

# Practical: 2

**AIM:- Create Android Application to demonstrate functions of Activity Life Cycle and Basic UI.**

Submitted By: Manthan Sharma  
Enrollment number: 22012011104



**Ganpat  
University**  
॥ विद्यया समाजोत्कर्षः ॥

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

**Department of Computer  
Engineering/Information Technology**

## Practical: 2

---

### 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:id="@+id/main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#ffff00"
    tools:context=".MainActivity">

    <TextView
        android:id="@+id/textView2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Hello World"
        android:textColor="@android:color/holo_blue_light"
        android:textSize="27sp"
        android:textStyle="bold|italic"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <TextView
        android:id="@+id/textView1"
        android:layout_width="412dp"
        android:layout_height="48dp"
        android:background="#7131E3"
        android:text="MAD-Practical1"
        android:textSize="30dp"
        android:textStyle="bold"
        android:textColor="#FBFBFB"
        android:paddingTop="7dp"
        android:paddingLeft="40dp"
```

## Practical: 2

---

```
android:layout_marginTop="0dp"
app:layout_constraintBottom_toTopOf="@+id/textView2"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintEnd_toEndOf="parent"
```

```
app:layout_constraintHorizontal_bias="0.0"
app:layout_constraintVertical_bias="0.0"
/>
```

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

### MainActivity.kt

```
package com.example.madprac_2
```

```
import android.nfc.Tag
import android.os.Bundle
import android.util.Log
import androidx.activity.enableEdgeToEdge
import android.widget.Toast
import com.google.android.material.snackbar.Snackbar
import androidx.appcompat.app.AppCompatActivity
import androidx.core.view.ViewCompat
import androidx.core.view.WindowInsetsCompat
```

```
class MainActivity : AppCompatActivity() {
    val TAG = "MainActivity"
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        enableEdgeToEdge()
        setContentView(R.layout.activity_main)
        ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main)) { v, insets ->
            val systemBars = insets.getInsets(WindowInsetsCompat.Type.systemBars())
            v.setPadding(systemBars.left, systemBars.top, systemBars.right, systemBars.bottom)
            insets
        }
        showMessage("onCreate function has been called.")
    }
    fun showMessage(msg: String){
        Log.i(TAG,msg)
        Toast.makeText(this, msg, Toast.LENGTH_SHORT).show()
    }
}
```

## Practical: 2

---

```
}  
fun showMessage2(msg: String){  
    Log.i(TAG,msg)  
    Snackbar.make(findViewById(R.id.main), msg, Snackbar.LENGTH_SHORT).show()  
}  
  
override fun onStart() {  
    super.onStart()  
    showMessage2("onStart function has been called.")  
}  
override fun onPause() {  
    super.onPause()  
    showMessage2("onPause function has been called.")  
}  
override fun onRestart() {  
    super.onRestart()  
    showMessage2("onRestart function has been called")  
}  
  
override fun onDestroy() {  
    super.onDestroy()  
    showMessage2("onDestroy function has been called")  
}  
  
override fun onResume() {  
    super.onResume()  
    showMessage2("onResume function has been called")  
}  
  
override fun onStop() {  
    super.onStop()  
    showMessage2("onStop function has been called")  
}  
}
```

## Practical: 2

---

Output:

