Ex. No.   :   01   **Date: 28/01/2025**

Register No.:   221701023   **Name: JANAKIRAMAN K**

GUI Components

Aim

 Develop an application to change the font and color of the text and display toast message when the user presses the button.

Procedure:

Step 1 : File -> NewProject

Provide the application name and Click "Next"

Step 2 : Select the target android devices

Select the minimum SDK to run the application. Click "Next".

Step 3 : Choose the activity for the application (By default choose "Blank Activity).

Click "Next".

Step 4 : Enter activity name and click.

Step 5 : Edit the program.

Step 6 : Run the application, 2-ways to run the application.

1. Running through emulator

2. Running through mobile device

AndroidManifest.xml

```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:supportsRtl="true"
        android:theme="@style/Theme._1exp23"
        tools:targetApi="31">
        <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>
```

Activity_main.xml

```xml
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout 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:id="@+id/main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:gravity="center"
    android:padding="16dp"
    tools:context=".MainActivity">
```

```xml
    <TextView
        android:id="@+id/tv1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Hello Kotlin"
        android:textAlignment="center"
        android:textSize="30sp" />
    <Button
        android:id="@+id/btn1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Text Size" />
    <Button
        android:id="@+id/btn2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Text Color" />
    <Button
        android:id="@+id/btn3"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="bg Color" />
</LinearLayout>
```

MainActivity.kt

```kotlin
package com.example.a1exp23

import android.annotation.SuppressLint
import android.graphics.Color
import android.os.Bundle
import android.widget.Button
import android.widget.TextView
import android.widget.LinearLayout
import androidx.activity.enableEdgeToEdge
import androidx.appcompat.app.AppCompatActivity
import androidx.core.view.ViewCompat

import androidx.core.view.WindowInsetsCompat
class MainActivity : AppCompatActivity() {
    @SuppressLint("MissingInflatedId")
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        enableEdgeToEdge()
        setContentView(R.layout.activity_main)
```
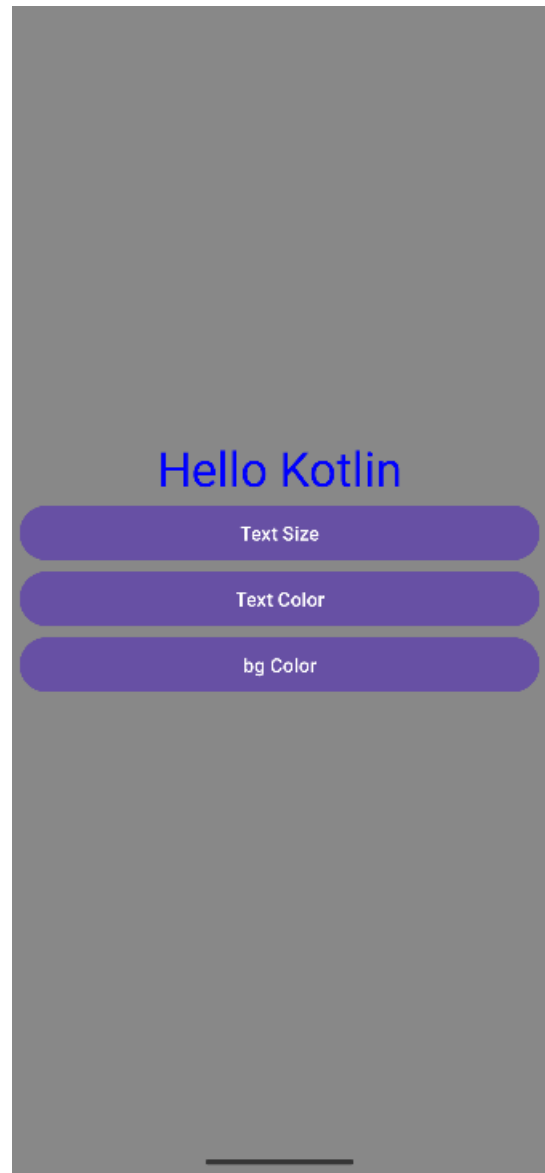
```kotlin
        var s = 0
        var c = 0
        val textSizes = arrayOf(25f, 30f, 35f)
        val colors = arrayOf(Color.RED, Color.BLUE, Color.GREEN)
        var isGray = true
        val tv1 : TextView = findViewById(R.id.tv1)
        val btn1 : Button = findViewById(R.id.btn1)
        val btn2 : Button = findViewById(R.id.btn2)
        val btn3 : Button = findViewById(R.id.btn3)
        val linearLayout: LinearLayout = findViewById(R.id.main)
        linearLayout.setBackgroundColor(Color.GRAY)
        btn1.setOnClickListener {
            tv1.textSize = textSizes[s]
            s = (s + 1) % textSizes.size
        }
        btn2.setOnClickListener {
            tv1.setTextColor(colors[c])
            c = (c + 1) % colors.size
        }
        btn3.setOnClickListener {
// Toggle background color
            if (isGray) {
                linearLayout.setBackgroundColor(Color.WHITE)
            }
            else {
                linearLayout.setBackgroundColor(Color.GRAY)
            }
            isGray = !isGray // Switch the state
        }
    }
}
```

Result:

The GUI components experiment has been successfully completed