

Practical: 2

AIM: TextView and its properties, Toast Message, Snackbar Message, Android in-built resources like color, Activity Life Cycle, Log Message in logcat, Properties of Layout ConstraintLayout, Generate id of TextView.

Submitted By: Tejasvi Patel
En.No: 21012022043



**Ganpat
University**
॥ विद्यया समाजोत्कर्षः ॥

**U.V. Patel
College of
Engineering**

**Department of Computer
Engineering/Information Technology**

Practical: 2

- Create Activity to display “Hello World” in TextView in the center of Activity screen with yellow (android:background="#FFFF00") layout background of Activity. TextView properties should be Holo Blue color (android:textColor=@android:color/holo_blue_bright), 27sp font size (android:textSize="27sp") and make text style bold and italic (android:textStyle="bold|italic"). After creating Activity, demonstrate functions of Activity Life Cycle by using Log message, Toast message & Snackbar message. All methods of Activity life cycle should be printed in Logcat.

Activity_main.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    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"
    android:background="#ffff00"
    android:id="@+id/mycl">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Hello World!"
        android:id="@+id/openActivity"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        android:textColor="@color/holo_blue_bright"
        android:textSize="27sp"
        android:textStyle="bold|italic"/>

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

MainActivity.kt

```
package com.example.madpractical2_21012022043

import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.util.Log
import android.widget.Toast
import androidx.constraintlayout.widget.ConstraintLayout
import com.google.android.material.snackbar.Snackbar

class MainActivity :
    AppCompatActivity() { private val
    TAG = "MainActivity"

    private lateinit var mycl:ConstraintLayout
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
        mycl=findViewById(R.id.mycl)
        Log.i(TAG, "onCreate: function called")
        Toast.makeText(this, "onCreate: function
        called", Toast.LENGTH_LONG).show()
        Snackbar.make(mycl, "onCreate: function called",
        Snackbar.LENGTH_LONG).show()
    }
    override fun onStart() {
        Log.i(TAG, "onStart: function called")
        Toast.makeText(this, "onStart: function
        called", Toast.LENGTH_LONG).show()
        Snackbar.make(mycl, "onStart: function called",
        Snackbar.LENGTH_LONG).show()
        super.onStart()
    }

    override fun
        onRestart() {
        super.onRestart()
        Log.i(TAG, "onRestart: function called")
        Toast.makeText(this, "onRestart: function
        called", Toast.LENGTH_LONG).show()
        Snackbar.make(mycl, "onRestart: function called",
```

```
        Snackbar.LENGTH_LONG).show()
    }

    override fun onResume() {
        super.onResume()
        Log.i(TAG, "onResume: function called")
        Toast.makeText(this, "onResume: function called", Toast.LENGTH_LONG).show()
        Snackbar.make(myCl, "onResume: function called", Snackbar.LENGTH_LONG).show()
    }

    override fun onPause() {
        super.onPause()
        Log.i(TAG, "onPause: function called")
        Toast.makeText(this, "onPause: function called", Toast.LENGTH_LONG).show()
        Snackbar.make(myCl, "onPause: function called", Snackbar.LENGTH_LONG).setDuration(10000)
    }

    override fun onStop() {
        super.onStop()
        Log.i(TAG, "onStop: function called")
        Toast.makeText(this, "onStop: function called", Toast.LENGTH_LONG).show()
        Snackbar.make(myCl, "onStop: function called", Snackbar.LENGTH_LONG).show()
    }

    override fun onDestroy() {
        super.onDestroy()
        Log.i(TAG, "onDestroy: function called")
        Toast.makeText(this, "onDestroy: function called", Toast.LENGTH_LONG).show()
        Snackbar.make(myCl, "onDestroy: function called", Snackbar.LENGTH_LONG).show()
    }
}
```

Practical: 2

Output:

