

Contador

1. Les parts més important d'un projecte android son:

- Fitxer AndroidManifest.xml ubicat en la carpeta manifests on es descriuran les propietats del projecte, com la api que utilitza i les activitats que te.
- En la carpeta java, estaran definides les activitats, separades per archius kt.
- Existeix una carpeta anomenada res que serveix per a guardar els recursos que es gastaran en la app. Les carpetes més important son drawable i layout. En la carpeta drawable estan els xml de cada activitat. En aquest se definix la estructura de la activitat i de que parts consta. En Android Studio, hi ha un apartat per modificar aquest arxiu amb codi i també un per a fer-ho gràficament.

2. Per a crear una nova activitat, s'afegirà al Android Manifest l'etiqueta "activity".

```
<activity
    android:name=".MainActivity"
    android:exported="true">
    <intent-filter>
        <action android:name="android.intent.action.MAIN"/>
        <category android:name="android.intent.category.LAUNCHER"/>
    </intent-filter>
</activity>
```

- A més, en la carpeta java, s'ha d'afegir un arxiu kt amb el nom de l'activitat (En aquest cas, MainActivity).
- En la carpeta res/layout es crearan els xml que definiran les activitats (botons, text view...).

3.

4. L'aplicació presenta un problema, el text view perd el valor del comptador a l'hora de rotar el dispositiu.

5. Per a solucionar el error descrit abans, em d'afegir aquest codi que comprovarà si existeix algun valor en la variable comptador (la que utilitza el TextView):

```
// Es comprova si existeix algun estat, si existeix es restaura el
comptador
if (savedInstanceState != null) {
    comptador = savedInstanceState.getInt("comptador", 0)
}
```

Es crea una funció que guardara l'estat de la variable comptador i es relaciona en la clau "comptador":

```
override fun onSaveInstanceState(outState: Bundle) {
    outState.putInt("comptador", comptador)
    super.onSaveInstanceState(outState)
}
```

6. Per a restar i resetejar hem d'afegir 2 botons semblant al de afegir:

- Afegirem aquest codi al MainActivity.kt

```
// Boto per a restar
val btSubtract = findViewById<Button>(R.id.btSubtract)

//Boto Reset
val btReset= findViewById<Button>(R.id.btReset)

//Métode per a restar
btSubtract.setOnClickListener {
    comptador--
    textViewContador.text = comptador.toString()
}

//Métode per a resetejar a 0
btReset.setOnClickListener{
    comptador=0
    textViewContador.setText(comptador.toString())
}
```

- Després, en la carpeta res/layout, afegirem aquest codi:

```
<Button
    android:id="@+id/btSubtract"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="-"
    android:textSize="40sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.625"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/textViewComptador"
    app:layout_constraintVertical_bias="0.109" />

<Button
    android:id="@+id/btReset"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Reset"
    android:textSize="40sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/textViewComptador"
    app:layout_constraintVertical_bias="0.383" />
```

- En el codi anterior es declara la clau de cada boto, la posició d'aquests i el tamany entre altres.