



DESARROLLO DE APLICACIONES MÓVILES AVANZADAS



“USO DE LIBRERÍAS PARA IMÁGENES EN ANDROID”

Resultados de Aprendizaje:

- Comprender los elementos necesarios para iniciarse en el desarrollo de aplicaciones basadas en Android.
- Desarrollar proyectos haciendo uso de librerías avanzadas que proporciona Android.

INDICE

INTRODUCCIÓN	3
CREANDO NUESTRO PROYECTO	3
PRÁCTICA POR REALIZAR	12
RUBRICA DE EVALUACIÓN	13

INTRODUCCIÓN

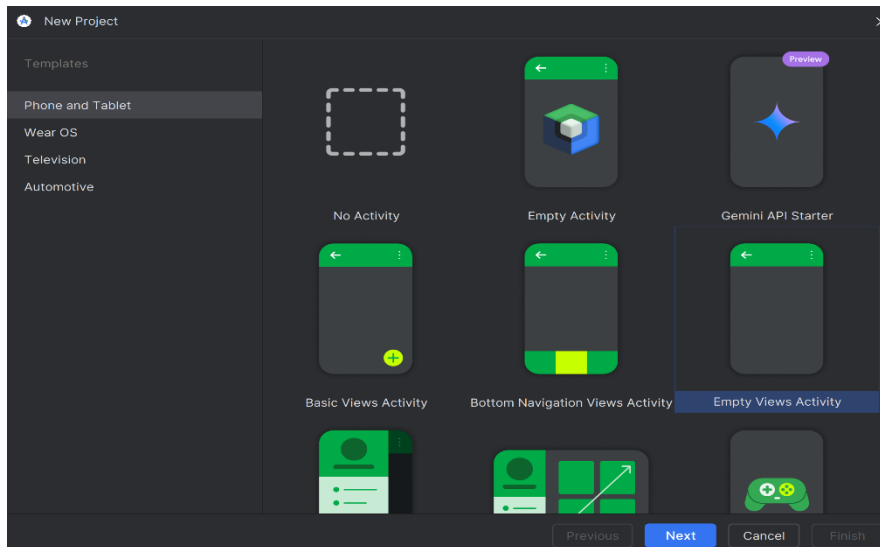
Para desarrollar esta práctica usted deberá tener los siguientes paquetes de software:

- Android Studio Versión 4 o superior
- Equipo Móvil con Android (Puede ser Tablet o Smartphone) con **modo desarrollador habilitado**.

(Ver <https://www.youtube.com/watch?v=wLJS8CKo95o>)

