

Slips 5

Q.1) Java Android Program to Demonstrate Alert Dialog Box.

Ans.

```
<?xml version="1.0" encoding="utf-8"?>

<RelativeLayout
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"    tools:context=".MainActivity">

    <Button        android:id="@+id/buttonShowDialog"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Show Dialog"
android:layout_centerInParent="true"
android:padding="16dp" />

</RelativeLayout>
```

MainActivity.java- package

```
com.example.myapplication;
```

```
import android.content.DialogInterface; import
android.os.Bundle;
```

```
import android.view.View; import android.widget.Button;
import androidx.appcompat.app.AlertDialog; import
androidx.appcompat.app.AppCompatActivity;
```

```

public class MainActivity extends AppCompatActivity {

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

        Button buttonShowDialog = findViewById(R.id.buttonShowDialog);
        buttonShowDialog.setOnClickListener(new
View.OnClickListener() {            @Override
        public void onClick(View v) {
showAlertDialog();
        }
        });
    }

    private void showAlertDialog() {
        AlertDialog.Builder builder = new AlertDialog.Builder(this);
builder.setTitle("Alert Dialog Example")

        .setMessage("This is a sample Alert Dialog!")
        .setPositiveButton("OK", new DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialog, int which)
{
                // Do something when OK button is clicked
            }
        })

        .setNegativeButton("Cancel", new
DialogInterface.OnClickListener() {
            @Override

```

```

        public void onClick(DialogInterface dialog, int which)
    {
        // Do something when Cancel button is clicked
    }
})
.setNeutralButton("Neutral", new
DialogInterface.OnClickListener() {
    @Override
    public void onClick(DialogInterface dialog, int which)
    {
        // Do something when Neutral button is clicked
    }
})
.show();
}
}

```

Q.2) Create an Android application which will ask the user to input his / her name. A message should display the two items concatenated in a label. Change the format of the label using radio buttons and check boxes for selection. The user can make the label text bold, underlined or italic as well as change its color. Also include buttons to display the message in the label, clear the text boxes as well as label. Finally exit.

Ans.

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
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"    tools:context=".MainActivity">
```

```
<EditText    android:id="@+id/editTextName"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:layout_margin="16dp"
android:hint="Enter your name" />
```

```
<Button    android:id="@+id/buttonDisplayMessage"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Display Message"
android:layout_below="@id/editTextName"
android:layout_marginTop="16dp"
android:layout_marginStart="16dp"
android:onClick="displayMessage" />
```

```
<Button    android:id="@+id/buttonClear"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Clear"
    android:layout_toEndOf="@id/buttonDisplayMessage"
android:layout_below="@id/editTextName"
android:layout_marginTop="16dp"    android:layout_marginStart="16dp"
android:onClick="clearFields" />
```

```
<TextView    android:id="@+id/textViewMessage"
android:layout_width="wrap_content"
```

```
android:layout_height="wrap_content"
android:layout_below="@id/buttonDisplayMessage"
android:layout_marginTop="16dp"
android:layout_marginStart="16dp"
    android:text=""    android:textSize="18sp"
/>
```

```
<RadioGroup
    android:id="@+id/radioGroupFormat"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@id/textViewMessage"
    android:layout_marginTop="16dp"    android:layout_marginStart="16dp"
    android:orientation="horizontal">
```

```
<RadioButton    android:id="@+id/radioButtonBold"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Bold" />
```

```
<RadioButton    android:id="@+id/radioButtonUnderline"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"    android:text="Underline"
/>
```

```
<RadioButton    android:id="@+id/radioButtonItalic"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Italic" />
```

</RadioGroup>

```
<CheckBox      android:id="@+id/checkBoxColor"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Change Color"
android:layout_below="@id/textViewMessage"
android:layout_toEndOf="@id/radioGroupFormat"
android:layout_marginTop="16dp"
android:layout_marginStart="16dp" />
```

```
<Button
    android:id="@+id/buttonApply"
    android:layout_width="120dp"      android:layout_height="59dp"
    android:layout_below="@id/textViewMessage"
    android:layout_marginStart="-110dp"
    android:layout_marginTop="81dp"
    android:layout_toEndOf="@id/checkBoxColor"
    android:onClick="applyChanges"    android:text="Apply" />
```

</RelativeLayout>

MainActivity.java- package

```
com.example.myapplication;
```

```
import android.graphics.Color;
import android.graphics.Typeface; import
android.os.Bundle; import
android.view.View; import
android.widget.CheckBox; import
```

```

android.widget.EditText; import
android.widget.RadioButton; import
android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {
    private EditText editTextName;    private TextView
    textViewMessage;

    private RadioButton radioButtonBold, radioButtonUnderline,
    radioButtonItalic;

    private CheckBox checkBoxColor;

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

        editTextName = findViewById(R.id.editTextName);
    textViewMessage = findViewById(R.id.textViewMessage);
    radioButtonBold = findViewById(R.id.radioButtonBold);
    radioButtonUnderline = findViewById(R.id.radioButtonUnderline);

        radioButtonItalic = findViewById(R.id.radioButtonItalic);
    checkBoxColor = findViewById(R.id.checkBoxColor);
    }

    public void displayMessage(View view) {
        String name = editTextName.getText().toString().trim();
    textViewMessage.setText(name);

```

```
}
```

```
    public void clearFields(View view) {        editTextName.setText("");
textViewMessage.setText("");        radioButtonBold.setChecked(false);
radioButtonUnderline.setChecked(false);
radioButtonItalic.setChecked(false);
checkBoxColor.setChecked(false);
textViewMessage.setTextColor(Color.BLACK);
textViewMessage.setTypeface(null, Typeface.NORMAL);
textViewMessage.setTypeface(null, Typeface.BOLD);
textViewMessage.setTypeface(null, Typeface.ITALIC);
textViewMessage.setTextColor(Color.GREEN);
    }
```

```
    public void applyChanges(View view) {
        int style = 0;

        if (radioButtonBold.isChecked()) {            style |=
android.graphics.Typeface.BOLD;
        }

        if (radioButtonUnderline.isChecked()) {

textViewMessage.setPaintFlags(textViewMessage.getPaintFlags() |
android.graphics.Paint.UNDERLINE_TEXT_FLAG);
        }

        if (radioButtonItalic.isChecked()) {            style |=
android.graphics.Typeface.ITALIC;
        }

        textViewMessage.setTypeface(null, style);

        if (checkBoxColor.isChecked()) {
```



```
        textViewMessage.setTextColor(Color.RED); // Change color to RED  
(You can change as per requirement)
```

```
    } else {
```

```
        textViewMessage.setTextColor(Color.BLACK); // Reset color to  
BLACK
```

```
    }
```

```
}
```

```
}
```