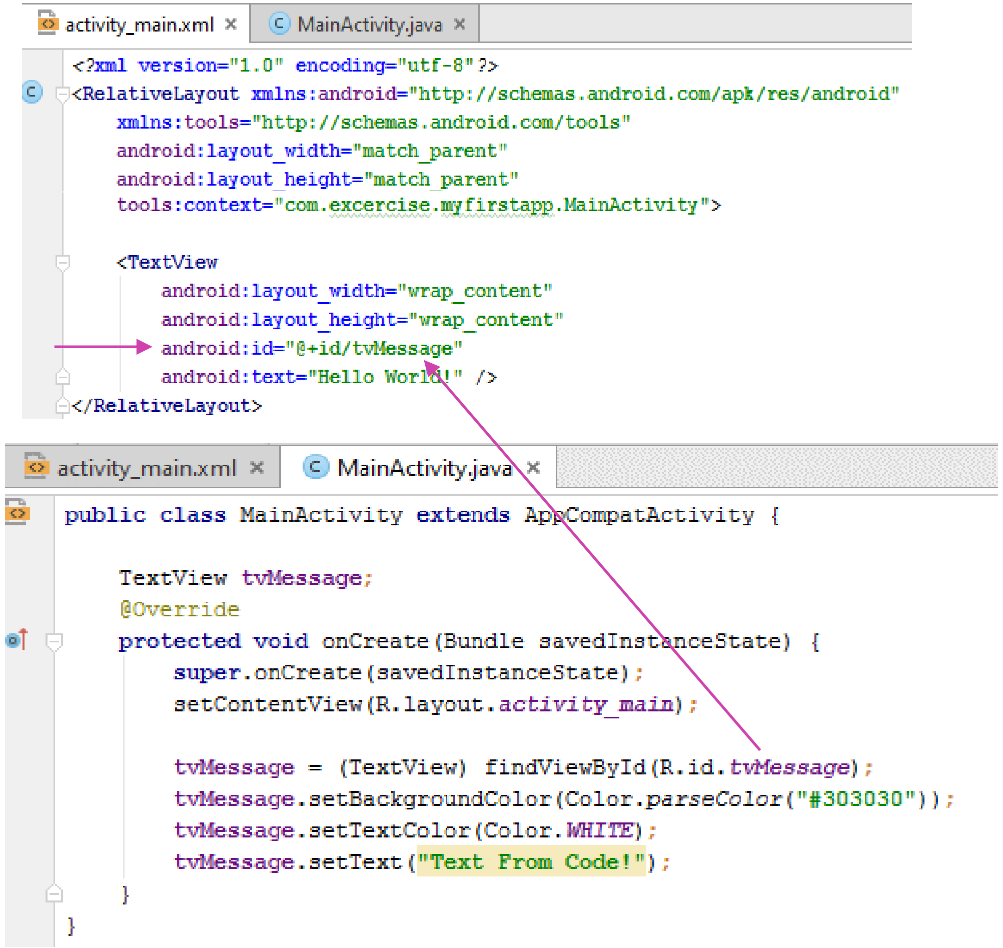


```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="16dp"
    android:paddingLeft="16dp"
    android:paddingRight="16dp"
    android:paddingTop="16dp"
    tools:context="com.excercise.myfirstapp.MainActivity">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Hello World!" />
</RelativeLayout>
```

```
package com.excercise.myfirstapp;

import ...

public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }
}
```

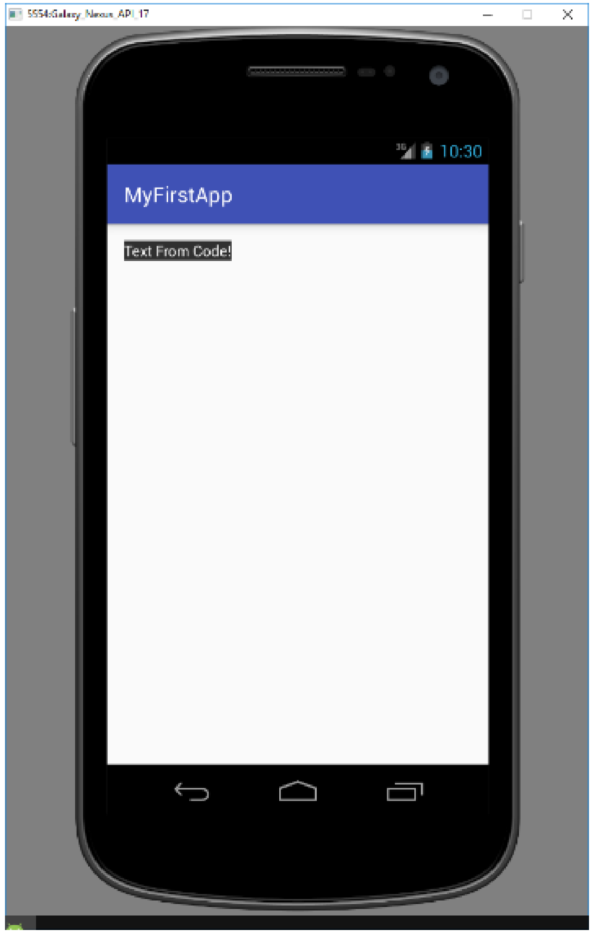


Access to views (controls)



```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout 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"
    tools:context="com.excercise.myfirstapp.MainActivity">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/tvMessage"
        android:text="Hello World!" />
</RelativeLayout>
```

```
public class MainActivity extends AppCompatActivity {

    TextView tvMessage;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        tvMessage = (TextView) findViewById(R.id.tvMessage);
        tvMessage.setBackgroundColor(Color.parseColor("#303030"));
        tvMessage.setTextColor(Color.WHITE);
        tvMessage.setText("Text From Code!");
    }
}
```



Intent

- Communication mechanism
  - Between components of the same application
  - With components of other applications

Implicit

```
Intent intent =
            new Intent (Intent.ACTION_DIAL, Uri.parse("tel:0981234567"));
this.startActivity (intent);
```

Explicit

```
Intent intent = new Intent (this, SecondActivity.class);
this.startActivity (intent);
```

Broadcast receiver

- ~ system wide event handler
- Receives and processes intents
- Must be declared in the manifest file

```
public class MySMSReceiver extends BroadcastReceiver {
@Override
  public void onReceive(Context context, Intent intent) {
    Toast.makeText(context,
     "You have a new message!", Toast.LENGTH_LONG).show();
  }
}
```

```
<receiver android:name=".MySMSReceiver">
 <intent-filter>
   <action android:name="android.provider.Telephony.SMS_RECEIVED" />
 </intent-filter>
</receiver>
```

Development environment

Software

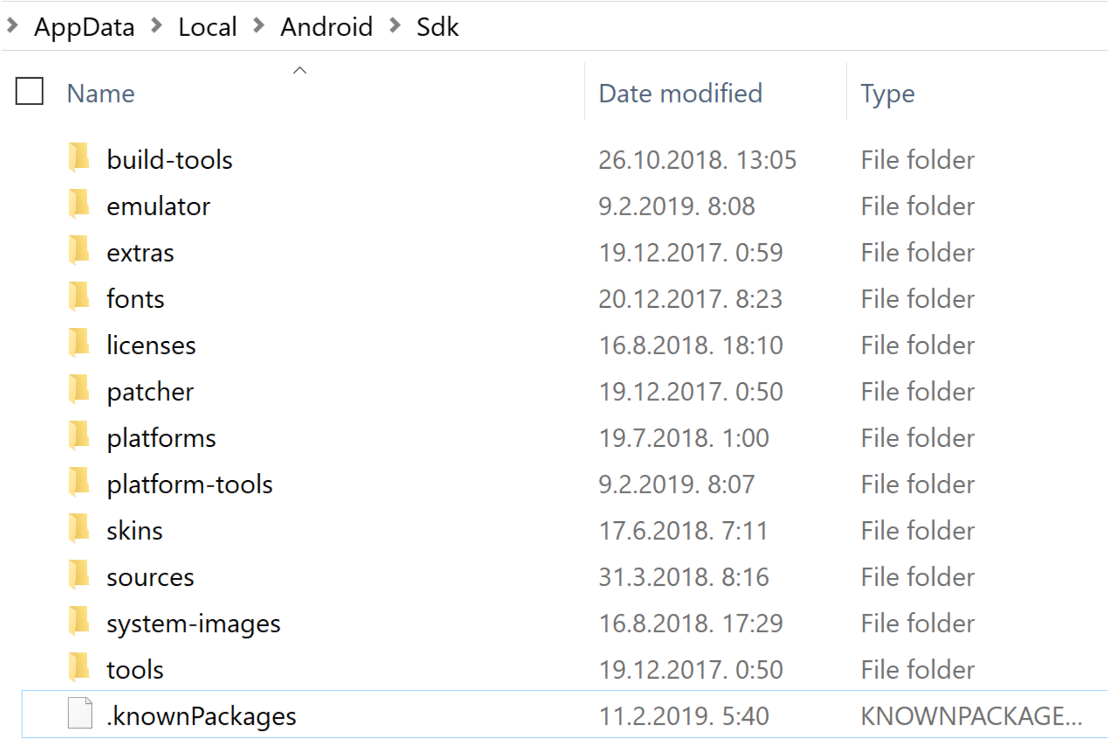
- Operating system
  - Microsoft Windows
  - Mac OS X 10.5.8
  - Linux

- Development tools
  - Java Development Kit (JDK)
  - Android SDK

Android SDK

- Set of tools, components and documentation that are required for Android application development

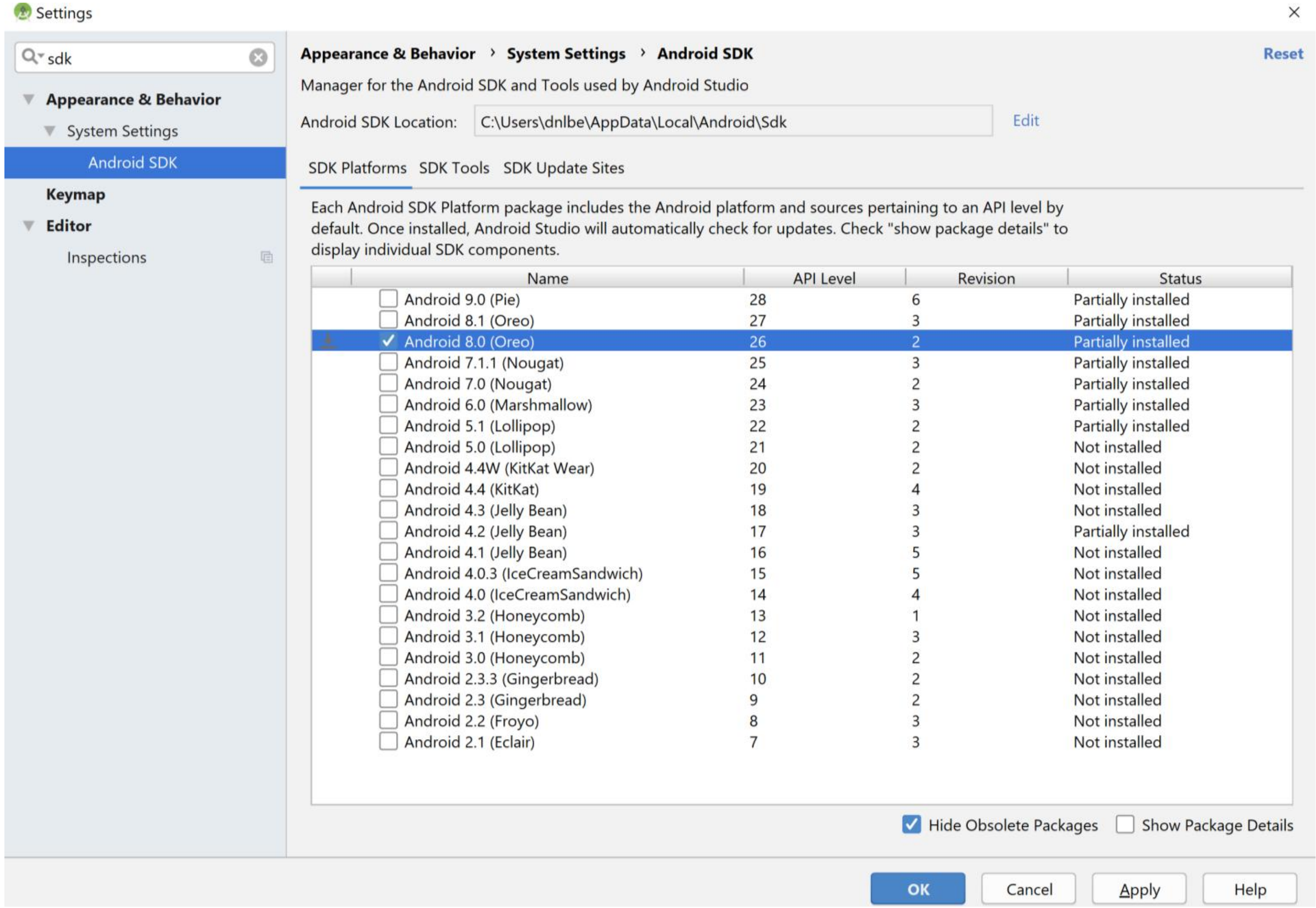
1. Android API
2. Development tools
3. AVD Manager
4. Emulator
5. Documentation
6. Code examples



› AppData › Local › Android › Sdk

□ Name	Date modified	Type
build-tools	26.10.2018. 13:05	File folder
emulator	9.2.2019. 8:08	File folder
extras	19.12.2017. 0:59	File folder
fonts	20.12.2017. 8:23	File folder
licenses	16.8.2018. 18:10	File folder
patcher	19.12.2017. 0:50	File folder
platforms	19.7.2018. 1:00	File folder
platform-tools	9.2.2019. 8:07	File folder
skins	17.6.2018. 7:11	File folder
sources	31.3.2018. 8:16	File folder
system-images	16.8.2018. 17:29	File folder
tools	19.12.2017. 0:50	File folder
.knownPackages	11.2.2019. 5:40	KNOWNPACKAGE...

Android SDK - installation



Appearance & Behavior › System Settings › Android SDK

Manager for the Android SDK and Tools used by Android Studio

Android SDK Location: C:\Users\dnlbe\AppData\Local\Android\Sdk

Each Android SDK Platform package includes the Android platform and sources pertaining to an API level by default. Once installed, Android Studio will automatically check for updates. Check "show package details" to display individual SDK components.

Name	API Level	Revision	Status
□ Android 9.0 (Pie)	28	6	Partially installed
□ Android 8.1 (Oreo)	27	3	Partially installed
☑ Android 8.0 (Oreo)	26	2	Partially installed
□ Android 7.1.1 (Nougat)	25	3	Partially installed
□ Android 7.0 (Nougat)	24	2	Partially installed
□ Android 6.0 (Marshmallow)	23	3	Partially installed
□ Android 5.1 (Lollipop)	22	2	Partially installed
□ Android 5.0 (Lollipop)	21	2	Not installed
□ Android 4.4W (KitKat Wear)	20	2	Not installed
□ Android 4.4 (KitKat)	19	4	Not installed
□ Android 4.3 (Jelly Bean)	18	3	Not installed
□ Android 4.2 (Jelly Bean)	17	3	Partially installed
□ Android 4.1 (Jelly Bean)	16	5	Not installed
□ Android 4.0.3 (IceCreamSandwich)	15	5	Not installed
□ Android 4.0 (IceCreamSandwich)	14	4	Not installed
□ Android 3.2 (Honeycomb)	13	1	Not installed
□ Android 3.1 (Honeycomb)	12	3	Not installed
□ Android 3.0 (Honeycomb)	11	2	Not installed
□ Android 2.3.3 (Gingerbread)	10	2	Not installed
□ Android 2.3 (Gingerbread)	9	2	Not installed
□ Android 2.2 (Froyo)	8	3	Not installed
□ Android 2.1 (Eclair)	7	3	Not installed

☑ Hide Obsolete Packages □ Show Package Details

1. development method

Use SDK tools

1. Write Java source code (e.g. Notepad)
2. Compile source code to Java bytecode (javac)
3. Compile Java bytecode to Dalvik (dx)
4. Copy .dex file to the device / emulator
5. Run application

```
1. ....
2. javac Hello.java
3. dx --dex --output=hello.dex Hello.class
4. adb push hello.dex /sdcard
5. adb shell dalvikvm -cp /sdcard/hello.dex Hello
```

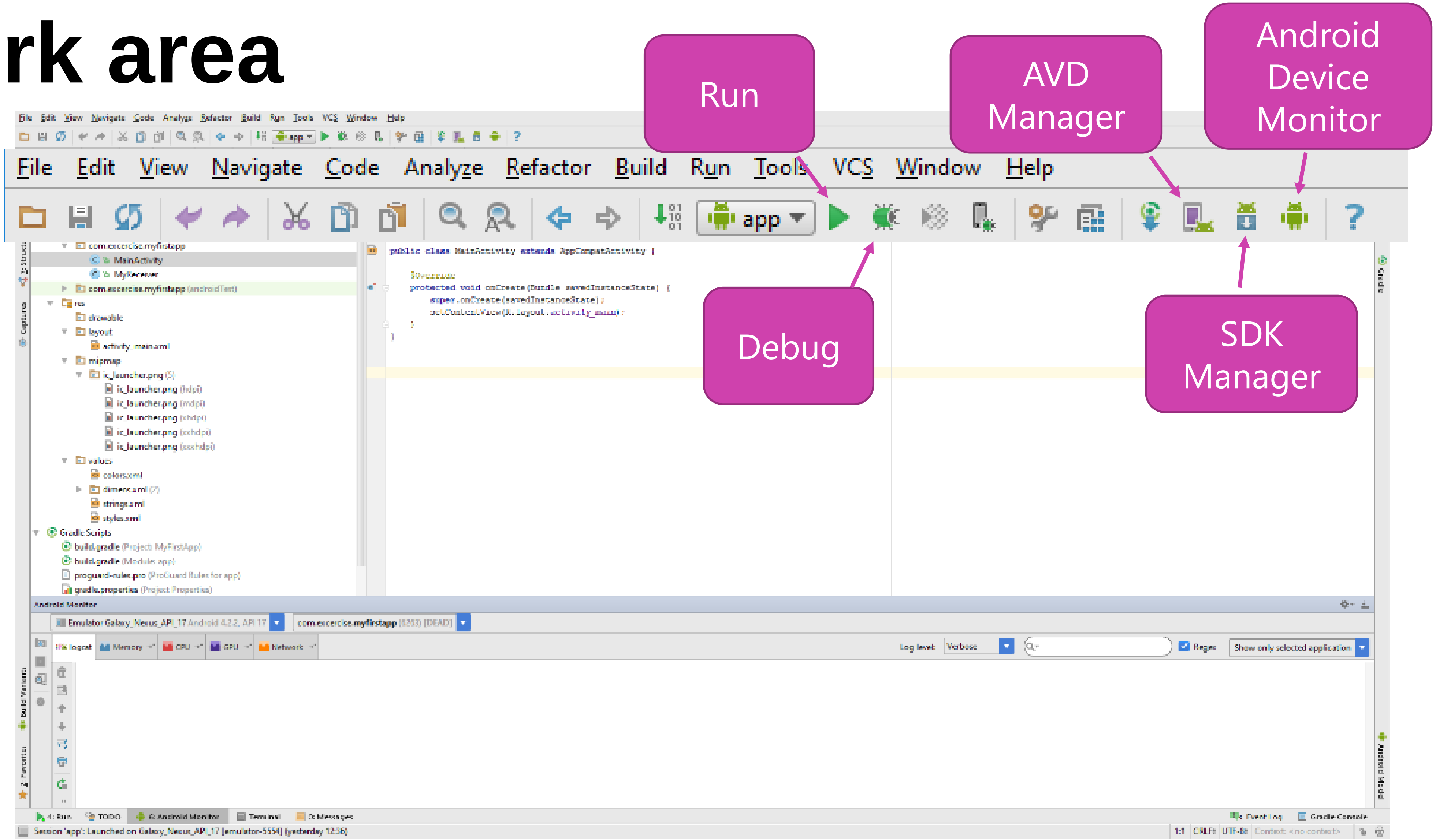
2. development method

~~Eclipse + Android Developer Tools (ADT) Plug-In~~

Android Studio

- Official IDE for Android application development
- Based on IntelliJ platform
- Eases and accelerates development
  - Creating project
  - Designing user interface
  - Debugging application
  - Running application
  - Hardware access
  - …

Work area



```
public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }
}
```

Emulator Galaxy_Nexus_API_17 Android 4.2.2, API 17

com.excercise.myfirstapp (8253) [DEAD]

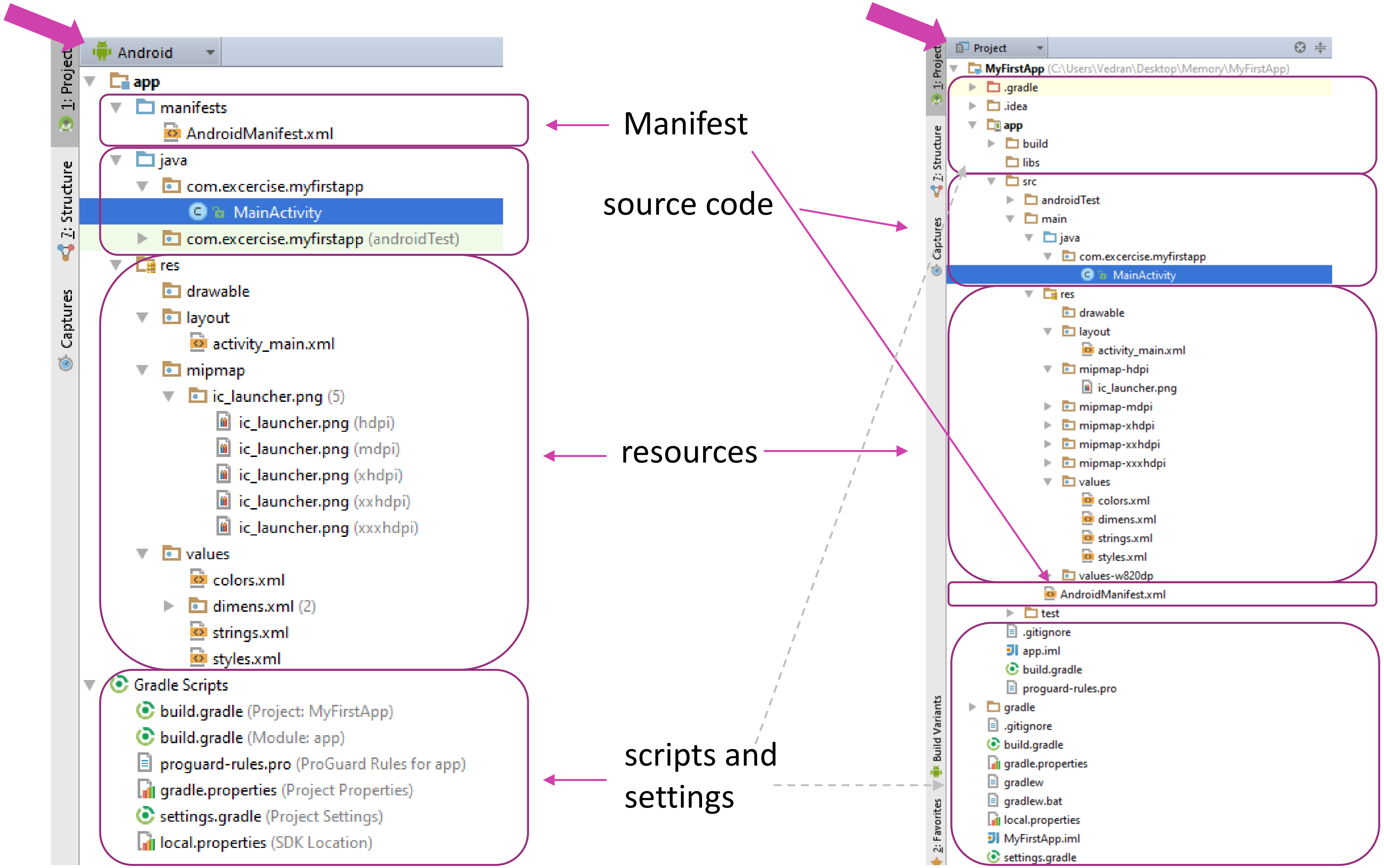
Log level: Verbose

☑ Regex

Show only selected application

Session 'app': Launched on Galaxy_Nexus_API_17 [emulator-5554] (yesterday 12:36)

1:1 CRLF UTF-8 Context: <no context>



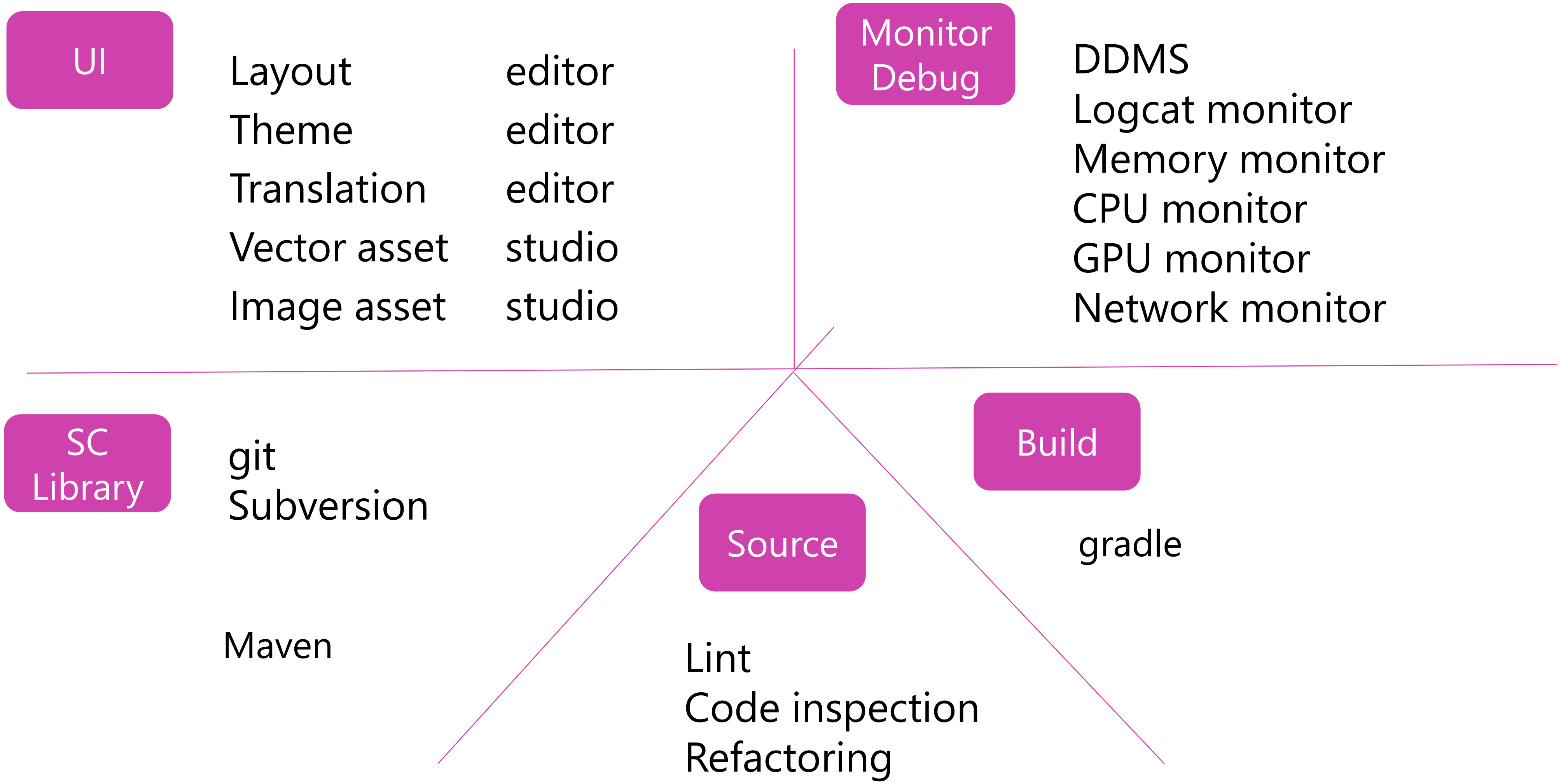
Manifest

source code

resources

scripts and settings

Tools and plugins



UI

Layout editor
Theme editor
Translation editor
Vector asset studio
Image asset studio

Monitor Debug

DDMS
Logcat monitor
Memory monitor
CPU monitor
GPU monitor
Network monitor

SC Library

git
Subversion

Maven

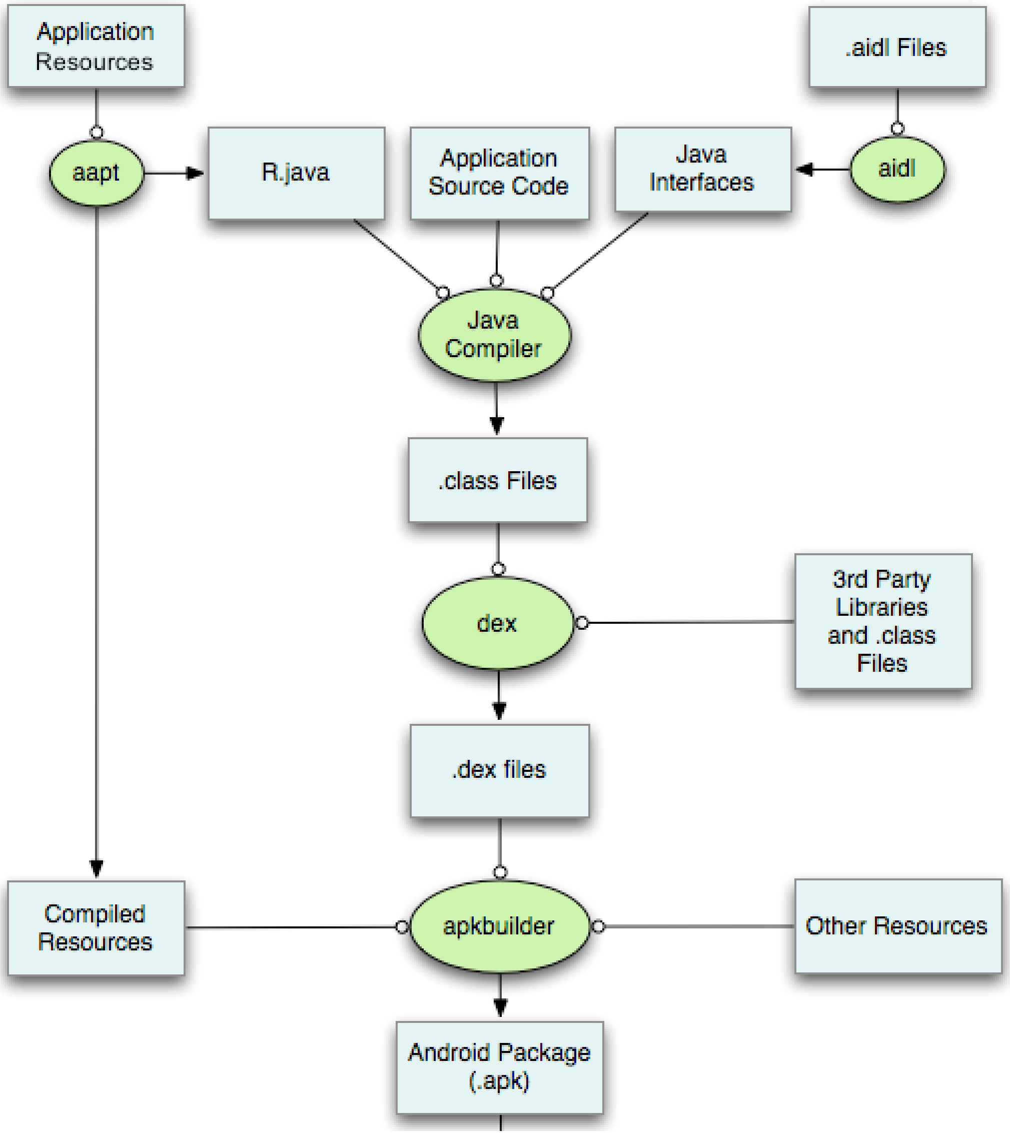
Source

Lint
Code inspection
Refactoring

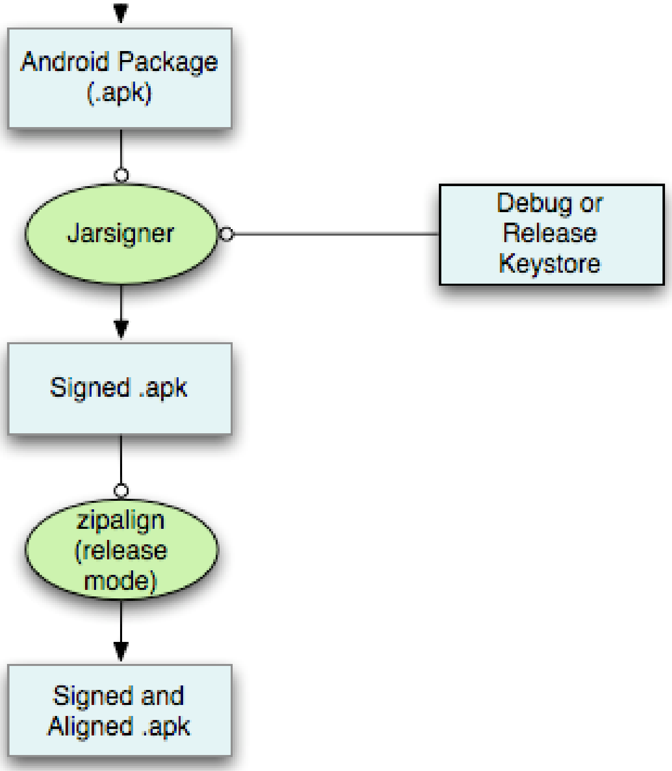
Build

gradle

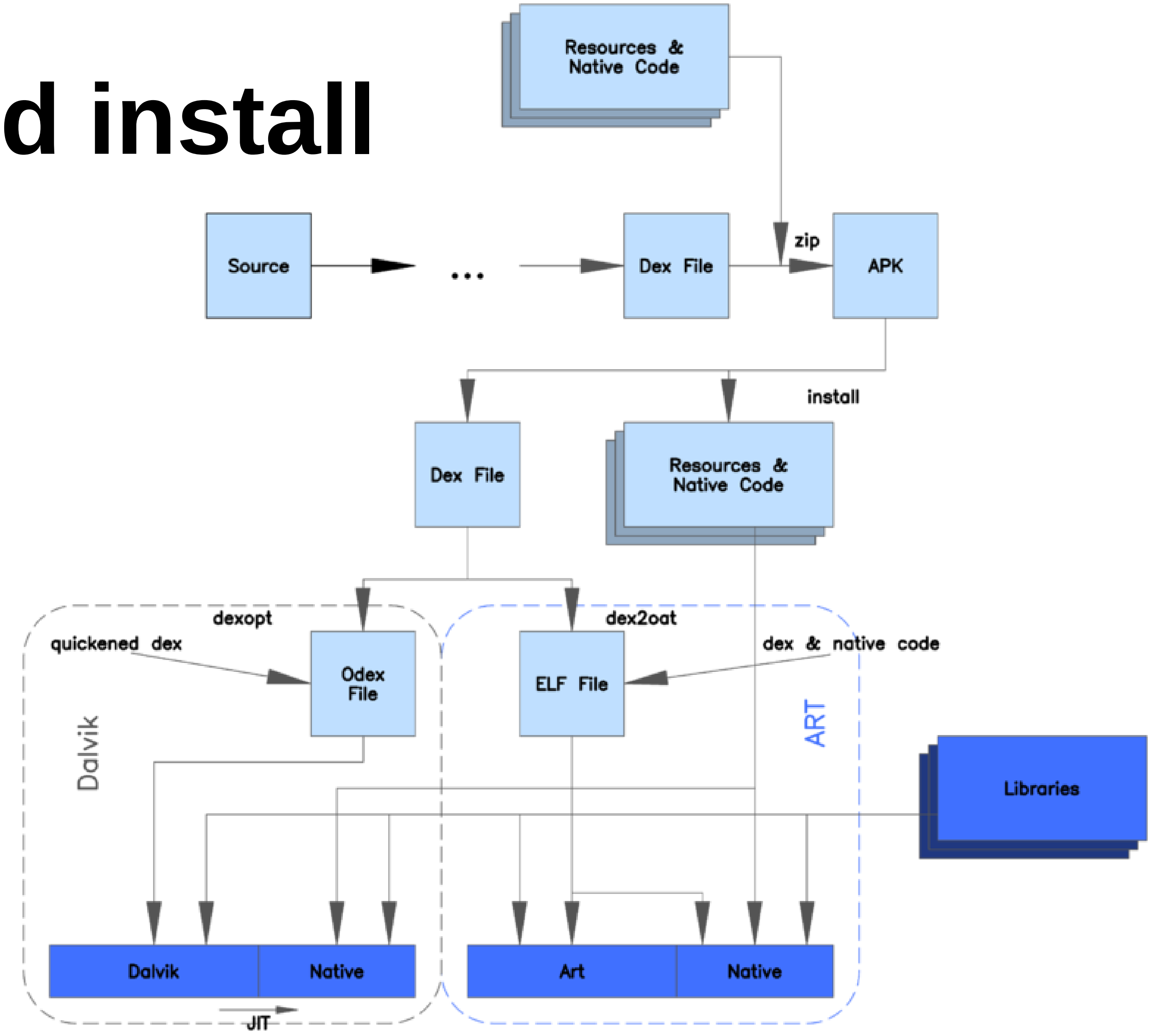
Build process 1 / 2



Build process 2 / 2



Android install



Android build

- Consists of Android Gradle plugin
- Automated build toolkit
- Allows
  - Dependency definition
  - Build variants configuration (debug, release, *flavors*)
  - Definition of manifest data (e.g. API level)
  - signing
  - ProGuard
  - …

Gradle script

Groovy programming language

build.gradle

```
task helloWorld <<
 { print 'hello world' }
```

```
$ gradle -q helloWorld

hello world
```

---

build.gradle

```
task hello <<
 { print 'hello' }

task world(dependsOn: hello) <<
  { println 'world' }
```

```
$ gradle -q world

hello world
```

Plugin usage

build.gradle

```
apply plugin: 'java'
```

```
$ gradle build

:compileJava
:processResources
:classes
:jar
:assemble
:compileTestJava
:processTestResources
:testClasses
:test
:check
:build

BUILD SUCCESSFUL

Total time: 1 secs
```

Android build.gradle

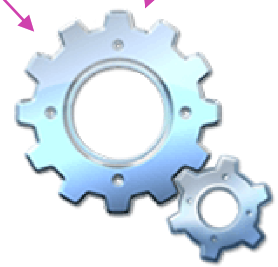
```
apply plugin: 'com.android.application'

android {
    compileSdkVersion 23
    buildToolsVersion "23.0.2"

    defaultConfig {
        applicationId "com.excercise.myfirstapp"
        minSdkVersion 17
        targetSdkVersion 23
        versionCode 1
        versionName "1.0"
    }
    buildTypes {
        release {
            minifyEnabled false
            proguardFiles getDefaultProguardFile
                         ('proguard-android.txt'), 'proguard-rules.pro'
        }
    }
}
```

AndroidManifest.xml(project)

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="..."
    package="com.excercise.myfirstapp">

    <application
        android:label="@string/app_name" ... >
    </application>
</manifest>
```

build.gradle

```
defaultConfig {
  minSdkVersion 17
  targetSdkVersion 23
  versionCode 1
  versionName "1.0"
}
```



Gradle

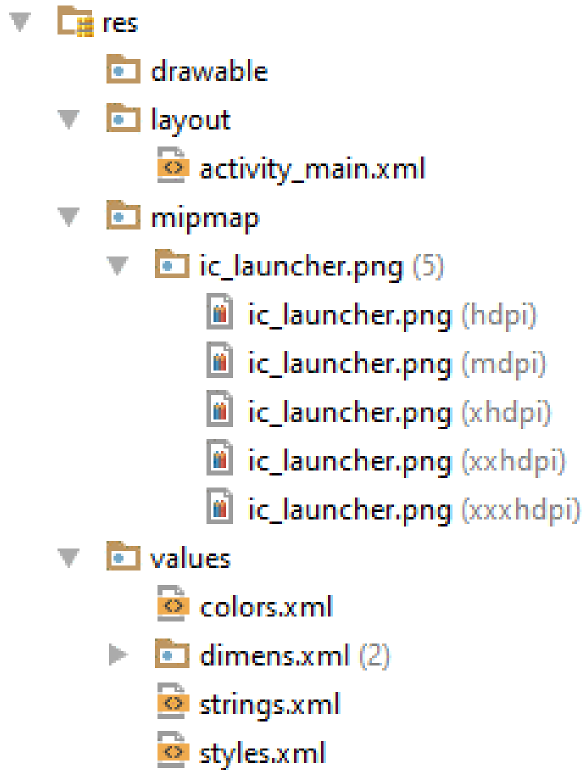
AndroidManifest.xml(merged)

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="…"
    package="com.excercise.myfirstapp"
    android:versionCode="1"
    android:versionName="1.0" >
    <uses-sdk
        android:minSdkVersion="17"
        android:targetSdkVersion="23" />
    ...
```

Resources

Resource types

Type	Folder (~/app/res/)
Simple values	values
Layouts	layout
Graphics	drawable mipmap
Menus	menu
Animations	animator anim
Colors	values
XML	xml
Styles	values
Other resources	raw
Assets	assets



```
▼ res
    drawable
  ▼ layout
      activity_main.xml
  ▼ mipmap
    ▼ ic_launcher.png (5)
        ic_launcher.png (hdpi)
        ic_launcher.png (mdpi)
        ic_launcher.png (xhdpi)
        ic_launcher.png (xxhdpi)
        ic_launcher.png (xxxhdpi)
  ▼ values
      colors.xml
    ▶ dimens.xml (2)
      strings.xml
      styles.xml
```

Definition and usage

res/values/colors.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <color name="myColor">#303030</color>
</resources>
```

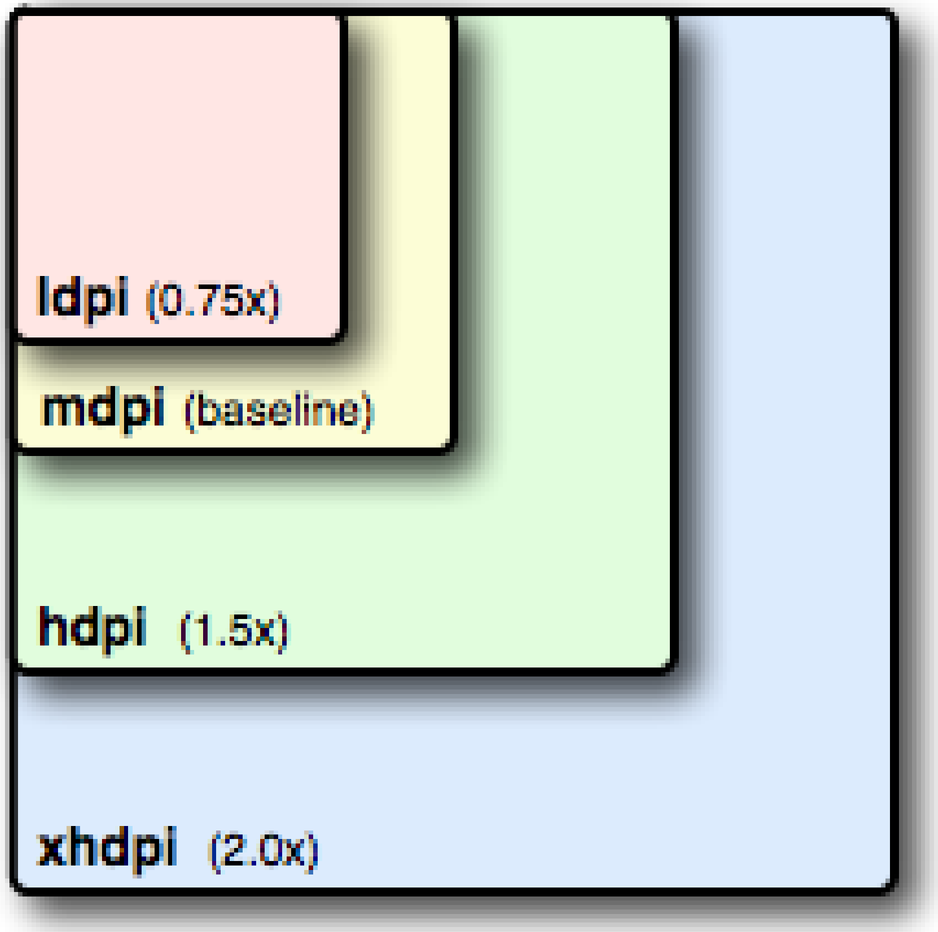
Resources

```
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/tvMessage"
        android:text="Hello World!"
        android:background="@color/myColor"/>
```

Code

```
    int color = getResources().getColor(R.color.myColor);
```

Images



Definition and usage

res/values/strings.xml

```
<resources>
    <string name="myString">My Text</string>
</resources>
```

Resources

```
<TextView ...
    android:text="@string/myString"
    android:background="@color/myColor"/>
```

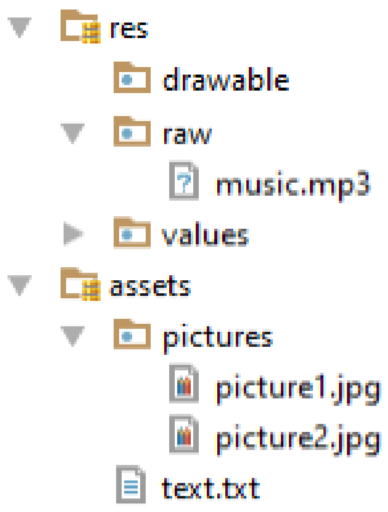
Code

```
textView.setText(getResources().getText(R.string.myString));

textView.setText(R.string.myString);
```

Other resource types

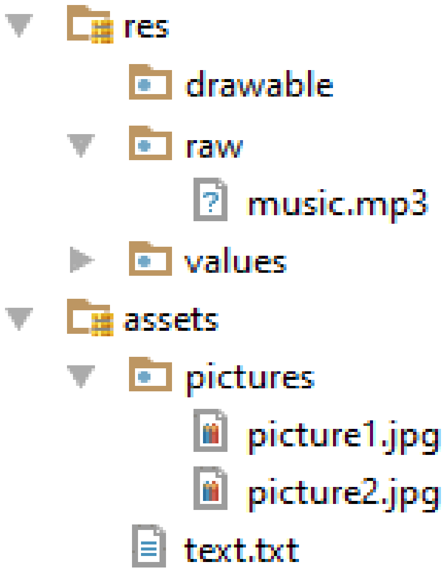
Audio, Video, Text, Excel, etc.

	/res/raw	/assets
Read-only	Yes	Yes
Resource types	All	All
Hijerarchy	No (one folder)	Yes
Resource access based on	Identifier	Name / file path
	Static	Dynamic
Identifiers in the R class	Yes	No
Fetching	Context.getResources()	Context.getAssets()
Usage	getResources().openRawResource()	getAssets().open()
Resource adjustment (e.g. localization)	Yes *e.g. raw-en*	No



```
res
  drawable
  raw
    music.mp3
  values
assets
  pictures
    picture1.jpg
    picture2.jpg
  text.txt
```

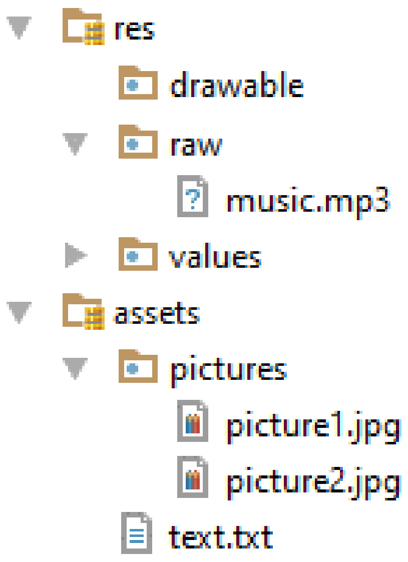
Raw - example



```
res
    drawable
    raw
        music.mp3
    values
assets
    pictures
        picture1.jpg
        picture2.jpg
    text.txt
```

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    MediaPlayer mediaPlayer = MediaPlayer.create(this, R.raw.music);
    mediaPlayer.start();
}
```

Assets - Example 1

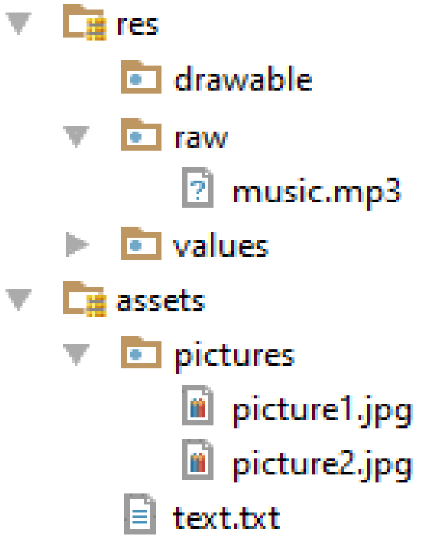


```
res
    drawable
    raw
        music.mp3
    values
assets
    pictures
        picture1.jpg
        picture2.jpg
    text.txt
```

```
protected void onCreate(Bundle savedInstanceState) {
    ...
    AssetManager am = this.getAssets();
    BufferedReader br = null;
    try {
        InputStream is = am.open("text.txt");
        br = new BufferedReader(new InputStreamReader(is));
        String text = br.readLine();
    } catch (IOException e) {
        e.printStackTrace();
    } finally {
       if (br != null){
          br.close();
       }
    }
}
```

Assets - Example 2

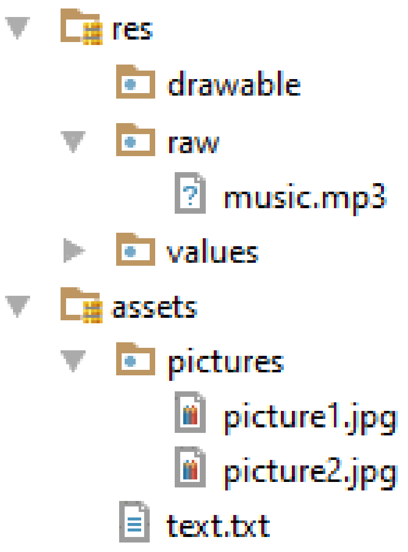
```
protected void onCreate(Bundle savedInstanceState) {
    ...
    AssetManager am = this.getAssets();
    InputStream is = null;
    try {
        is = am.open("pictures/picture1.jpg");
        Bitmap bitmap = BitmapFactory.decodeStream(is);
        imageView.setImageBitmap(bitmap);
    } catch (IOException e) {
        e.printStackTrace();
    } finally {
       if (is != null){
          is.close();
       }
    }
}
```



```
res
  drawable
  raw
    music.mp3
  values
assets
  pictures
    picture1.jpg
    picture2.jpg
  text.txt
```

Assets - Example 3

```
protected void onCreate(Bundle savedInstanceState) {
    ...
    AssetManager am = this.getAssets();
    InputStream is = null;
    try {
        for (String fileName : am.list("pictures")) {
            InputStream is = am.open("pictures/" + fileName);
            Bitmap bitmap = BitmapFactory.decodeStream(is);
            ...
         }
    } catch (IOException e) {
        e.printStackTrace();
    } finally {
       if (is != null){
          is.close();
       }
    }
}
```



```
res
  drawable
  raw
    music.mp3
  values
assets
  pictures
    picture1.jpg
    picture2.jpg
  text.txt
```

Styles and themes

Style

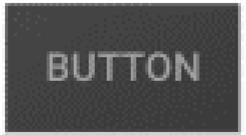
- Set of properties that define appearance
  - width, height
  - color, padding, text
- ~CSS but simpler(e.g. there is no *btn btn-primary*)
- Defined in separate file
  - Folder *values*
  - Mainly in styles.xml

```
<style name="<StyleName>">
    <item name="<ItemName1>">value1</item>
    <item name="<ItemName2>">value2</item>
</style>
```

Example

layout/layout.xml

```
<Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Button"
    android:textSize="15sp"
    android:textColor="#9B9A9A"
    android:background="#303030"
/>
```



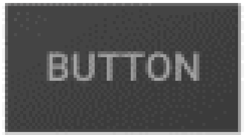
values/styles.xml

```
<resources>
    <style name="myButton">
        <item name="android:background">#303030</item>
        <item name="android:textColor">#9B9A9A</item>
        <item name="android:textSize">15sp</item>
    </style>
</resources>
```

Example

layout/layout.xml

```
<Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Button"
    style="@style/myButton"
/>
```



values/styles.xml

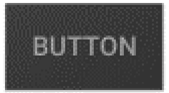
```
<resources>
    <style name="myButton">
        <item name="android:background">#303030</item>
        <item name="android:textColor">#9B9A9A</item>
        <item name="android:textSize">15sp</item>
    </style>
</resources>
```

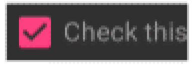
Defining a style

- Can use properties from the android.R.attr class
  - http://developer.android.com/reference/android/R.attr.html
  - Only certain properties are applied

```
<style name="myButton">
    <item name="android:background">#303030</item>
    ...
    <item name="android:checked">true</item>
</style>
```

```
<Button ...
    android:text="Button"
    style="@style/myButton"/>

<CheckBox
    android:text="Check this"
    style="@style/myButton"/>
```





Inheritance

1. Attribute parent

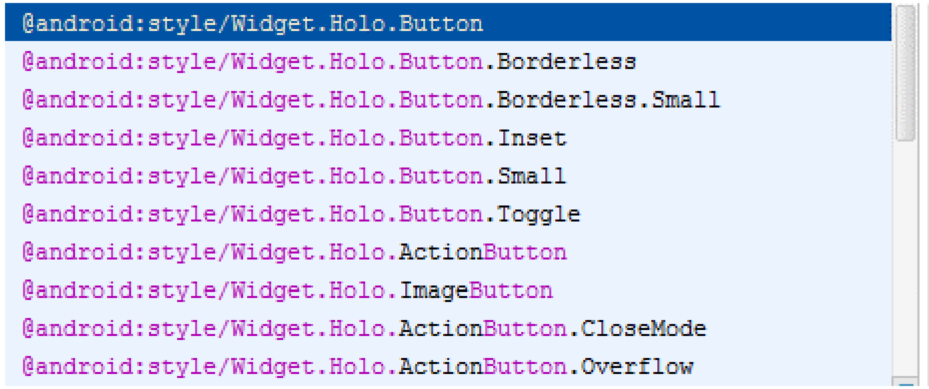
```
<style name="myBigButton" parent="@style/myButton">
    <item name="android:textSize">20sp</item>
</style>

<Button ...
    style="@style/myBigButton" />
```

2. Prefix

```
<style name="myButton.myBigButton">
    <item name="android:textSize">20sp</item>
</style>

<Button ...
    style="@style/myButton.myBigButton" />
```

Android style inheritance

```
<style name="myButton"
       parent="@android:style/...">
</style>
```



```
@android:style/Widget.Holo.Button
@android:style/Widget.Holo.Button.Borderless
@android:style/Widget.Holo.Button.Borderless.Small
@android:style/Widget.Holo.Button.Inset
@android:style/Widget.Holo.Button.Small
@android:style/Widget.Holo.Button.Toggle
@android:style/Widget.Holo.ActionButton
@android:style/Widget.Holo.ImageButton
@android:style/Widget.Holo.ActionButton.CloseMode
@android:style/Widget.Holo.ActionButton.Overflow
```

```
<style name="myButton"
       parent="@android:style/Widget.Button">
    <item name="android:textColor">#FFFFFF</item>
</style>
```

It is recommended to exend from Android Support Library:

```
<style name="myButton"
       parent="Widget.AppCompat.Button">
    <item name="android:textColor">#FFFFFF</item>
</style>
```

Theme

- Set of properties that define appearance of activities

```
<style name="ThemeName">
    <item name="<ItemName1>">value1</item>
    <item name="<ItemName2>">value2</item>
</style>
```

- Can be applied to an application or an activity

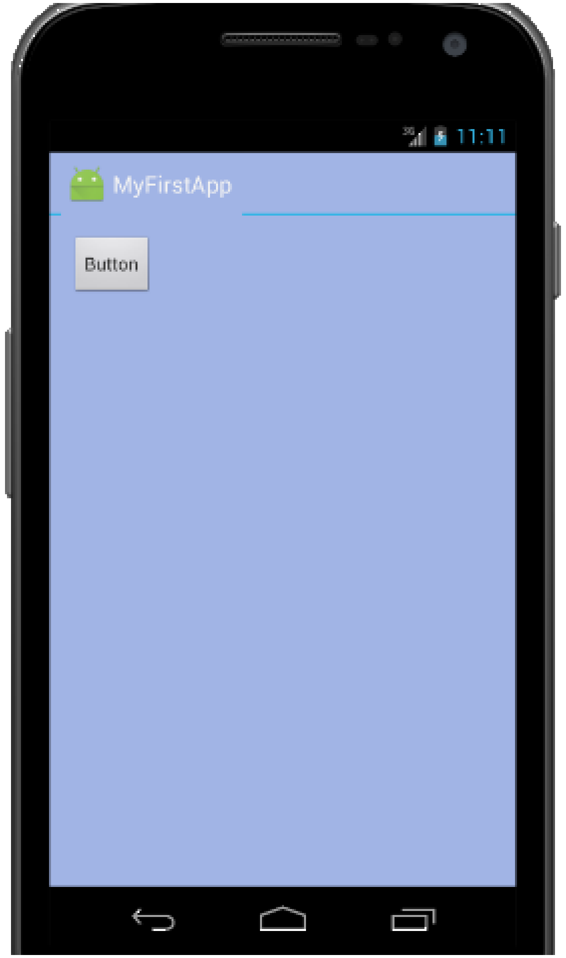
(>=API 21 – can be applied to any parent View)

```
<application ...
    android:theme="@style/myTheme">

<activity ...
    android:theme="@style/myTheme">
```

Example

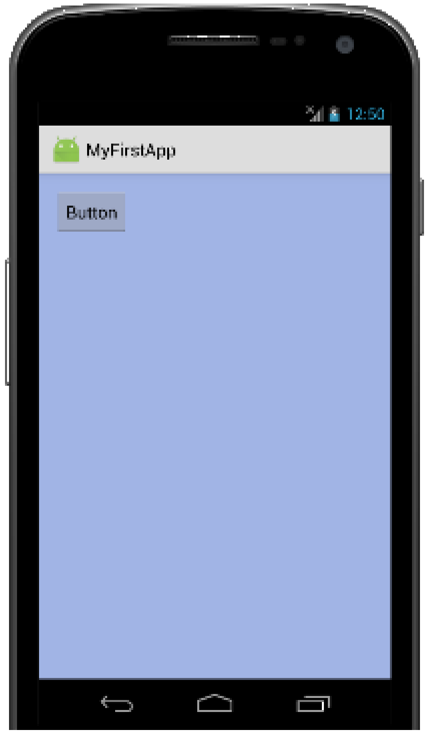
```
<style name="myTheme" >
    <item name="android:background">#A1B4E5</item>
</style>

<application ...
    android:theme="@style/myTheme">
```

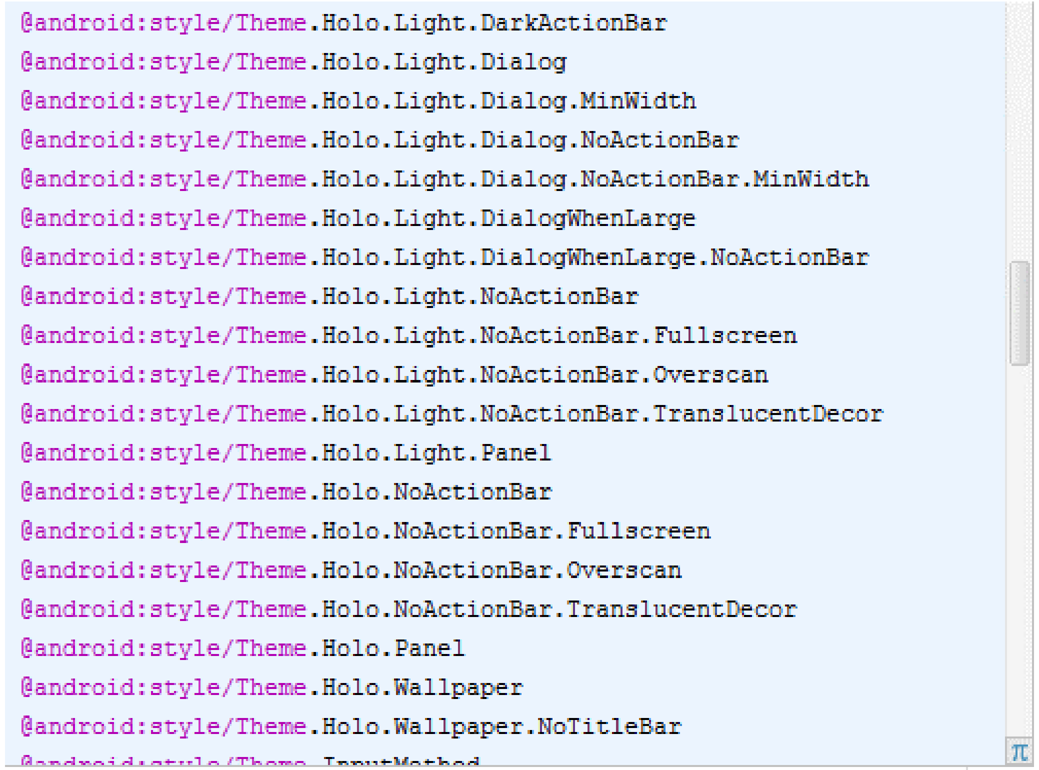
- Properties from the android.R.attr class
- Themes can also be inherited



Inheritance and properties

```
<style name="myTheme" parent="@android:style/Theme.Holo.Light">
    <item name="android:windowBackground">@color/myColor</item>
</style>
<color name="myColor">#A1B4E5</color>
```

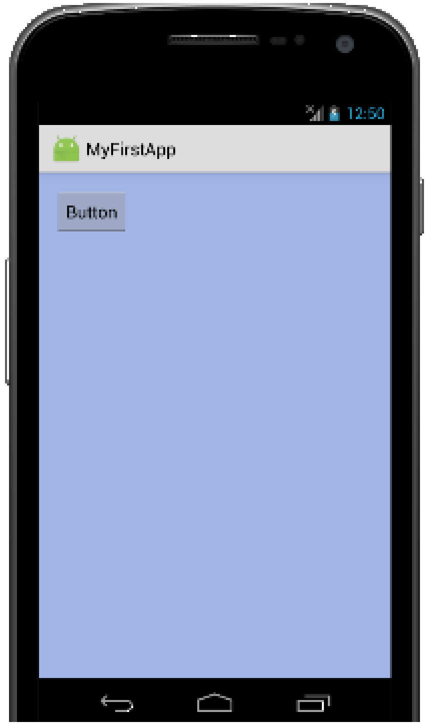




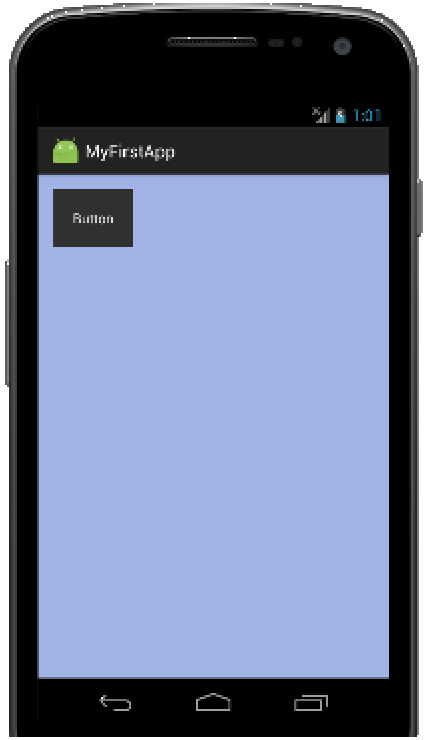
```
@android:style/Theme.Holo.Light.DarkActionBar
@android:style/Theme.Holo.Light.Dialog
@android:style/Theme.Holo.Light.Dialog.MinWidth
@android:style/Theme.Holo.Light.Dialog.NoActionBar
@android:style/Theme.Holo.Light.Dialog.NoActionBar.MinWidth
@android:style/Theme.Holo.Light.DialogWhenLarge
@android:style/Theme.Holo.Light.DialogWhenLarge.NoActionBar
@android:style/Theme.Holo.Light.NoActionBar
@android:style/Theme.Holo.Light.NoActionBar.Fullscreen
@android:style/Theme.Holo.Light.NoActionBar.Overscan
@android:style/Theme.Holo.Light.NoActionBar.TranslucentDecor
@android:style/Theme.Holo.Light.Panel
@android:style/Theme.Holo.NoActionBar
@android:style/Theme.Holo.NoActionBar.Fullscreen
@android:style/Theme.Holo.NoActionBar.Overscan
@android:style/Theme.Holo.NoActionBar.TranslucentDecor
@android:style/Theme.Holo.Panel
@android:style/Theme.Holo.Wallpaper
@android:style/Theme.Holo.Wallpaper.NoTitleBar
@android:style/Theme.InputMethod
```

Android themes

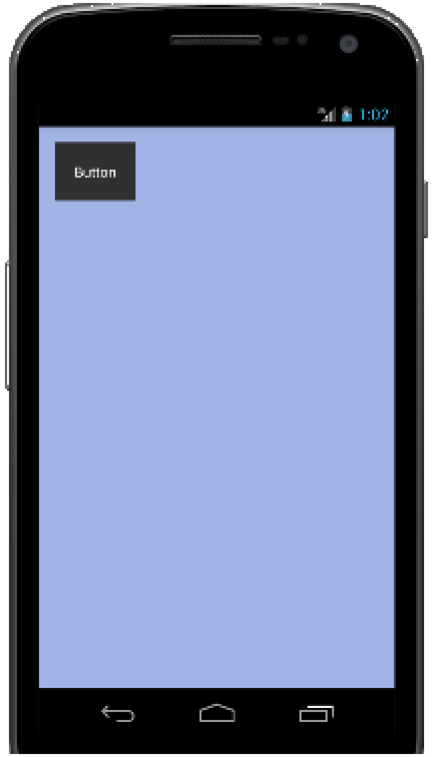
```
<style name="myTheme" parent="@android:style/Theme.Holo.???">
    <item name="android:windowBackground">@color/myColor</item>
</style>
```



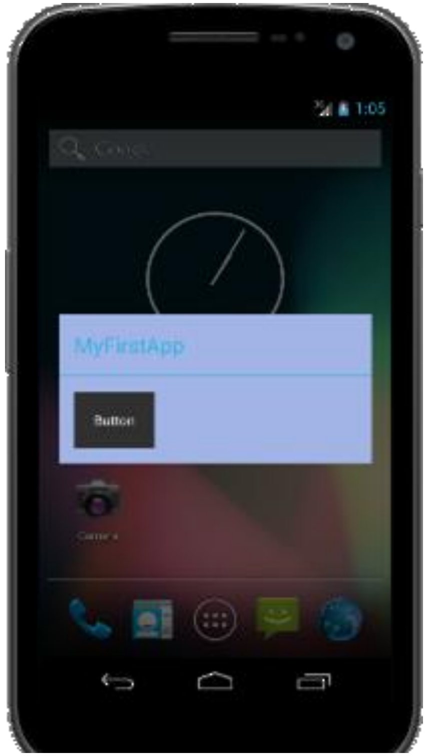
Holo.Light



Holo.Light.DarkActionBar



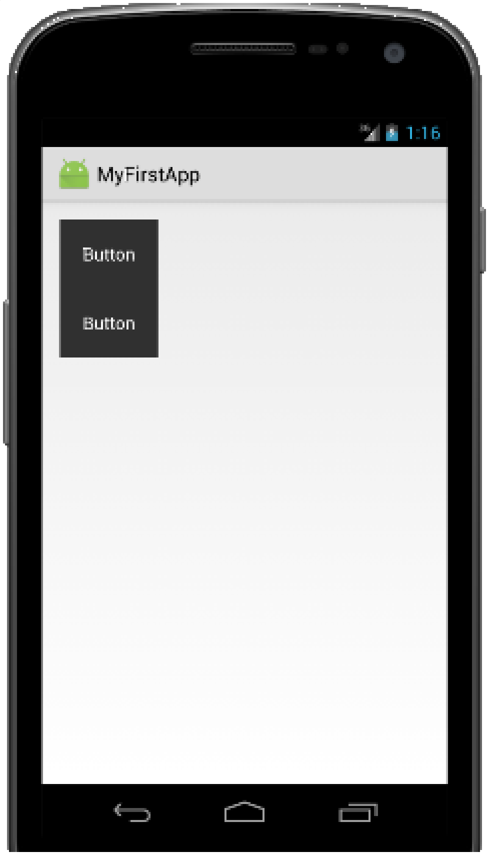
Holo.Light.NoActionBar



Holo.Light.Dialog

Using styles in a theme

```
<style name="myTheme" parent="@android:style/Theme.Holo.Light">
    <item name="android:buttonStyle">@style/myButton</item>
</style>

<style name="myButton" parent="...">
    <item name="android:textColor">#FFFFFF</item>
    <item name="android:padding">20dp</item>
    <item name="android:background">#303030</item>
</style>
```



Thank you for your attention!