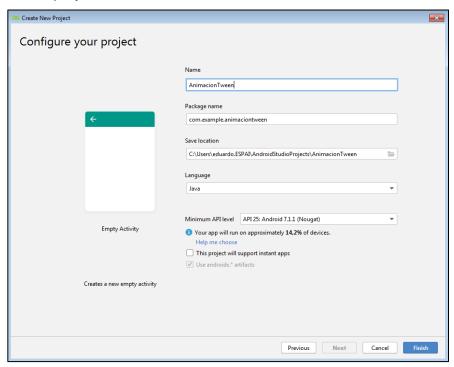
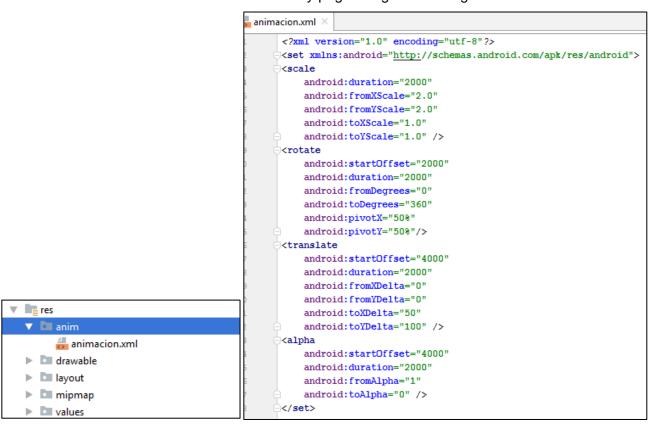
PRACTICA 4: ANIMACIONTWEEN

Parte I: Creación de una animación Tween para animar una vista

Paso 1. Crea un nuevo proyecto con nombre AnimacionTween.



Paso 2. Crea el fichero res/anim/animacion.xml y pega el siguiente código:



Paso 3. Abre el fichero res/Layout/main.xml y añade el siguiente atributo a la vista de tipo TextView: android:id="@+id/textView"

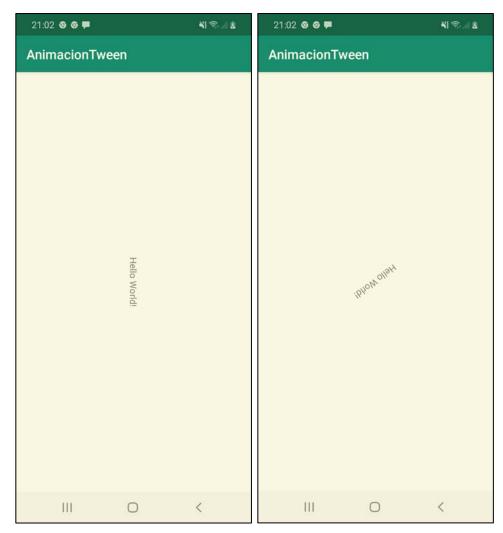
```
activity_main.xml
     <?xml version="1.0" encoding="utf-8"?>
   <androidx.constraintlayout.widget.ConstraintLayout</p>
         xmlns:android="http://schemas.android.com/apk/res/android"
         xmlns:app="http://schemas.android.com/apk/res-auto"
         xmlns:tools="http://schemas.android.com/tools"
         android:layout width="match parent"
         android:layout height="match parent"
         tools:context=".MainActivity">
        <TextView
             android:id="@+id/textView"
             android:layout width="wrap content"
             android:layout height="wrap content"
             android:text="Hello World!"
             app:layout constraintBottom toBottomOf="parent"
             app:layout constraintLeft toLeftOf="parent"
             app:layout_constraintRight_toRightOf="parent"
             app:layout constraintTop toTopOf="parent" />
   </androidx.constraintlayout.widget.ConstraintLayout>
```

Paso 4. Abre la actividad del proyecto y añade las líneas marcadas en negrita al método OnCreate().

```
public class MainActivity extends AppCompatActivity {

@Override public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    TextView texto = (TextView) findViewById(R.id.textView);
    Animation animacion = AnimationUtils.loadAnimation(context this, R.anim.animacion);
    texto.startAnimation(animacion);
}
```

Paso 5. Ejecuta la aplicación.



Como podrás ver, el TextView comienza haciéndose más pequeño (etiqueta <scale>), después rota sobre sí mismo (etiqueta <rotate>) y finalmente, se desplaza (etiqueta <translate>) a la vez que se hace transparente (etiqueta <alpha>). Al finalizar la animación, vuelve a su posición y estado inicial, sin importar dónde ni como haya acabado.