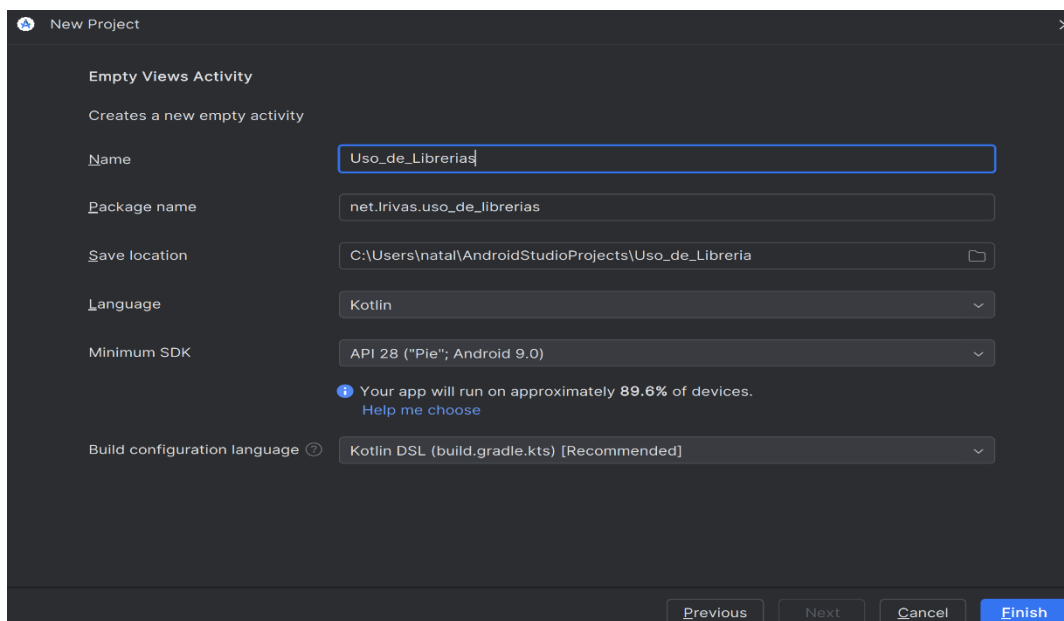
CREANDO NUESTRO PROYECTO

1. Iniciamos nuestro Android Studio.
2. Elegimos la plantilla **Empty Views Activity**.

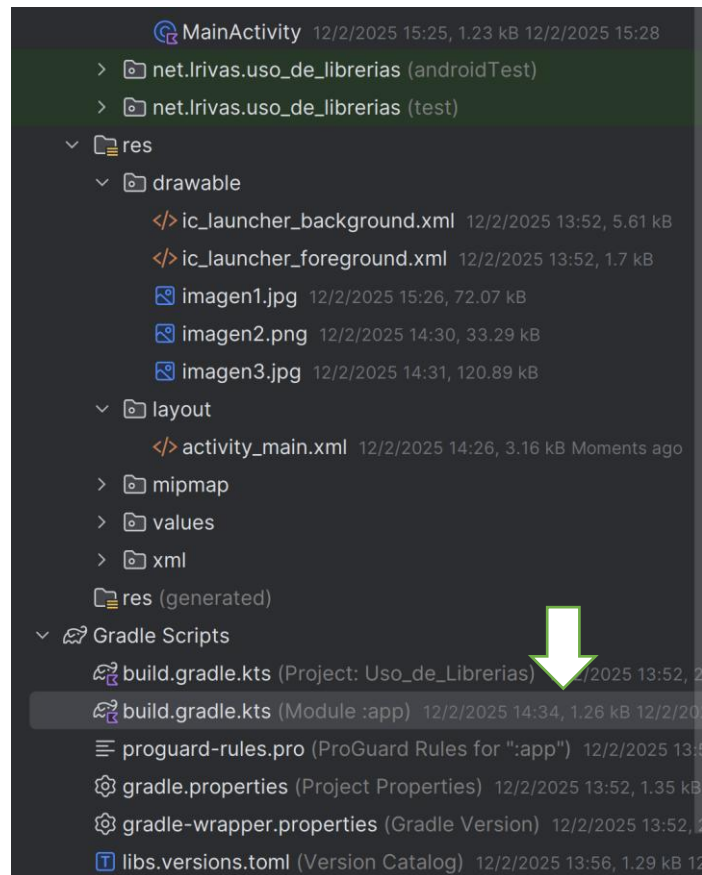


3. Luego establezca las siguientes propiedades al proyecto:

- a. Nombre: **Uso_de_Libreria**
- b. Package Name: **net.lrivas.usodelibrerias**
- c. Lenguaje: **Kotlin**
- d. Minimum Api: **28**



4. Implemente la librería Picasso, para ello agregue las siguientes líneas en el archivo Gradle.

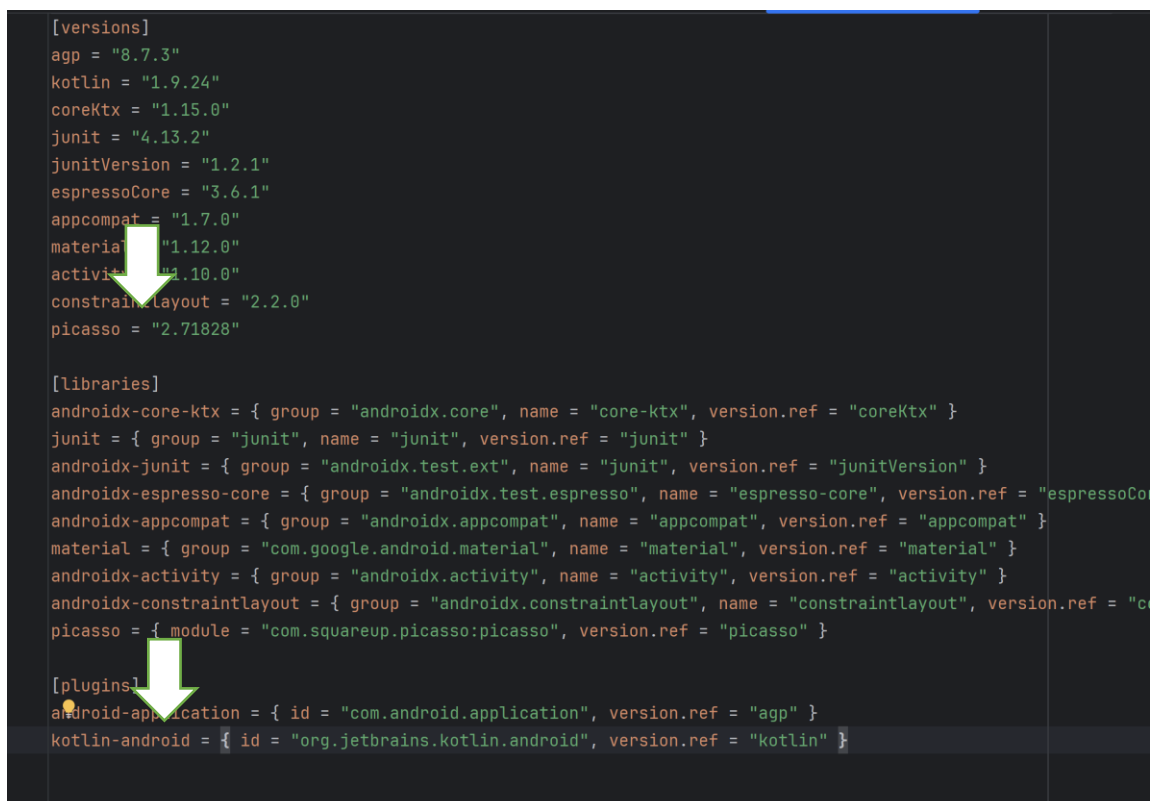
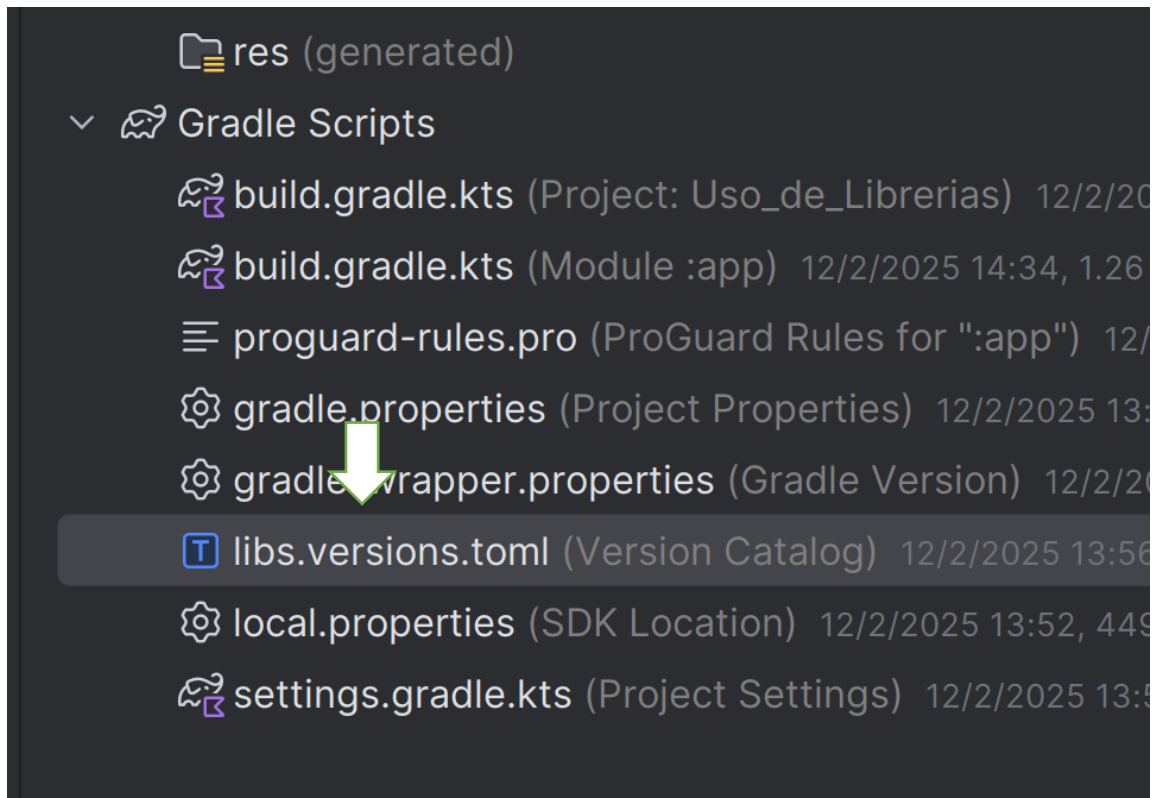


```
dependencies {

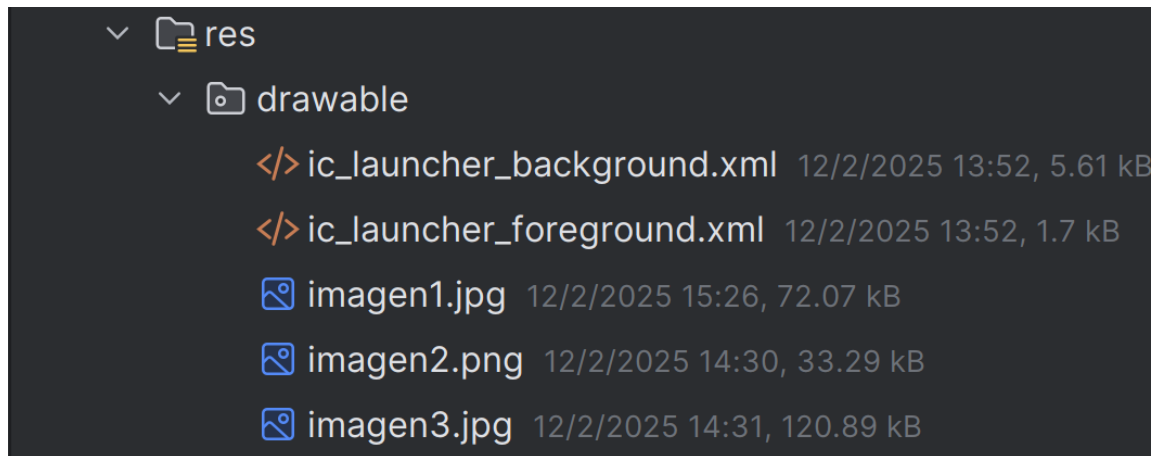
    implementation(libs.androidx.core.ktx)
    implementation(libs.androidx.appcompat)
    implementation(libs.material)
    implementation(libs.androidx.activity)
    implementation(libs.androidx.constraintlayout)
    testImplementation(libs.junit)
    androidTestImplementation(libs.androidx.junit)
    androidTestImplementation(libs.androidx.espresso.core)
    implementation(libs.picasso)

}
```

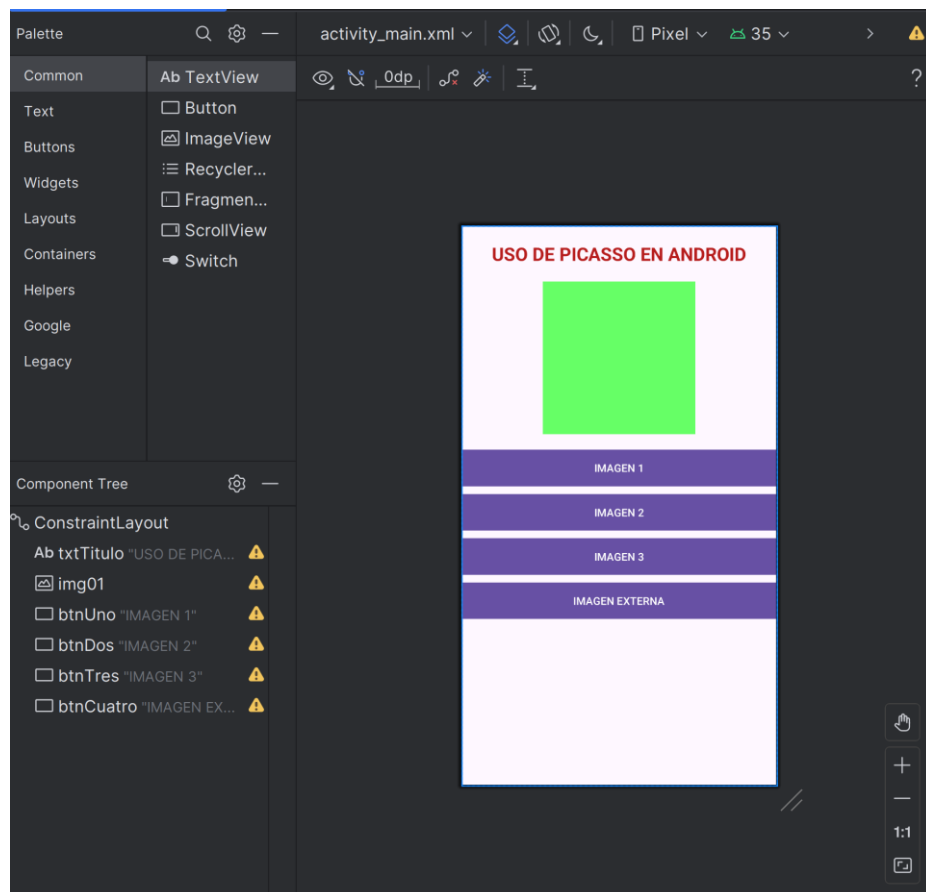
5. Agregue la librería al archivo libs.versions.toml con los siguientes códigos y de clic en Sync Now.



6. Asegúrese de descargar los recursos de esta práctica, copie y pegue las imágenes en el directorio drawable, tal como se muestra a continuación.



7. En el archivo **activity_main**, vamos a construir la siguiente interfaz:



```
1 <?xml version="1.0" encoding="utf-8"?>
2 <androidx.constraintlayout.widget.ConstraintLayout
3     xmlns:android="http://schemas.android.com/apk/res/android"
4     xmlns:app="http://schemas.android.com/apk/res-auto"
5     xmlns:tools="http://schemas.android.com/tools"
6     android:layout_width="match_parent"
7     android:layout_height="match_parent"
8     tools:context=".MainActivity">
9
10    <!-- Título -->
11    <TextView
12        android:id="@+id/txtTítulo"
13        android:layout_width="wrap_content"
14        android:layout_height="wrap_content"
15        android:text="USO DE PICASSO EN ANDROID"
16        android:textColor="#B71C1C"
17        android:textSize="24sp"
18        android:textStyle="bold"
19        app:layout_constraintTop_toTopOf="parent"
20        app:layout_constraintStart_toStartOf="parent"
21        app:layout_constraintEnd_toEndOf="parent"
22        android:layout_marginTop="20dp"/>
23
24    <!-- ImageView -->
25    <ImageView
26        android:id="@+id/img01"
27        android:layout_width="200dp"
28        android:layout_height="200dp"
29        android:background="#66FF66"
30        app:layout_constraintTop_toBottomOf="@id/txtTítulo"
31        app:layout_constraintStart_toStartOf="parent"
32        app:layout_constraintEnd_toEndOf="parent"
```

```

<androidx.constraintlayout.widget.ConstraintLayout
    <ImageView
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        android:layout_marginTop="20dp"/>

<!-- Botón 1 -->
<Button
    android:id="@+id/btnUno"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:text="IMAGEN 1"
    android:textColor="#FFFFFF"
    android:background="#6200EE"
    app:layout_constraintTop_toBottomOf="@id/img01"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    android:layout_marginTop="20dp"/>

<!-- Botón 2 -->
<Button
    android:id="@+id/btnDos"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:text="IMAGEN 2"
    android:textColor="#FFFFFF"
    android:background="#6200EE"
    app:layout_constraintTop_toBottomOf="@id/btnUno"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    android:layout_marginTop="10dp"/>

```



```
        android:layout_marginTop="10dp"/>

<!-- Botón 3 -->
<Button
    android:id="@+id/btnTres"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:text="IMAGEN 3"
    android:textColor="#FFFFFF"
    android:background="#6200EE"
    app:layout_constraintTop_toBottomOf="@id/btnDos"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    android:layout_marginTop="10dp"/>

<!-- Botón 4 -->
<Button
    android:id="@+id/btnCuatro"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:text="IMAGEN EXTERNA"
    android:textColor="#FFFFFF"
    android:background="#6200EE"
    app:layout_constraintTop_toBottomOf="@id/btnTres"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    android:layout_marginTop="10dp"/>

</androidx.constraintlayout.widget.ConstraintLayout>
```

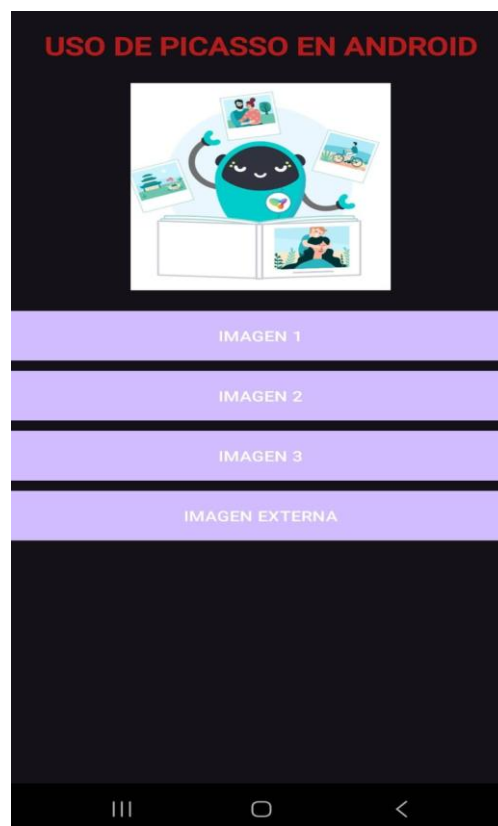
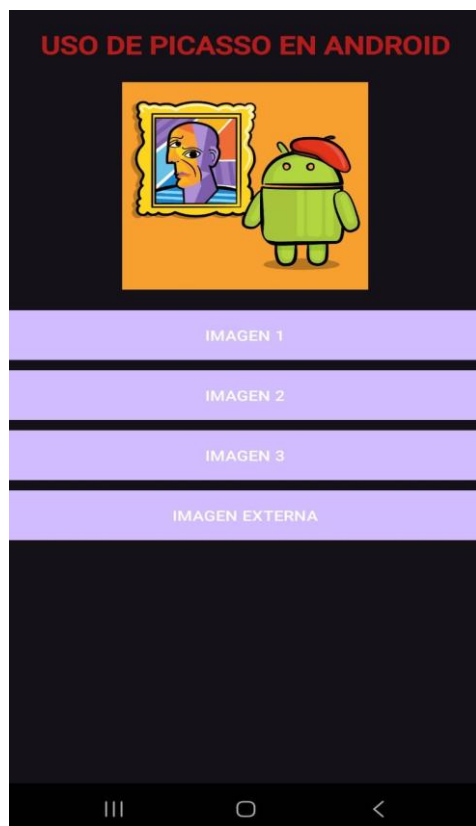
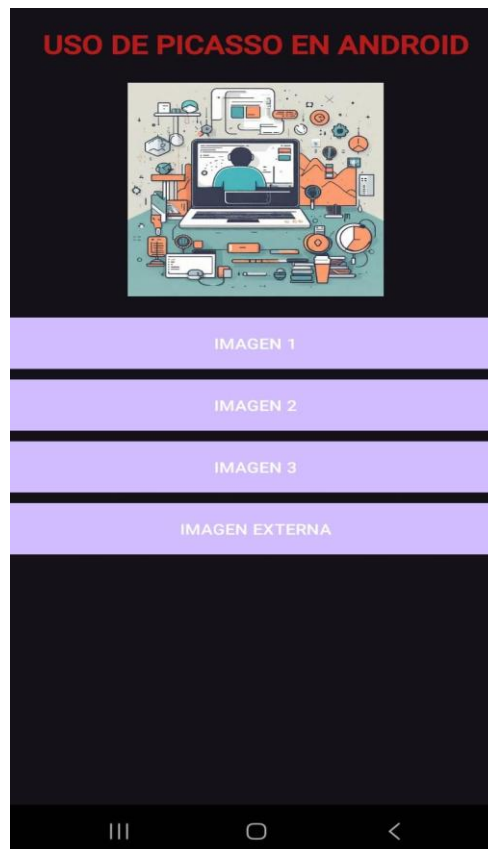
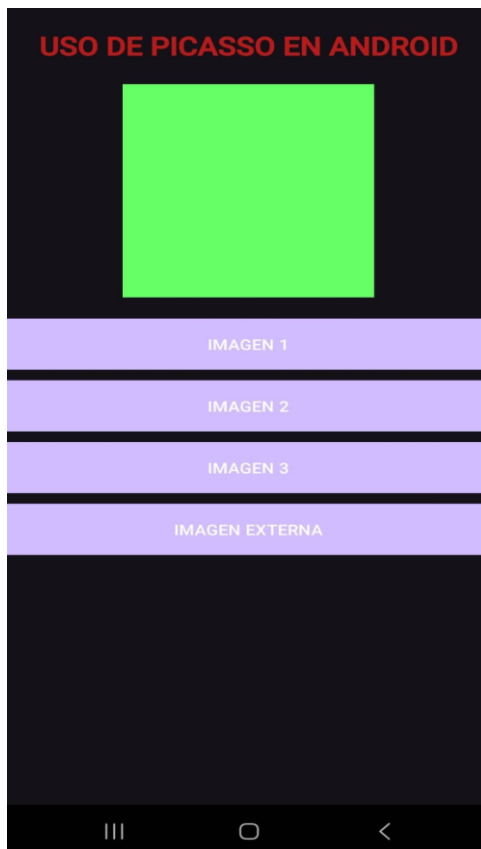
8. Luego en nuestro código del **MainActivity** deberemos configurar los eventos clic y las referencias para cada objeto de nuestra interfaz.

```

</> activity_main.xml      MainActivity.kt  ×
1      package net.lrivas.uso_de_librerias
2
3      import android.os.Bundle
4      import android.widget.Button
5      import android.widget.ImageView
6      import androidx.appcompat.app.AppCompatActivity
7      import com.squareup.picasso.Picasso
8
9      class MainActivity : AppCompatActivity() {
10         override fun onCreate(savedInstanceState: Bundle?) {
11             super.onCreate(savedInstanceState)
12             setContentView(R.layout.activity_main)
13
14             val imgView: ImageView = findViewById(R.id.img01)
15             val btnUno: Button = findViewById(R.id.btnUno)
16             val btnDos: Button = findViewById(R.id.btnDos)
17             val btnTres: Button = findViewById(R.id.btnTres)
18             val btnCuatro: Button = findViewById(R.id.btnCuatro)
19
20             btnUno.setOnClickListener {
21                 imgView.setImageResource(R.drawable.imagen1)
22             }
23
24             btnDos.setOnClickListener {
25                 imgView.setImageResource(R.drawable.imagen2)
26             }
27
28             btnTres.setOnClickListener {
29                 imgView.setImageResource(R.drawable.imagen3)
30             }
31
32             btnCuatro.setOnClickListener {
33                 Picasso.get()
34                     .load(path: "https://web-design-eastbourne.co.uk/News/wp-content/uploads/2023/11/Backend-Developer.jpeg")
35                     .into(imgView)
36             }
37         }
38     }

```

9. Compile su aplicación, deberá cargarle de la siguiente manera:




PRÁCTICA POR REALIZAR

NOMBRE: Uso de Librerías

Indicación: haciendo uso del contenido visto en la semana, haga un documento **pdf** que contenga **capturas de pantallas y el código fuente** del siguiente requerimiento:

Diseño de la Interfaz por crear



NOMBRE DE LA SERIE O PELICULA:
XXXXXX

FECHA DE LANZAMIENTO:
XXXXXX

AUTORES PRINCIPALES:
— XXXXXXXXXXXXX
— XXXXXXXXXXXXX
— XXXXXXXXXXXXX

DESCRIPCIÓN DE LA SERIE O PELICULA:
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

SITIO WEB

- A. Deberá crear un **Auto Image Slider**, puedes auxiliarte de este recurso multimedia https://youtu.be/mP-M_k8c2O4?si=DNeaxVeg3xvm11OK. El control Auto Image Slider deberá contener 6 imágenes que pueden ser locales o remotas sobre una temática de serie o película de su preferencia.
- B. Deberá debajo del Image Slider agregar información relacionada a la película o serie de su preferencia, entre los datos están **Nombre de la Película o Serie, Fecha en que se lanzón, Autores Principales y Descripción de la película o serie. (Quedará a su criterio el diseño de la interfaz, colores, distribución del contenido, etc).**
- C. Agregue un botón que al darle clic le abra el navegador con la URL oficial de la película o serie.

FORMA DE ENTREGA: Se deberá enviar al buzón de tarea llamado **Uso de Librerías para Imágenes**, de manera individual.

TIPO DE ENTREGA: Enviar un documento en **PDF o DOCX** al final de la semana antes de las **23:59**.

RUBRICA DE EVALUACIÓN

Indicación: a continuación, se establecen los criterios de evaluación para la actividad de la semana.

#	CRITERIO	PTS.
1	Entrega el proyecto en la fecha establecida.	2.0
2	Diseña la interfaz principal de acuerdo con el requerimiento.	0.5
3	Implementa el Auto Image Slider con sus respectivas imágenes.	3.0
4	Diseña el contenido de la serie elegida.	0.5
5	Agrega la funcionalidad de visitar el sitio web oficial de la serie.	0.5
6	Elabora el ejemplo de la práctica.	1.5
7	Sube el proyecto a git y anexa el enlace en el documento.	1.0
8	El documento está ordenado y sin errores de ortografía.	1.0