



Sistema Universidad Abierta

UNIDAD 1 ACTIVIDAD 3

Materia:

PROGRAMACIÓN DE DISPOSITIVOS MÓVILES

Asesor:

Cristian Cardoso Arellano

Semestre:

**6°
Semestre**

Alumno:

Sealtiel Esteban Solano Flores

Unidad 1.

Actividad 4.

1. Crear tu primer activity
2. En lenguaje de programación JAVA dentro de Android Studio, crea una app vacía.
3. Crear un arreglo con 1 millón de elementos enteros de manera aleatoria.
4. Ordenar los elementos.
5. Medir el tiempo de inicio a fin.
6. Imprimir en el logcat los resultados obtenidos.

Activity Main XML:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout 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:orientation="vertical"
    android:gravity="center"
    android:padding="24dp"
    android:background="#F0F4F8"
    tools:context=".MainActivity">

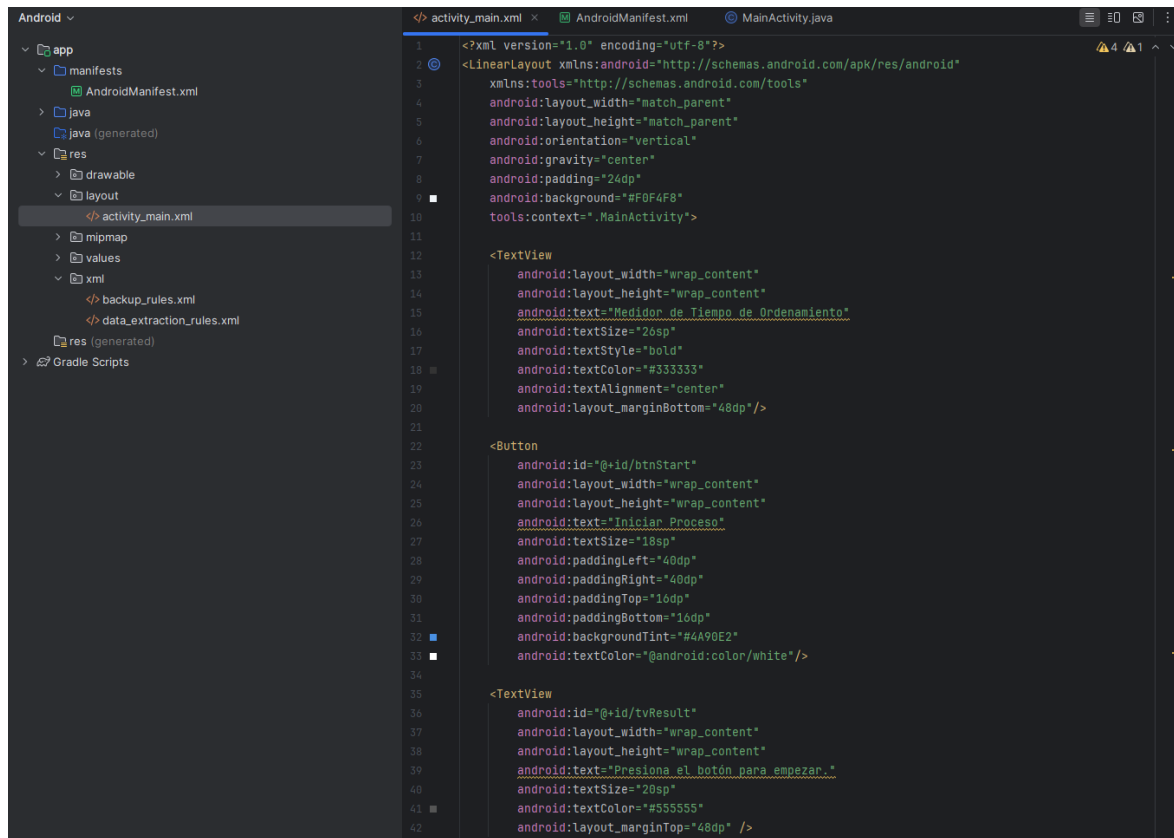
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Medidor de Tiempo de Ordenamiento"
        android:textSize="26sp"
        android:textStyle="bold"
        android:textColor="#333333"
        android:textAlignment="center"
        android:layout_marginBottom="48dp"/>

    <Button
        android:id="@+id/btnStart"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Iniciar Proceso"
        android:textSize="18sp"
        android:paddingLeft="40dp"
        android:paddingRight="40dp"
        android:paddingTop="16dp"
        android:paddingBottom="16dp"
        android:backgroundTint="#4A90E2"
        android:textColor="@android:color/white"/>

    <TextView
        android:id="@+id/tvResult"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Presiona el botón para empezar."
        android:textSize="20sp"
        android:textColor="#555555"
```

```
android:layout_marginTop="48dp" />
```

```
</LinearLayout>
```



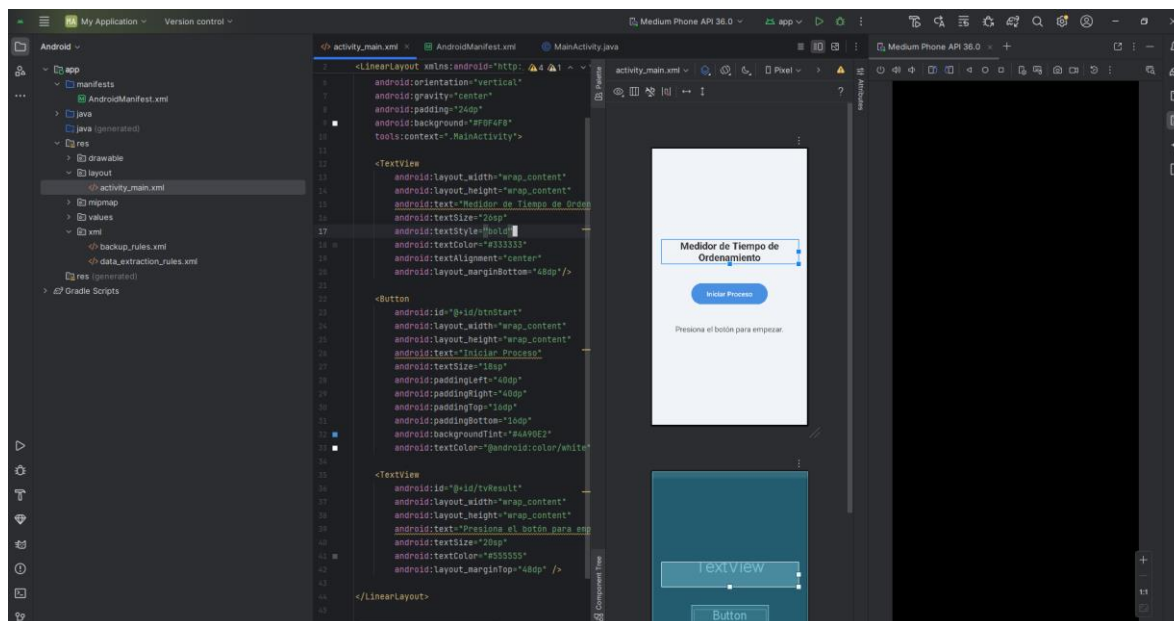
```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:gravity="center"
    android:padding="24dp"
    android:background="#F0F4F8"
    tools:context=".MainActivity">

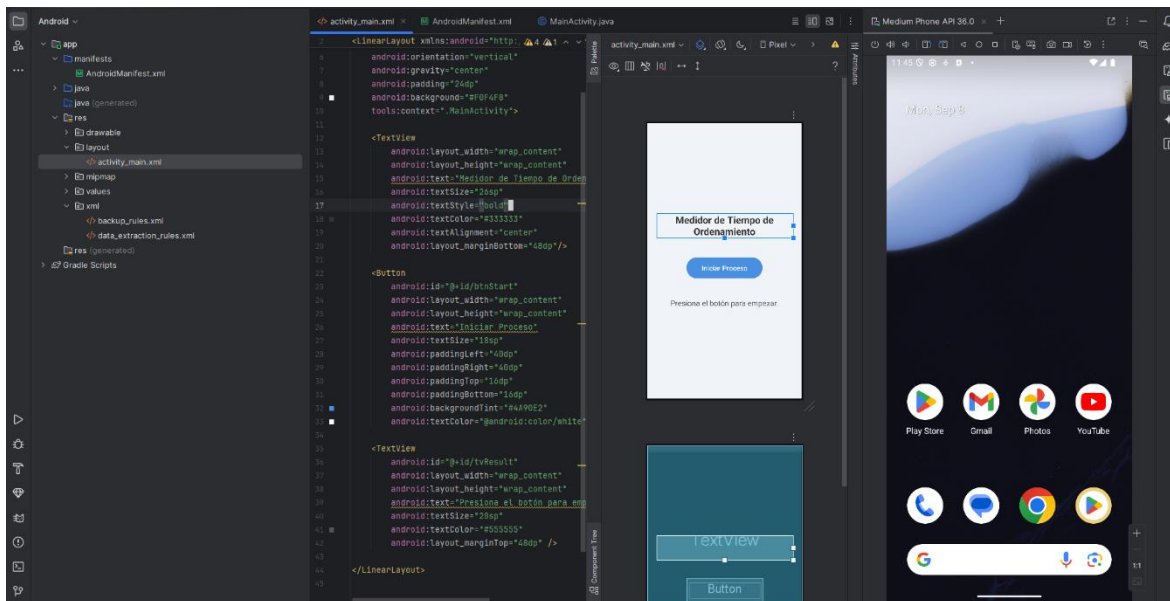
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Medidor de Tiempo de Ordenamiento"
        android:textSize="26sp"
        android:textStyle="bold"
        android:textColor="#333333"
        android:textAlignment="center"
        android:layout_marginBottom="48dp"/>

