# Experiment 10

Name:Hayden Cordeiro BECOMP RollNo:05

**10.1 AIM:**

Develop an application that uses GUI components.

**10.2 THEORY:**

In android **ui** or **input** controls are the interactive or View components which are used to design the user interface of an application. In android we have a wide variety of UI or input controls available, those are TextView, EditText, Buttons, Checkbox, Progressbar, Spinners, etc.

Generally, in android the user interface of an app is made with a collection of **View** and **ViewGroup** objects.

The **View** is a base class for all UI components in android and it is used to create an interactive UI components such as TextView, EditText, Checkbox, Radio Button, etc. and it responsible for event handling and drawing.

The **ViewGroup** is a subclass of **View** and it will act as a base class for layouts and layout parameters. The ViewGroup will provide an invisible containers to hold other Views or ViewGroups and to define the layout properties.

In android, we can define a UI or input controls in two ways, those are

- Declare UI elements in XML
- Create UI elements at runtime

**Android Different Types of UI Controls**

We have a different type of UI controls available in android to implement user interface for our android applications.

Following are the commonly used UI or input controls in android applications.

TextView

EditText

AutoCompleteTextView

Button

ImageButton

ToggleButton

CheckBox

RadioButton

RadioGroup

ProgressBar

Spinner

TimePicker

DatePicker

SeekBar

AlertDialog

Switch

RatingBar

In this experiment we will be implementing font and color changes of a text using GUI components.

### 10.3 ANDROID CODE:

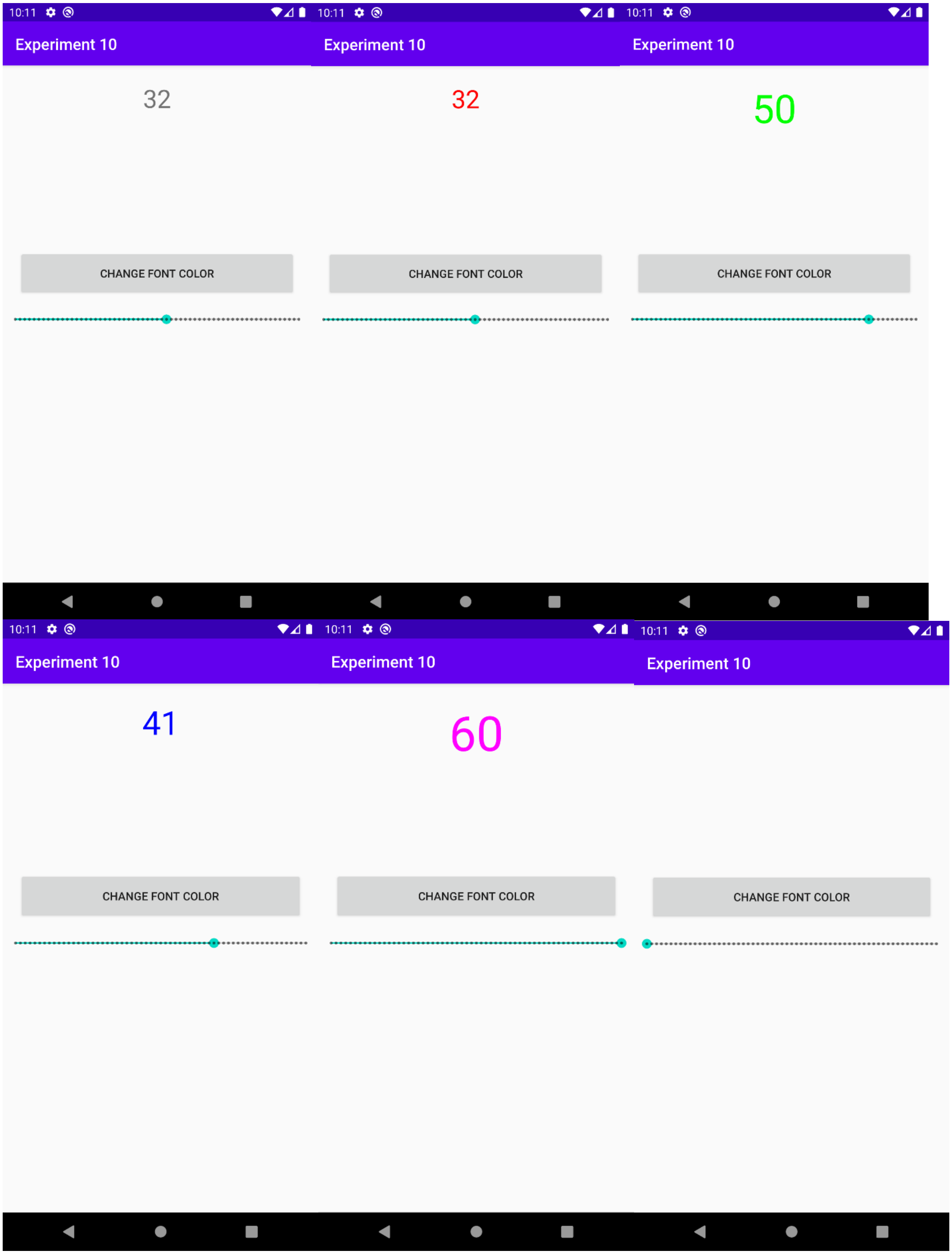
**activity_main.xml**

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.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"
    android:layout_gravity="center_horizontal"
    android:foregroundGravity="clip_horizontal"
    app:layout_anchorGravity="center_horizontal"
    tools:context=".MainActivity">

    <LinearLayout
        android:orientation="vertical"
        android:layout_width="match_parent"
android:layout_height="match_parent">

        <TextView
            android:id="@+id/textInp"
            android:layout_width="match_parent"
            android:layout_height="66dp"
            android:layout_margin="20dp"
            android:foregroundGravity="center"
            android:gravity="center_horizontal"
            android:textSize="34sp" />

        <Space
            android:layout_width="match_parent"
            android:layout_height="108dp" />

        <Button
            android:id="@+id/FontColor"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_margin="20dp"
            android:layout_marginStart="20dp"
            android:layout_marginLeft="20dp"
            android:layout_marginTop="0dp"
            android:layout_marginEnd="20dp"
            android:layout_marginRight="20dp"
            android:layout_marginBottom="0dp"
            android:padding="20dp"
            android:paddingLeft="20dp"
            android:paddingTop="10dp"
            android:paddingRight="20dp"
            android:paddingBottom="20dp"
            android:text="Change Font Color" />

        <SeekBar
            android:id="@+id/FontSize"
            style="@style/Widget.AppCompat.SeekBar.Discrete"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:max="10"
            android:progress="3" />
    </LinearLayout>


</android.support.constraint.ConstraintLayout>
```

### MainActivity.java

```
package com.example.experiment10;

import android.graphics.Color;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.SeekBar;
import android.widget.TextView;

public class MainActivity extends AppCompatActivity {
    int color=1;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        final TextView textEle = findViewById(R.id.textInp);

        final SeekBar FontSize = findViewById(R.id.FontSize);
        FontSize.setMax(60);
        final Button FontColor = findViewById(R.id.FontColor);
        FontColor.setOnClickListener(new View.OnClickListener() {
            public void onClick(View v) {
                switch (color) {
                    case 1:
                        textEle.setTextColor(Color.RED); break;
                    case 2:
                        textEle.setTextColor(Color.GREEN); break;
                    case 3:
                        textEle.setTextColor(Color.BLUE); break;
                    case 4:
                        textEle.setTextColor(Color.CYAN); break;
                    case 5:
                        textEle.setTextColor(Color.YELLOW); break;
                    case 6:
                        textEle.setTextColor(Color.MAGENTA); break;
                }
                color++;
                if(color>=7){
                    color=1;
                }

            }
        });



//       seekbar tracking
        FontSize.setOnSeekBarChangeListener(new
SeekBar.OnSeekBarChangeListener() {
            int progressChangedValue = 0;

            public void onProgressChanged(SeekBar seekBar, int progress,
boolean fromUser) {
                progressChangedValue = progress;
            }

            public void onStartTrackingTouch(SeekBar seekBar) {
                // TODO Auto-generated method stub
            }

            public void onStopTrackingTouch(SeekBar seekBar) {
                textEle.setText(String.valueOf(progressChangedValue));
                textEle.setTextSize(progressChangedValue);
            }
        });

    }

    }
```

## 10.4 OUTPUT:



## Conclusion

Thus using Android Studio the GUI components could be demonstrated and font size and color of the text could be changed.