

Bootcamp X Fundamentos de Android





Android Studio



- IDE Oficial
- IntelliJ IDEA + Plugins

<https://developer.android.com/studio>



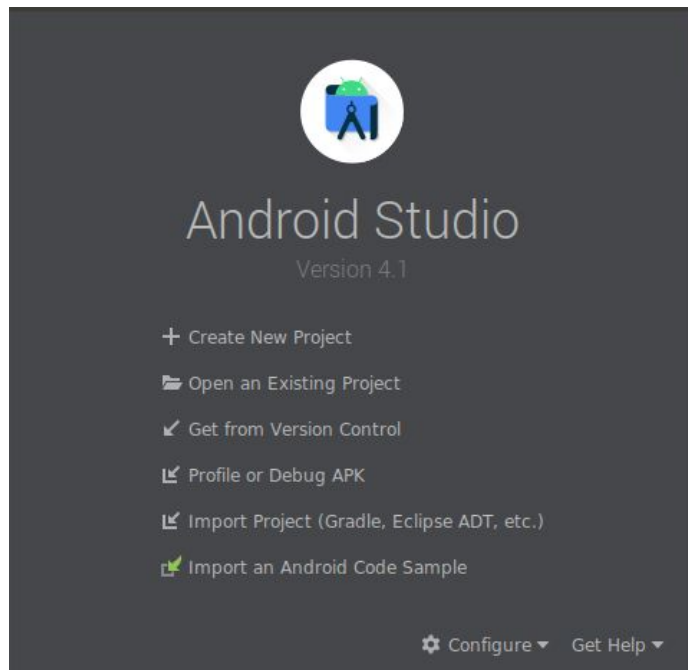
Android SDK



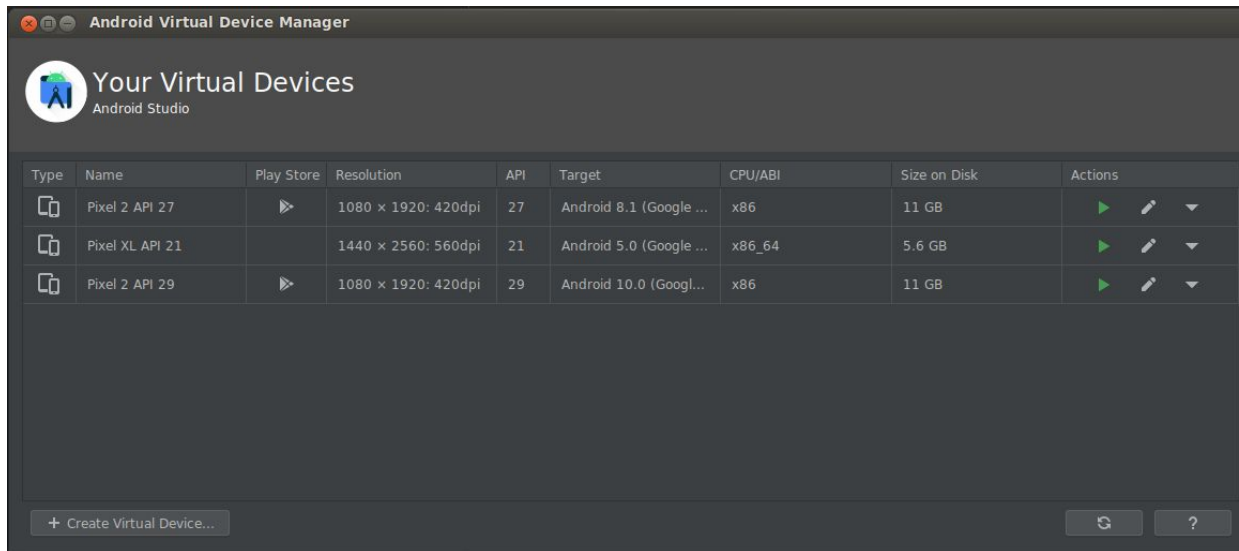
- Entorno de desarrollo
- Android API Versions

<https://developer.android.com/docs>

Welcom to Androdi Studio



Emuladores



Gradle

- Build Automation Tools
- Dependencies Manager
- Scripts
- Groovy/Kotlin DSL

```
1  plugins {  
2      id 'com.android.application'  
3      id 'kotlin-android'  
4  }  
5  
6  android {  
7      compileSdkVersion 30  
8      buildToolsVersion "30.0.2"  
9  
10     defaultConfig {  
11         applicationId "io.keepcoding.recyclerviewsampe"  
12         minSdkVersion 21  
13         targetSdkVersion 30  
14         versionCode 1  
15         versionName "1.0"  
16  
17         testInstrumentationRunner "androidx.test.runner.AndroidJUnitRunner"  
18     }  
19  
20     buildTypes {  
21         release {  
22             minifyEnabled false  
23             proguardFiles getDefaultProguardFile('proguard-android-optimize.txt'), 'proguard-rules.pro'  
24         }  
25     }  
26     compileOptions {  
27         sourceCompatibility JavaVersion.VERSION_1_8  
28         targetCompatibility JavaVersion.VERSION_1_8  
29     }  
30     kotlinOptions {  
31         jvmTarget = '1.8'  
32     }  
33 }  
34  
35 dependencies {  
36     implementation 'org.jetbrains.kotlin:kotlin-stdlib:$kotlin_version'  
37     implementation 'androidx.core:core-ktx:1.3.2'  
38     implementation 'androidx.appcompat:appcompat:1.2.0'  
39     implementation 'com.google.android.material:material:1.3.0'  
40     implementation 'androidx.constraintlayout:constraintlayout:2.0.4'  
41     testImplementation 'junit:junit:4.12.0'  
42     androidTestImplementation 'androidx.test.ext:junit:1.1.2'  
43     androidTestImplementation 'androidx.test.espresso:espresso-core:3.3.0'  
44 }  
45 }
```

Android Manifest

- Define la Información esencial de la aplicación
- Icono de la App
- Nombre de la App
- “Pantallas” (Activities)
- Servicios
- Permisos

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <manifest xmlns:android="http://schemas.android.com/apk/res/android"
3     package="io.keepcoding.recyclerviewssampe">
4
5     <application
6         android:allowBackup="true"
7         android:icon="@mipmap/ic_launcher"
8         android:label="Recycler Views Sampe"
9         android:roundIcon="@mipmap/ic_launcher_round"
10        android:supportRtl="true"
11        android:theme="@style/Theme.RecyclerViewsSampe">
12        <activity android:name=".MainActivity">
13            <intent-filter>
14                <action android:name="android.intent.action.MAIN" />
15                <category android:name="android.intent.category.LAUNCHER" />
16            </intent-filter>
17        </activity>
18    </application>
19
20 </manifest>
21
```



Kotlin

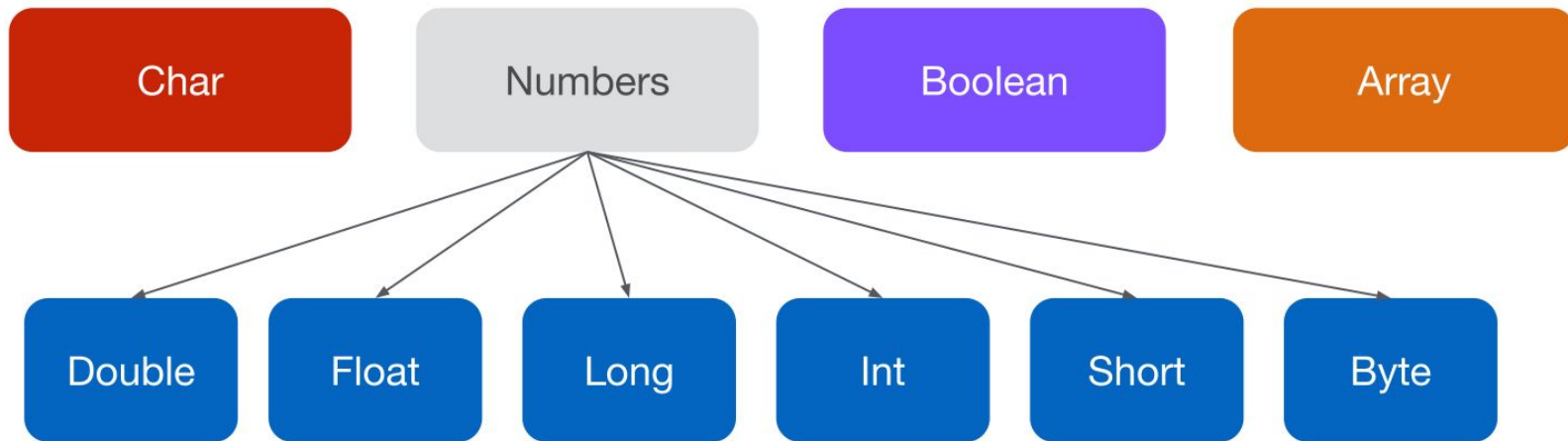
- JVM
- Interoperabilidad con Java
- Tipado Estático
- Inferencia de tipos
- Programación orientada a Objetos
- Conceptos de Lenguajes funcionales

Kotlin - Mutabilidad

```
// Mutable variable  
var variable: String = "Hello World"
```

```
// Immutable variable  
val constant: String = "Hello World"
```

Kotlin - Tipos Básicos



Kotlin - String Interpolation

```
val name = "Jc Miñarro"
```

```
val age = 31
```

```
val text = "Hello, \nmy name is $name and I am $age years old"
```

```
println("My birthday is in September and I will be ${age + 1} years old")
```

```
val message = """
```

```
    Hello,
```

```
    How are you?
```

```
    Where are you from?
```

```
""".trimIndent()
```



Kotlin - Class

- Class
- Interface
- Abstract Class
- Data Class
- Enum
- Sealed Class
- Object

Kotlin - Nullability

- Compile-Time vs Runtime
- Identificador de Nullabilidad -> ?
- Safe-Navigation Operator -> ?.
- Elvis Operator -> ?:

```
val nullableString = String? = generateStringOrNull()  
val size = nullableString?.size ?: 0
```

Kotlin - Cast

- Operador **is** y **!is**
- Operador **as** y **as?**
- Smart Cast

```
val something: Any = "Hello"  
if(something is String)  
    something.size
```

Kotlin - When

- Condition Expression

```
when (x) {  
    1 -> print("x == 1")  
    2 -> print("x == 2")  
    else -> { // Note the block  
        print("x is neither 1 nor 2")  
    }  
}
```

Kotlin - Extensions Function

```
fun ThirdPartyClass.someFunction(): String {  
    // `this` represent the instance  
    return "Hello $this"  
}
```


Kotlin - Extension Properties

- Last Index of a generic list

```
val <T> List<T>.lastIndex: Int  
    get() = size - 1
```

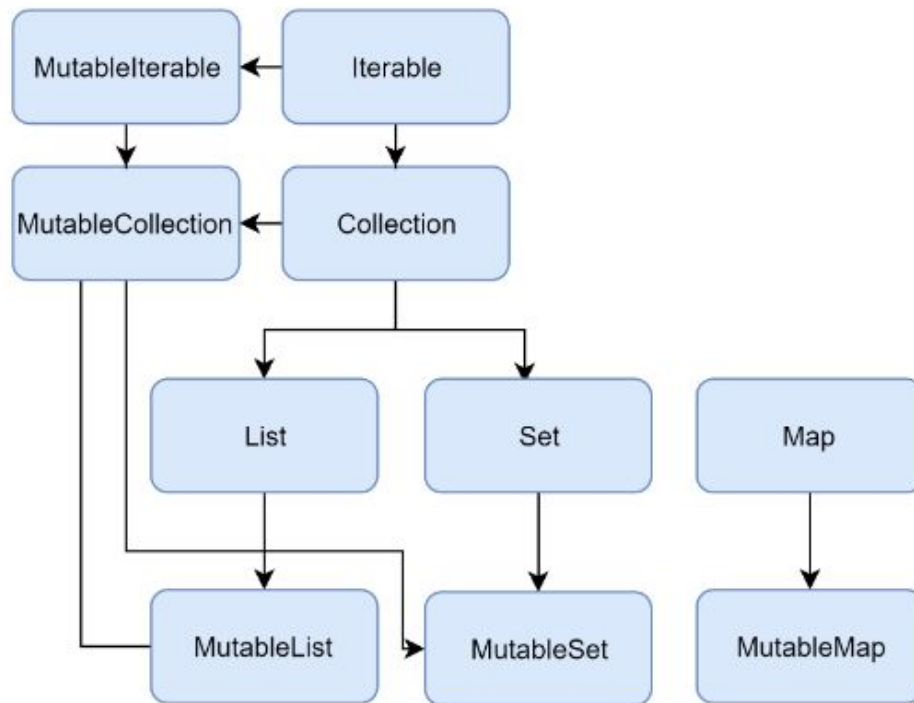
- Visibility of a View

```
var View.isVisible: Boolean  
    get() = visibility == View.VISIBLE  
    set(value) {  
        visibility = if (value) View.VISIBLE else View.GONE  
    }
```

Kotlin - Extension Function

- `fun <T, R> T.let(block: (T) -> R): R`
- `fun <T> T.apply(block: T.() -> Unit): T`
- `fun <T, R> with(receiver: T, block: T.() -> R): R`
- `fun <T, R> T.run(block: T.() -> R): R`
- `fun <T> T.also(block: (T) -> Unit): T`

Kotlin - Collections



Kotlin - Collections

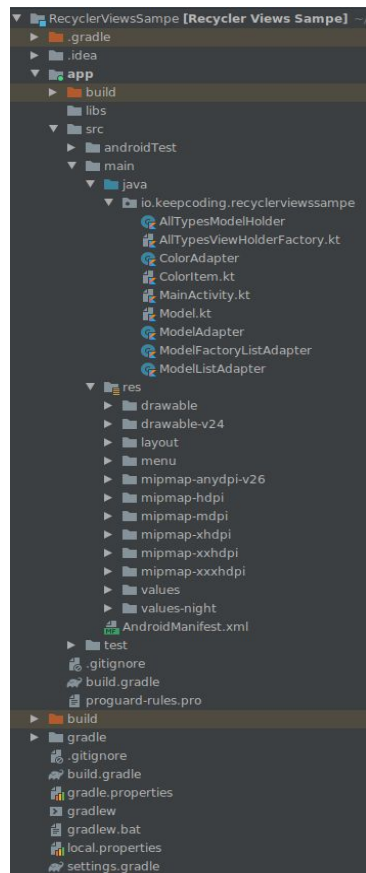
- `listOf(1, 2, 3, 4) // [1, 2, 3, 4]`
- `mutableListOf("a", "b", "c") // ["a", "b", "c"]`
- `emptyList<Double>() // []`
- `setOfNotNull(1, null, 3, null) // [1, 3]`
- `mutableMapOf(1 to "first", 2 to "second", 3 to "third")
// [1:"first", 2:"second", 3:"third"]`

Kotlin - Collections

- `forEach { }`
- `map { }`
- `flatMap { }`
- `fold { }`

Estructura del proyecto

- Proyecto
 - Modulo App
 - src
 - main
 - java
 - res
 - test
 - build.gradle (modulo)
 - proguard
 - build.gradle (Global del proyecto)
 - settings.gradle
 - gradlew





Resources

- String
- Color
- Dimens
- Styles
- Drawables
- Layout
- Menu



Activity



- Punto de entrada
- Define “la pantalla” que se muestra
- Debe estar incluida en el Manifest



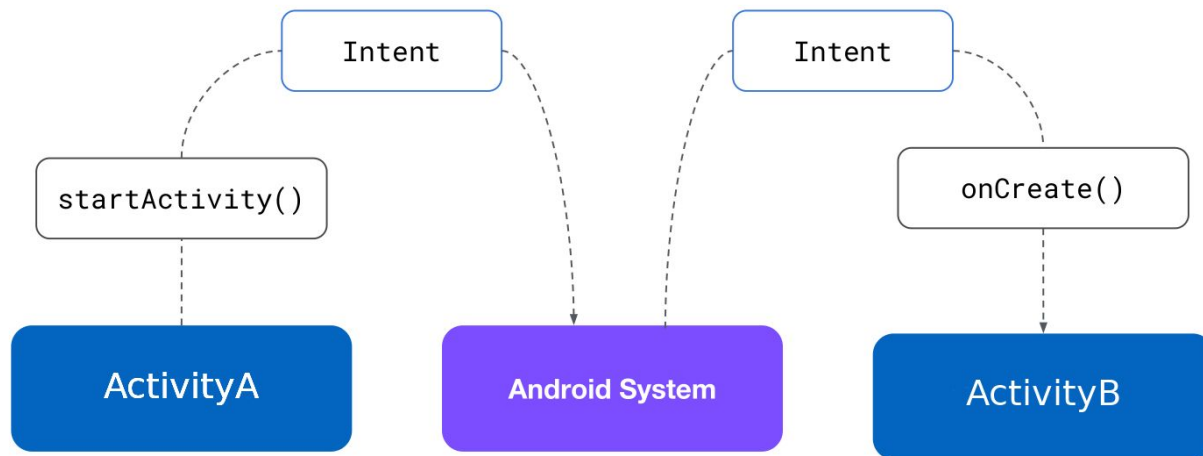
Context



- Obtención de recursos específicos de la aplicación
- Hay diferentes tipos:
 - Application
 - Activity
 - Service
 - BroadcastReceiver
 - ContentProvider

Intent

- Objeto de intercambio de información
- Define Acciones a realizar
- Intent-Filter
- Extras



Activity Lifecycle



onCreate()

onStart()

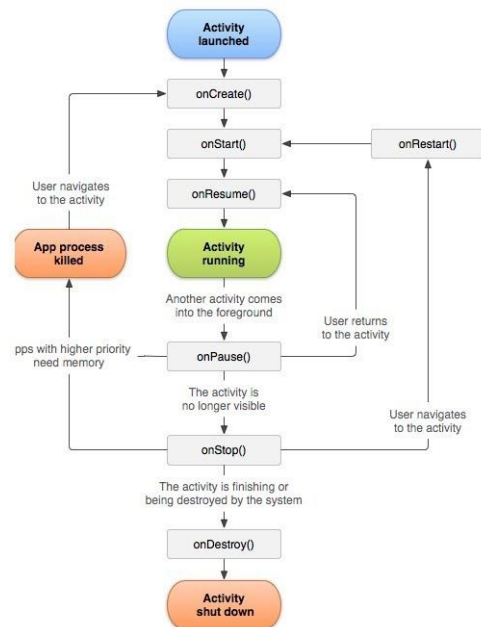
onRestart()

onResume()

onPause()

onStop()

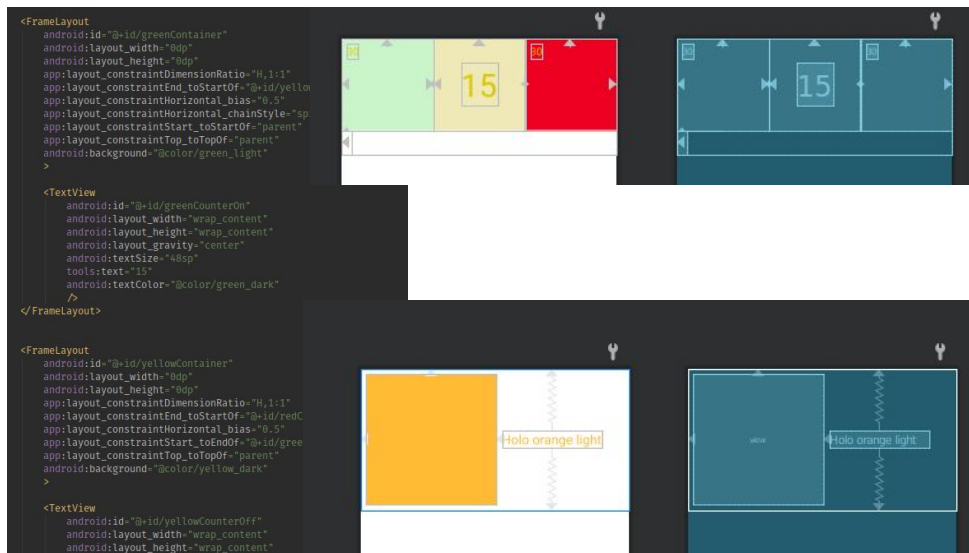
onDestroy()



Vistas en Android

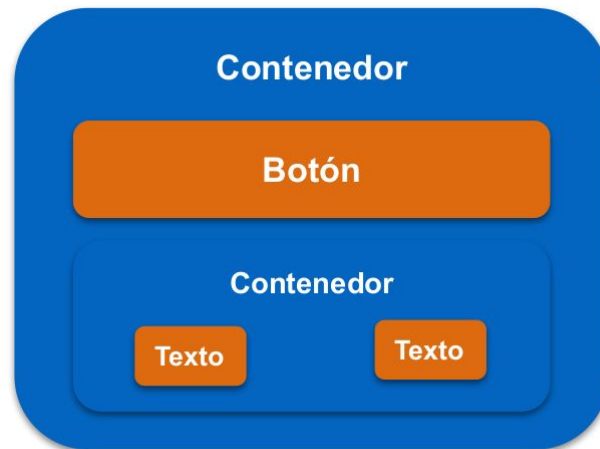
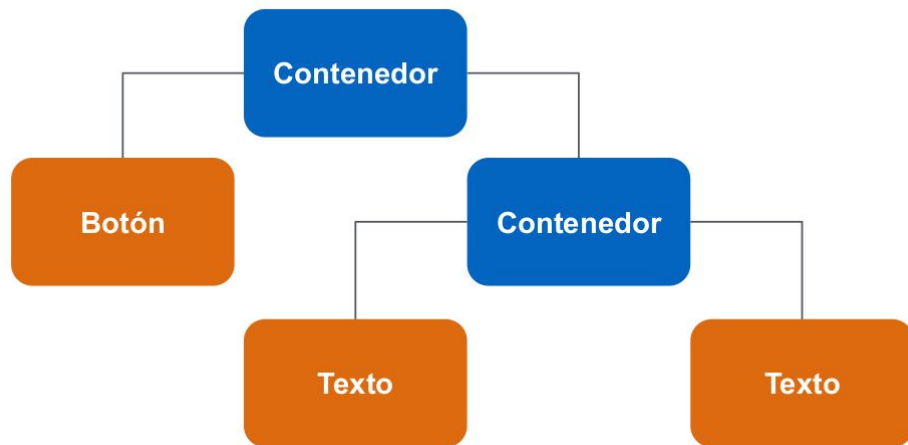
● View

- TextView
- EditText
- Button
- ImageView
- Snackbar
- ViewGroup
 - LinearLayout
 - RelativeLayout
 - FrameLayout
 - ConstraintLayout



View

Jerarquía de Vistas



View

```
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    >

    <TextView
        android:id="@+id/viewId"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/text"
        android:layout_margin="8dp"
        android:textSize="@dimen/text_size"
    />
</FrameLayout>
```

TextView



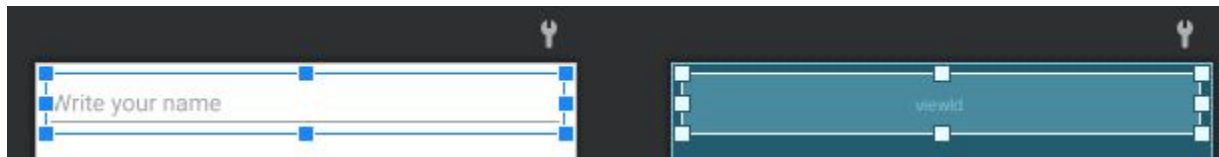
<TextView

```
    android:id="@+id/viewId"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="@string/text"  
    android:layout_margin="8dp"  
    android:textSize="@dimen/text_size"  
    android:textColor="@android:color/holo_purple"  
    android:background="@android:color/holo_orange_dark"  
/>
```

EditText

<EditText

```
    android:id="@+id/viewId"  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:layout_margin="8dp"  
    android:hint="@dimen/hint_name"  
    android:inputType="text|textCapWords"  
/>
```



Button

```
<?xml version="1.0" encoding="UTF-8"?>
<shape xmlns:android="http://schemas.android.com/apk/res/android">
    <stroke android:width="2dp" android:color="@color/yellow_dark" />
    <corners android:radius="5dp" />
    <solid android:color="@color/red_dark" />
</shape>
```

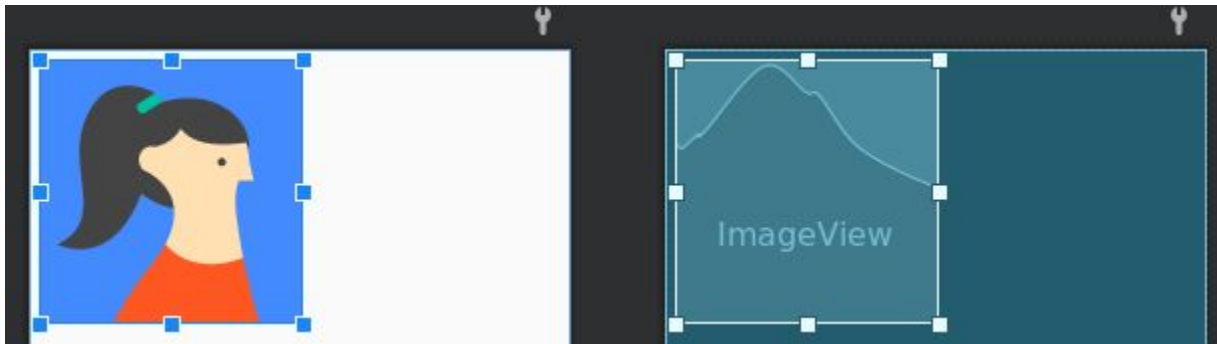
<Button

```
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_margin="8dp"
    android:text="@string/not_click_me"
    android:textColor="@color/yellow_dark"
    android:background="@drawable/button_background"
/>
```



ImageView

```
<ImageView  
    android:id="@+id/avatar"  
    android:layout_width="200dp"  
    android:layout_height="200dp"  
    android:layout_margin="8dp"  
    tools:src="@tools:sample/avatars"  
/>
```



Snackbar

Snackbar

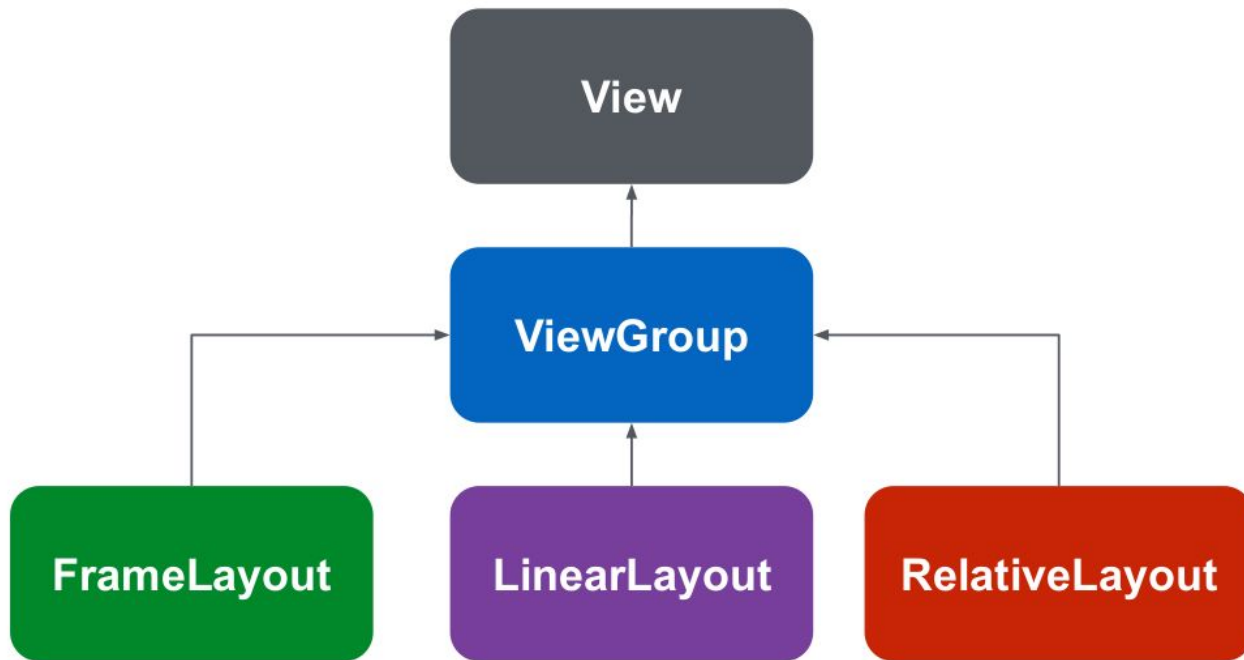
```
.make(targetView, "Hello, I am an Snackbar", Snackbar.LENGTH_SHORT)  
.show()
```



Rg0JJQU1FclqjCka1jcY

Hello, I am an Snackbar

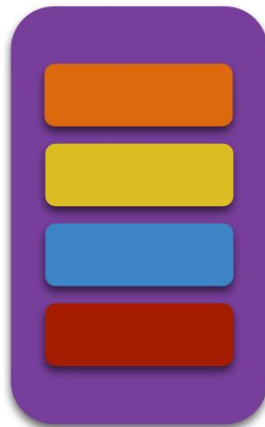
ViewGroup



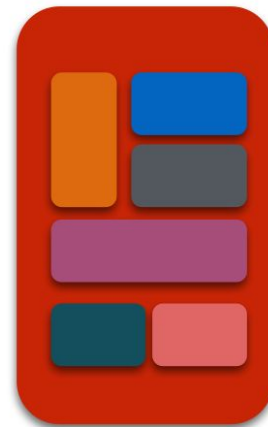
ViewGroup



FrameLayout



LinearLayout



RelativeLayout /
ConstraintLayout



Styles and Theme

- Apariencia de nuestra App
- Define:
 - Colores
 - Fuentes
 - Apariencia de Textos
 - Dimensions
 - Gravity

View Listeners

```
button.setOnClickListener { view ->
    // `view` represent the view that was clicked
    performAction()
}
```

```
button.setOnLongClickListener {
    // Long Click over the button
    performOtherAction()
}
```

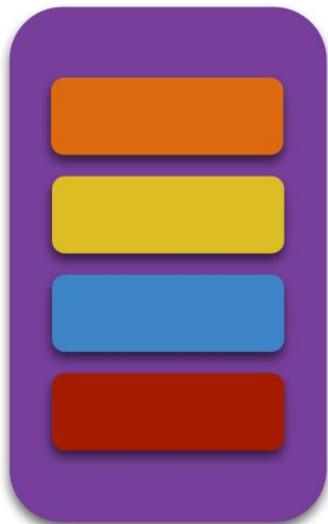


RecyclerView

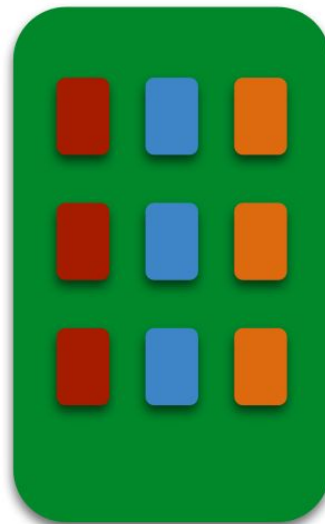


- Listado de datos
- Renderización optimizada
- Reutiliza vistas

RecyclerView - LayoutManager



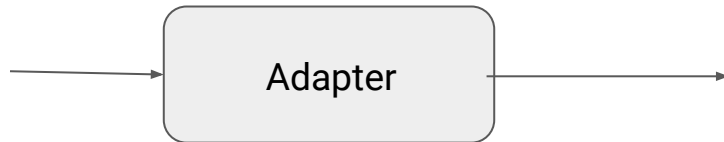
LinearLayoutManager



GridLayoutManager

RecyclerView - Adapter

Orange
Yellow
Blue
Red





SharedPreferences



- Internal XML File
- Datos Primitivos
- Tamaño Limitado
- Lento



Jetpack DataStore

- Primitives
- Protocol buffer
- Flow
- Beta Version



DataBase - SQLite

- Gestor DB “de fábrica”
- Operaciones limitadas
- Procesamiento lento
- Complejo de configurar (Queries, SQLiteOpenHelper...)



DataBase - Realm



- No-SQL DataBase
- Object-Oriented Data Mode
- Muy rápida
- Multiplatform
- Online Sync



DataBase - Room

- SQLite Wrapper
- Object-Oriented Data Mode
- Fácil de configurar
- Asincronía



KEEPCODING

Tech School

Madrid | Barcelona | Bogotá

Datos de contacto