    <Button
        android:id="@+id/btnStart"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Iniciar Proceso"
        android:textSize="18sp"
        android:paddingLeft="40dp"
        android:paddingRight="40dp"
        android:paddingTop="16dp"
        android:paddingBottom="16dp"
        android:backgroundTint="#4A90E2"
        android:textColor="@android:color/white"/>

    <TextView
        android:id="@+id/tvResult"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Presiona el botón para empezar."
        android:textSize="20sp"
        android:textColor="#555555"
        android:layout_marginTop="48dp" />

</LinearLayout>
```





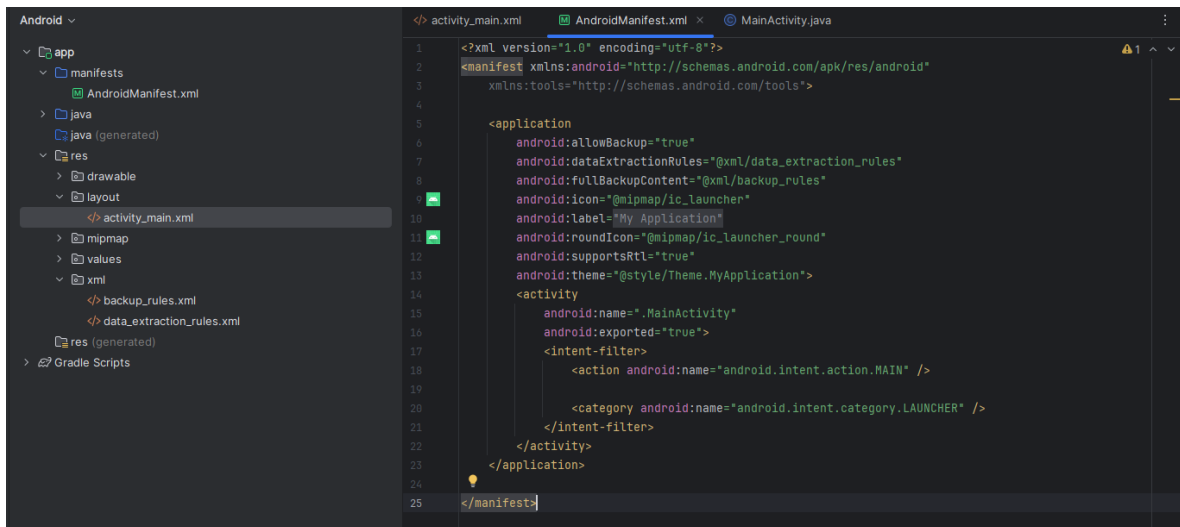
Android manifest XML:

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

    <application
        android:allowBackup="true"
        android:dataExtractionRules="@xml/data_extraction_rules"
        android:fullBackupContent="@xml/backup_rules"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportRtl="true"
        android:theme="@style/Theme.MyApplication">
        <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>
    </application>

</manifest>
```



Main Activity:

```
package com.example.myapplication;

import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;
import java.util.Arrays;
import java.util.Random;
import java.util.concurrent.ExecutorService;
import java.util.concurrent.Executors;

public class MainActivity extends AppCompatActivity {

    private TextView tvResult;
    private Button btnStart;

    private final ExecutorService executor =
        Executors.newSingleThreadExecutor();

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

        tvResult = findViewById(R.id.tvResult);
        btnStart = findViewById(R.id.btnStart);

        btnStart.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                startSortProcess();
            }
        });
    }
}
```

```

    }

    private void startSortProcess() {

        btnStart.setEnabled(false);
        tvResult.setText("Generando y ordenando 1 millón de
elementos...");

        executor.execute(() -> {

            final int ARRAY_SIZE = 1_000_000;
            int[] numbers = new int[ARRAY_SIZE];
            Random random = new Random();

            for (int i = 0; i < ARRAY_SIZE; i++) {
                numbers[i] = random.nextInt();
            }

            long startTime = System.nanoTime();

            Arrays.sort(numbers);

            long endTime = System.nanoTime();

            double durationInMillis = (endTime - startTime) /
1_000_000.0;
            final String resultText = String.format("Tiempo de
ordenamiento: %.2f ms", durationInMillis);

            runOnUiThread(() -> {
                tvResult.setText(resultText);

                btnStart.setEnabled(true);
            });
        });
    }

    @Override
    protected void onDestroy() {
        super.onDestroy();

        executor.shutdown();
    }
}

```

Ejecución:

