Slip 1

Q.1) Write an application to create a splash screen.
Solution:
// MainActivity.java

```
package com.example.splashscreen;
import android.content.Intent;
import android.os.Bundle;
import android.os.Handler;
import android.support.v7.app.AppCompatActivity;
public class MainActivity extends AppCompatActivity {
   // Splash screen timer
   private static int SPLASH_TIME_OUT = 3000; // 3 seconds

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

      // Using a handler to delay the intent to the next activity
      new Handler().postDelayed(new Runnable() {
         @Override
         public void run() {
            // This method will be executed once the timer is over
            // Start your app's main activity
            Intent intent = new Intent(MainActivity.this, HomeActivity.class);
            startActivity(intent);

            // Close this activity
            finish();
         }
      }, SPLASH_TIME_OUT);
   }
}


}
```

<!-- activity_main.xml -->

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
   android:layout_width="match_parent"
   android:layout_height="match_parent"
   android:background="@color/colorPrimaryDark">

   <ImageView
      android:id="@+id/splash_logo"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:layout_centerInParent="true"
      android:src="@drawable/ic_launcher_foreground" />

</RelativeLayout>
```

Q.2) Create table Student (roll_no, name, address, percentage). Create Application for performing the following operation on the table. (Using SQLite database). i] Insert record of 5 new student details. ii] Show all the student details.
Solution:
File Name: DatabaseHelper.java
Location: app/src/main/java/com/example/package/DatabaseHelper.java

Code:

```
import android.content.Context;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;

public class DatabaseHelper extends SQLiteOpenHelper {

    private static final String DATABASE_NAME = "student_database";
    private static final int DATABASE_VERSION = 1;

    // Table name and columns
    private static final String TABLE_STUDENT = "student";
    private static final String COLUMN_ROLL_NO = "roll_no";
    private static final String COLUMN_NAME = "name";
    private static final String COLUMN_ADDRESS = "address";
    private static final String COLUMN_PERCENTAGE = "percentage";

    // Create table query
    private static final String CREATE_TABLE_STUDENT = "CREATE TABLE " + TABLE_STUDENT + "("
            + COLUMN_ROLL_NO + " INTEGER PRIMARY KEY,"
            + COLUMN_NAME + " TEXT,"
            + COLUMN_ADDRESS + " TEXT,"
            + COLUMN_PERCENTAGE + " REAL"
            + ")";

    public DatabaseHelper(Context context) {
        super(context, DATABASE_NAME, null, DATABASE_VERSION);
    }

    @Override
    public void onCreate(SQLiteDatabase db) {
        db.execSQL(CREATE_TABLE_STUDENT);
    }

    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
        db.execSQL("DROP TABLE IF EXISTS " + TABLE_STUDENT);
        onCreate(db);
    }
}
```

File Name: Student.java
Location: app/src/main/java/com/example/package/Student.java
Code:

```
public class Student {

    private int rollNo;
    private String name;
    private String address;
    private double percentage;

    public Student(int rollNo, String name, String address, double percentage) {
        this.rollNo = rollNo;
        this.name = name;
        this.address = address;
        this.percentage = percentage;
    }

    // Getters and setters
}
```

Slip2

Q.1) Create an application that allows the user to enter a number in the textbox. Check whether the number in the textbox is perfect number or not. Print the message using Toast control.

Solution:

activity_main.xml:

```
<?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/editTextNumber"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="50dp"
        android:hint="Enter a number"
        android:inputType="number"/>

    <Button
        android:id="@+id/buttonCheck"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/editTextNumber"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="20dp"
        android:text="Check"/>
</RelativeLayout>
```

MainActivity.java:

```
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

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

```
        EditText editTextNumber = findViewById(R.id.editTextNumber);
        Button buttonCheck = findViewById(R.id.buttonCheck);

        buttonCheck.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String numberStr = editTextNumber.getText().toString();
                if (!numberStr.isEmpty()) {
                    int number = Integer.parseInt(numberStr);
                    if (isPerfectNumber(number)) {
                        showToast(number + " is a perfect number");
                    } else {
                        showToast(number + " is not a perfect number");
                    }
                } else {
                    showToast("Please enter a number");
                }
            }
        });
    }

    private boolean isPerfectNumber(int number) {
        int sum = 1;
        for (int i = 2; i <= Math.sqrt(number); i++) {
            if (number % i == 0) {
                sum += i;
                if (i != number / i) {
                    sum += number / i;
                }
            }
        }
        return sum == number;
    }

    private void showToast(String message) {
        Toast.makeText(this, message, Toast.LENGTH_SHORT).show();
    }
}
```

Q.2) Java Android Program to perform all arithmetic Operations using Calculator.

Solution:

activity_main.xml:

```
<?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">

    <TextView
        android:id="@+id/textViewResult"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="20dp"
        android:textSize="24sp"
        android:textAlignment="center"
        android:text="0"
        android:padding="10dp"/>

    <GridLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@id/textViewResult"
```

```
        android:rowCount="5"
        android:columnCount="4"
        android:padding="10dp">

        <!-- Buttons for numbers -->
        <Button
            android:id="@+id/button1"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_columnWeight="1"
            android:text="1"
            android:onClick="onNumberClick"/>

        <!-- Add buttons for numbers 2 to 9 -->

        <Button
            android:id="@+id/button0"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_columnWeight="1"
            android:text="0"
            android:onClick="onNumberClick"/>

        <!-- Buttons for arithmetic operations -->
        <Button
            android:id="@+id/buttonPlus"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_columnWeight="1"
            android:text="+"
            android:onClick="onOperationClick"/>

        <!-- Add buttons for subtraction, multiplication, division -->

        <Button
            android:id="@+id/buttonEquals"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_columnSpan="2"
            android:layout_columnWeight="2"
            android:text="="
            android:onClick="onEqualsClick"/>
    </GridLayout>
</RelativeLayout>
```

MainActivity.java:

```
import android.os.Bundle;
import android.view.View;
import android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

    private TextView textViewResult;
    private String input = "";
    private double operand1 = Double.NaN;
    private String operator = "";

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
```

```
        setContentView(R.layout.activity_main);

        textViewResult = findViewById(R.id.textViewResult);
    }

    public void onNumberClick(View view) {
        String number = ((TextView) view).getText().toString();
        input += number;
        textViewResult.setText(input);
    }

    public void onOperationClick(View view) {
        if (!input.isEmpty()) {
            operand1 = Double.parseDouble(input);
            operator = ((TextView) view).getText().toString();
            input = "";
        }
    }

    public void onEqualsClick(View view) {
        if (!input.isEmpty() && !Double.isNaN(operand1)) {
            double operand2 = Double.parseDouble(input);
            double result = performOperation(operand1, operand2, operator);
            textViewResult.setText(String.valueOf(result));
            input = "";
            operand1 = Double.NaN;
            operator = "";
        }
    }

    private double performOperation(double operand1, double operand2, String operator) {
        switch (operator) {
            case "+":
                return operand1 + operand2;
            case "-":
                return operand1 - operand2;
            case "*":
                return operand1 * operand2;
            case "/":
                if (operand2 != 0) {
                    return operand1 / operand2;
                } else {
                    return Double.NaN; // Division by zero
                }
            default:
                return Double.NaN; // Invalid operator
        }
    }
}
```

Slip3

Q.1) Create an application that allows the user to enter a number in the textbox. Check whether the number in the textbox is Armstrong or not. Print the message accordingly in the label control.

Solution:

activity_main.xml:

```
<?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/editTextNumber"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="50dp"
        android:hint="Enter a number"
        android:inputType="number"/>

    <Button
        android:id="@+id/buttonCheck"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/editTextNumber"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="20dp"
        android:text="Check"
        android:onClick="checkArmstrongNumber"/>

    <TextView
        android:id="@+id/textViewResult"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/buttonCheck"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="20dp"
        android:textSize="18sp"
        android:textColor="@android:color/black"/>
</RelativeLayout>
```

MainActivity.java:

```
import android.os.Bundle;
import android.view.View;
import android.widget.EditText;
import android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

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

```
    public void checkArmstrongNumber(View view) {
        EditText editTextNumber = findViewById(R.id.editTextNumber);
        TextView textViewResult = findViewById(R.id.textViewResult);

        String numberStr = editTextNumber.getText().toString();
        if (!numberStr.isEmpty()) {
            int number = Integer.parseInt(numberStr);
            if (isArmstrong(number)) {
                textViewResult.setText(number + " is an Armstrong number");
            } else {
                textViewResult.setText(number + " is not an Armstrong number");
            }
        } else {
            textViewResult.setText("Please enter a number");
        }
    }

    private boolean isArmstrong(int number) {
        int originalNumber, remainder, result = 0, n = 0;
        originalNumber = number;

        // Count digits
        while (originalNumber != 0) {
            originalNumber /= 10;
            ++n;
        }

        originalNumber = number;

        // Calculate result
        while (originalNumber != 0) {
            remainder = originalNumber % 10;
            result += Math.pow(remainder, n);
            originalNumber /= 10;
        }

        return result == number;
    }
}
```

Q.2) Create an Android application which examine a phone number entered by a user with the given format. • Area code should be one of the following: 040, 041, 050, 0400, 044 • There should 6 - 8 numbers in telephone number (+ area code).

Solution:

activity_main.xml:

```
<?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/editTextPhoneNumber"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="50dp"
        android:layout_marginStart="20dp"
        android:layout_marginEnd="20dp"
        android:hint="Enter phone number"
        android:inputType="phone"/>
```

```
    <Button
        android:id="@+id/buttonCheck"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/editTextPhoneNumber"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="20dp"
        android:text="Check"
        android:onClick="checkPhoneNumberFormat"/>

    <TextView
        android:id="@+id/textViewResult"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/buttonCheck"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="20dp"
        android:textSize="18sp"
        android:textColor="@android:color/black"/>
</RelativeLayout>
```

MainActivity.java:

```
import android.os.Bundle;
import android.text.TextUtils;
import android.view.View;
import android.widget.EditText;
import android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

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

    public void checkPhoneNumberFormat(View view) {
        EditText editTextPhoneNumber = findViewById(R.id.editTextPhoneNumber);
        TextView textViewResult = findViewById(R.id.textViewResult);

        String phoneNumber = editTextPhoneNumber.getText().toString().trim();

        if (isValidPhoneNumber(phoneNumber)) {
            textViewResult.setText("Phone number format is valid");
        } else {
            textViewResult.setText("Invalid phone number format");
        }
    }

    private boolean isValidPhoneNumber(String phoneNumber) {
        // Check if phone number starts with one of the area codes
        if (phoneNumber.startsWith("040") || phoneNumber.startsWith("041") ||
                phoneNumber.startsWith("050") || phoneNumber.startsWith("0400") ||
                phoneNumber.startsWith("044")) {
            // Remove area code and check if remaining length is between 6 and 8
            String numberWithoutAreaCode = phoneNumber.substring(3);
            return numberWithoutAreaCode.length() >= 6 && numberWithoutAreaCode.length() <= 8 &&
TextUtils.isDigitsOnly(numberWithoutAreaCode);
        }
        return false;
```

```
    }
}
```

Slip 4

Q.1) Construct image switcher using setFactory().

Solution:

activity_main.xml:

```
<?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">

    <ImageSwitcher
        android:id="@+id/imageSwitcher"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_centerInParent="true"/>

    <Button
        android:id="@+id/buttonPrevious"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentStart="true"
        android:layout_centerVertical="true"
        android:text="Previous"/>

    <Button
        android:id="@+id/buttonNext"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentEnd="true"
        android:layout_centerVertical="true"
        android:text="Next"/>
</RelativeLayout>
```

MainActivity.java:

```
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.ImageSwitcher;
import android.widget.ImageView;
import android.widget.ViewSwitcher;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

    private ImageSwitcher imageSwitcher;
    private int[] images = {R.drawable.image1, R.drawable.image2, R.drawable.image3};
    private int currentIndex = 0;

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

        imageSwitcher = findViewById(R.id.imageSwitcher);
        imageSwitcher.setFactory(new ViewSwitcher.ViewFactory() {
            @Override
            public View makeView() {
                ImageView imageView = new ImageView(getApplicationContext());
                imageView.setScaleType(ImageView.ScaleType.CENTER_CROP);
                imageView.setLayoutParams(new ImageSwitcher.LayoutParams(
                        ImageSwitcher.LayoutParams.MATCH_PARENT,
                        ImageSwitcher.LayoutParams.MATCH_PARENT));
                return imageView;
            }
        });

        // Set initial image
        imageSwitcher.setImageResource(images[currentIndex]);

        // Set click listeners for previous and next buttons
        Button buttonPrevious = findViewById(R.id.buttonPrevious);
        Button buttonNext = findViewById(R.id.buttonNext);

        buttonPrevious.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                if (currentIndex > 0) {
                    currentIndex--;
                    imageSwitcher.setImageResource(images[currentIndex]);
                }
            }
        });

        buttonNext.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                if (currentIndex < images.length - 1) {
                    currentIndex++;
                    imageSwitcher.setImageResource(images[currentIndex]);
                }
            }
        });
    }
}
```

Q.2) Write a program to search a specific location on Google Map.

Solution:

Obtain an API key for the Google Maps API from the Google Cloud Console (https://console.cloud.google.com/) add it to your project's **AndroidManifest.xml** file within the **<application>** tag:

Code:

```
<meta-data
   android:name="com.google.android.geo.API_KEY"
   android:value="API_KEY"/>
```

activity_main.xml:

```
<?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">

   <com.google.android.gms.maps.MapView
      android:id="@+id/mapView"
      android:layout_width="match_parent"
      android:layout_height="match_parent"
      android:layout_above="@id/editTextLocation"/>

   <EditText
      android:id="@+id/editTextLocation"
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      android:layout_alignParentBottom="true"
      android:hint="Enter location to search"
      android:inputType="text"
      android:imeOptions="actionSearch"/>
</RelativeLayout>
```

MainActivity.java:

```
import android.os.Bundle;
import android.view.KeyEvent;
import android.view.inputmethod.EditorInfo;
import android.widget.EditText;

import androidx.appcompat.app.AppCompatActivity;

import com.google.android.gms.maps.CameraUpdateFactory;
import com.google.android.gms.maps.GoogleMap;
import com.google.android.gms.maps.OnMapReadyCallback;
import com.google.android.gms.maps.SupportMapFragment;
import com.google.android.gms.maps.model.LatLng;
import com.google.android.gms.maps.model.MarkerOptions;

public class MainActivity extends AppCompatActivity implements OnMapReadyCallback {

   private GoogleMap googleMap;
   private EditText editTextLocation;

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

```
        // Initialize the map
        SupportMapFragment mapFragment = (SupportMapFragment) getSupportFragmentManager()
                .findFragmentById(R.id.mapView);
        mapFragment.getMapAsync(this);

        // Initialize EditText
        editTextLocation = findViewById(R.id.editTextLocation);
        editTextLocation.setOnEditorActionListener((v, actionId, event) -> {
            if (actionId == EditorInfo.IME_ACTION_SEARCH ||
                    (event.getAction() == KeyEvent.ACTION_DOWN &&
                            event.getKeyCode() == KeyEvent.KEYCODE_ENTER)) {
                searchLocation();
                return true;
            }
            return false;
        });
    }

    @Override
    public void onMapReady(GoogleMap map) {
        googleMap = map;
    }

    private void searchLocation() {
        String location = editTextLocation.getText().toString().trim();
        if (!location.isEmpty()) {
            // Perform geocoding to get the coordinates of the location
            // For simplicity, let's use a dummy location (New York City)
            LatLng latLng = new LatLng(40.7128, -74.0060);

            // Add marker to the map
            googleMap.clear();
            googleMap.addMarker(new MarkerOptions().position(latLng).title(location));

            // Move camera to the location
            googleMap.moveCamera(CameraUpdateFactory.newLatLngZoom(latLng, 12));
        }
    }
}
```

Slip 5

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

Code:

MainActivity.java:

```
import android.os.Bundle;
import android.view.View;

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);
    }
```

```
    public void showAlert(View view) {
        AlertDialog.Builder builder = new AlertDialog.Builder(this);
        builder.setTitle("Alert Dialog")
                .setMessage("This is an example of an AlertDialog.")
                .setPositiveButton("OK", (dialog, which) -> {
                    // Positive button clicked
                    dialog.dismiss(); // Dismiss the dialog
                })
                .setNegativeButton("Cancel", (dialog, which) -> {
                    // Negative button clicked
                    dialog.dismiss(); // Dismiss the dialog
                });
        AlertDialog alertDialog = builder.create();
        alertDialog.show();
    }
}
```

activity_main.xml:

```
<?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/buttonShowAlert"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Show Alert"
        android:layout_centerInParent="true"
        android:onClick="showAlert"/>
</RelativeLayout>
```

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.

Solution:

activity_main.xml:

```
<?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"
    android:padding="16dp"
    tools:context=".MainActivity">

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

    <RadioGroup
        android:id="@+id/radioGroupFormat"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal"
        android:layout_below="@id/editTextName"
```

```
        android:layout_marginBottom="16dp">

        <RadioButton
            android:id="@+id/radioButtonNormal"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Normal"/>

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

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

    <CheckBox
        android:id="@+id/checkBoxUnderline"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Underline"
        android:layout_below="@id/radioGroupFormat"/>

    <Button
        android:id="@+id/buttonDisplay"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Display Message"
        android:layout_below="@id/checkBoxUnderline"/>

    <Button
        android:id="@+id/buttonClear"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Clear"
        android:layout_below="@id/buttonDisplay"
        android:layout_marginTop="16dp"/>

    <Button
        android:id="@+id/buttonExit"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Exit"
        android:layout_below="@id/buttonClear"
        android:layout_marginTop="16dp"/>

    <TextView
        android:id="@+id/textViewMessage"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/buttonExit"
        android:layout_marginTop="24dp"/>
</RelativeLayout>
```

MainActivity.java:

```
import android.graphics.Color;
import android.os.Bundle;
```

```
import android.view.View;
import android.widget.CheckBox;
import android.widget.EditText;
import android.widget.RadioButton;
import android.widget.RadioGroup;
import android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

    private EditText editTextName;
    private RadioGroup radioGroupFormat;
    private CheckBox checkBoxUnderline;
    private TextView textViewMessage;

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

        editTextName = findViewById(R.id.editTextName);
        radioGroupFormat = findViewById(R.id.radioGroupFormat);
        checkBoxUnderline = findViewById(R.id.checkBoxUnderline);
        textViewMessage = findViewById(R.id.textViewMessage);
    }

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

        if (!name.isEmpty()) {
            String formattedText = name;

            // Apply selected formatting options
            int selectedRadioButtonId = radioGroupFormat.getCheckedRadioButtonId();
            RadioButton radioButton = findViewById(selectedRadioButtonId);
            if (radioButton != null) {
                String textStyle = radioButton.getText().toString();
                switch (textStyle) {
                    case "Bold":
                        formattedText = "<b>" + formattedText + "</b>";
                        break;
                    case "Italic":
                        formattedText = "<i>" + formattedText + "</i>";
                        break;
                }
            }

            if (checkBoxUnderline.isChecked()) {
                formattedText = "<u>" + formattedText + "</u>";
            }

            // Set formatted text to the TextView
            textViewMessage.setText(HtmlCompat.fromHtml(formattedText, HtmlCompat.FROM_HTML_MODE_LEGACY));
            textViewMessage.setTextColor(Color.BLUE); // Change text color
        } else {
            textViewMessage.setText("Please enter your name.");
        }
    }

    public void clearText(View view) {
        editTextName.getText().clear();
```

```
        radioGroupFormat.clearCheck();
        checkBoxUnderline.setChecked(false);
        textViewMessage.setText("");
    }

    public void exitApp(View view) {
        finish();
    }
}
```

Slip 6

Q.1) Java Android Program to demonstrate login form with validation.

Solution:

activity_main.xml:

```
<?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"
    android:padding="16dp"
    tools:context=".MainActivity">

    <EditText
        android:id="@+id/editTextUsername"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Username"
        android:layout_marginBottom="16dp"/>

    <EditText
        android:id="@+id/editTextPassword"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Password"
        android:layout_below="@id/editTextUsername"
        android:layout_marginBottom="16dp"
        android:inputType="textPassword"/>

    <Button
        android:id="@+id/buttonLogin"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Login"
        android:layout_below="@id/editTextPassword"
        android:layout_marginBottom="16dp"
        android:onClick="login"/>

    <TextView
        android:id="@+id/textViewError"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:textColor="@android:color/red"
        android:layout_below="@id/buttonLogin"/>
</RelativeLayout>
```

MainActivity.java:

```
import android.os.Bundle;
import android.text.TextUtils;
import android.view.View;
import android.widget.EditText;
import android.widget.TextView;
```

```
import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

    private EditText editTextUsername;
    private EditText editTextPassword;
    private TextView textViewError;

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

        editTextUsername = findViewById(R.id.editTextUsername);
        editTextPassword = findViewById(R.id.editTextPassword);
        textViewError = findViewById(R.id.textViewError);
    }

    public void login(View view) {
        String username = editTextUsername.getText().toString().trim();
        String password = editTextPassword.getText().toString().trim();

        // Reset error messages
        textViewError.setText("");

        // Validate username and password
        if (TextUtils.isEmpty(username)) {
            textViewError.setText("Username cannot be empty");
            return;
        }

        if (TextUtils.isEmpty(password)) {
            textViewError.setText("Password cannot be empty");
            return;
        }

        // Perform login operation (replace with your actual login logic)
        if (username.equals("admin") && password.equals("password")) {
            // Successful login
            textViewError.setText("Login successful");
        } else {
            // Invalid credentials
            textViewError.setText("Invalid username or password");
        }
    }
}
```

Q.2) Write a program to search a specific location on Google Map.

Solution:

activity_main.xml:

```
<?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/editTextLocation"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
```

```
        android:layout_marginTop="16dp"
        android:hint="Enter location to search"
        android:inputType="text"/>

    <Button
        android:id="@+id/buttonSearch"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/editTextLocation"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="16dp"
        android:text="Search"
        android:onClick="searchLocation"/>

    <com.google.android.gms.maps.MapView
        android:id="@+id/mapView"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_below="@id/buttonSearch"
        android:layout_marginTop="16dp"/>
</RelativeLayout>
```

MainActivity.java:

```
import android.os.Bundle;
import android.view.View;
import android.widget.EditText;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

import com.google.android.gms.maps.CameraUpdateFactory;
import com.google.android.gms.maps.GoogleMap;
import com.google.android.gms.maps.OnMapReadyCallback;
import com.google.android.gms.maps.SupportMapFragment;
import com.google.android.gms.maps.model.LatLng;
import com.google.android.gms.maps.model.MarkerOptions;

public class MainActivity extends AppCompatActivity {

    private GoogleMap googleMap;
    private EditText editTextLocation;

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

        editTextLocation = findViewById(R.id.editTextLocation);

        // Initialize the map
        SupportMapFragment mapFragment = (SupportMapFragment) getSupportFragmentManager()
                .findFragmentById(R.id.mapView);
        mapFragment.getMapAsync(new OnMapReadyCallback() {
            @Override
            public void onMapReady(GoogleMap map) {
                googleMap = map;
            }
        });
    }

    public void searchLocation(View view) {
        String location = editTextLocation.getText().toString().trim();
```

```
        if (!location.isEmpty()) {
            // Perform geocoding to get the coordinates of the location
            // For simplicity, let's use a dummy location (New York City)
            LatLng latLng = new LatLng(40.7128, -74.0060);

            // Add marker to the map
            googleMap.clear();
            googleMap.addMarker(new MarkerOptions().position(latLng).title(location));

            // Move camera to the location
            googleMap.moveCamera(CameraUpdateFactory.newLatLngZoom(latLng, 12));
        } else {
            Toast.makeText(this, "Please enter a location to search", Toast.LENGTH_SHORT).show();
        }
    }
}
```

Slip 7

Q.1] Java Android Program to Demonstrate ProgressBar.

Solution:

activity_main.xml:

```
<?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"
    android:padding="16dp"
    tools:context=".MainActivity">

    <Button
        android:id="@+id/startButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Start Progress"
        android:layout_centerHorizontal="true"
        android:layout_centerVertical="true"
        android:onClick="startProgress"/>

    <ProgressBar
        android:id="@+id/progressBar"
        style="?android:attr/progressBarStyleHorizontal"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@id/startButton"
        android:layout_marginTop="16dp"
        android:max="100"/>

</RelativeLayout>
```

MainActivity.java:

```
import android.os.Bundle;
import android.os.Handler;
import android.view.View;
import android.widget.ProgressBar;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {
```

```
    private ProgressBar progressBar;
    private int progressStatus = 0;
    private Handler handler = new Handler();

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

        progressBar = findViewById(R.id.progressBar);
    }

    public void startProgress(View view) {
        progressStatus = 0;
        new Thread(new Runnable() {
            public void run() {
                while (progressStatus < 100) {
                    progressStatus += 1;
                    handler.post(new Runnable() {
                        public void run() {
                            progressBar.setProgress(progressStatus);
                        }
                    });
                    try {
                        // Sleep for 50 milliseconds.
                        Thread.sleep(50);
                    } catch (InterruptedException e) {
                        e.printStackTrace();
                    }
                }
            }
        }).start();
    }
}
```

Q.2] Create table Employee (E_id, name, address, ph_no). Create Application for performing the following operation on the table. (Using SQLite database). i] Insert record of 5 new Employees. ii] Show all the details of Employee.

Solution:

DBHelper.java:

```
import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;

public class DBHelper extends SQLiteOpenHelper {

    private static final String DATABASE_NAME = "EmployeeDB";
    private static final int DATABASE_VERSION = 1;
    private static final String TABLE_NAME = "Employee";
    private static final String COLUMN_ID = "E_id";
    private static final String COLUMN_NAME = "name";
    private static final String COLUMN_ADDRESS = "address";
    private static final String COLUMN_PH_NO = "ph_no";

    public DBHelper(Context context) {
        super(context, DATABASE_NAME, null, DATABASE_VERSION);
    }

    @Override
    public void onCreate(SQLiteDatabase db) {
```

```
        String CREATE_TABLE_QUERY = "CREATE TABLE " + TABLE_NAME + "("
                + COLUMN_ID + " INTEGER PRIMARY KEY AUTOINCREMENT,"
                + COLUMN_NAME + " TEXT,"
                + COLUMN_ADDRESS + " TEXT,"
                + COLUMN_PH_NO + " TEXT" + ")";
        db.execSQL(CREATE_TABLE_QUERY);
    }

    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
        db.execSQL("DROP TABLE IF EXISTS " + TABLE_NAME);
        onCreate(db);
    }

    public long insertEmployee(String name, String address, String phone) {
        SQLiteDatabase db = this.getWritableDatabase();
        ContentValues values = new ContentValues();
        values.put(COLUMN_NAME, name);
        values.put(COLUMN_ADDRESS, address);
        values.put(COLUMN_PH_NO, phone);
        long id = db.insert(TABLE_NAME, null, values);
        db.close();
        return id;
    }

    public Cursor getAllEmployees() {
        SQLiteDatabase db = this.getReadableDatabase();
        return db.rawQuery("SELECT * FROM " + TABLE_NAME, null);
    }
}
```

MainActivity.java:

```
import android.database.Cursor;
import android.os.Bundle;
import android.view.View;
import android.widget.TextView;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

    private DBHelper dbHelper;
    private TextView textView;

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

        dbHelper = new DBHelper(this);
        textView = findViewById(R.id.textView);
    }

    public void insertEmployees(View view) {
        long id;
        for (int i = 1; i <= 5; i++) {
            id = dbHelper.insertEmployee("Employee " + i, "Address " + i, "123456789" + i);
            if (id != -1) {
                Toast.makeText(this, "Employee " + i + " inserted successfully", Toast.LENGTH_SHORT).show();
            } else {
                Toast.makeText(this, "Error inserting Employee " + i, Toast.LENGTH_SHORT).show();
            }
```

```
        }
    }

    public void showEmployees(View view) {
        StringBuilder stringBuilder = new StringBuilder();
        Cursor cursor = dbHelper.getAllEmployees();
        if (cursor.moveToFirst()) {
            do {
                stringBuilder.append("ID: ").append(cursor.getInt(0)).append("\n");
                stringBuilder.append("Name: ").append(cursor.getString(1)).append("\n");
                stringBuilder.append("Address: ").append(cursor.getString(2)).append("\n");
                stringBuilder.append("Phone: ").append(cursor.getString(3)).append("\n\n");
            } while (cursor.moveToNext());
        }
        cursor.close();
        textView.setText(stringBuilder.toString());
    }
}
```

activity_main.xml:

```
<?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"
    android:padding="16dp"
    tools:context=".MainActivity">

    <Button
        android:id="@+id/insertButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Insert Employees"
        android:layout_centerHorizontal="true"
        android:onClick="insertEmployees"/>

    <Button
        android:id="@+id/showButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Show Employees"
        android:layout_below="@id/insertButton"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="16dp"
        android:onClick="showEmployees"/>

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/showButton"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="16dp"/>

</RelativeLayout>
```

AndroidManifest.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.employeeapp">

    <application
```

```
    android:allowBackup="true"
    android:icon="@mipmap/ic_launcher"
    android:label="@string/app_name"
    android:roundIcon="@mipmap/ic_launcher_round"
    android:supportsRtl="true"
    android:theme="@style/AppTheme">
    <activity android:name=".MainActivity">
      <intent-filter>
        <action android:name="android.intent.action.MAIN" />

        <category android:name="android.intent.category.LAUNCHER" />
      </intent-filter>
    </activity>
  </application>

</manifest>
```

Slip 8

Q.1] Create a Application which shows Life Cycle of Activity.

Solution:

MainActivity.java:

```
import android.os.Bundle;
import android.util.Log;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

  private static final String TAG = "MainActivity";

  @Override
  protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    Log.d(TAG, "onCreate: Activity created");
  }

  @Override
  protected void onStart() {
    super.onStart();
    Log.d(TAG, "onStart: Activity started");
  }

  @Override
  protected void onResume() {
    super.onResume();
    Log.d(TAG, "onResume: Activity resumed");
  }

  @Override
  protected void onPause() {
    super.onPause();
    Log.d(TAG, "onPause: Activity paused");
  }

  @Override
  protected void onStop() {
    super.onStop();
    Log.d(TAG, "onStop: Activity stopped");
  }
```

```
    @Override
    protected void onDestroy() {
        super.onDestroy();
        Log.d(TAG, "onDestroy: Activity destroyed");
    }

    @Override
    protected void onRestart() {
        super.onRestart();
        Log.d(TAG, "onRestart: Activity restarted");
    }
}
```

activity_main.xml:

```
<?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"
    android:padding="16dp"
    tools:context=".MainActivity">

    <!-- Add any UI components you want to display in the activity here -->

</RelativeLayout>
```

Q.2] Create table Customer (id, name, address, ph_no). Create Application for performing the following operation on the table. (Using SQLite database). i] Insert new customer details (At least 5 records). ii] Show all the customer details.

Solution:

Customer.java:

```
public class Customer {

    private int id;
    private String name;
    private String address;
    private String phoneNumber;

    public Customer() {
    }

    public Customer(String name, String address, String phoneNumber) {
        this.name = name;
        this.address = address;
        this.phoneNumber = phoneNumber;
    }

    public int getId() {
        return id;
    }

    public void setId(int id) {
        this.id = id;
    }

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }
```

```
    public String getAddress() {
        return address;
    }

    public void setAddress(String address) {
        this.address = address;
    }

    public String getPhoneNumber() {
        return phoneNumber;
    }

    public void setPhoneNumber(String phoneNumber) {
        this.phoneNumber = phoneNumber;
    }
}
```

DBHelper.java:

```
import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;

import java.util.ArrayList;
import java.util.List;

public class DBHelper extends SQLiteOpenHelper {

    private static final String DATABASE_NAME = "CustomerDB";
    private static final int DATABASE_VERSION = 1;
    private static final String TABLE_NAME = "Customer";
    private static final String COLUMN_ID = "id";
    private static final String COLUMN_NAME = "name";
    private static final String COLUMN_ADDRESS = "address";
    private static final String COLUMN_PH_NO = "ph_no";

    public DBHelper(Context context) {
        super(context, DATABASE_NAME, null, DATABASE_VERSION);
    }

    @Override
    public void onCreate(SQLiteDatabase db) {
        String CREATE_TABLE_QUERY = "CREATE TABLE " + TABLE_NAME + "("
                + COLUMN_ID + " INTEGER PRIMARY KEY AUTOINCREMENT,"
                + COLUMN_NAME + " TEXT,"
                + COLUMN_ADDRESS + " TEXT,"
                + COLUMN_PH_NO + " TEXT" + ")";
        db.execSQL(CREATE_TABLE_QUERY);
    }

    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
        db.execSQL("DROP TABLE IF EXISTS " + TABLE_NAME);
        onCreate(db);
    }

    public long insertCustomer(Customer customer) {
        SQLiteDatabase db = this.getWritableDatabase();
        ContentValues values = new ContentValues();
        values.put(COLUMN_NAME, customer.getName());
```

```
        values.put(COLUMN_ADDRESS, customer.getAddress());
        values.put(COLUMN_PH_NO, customer.getPhoneNumber());
        long id = db.insert(TABLE_NAME, null, values);
        db.close();
        return id;
    }

    public List<Customer> getAllCustomers() {
        List<Customer> customersList = new ArrayList<>();
        String selectQuery = "SELECT * FROM " + TABLE_NAME;
        SQLiteDatabase db = this.getWritableDatabase();
        Cursor cursor = db.rawQuery(selectQuery, null);
        if (cursor.moveToFirst()) {
            do {
                Customer customer = new Customer();
                customer.setId(cursor.getInt(cursor.getColumnIndex(COLUMN_ID)));
                customer.setName(cursor.getString(cursor.getColumnIndex(COLUMN_NAME)));
                customer.setAddress(cursor.getString(cursor.getColumnIndex(COLUMN_ADDRESS)));
                customer.setPhoneNumber(cursor.getString(cursor.getColumnIndex(COLUMN_PH_NO)));
                customersList.add(customer);
            } while (cursor.moveToNext());
        }
        cursor.close();
        return customersList;
    }
}
```

MainActivity.java:

```
import android.os.Bundle;
import android.view.View;
import android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;

import java.util.List;

public class MainActivity extends AppCompatActivity {

    private DBHelper dbHelper;
    private TextView resultTextView;

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

        dbHelper = new DBHelper(this);
        resultTextView = findViewById(R.id.resultTextView);
    }

    public void insertCustomers(View view) {
        for (int i = 1; i <= 5; i++) {
            Customer customer = new Customer("Customer " + i, "Address " + i, "123456789" + i);
            long id = dbHelper.insertCustomer(customer);
            if (id != -1) {
                displayMessage("Customer " + i + " inserted successfully");
            } else {
                displayMessage("Error inserting Customer " + i);
            }
        }
    }
```

```
    public void showCustomers(View view) {
        List<Customer> customersList = dbHelper.getAllCustomers();
        StringBuilder stringBuilder = new StringBuilder();
        for (Customer customer : customersList) {
            stringBuilder.append("ID: ").append(customer.getId()).append("\n");
            stringBuilder.append("Name: ").append(customer.getName()).append("\n");
            stringBuilder.append("Address: ").append(customer.getAddress()).append("\n");
            stringBuilder.append("Phone Number: ").append(customer.getPhoneNumber()).append("\n\n");
        }
        resultTextView.setText(stringBuilder.toString());
    }

    private void displayMessage(String message) {
        resultTextView.append(message + "\n");
    }
}
```

activity_main.xml:

```
<?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"
    android:padding="16dp"
    tools:context=".MainActivity">

    <Button
        android:id="@+id/insertButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Insert Customers"
        android:layout_centerHorizontal="true"
        android:onClick="insertCustomers"/>

    <Button
        android:id="@+id/showButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Show Customers"
        android:layout_below="@id/insertButton"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="16dp"
        android:onClick="showCustomers"/>

    <TextView
        android:id="@+id/resultTextView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/showButton"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="16dp"/>

</RelativeLayout>
```

AndroidManifest.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.customerapp">

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
```

```
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>
```

Slip 9

Q.1] Create an application that allows the user to enter a number in the textbox named „getnum". Check whether the number in the textbox „getnum" is Palindrome or not. Print the message accordingly in the label when the user clicks on the button „Check".

Solution:

activity_main.xml:

```
<?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"
    android:padding="16dp"
    tools:context=".MainActivity">

    <EditText
        android:id="@+id/getnum"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter a number"
        android:inputType="number"/>

    <Button
        android:id="@+id/checkButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Check"
        android:layout_below="@id/getnum"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="16dp"
        android:onClick="checkPalindrome"/>

    <TextView
        android:id="@+id/resultTextView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/checkButton"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="16dp"/>
</RelativeLayout>
```

MainActivity.java:

```
import android.os.Bundle;
import android.view.View;
import android.widget.EditText;
import android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;
```

```
public class MainActivity extends AppCompatActivity {

    private EditText getNumEditText;
    private TextView resultTextView;

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

        getNumEditText = findViewById(R.id.getnum);
        resultTextView = findViewById(R.id.resultTextView);
    }

    public void checkPalindrome(View view) {
        String numString = getNumEditText.getText().toString().trim();
        if (numString.isEmpty()) {
            resultTextView.setText("Please enter a number.");
            return;
        }

        int num = Integer.parseInt(numString);
        if (isPalindrome(num)) {
            resultTextView.setText(num + " is a palindrome.");
        } else {
            resultTextView.setText(num + " is not a palindrome.");
        }
    }

    private boolean isPalindrome(int num) {
        int originalNum = num;
        int reverseNum = 0;
        while (num != 0) {
            int remainder = num % 10;
            reverseNum = reverseNum * 10 + remainder;
            num /= 10;
        }
        return originalNum == reverseNum;
    }
}
```

Q.2] Java android program to create simple calculator.

Solution:

activity_main.xml:

```
<?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"
    android:padding="16dp"
    tools:context=".MainActivity">

    <EditText
        android:id="@+id/number1EditText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter number 1"
        android:inputType="numberDecimal"/>

    <EditText
        android:id="@+id/number2EditText"
```

```
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter number 2"
        android:layout_below="@id/number1EditText"
        android:layout_marginTop="16dp"
        android:inputType="numberDecimal"/>

    <Button
        android:id="@+id/addButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Add"
        android:layout_below="@id/number2EditText"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="16dp"
        android:onClick="performAddition"/>

    <Button
        android:id="@+id/subtractButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Subtract"
        android:layout_below="@id/addButton"
        android:layout_marginTop="16dp"
        android:layout_alignLeft="@id/addButton"
        android:onClick="performSubtraction"/>

    <Button
        android:id="@+id/multiplyButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Multiply"
        android:layout_below="@id/subtractButton"
        android:layout_marginTop="16dp"
        android:layout_alignLeft="@id/addButton"
        android:onClick="performMultiplication"/>

    <Button
        android:id="@+id/divideButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Divide"
        android:layout_below="@id/multiplyButton"
        android:layout_marginTop="16dp"
        android:layout_alignLeft="@id/addButton"
        android:onClick="performDivision"/>

    <TextView
        android:id="@+id/resultTextView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/divideButton"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="16dp"/>
</RelativeLayout>
```

MainActivity.java:

```
import android.os.Bundle;
import android.view.View;
import android.widget.EditText;
import android.widget.TextView;
```

```
import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

    private EditText number1EditText, number2EditText;
    private TextView resultTextView;

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

        number1EditText = findViewById(R.id.number1EditText);
        number2EditText = findViewById(R.id.number2EditText);
        resultTextView = findViewById(R.id.resultTextView);
    }

    public void performAddition(View view) {
        String num1Str = number1EditText.getText().toString();
        String num2Str = number2EditText.getText().toString();

        if (num1Str.isEmpty() || num2Str.isEmpty()) {
            resultTextView.setText("Please enter numbers.");
            return;
        }

        double num1 = Double.parseDouble(num1Str);
        double num2 = Double.parseDouble(num2Str);
        double result = num1 + num2;

        resultTextView.setText("Result: " + result);
    }

    public void performSubtraction(View view) {
        String num1Str = number1EditText.getText().toString();
        String num2Str = number2EditText.getText().toString();

        if (num1Str.isEmpty() || num2Str.isEmpty()) {
            resultTextView.setText("Please enter numbers.");
            return;
        }

        double num1 = Double.parseDouble(num1Str);
        double num2 = Double.parseDouble(num2Str);
        double result = num1 - num2;

        resultTextView.setText("Result: " + result);
    }

    public void performMultiplication(View view) {
        String num1Str = number1EditText.getText().toString();
        String num2Str = number2EditText.getText().toString();

        if (num1Str.isEmpty() || num2Str.isEmpty()) {
            resultTextView.setText("Please enter numbers.");
            return;
        }

        double num1 = Double.parseDouble(num1Str);
        double num2 = Double.parseDouble(num2Str);
        double result = num1 * num2;
```

```
        resultTextView.setText("Result: " + result);
    }

    public void performDivision(View view) {
        String num1Str = number1EditText.getText().toString();
        String num2Str = number2EditText.getText().toString();

        if (num1Str.isEmpty() || num2Str.isEmpty()) {
            resultTextView.setText("Please enter numbers.");
            return;
        }

        double num1 = Double.parseDouble(num1Str);
        double num2 = Double.parseDouble(num2Str);

        if (num2 == 0) {
            resultTextView.setText("Cannot divide by zero.");
            return;
        }

        double result = num1 / num2;
        resultTextView.setText("Result: " + result);
    }
}
```

Slip 10

Q.1] Create an application that allows the user to enter a number in the textbox named getnum. Check whether the number in the textbox getnum is Armstrong or not. Print the message using Toast control when the user clicks on the button Check.

Solution:

activity_main.xml:

```
<?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"
    android:padding="16dp"
    tools:context=".MainActivity">

    <EditText
        android:id="@+id/getnum"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter a number"
        android:inputType="number"/>

    <Button
        android:id="@+id/checkButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Check"
        android:layout_below="@id/getnum"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="16dp"
        android:onClick="checkArmstrongNumber"/>
</RelativeLayout>
```

MainActivity.java:

```
import android.os.Bundle;
import android.view.View;
import android.widget.EditText;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

    private EditText getNumEditText;

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

        getNumEditText = findViewById(R.id.getnum);
    }

    public void checkArmstrongNumber(View view) {
        String numString = getNumEditText.getText().toString().trim();
        if (numString.isEmpty()) {
            showToast("Please enter a number.");
            return;
        }

        int num = Integer.parseInt(numString);
        if (isArmstrongNumber(num)) {
            showToast(num + " is an Armstrong number.");
        } else {
            showToast(num + " is not an Armstrong number.");
        }
    }

    private boolean isArmstrongNumber(int num) {
        int originalNum = num;
        int result = 0;
        int power = String.valueOf(num).length();
        while (num != 0) {
            int digit = num % 10;
            result += Math.pow(digit, power);
            num /= 10;
        }
        return result == originalNum;
    }

    private void showToast(String message) {
        Toast.makeText(this, message, Toast.LENGTH_SHORT).show();
    }
}
```

Q.2] Write a program to draw GUI by using Spinner, Buttons.

Solution:

activity_main.xml:

```
<?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"
    android:padding="16dp"
```

```
    tools:context=".MainActivity">

    <Spinner
        android:id="@+id/spinner"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="16dp"/>

    <Button
        android:id="@+id/submitButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Submit"
        android:layout_below="@id/spinner"
        android:layout_marginTop="16dp"
        android:layout_centerHorizontal="true"
        android:onClick="onSubmitClicked"/>

    <Button
        android:id="@+id/resetButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Reset"
        android:layout_below="@id/submitButton"
        android:layout_marginTop="16dp"
        android:layout_centerHorizontal="true"
        android:onClick="onResetClicked"/>
</RelativeLayout>
```

MainActivity.java:

```
import android.os.Bundle;
import android.view.View;
import android.widget.AdapterView;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.Spinner;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

    private Spinner spinner;
    private Button submitButton, resetButton;

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

        spinner = findViewById(R.id.spinner);
        submitButton = findViewById(R.id.submitButton);
        resetButton = findViewById(R.id.resetButton);

        // Create an ArrayAdapter using the string array and a default spinner layout
        ArrayAdapter<CharSequence> adapter = ArrayAdapter.createFromResource(this,
                R.array.planets_array, android.R.layout.simple_spinner_item);
        // Specify the layout to use when the list of choices appears
        adapter.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);
        // Apply the adapter to the spinner
        spinner.setAdapter(adapter);
```

```
        // Set item click listener for spinner
        spinner.setOnItemSelectedListener(new AdapterView.OnItemSelectedListener() {
            @Override
            public void onItemSelected(AdapterView<?> parent, View view, int position, long id) {
                String selectedPlanet = parent.getItemAtPosition(position).toString();
                showToast("Selected Planet: " + selectedPlanet);
            }

            @Override
            public void onNothingSelected(AdapterView<?> parent) {
                // Do nothing
            }
        });
    }

    public void onSubmitClicked(View view) {
        showToast("Submit Button Clicked");
    }

    public void onResetClicked(View view) {
        showToast("Reset Button Clicked");
        spinner.setSelection(0); // Reset spinner selection
    }

    private void showToast(String message) {
        Toast.makeText(this, message, Toast.LENGTH_SHORT).show();
    }
}
```

Slip 11

Q.1] Create an Android Application to accept two numbers to calculate its Power and Average. Create two buttons: Power and Average. Display the appropriate result on the next activity on Button click.

Solution:

activity_main.xml

```
<?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"
    android:padding="16dp"
    tools:context=".MainActivity">

    <EditText
        android:id="@+id/num1EditText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter number 1"
        android:inputType="numberDecimal"/>

    <EditText
        android:id="@+id/num2EditText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@id/num1EditText"
        android:layout_marginTop="16dp"
        android:hint="Enter number 2"
        android:inputType="numberDecimal"/>

    <Button
        android:id="@+id/powerButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
```

```
        android:text="Calculate Power"
        android:layout_below="@id/num2EditText"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="16dp"
        android:onClick="calculatePower"/>

    <Button
        android:id="@+id/averageButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Calculate Average"
        android:layout_below="@id/powerButton"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="16dp"
        android:onClick="calculateAverage"/>

</RelativeLayout>
```

activity_result.xml

```
<?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"
    android:padding="16dp"
    tools:context=".ResultActivity">

    <TextView
        android:id="@+id/powerResultTextView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Power Result"
        android:layout_centerInParent="true"
        android:textSize="18sp"/>

    <TextView
        android:id="@+id/averageResultTextView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Average Result"
        android:layout_below="@id/powerResultTextView"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="16dp"
        android:textSize="18sp"/>

</RelativeLayout>
```

ResultActivity.java

```
import android.os.Bundle;
import android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;

public class ResultActivity extends AppCompatActivity {

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

        TextView powerResultTextView = findViewById(R.id.powerResultTextView);
        TextView averageResultTextView = findViewById(R.id.averageResultTextView);
```

```
        // Retrieve data from intent
        double powerResult = getIntent().getDoubleExtra("powerResult", 0);
        double averageResult = getIntent().getDoubleExtra("averageResult", 0);

        // Set results in TextViews
        powerResultTextView.setText("Power Result: " + powerResult);
        averageResultTextView.setText("Average Result: " + averageResult);
    }
}
```

MainActivity.java

```
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.EditText;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

    private EditText num1EditText, num2EditText;

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

        num1EditText = findViewById(R.id.num1EditText);
        num2EditText = findViewById(R.id.num2EditText);
    }

    public void calculatePower(View view) {
        double num1 = Double.parseDouble(num1EditText.getText().toString());
        double num2 = Double.parseDouble(num2EditText.getText().toString());
        double powerResult = Math.pow(num1, num2);

        Intent intent = new Intent(this, ResultActivity.class);
        intent.putExtra("powerResult", powerResult);
        startActivity(intent);
    }

    public void calculateAverage(View view) {
        double num1 = Double.parseDouble(num1EditText.getText().toString());
        double num2 = Double.parseDouble(num2EditText.getText().toString());
        double averageResult = (num1 + num2) / 2;

        Intent intent = new Intent(this, ResultActivity.class);
        intent.putExtra("averageResult", averageResult);
        startActivity(intent);
    }
}
```

AndroidManifest.xml

```
<application
    android:allowBackup="true"
    android:icon="@mipmap/ic_launcher"
    android:label="@string/app_name"
    android:roundIcon="@mipmap/ic_launcher_round"
    android:supportsRtl="true"
    android:theme="@style/AppTheme">
    <activity android:name=".MainActivity">
```

```
        <intent-filter>
            <action android:name="android.intent.action.MAIN" />

            <category android:name="android.intent.category.LAUNCHER" />
        </intent-filter>
    </activity>
    <activity android:name=".ResultActivity"></activity>
</application>
```

Q.2] Create an Android Application to perform following string operation according to user selection of radio button

Solution:

activity_main.xml

```
<?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"
    android:padding="16dp"
    tools:context=".MainActivity">

    <EditText
        android:id="@+id/inputEditText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter a string"
        android:inputType="text"/>

    <RadioGroup
        android:id="@+id/radioGroup"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/inputEditText"
        android:layout_marginTop="16dp">

        <RadioButton
            android:id="@+id/uppercaseRadioButton"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Convert to Uppercase" />

        <RadioButton
            android:id="@+id/lowercaseRadioButton"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Convert to Lowercase" />

        <RadioButton
            android:id="@+id/reverseRadioButton"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Reverse String" />
    </RadioGroup>

    <Button
        android:id="@+id/submitButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Submit"
        android:layout_below="@id/radioGroup"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="16dp"
        android:onClick="performOperation"/>
```

```
</RelativeLayout>
```

MainActivity.java

```
import android.os.Bundle;
import android.view.View;
import android.widget.EditText;
import android.widget.RadioButton;
import android.widget.RadioGroup;
import android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

    private EditText inputEditText;
    private RadioGroup radioGroup;

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

        inputEditText = findViewById(R.id.inputEditText);
        radioGroup = findViewById(R.id.radioGroup);
    }

    public void performOperation(View view) {
        String inputString = inputEditText.getText().toString().trim();
        int selectedId = radioGroup.getCheckedRadioButtonId();

        if (inputString.isEmpty()) {
            showToast("Please enter a string.");
            return;
        }

        if (selectedId == -1) {
            showToast("Please select an operation.");
            return;
        }

        RadioButton radioButton = findViewById(selectedId);
        String operation = radioButton.getText().toString();
        String result;

        switch (operation) {
            case "Convert to Uppercase":
                result = inputString.toUpperCase();
                break;
            case "Convert to Lowercase":
                result = inputString.toLowerCase();
                break;
            case "Reverse String":
                result = new StringBuilder(inputString).reverse().toString();
                break;
            default:
                result = "";
        }

        // Start ResultActivity to display the result
        ResultActivity.start(this, result);
    }
```

```
    private void showToast(String message) {
        Toast.makeText(this, message, Toast.LENGTH_SHORT).show();
    }
}
```

ResultActivity.java

```
import android.content.Context;
import android.content.Intent;
import android.os.Bundle;
import android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;

public class ResultActivity extends AppCompatActivity {

    private static final String EXTRA_RESULT = "result";

    public static void start(Context context, String result) {
        Intent intent = new Intent(context, ResultActivity.class);
        intent.putExtra(EXTRA_RESULT, result);
        context.startActivity(intent);
    }

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

        TextView resultTextView = findViewById(R.id.resultTextView);

        String result = getIntent().getStringExtra(EXTRA_RESULT);
        resultTextView.setText(result);
    }
}
```

activity_result.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:padding="16dp">

    <TextView
        android:id="@+id/resultTextView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textSize="20sp"
        android:layout_centerInParent="true"/>

</RelativeLayout>
```

Slip 12

Q.1] Construct an Android app that toggles a light bulb ON and OFF when the user clicks on toggle button.

Solution:

activity_main.xml

```
<?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"
    android:padding="16dp"
    tools:context=".MainActivity">
```

```
    <ToggleButton
        android:id="@+id/lightToggleButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textOff="OFF"
        android:textOn="ON"
        android:layout_centerInParent="true"/>

</RelativeLayout>
```

MainActivity.java

```
import android.os.Bundle;
import android.widget.CompoundButton;
import android.widget.Toast;
import android.widget.ToggleButton;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

    private ToggleButton lightToggleButton;
    private boolean isLightOn = false;

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

        lightToggleButton = findViewById(R.id.lightToggleButton);

        // Set initial state of light bulb
        lightToggleButton.setChecked(isLightOn);

        // Set listener for toggle button
        lightToggleButton.setOnCheckedChangeListener(new CompoundButton.OnCheckedChangeListener() {
            @Override
            public void onCheckedChanged(CompoundButton buttonView, boolean isChecked) {
                if (isChecked) {
                    // Light bulb is ON
                    turnOnLight();
                } else {
                    // Light bulb is OFF
                    turnOffLight();
                }
            }
        });
    }

    private void turnOnLight() {
        // Simulate turning on the light bulb (You can add your own logic here)
        showToast("Light bulb is ON");
        isLightOn = true;
    }

    private void turnOffLight() {
        // Simulate turning off the light bulb (You can add your own logic here)
        showToast("Light bulb is OFF");
        isLightOn = false;
    }

    private void showToast(String message) {
        Toast.makeText(this, message, Toast.LENGTH_SHORT).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.

Solution:

activity_main.xml

```
<?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"
    android:padding="16dp"
    tools:context=".MainActivity">

    <EditText
        android:id="@+id/nameEditText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter your name"
        android:inputType="text"/>

    <RadioGroup
        android:id="@+id/formatRadioGroup"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@id/nameEditText"
        android:layout_marginTop="16dp">

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

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

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

    <CheckBox
        android:id="@+id/redCheckBox"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Red"
        android:layout_below="@id/formatRadioGroup"
        android:layout_marginTop="16dp"/>

    <CheckBox
        android:id="@+id/greenCheckBox"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
```

```
        android:text="Green"
        android:layout_below="@id/redCheckBox"
        android:layout_marginTop="8dp"/>

    <Button
        android:id="@+id/displayButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Display"
        android:layout_below="@id/greenCheckBox"
        android:layout_marginTop="16dp"
        android:onClick="displayMessage"/>

    <Button
        android:id="@+id/clearButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Clear"
        android:layout_toEndOf="@id/displayButton"
        android:layout_below="@id/greenCheckBox"
        android:layout_marginTop="16dp"
        android:layout_marginStart="16dp"
        android:onClick="clearText"/>

    <TextView
        android:id="@+id/messageTextView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textSize="18sp"
        android:textStyle="normal"
        android:textColor="@android:color/black"
        android:layout_below="@id/displayButton"
        android:layout_marginTop="16dp"/>

</RelativeLayout>
```

MainActivity.java

```
import android.graphics.Color;
import android.os.Bundle;
import android.text.Spannable;
import android.text.SpannableString;
import android.text.style.StyleSpan;
import android.text.style.UnderlineSpan;
import android.view.View;
import android.widget.CheckBox;
import android.widget.EditText;
import android.widget.RadioButton;
import android.widget.RadioGroup;
import android.widget.TextView;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

    private EditText nameEditText;
    private RadioGroup formatRadioGroup;
    private CheckBox redCheckBox, greenCheckBox;
    private TextView messageTextView;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
```

```
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        nameEditText = findViewById(R.id.nameEditText);
        formatRadioGroup = findViewById(R.id.formatRadioGroup);
        redCheckBox = findViewById(R.id.redCheckBox);
        greenCheckBox = findViewById(R.id.greenCheckBox);
        messageTextView = findViewById(R.id.messageTextView);
    }

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

        if (name.isEmpty()) {
            showToast("Please enter your name.");
            return;
        }

        // Get selected text style
        int style = 0;
        int checkedStyleId = formatRadioGroup.getCheckedRadioButtonId();
        if (checkedStyleId != -1) {
            RadioButton radioButton = findViewById(checkedStyleId);
            String styleText = radioButton.getText().toString();
            if (styleText.equals("Bold")) {
                style |= Typeface.BOLD;
            } else if (styleText.equals("Italic")) {
                style |= Typeface.ITALIC;
            }
        }

        // Get selected text color
        int color = Color.BLACK;
        if (redCheckBox.isChecked()) {
            color = Color.RED;
        } else if (greenCheckBox.isChecked()) {
            color = Color.GREEN;
        }

        // Create styled and colored text
        SpannableString spannableString = new SpannableString(name);
        spannableString.setSpan(new StyleSpan(style), 0, name.length(), Spannable.SPAN_EXCLUSIVE_EXCLUSIVE);
        spannableString.setSpan(new ForegroundColorSpan(color), 0, name.length(),
Spannable.SPAN_EXCLUSIVE_EXCLUSIVE);
        messageTextView.setText(spannableString);
    }

    public void clearText(View view) {
        nameEditText.setText("");
        messageTextView.setText("");
        formatRadioGroup.clearCheck();
        redCheckBox.setChecked(false);
        greenCheckBox.setChecked(false);
    }

    private void showToast(String message) {
        Toast.makeText(this, message, Toast.LENGTH_SHORT).show();
    }
}
```

Slip 13

Q.1] Java android program to demonstrate Registration form with validation.

Solution:

activity_registration.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <EditText
        android:id="@+id/usernameEditText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Username"
        android:inputType="text"/>

    <EditText
        android:id="@+id/emailEditText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@id/usernameEditText"
        android:layout_marginTop="16dp"
        android:hint="Email"
        android:inputType="textEmailAddress"/>

    <EditText
        android:id="@+id/passwordEditText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@id/emailEditText"
        android:layout_marginTop="16dp"
        android:hint="Password"
        android:inputType="textPassword"/>

    <Button
        android:id="@+id/registerButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/passwordEditText"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="24dp"
        android:text="Register"/>

</RelativeLayout>
```

RegistrationActivity.java

```
import android.os.Bundle;
import android.util.Patterns;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

    private EditText usernameEditText, emailEditText, passwordEditText;
    private Button registerButton;

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

```
        usernameEditText = findViewById(R.id.usernameEditText);
        emailEditText = findViewById(R.id.emailEditText);
        passwordEditText = findViewById(R.id.passwordEditText);
        registerButton = findViewById(R.id.registerButton);

        registerButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                register();
            }
        });
    }

    private void register() {
        String username = usernameEditText.getText().toString().trim();
        String email = emailEditText.getText().toString().trim();
        String password = passwordEditText.getText().toString().trim();

        if (username.isEmpty()) {
            usernameEditText.setError("Username is required");
            usernameEditText.requestFocus();
            return;
        }

        if (email.isEmpty()) {
            emailEditText.setError("Email is required");
            emailEditText.requestFocus();
            return;
        }

        if (!Patterns.EMAIL_ADDRESS.matcher(email).matches()) {
            emailEditText.setError("Enter a valid email address");
            emailEditText.requestFocus();
            return;
        }

        if (password.isEmpty()) {
            passwordEditText.setError("Password is required");
            passwordEditText.requestFocus();
            return;
        }

        if (password.length() < 6) {
            passwordEditText.setError("Password must be at least 6 characters");
            passwordEditText.requestFocus();
            return;
        }

        // Registration successful
        Toast.makeText(this, "Registration Successful", Toast.LENGTH_SHORT).show();
    }
}
```

Q.2] Write a Java Android Program to Demonstrate List View Activity with all operations Such as: Insert, Delete, Search

Solution:

activity_main.xml

```
<?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"
    android:padding="16dp"
    tools:context=".MainActivity">

    <EditText
        android:id="@+id/itemEditText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter item name"/>

    <Button
        android:id="@+id/addButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/itemEditText"
        android:layout_marginTop="16dp"
        android:text="Add"
        android:onClick="addItem"/>

    <EditText
        android:id="@+id/searchEditText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@id/addButton"
        android:layout_marginTop="16dp"
        android:hint="Search"/>

    <ListView
        android:id="@+id/listView"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@id/searchEditText"
        android:layout_marginTop="16dp"/>

</RelativeLayout>
```

ItemAdapter.java

```
import android.content.Context;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.ArrayAdapter;
import android.widget.TextView;

import java.util.ArrayList;

public class ItemAdapter extends ArrayAdapter<String> {

    private Context mContext;
    private int mResource;

    public ItemAdapter(Context context, int resource, ArrayList<String> items) {
        super(context, resource, items);
        mContext = context;
        mResource = resource;
    }

    @Override
    public View getView(int position, View convertView, ViewGroup parent) {
        String item = getItem(position);

        if (convertView == null) {
```

```
            LayoutInflater inflater = LayoutInflater.from(mContext);
            convertView = inflater.inflate(mResource, parent, false);
        }

        TextView textView = convertView.findViewById(R.id.textView);
        textView.setText(item);

        return convertView;
    }
}
```

MainActivity.java

```
import android.os.Bundle;
import android.text.Editable;
import android.text.TextWatcher;
import android.view.View;
import android.widget.AdapterView;
import android.widget.ArrayAdapter;
import android.widget.EditText;
import android.widget.ListView;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

import java.util.ArrayList;

public class MainActivity extends AppCompatActivity {

    private EditText itemEditText, searchEditText;
    private ListView listView;
    private ArrayList<String> itemList;
    private ArrayAdapter<String> adapter;

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

        itemEditText = findViewById(R.id.itemEditText);
        searchEditText = findViewById(R.id.searchEditText);
        listView = findViewById(R.id.listView);

        itemList = new ArrayList<>();
        adapter = new ItemAdapter(this, R.layout.item_layout, itemList);
        listView.setAdapter(adapter);

        // Add item to list view when "Add" button is clicked
        findViewById(R.id.addButton).setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                addItem();
            }
        });

        // Delete item from list view when item is clicked
        listView.setOnItemClickListener(new AdapterView.OnItemClickListener() {
            @Override
            public void onItemClick(AdapterView<?> parent, View view, int position, long id) {
                deleteItem(position);
            }
        });
```

```
        // Filter list view items based on search query
        searchEditText.addTextChangedListener(new TextWatcher() {
            @Override
            public void beforeTextChanged(CharSequence s, int start, int count, int after) {}

            @Override
            public void onTextChanged(CharSequence s, int start, int before, int count) {
                filterItems(s.toString());
            }

            @Override
            public void afterTextChanged(Editable s) {}
        });
    }

    private void addItem() {
        String item = itemEditText.getText().toString().trim();
        if (!item.isEmpty()) {
            itemList.add(item);
            adapter.notifyDataSetChanged();
            itemEditText.setText("");
        } else {
            showToast("Please enter an item.");
        }
    }

    private void deleteItem(int position) {
        itemList.remove(position);
        adapter.notifyDataSetChanged();
    }

    private void filterItems(String query) {
        ArrayList<String> filteredList = new ArrayList<>();
        for (String item : itemList) {
            if (item.toLowerCase().contains(query.toLowerCase())) {
                filteredList.add(item);
            }
        }
        adapter = new ItemAdapter(this, R.layout.item_layout, filteredList);
        listView.setAdapter(adapter);
    }

    private void showToast(String message) {
        Toast.makeText(this, message, Toast.LENGTH_SHORT).show();
    }
}
```

item_layout.xml

```
<?xml version="1.0" encoding="utf-8"?>
<TextView xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/textView"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:padding="16dp"
    android:textSize="18sp"/>
```

Slip 14

Q.1] Construct an Android application to accept a number and calculate and display Factorial of a given number in TextView.

Solution:

activity_main.xml

```
<?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"
    android:padding="16dp"
    tools:context=".MainActivity">

    <EditText
        android:id="@+id/numberEditText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter a number"
        android:inputType="number"/>

    <Button
        android:id="@+id/calculateButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/numberEditText"
        android:layout_marginTop="16dp"
        android:text="Calculate"
        android:onClick="calculateFactorial"/>

    <TextView
        android:id="@+id/resultTextView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/calculateButton"
        android:layout_marginTop="16dp"
        android:textSize="18sp"/>
</RelativeLayout>
```

MainActivity.java

```
import android.os.Bundle;
import android.view.View;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

    private EditText numberEditText;
    private TextView resultTextView;

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

        numberEditText = findViewById(R.id.numberEditText);
        resultTextView = findViewById(R.id.resultTextView);
    }

    public void calculateFactorial(View view) {
        String input = numberEditText.getText().toString().trim();

        if (input.isEmpty()) {
            showToast("Please enter a number.");
            return;
```

```
        }

        int number = Integer.parseInt(input);

        if (number < 0) {
            showToast("Please enter a non-negative number.");
            return;
        }

        long factorial = calculateFactorial(number);
        resultTextView.setText("Factorial of " + number + " is: " + factorial);
    }

    private long calculateFactorial(int number) {
        if (number == 0 || number == 1) {
            return 1;
        }

        long result = 1;
        for (int i = 2; i <= number; i++) {
            result *= i;
        }

        return result;
    }

    private void showToast(String message) {
        Toast.makeText(this, message, Toast.LENGTH_SHORT).show();
    }
}
```

Q.2] Create an Android application, which show Login Form. After clicking LOGIN button display the "Login Successful…" message if username and password is same else display "Invalid Login" message in Toast Control.

Solution:

activity_main.xml

```
<?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"
    android:padding="16dp"
    tools:context=".MainActivity">

    <EditText
        android:id="@+id/usernameEditText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Username"
        android:inputType="text"/>

    <EditText
        android:id="@+id/passwordEditText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@id/usernameEditText"
        android:layout_marginTop="16dp"
        android:hint="Password"
        android:inputType="textPassword"/>

    <Button
        android:id="@+id/loginButton"
        android:layout_width="wrap_content"
```

```
        android:layout_height="wrap_content"
        android:layout_below="@id/passwordEditText"
        android:layout_marginTop="24dp"
        android:text="Login"
        android:onClick="login"/>

</RelativeLayout>
```

MainActivity.java

```
package com.example.operation;

import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

    private EditText usernameEditText, passwordEditText;
    private Button loginButton;
    private TextView resultTextView;

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

        usernameEditText = findViewById(R.id.usernameEditText);
        passwordEditText = findViewById(R.id.passwordEditText);
        loginButton = findViewById(R.id.loginButton);
        resultTextView = findViewById(R.id.resultTextView);

        loginButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                loginUser();
            }
        });
    }

    private void loginUser() {
        String username = usernameEditText.getText().toString().trim();
        String password = passwordEditText.getText().toString().trim();

        // Check if username and password are correct
        if (username.equals("maheshkandekar") && password.equals("password")) {
            // Display login successful message
            Toast.makeText(MainActivity.this, "Login Successful", Toast.LENGTH_SHORT).show();
            resultTextView.setText("");
        } else {
            // Display invalid login message
            Toast.makeText(MainActivity.this, "Invalid Login", Toast.LENGTH_SHORT).show();
            resultTextView.setText("");
        }
    }
}
```

Q1] Construct an Android application to accept two numbers in two EditText, with four buttons as ADD, SUB, DIV and MULT and display Result using Toast Control.

Solution:

activity_main.xml

```
<?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"
    android:padding="16dp"
    tools:context=".MainActivity">

    <EditText
        android:id="@+id/number1EditText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter first number"
        android:inputType="numberDecimal"/>

    <EditText
        android:id="@+id/number2EditText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@id/number1EditText"
        android:layout_marginTop="16dp"
        android:hint="Enter second number"
        android:inputType="numberDecimal"/>

    <Button
        android:id="@+id/addButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/number2EditText"
        android:layout_marginTop="16dp"
        android:text="ADD"
        android:onClick="performAddition"/>

    <Button
        android:id="@+id/subtractButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/addButton"
        android:layout_marginTop="16dp"
        android:text="SUB"
        android:onClick="performSubtraction"/>

    <Button
        android:id="@+id/multiplyButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/subtractButton"
        android:layout_marginTop="16dp"
        android:text="MULT"
        android:onClick="performMultiplication"/>

    <Button
        android:id="@+id/divideButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/multiplyButton"
        android:layout_marginTop="16dp"
        android:text="DIV"
```

```
        android:onClick="performDivision"/>

</RelativeLayout>
```

MainActivity.java

```
import android.os.Bundle;
import android.view.View;
import android.widget.EditText;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

    private EditText number1EditText, number2EditText;

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

        number1EditText = findViewById(R.id.number1EditText);
        number2EditText = findViewById(R.id.number2EditText);
    }

    public void performAddition(View view) {
        performOperation("+");
    }

    public void performSubtraction(View view) {
        performOperation("-");
    }

    public void performMultiplication(View view) {
        performOperation("*");
    }

    public void performDivision(View view) {
        performOperation("/");
    }

    private void performOperation(String operator) {
        String number1Str = number1EditText.getText().toString().trim();
        String number2Str = number2EditText.getText().toString().trim();

        if (number1Str.isEmpty() || number2Str.isEmpty()) {
            showToast("Please enter both numbers.");
            return;
        }

        double number1 = Double.parseDouble(number1Str);
        double number2 = Double.parseDouble(number2Str);
        double result;

        switch (operator) {
            case "+":
                result = number1 + number2;
                showToast("Result of addition: " + result);
                break;
            case "-":
                result = number1 - number2;
                showToast("Result of subtraction: " + result);
```

```
            break;
        case "*":
            result = number1 * number2;
            showToast("Result of multiplication: " + result);
            break;
        case "/":
            if (number2 == 0) {
                showToast("Cannot divide by zero.");
            } else {
                result = number1 / number2;
                showToast("Result of division: " + result);
            }
            break;
        default:
            showToast("Invalid operator.");
            break;
    }
  }

  private void showToast(String message) {
    Toast.makeText(this, message, Toast.LENGTH_SHORT).show();
  }
}
```

Q2] Construct a bank app to display different menu like withdraw, deposit etc.

Solution:

activity_main.xml

```
<?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"
  android:padding="16dp"
  tools:context=".MainActivity">

  <Button
    android:id="@+id/withdrawButton"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Withdraw"
    android:layout_centerHorizontal="true"
    android:onClick="withdraw"/>

  <Button
    android:id="@+id/depositButton"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Deposit"
    android:layout_below="@id/withdrawButton"
    android:layout_marginTop="16dp"
    android:layout_centerHorizontal="true"
    android:onClick="deposit"/>

  <Button
    android:id="@+id/balanceButton"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Check Balance"
    android:layout_below="@id/depositButton"
    android:layout_marginTop="16dp"
    android:layout_centerHorizontal="true"
```

```
        android:onClick="checkBalance"/>

</RelativeLayout>
```

MainActivity.java

```
import android.os.Bundle;
import android.view.View;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

    private double balance = 1000; // Initial balance

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

    public void withdraw(View view) {
        if (balance >= 100) { // Minimum balance check
            balance -= 100; // Withdrawal amount
            showToast("Withdrawal Successful. Current Balance: " + balance);
        } else {
            showToast("Insufficient balance for withdrawal.");
        }
    }

    public void deposit(View view) {
        balance += 200; // Deposit amount
        showToast("Deposit Successful. Current Balance: " + balance);
    }

    public void checkBalance(View view) {
        showToast("Current Balance: " + balance);
    }

    private void showToast(String message) {
        Toast.makeText(this, message, Toast.LENGTH_SHORT).show();
    }
}
```

Slip 16

Q1] Create a Simple Android Application Which Send —Hello‖ message from one activity to another with help of Button (Use Intent).

Solution:

activity_main.xml

```
<?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"
    android:padding="16dp"
    tools:context=".MainActivity">
```

```
    <Button
        android:id="@+id/sendMessageButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Send Message"
        android:layout_centerInParent="true"
        android:onClick="sendMessage"/>

</RelativeLayout>
```

activity_display_message.xml

```
<?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"
    android:padding="16dp"
    tools:context=".DisplayMessageActivity">

    <TextView
        android:id="@+id/messageTextView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textSize="24sp"
        android:layout_centerInParent="true"/>

</RelativeLayout>
```

MainActivity.java

```
import android.content.Intent;
import android.os.Bundle;
import android.view.View;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

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

    public void sendMessage(View view) {
        Intent intent = new Intent(this, DisplayMessageActivity.class);
        intent.putExtra("message", "Hello"); // Send message as an extra
        startActivity(intent);
    }
}
```

DisplayMessageActivity.java

```
import android.content.Intent;
import android.os.Bundle;
import android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;

public class DisplayMessageActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
```

```
        setContentView(R.layout.activity_display_message);

        Intent intent = getIntent();
        String message = intent.getStringExtra("message"); // Receive message extra

        TextView messageTextView = findViewById(R.id.messageTextView);
        messageTextView.setText(message); // Set message in TextView
    }
}
```

AndroidManifest.xml

```
<activity android:name=".DisplayMessageActivity">
    <intent-filter>
        <action android:name="android.intent.action.MAIN" />
        <category android:name="android.intent.category.LAUNCHER" />
    </intent-filter>
</activity>
```

Q2] Create an Android application, with two activity first activity will have an EditText and a Button where the user can enter player name and after clicking on button the entered name will be display in another Activity. Second activity has the BACK button to transition to first activity (Using Intent).

Solution

activity_main.xml

```
<?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"
    android:padding="16dp"
    tools:context=".MainActivity">

    <EditText
        android:id="@+id/playerNameEditText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter player name"
        android:inputType="text"/>

    <Button
        android:id="@+id/sendButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/playerNameEditText"
        android:layout_marginTop="16dp"
        android:text="Send"
        android:onClick="sendMessage"/>

</RelativeLayout>
```

activity_display_name.xml

```
<?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"
    android:padding="16dp"
    tools:context=".DisplayNameActivity">

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

```
        android:textSize="24sp"
        android:layout_centerInParent="true"/>

    <Button
        android:id="@+id/backButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/displayNameTextView"
        android:layout_marginTop="16dp"
        android:text="Back"
        android:onClick="goBack"/>

</RelativeLayout>
```

MainActivity.java

```
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.EditText;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

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

    public void sendMessage(View view) {
        EditText playerNameEditText = findViewById(R.id.playerNameEditText);
        String playerName = playerNameEditText.getText().toString().trim();

        Intent intent = new Intent(this, DisplayNameActivity.class);
        intent.putExtra("playerName", playerName); // Send player name as an extra
        startActivity(intent);
    }
}
```

DisplayNameActivity.java

```
import android.content.Intent;
import android.os.Bundle;
import android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;

public class DisplayNameActivity extends AppCompatActivity {

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

        Intent intent = getIntent();
        String playerName = intent.getStringExtra("playerName"); // Receive player name extra

        TextView displayNameTextView = findViewById(R.id.displayNameTextView);
        displayNameTextView.setText(playerName); // Set player name in TextView
    }

    public void goBack(View view) {
```

```
        finish(); // Finish current activity to go back
    }
}
```

Slip 17

Q1] Write an Android Program to demonstrate Activity life Cycle.

activity_main.xml

```
<!-- activity_main.xml -->
<?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"
    android:padding="16dp"
    tools:context=".MainActivity">

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Activity Lifecycle Demo"
        android:textSize="24sp"
        android:layout_centerInParent="true"/>

</RelativeLayout>
```

MainActivity.java

```
// MainActivity.java
package com.example.activitylifecycledemo;

import android.os.Bundle;
import android.util.Log;
import android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

    private static final String TAG = "MainActivity";

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        Log.d(TAG, "onCreate: Activity created");
    }

    @Override
    protected void onStart() {
        super.onStart();
        Log.d(TAG, "onStart: Activity started");
    }

    @Override
    protected void onResume() {
        super.onResume();
        Log.d(TAG, "onResume: Activity resumed");
    }
```

```
    @Override
    protected void onPause() {
        super.onPause();
        Log.d(TAG, "onPause: Activity paused");
    }

    @Override
    protected void onStop() {
        super.onStop();
        Log.d(TAG, "onStop: Activity stopped");
    }

    @Override
    protected void onDestroy() {
        super.onDestroy();
        Log.d(TAG, "onDestroy: Activity destroyed");
    }

    @Override
    protected void onRestart() {
        super.onRestart();
        Log.d(TAG, "onRestart: Activity restarted");
    }
}
```

AndroidManifest.xml

```
<!-- AndroidManifest.xml -->
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.activitylifecycledemo">

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>
```

Q2] Write a PhoneGap application to create a contact. Options are: • Searching for Contacts • Cloning Contacts • Removing Contacts.

Solution:

index.html

```
<!-- index.html -->
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Contact Management</title>
    <style>
        /* Add your CSS styles here */
        /* For simplicity, you can add inline styles or use a separate CSS file */
```

```
    </style>
</head>
<body>
    <h1>Contact Management</h1>

    <button id="searchBtn">Search Contacts</button>
    <button id="cloneBtn">Clone Contacts</button>
    <button id="removeBtn">Remove Contacts</button>

    <script src="phonegap.js"></script>
    <script src="index.js"></script>
</body>
</html>
```

index.js

```
// index.js
document.addEventListener('deviceready', onDeviceReady, false);

function onDeviceReady() {
    console.log('Device is ready');

    // Add event listeners for buttons
    document.getElementById('searchBtn').addEventListener('click', searchContacts);
    document.getElementById('cloneBtn').addEventListener('click', cloneContacts);
    document.getElementById('removeBtn').addEventListener('click', removeContacts);
}

function searchContacts() {
    // Implement logic to search for contacts
}

function cloneContacts() {
    // Implement logic to clone contacts
}

function removeContacts() {
    // Implement logic to remove contacts
}
```

Slip 18

Q1] Create an Android Application that will change color of the screen and change the font size of text view using xml.

Solution:

activity_main.xml

```
<!-- activity_main.xml -->
<?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"
    android:padding="16dp"
    tools:context=".MainActivity">

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Hello, World!"
        android:textSize="24sp"
        android:layout_centerInParent="true"/>
```

```
    <Button
        android:id="@+id/changeButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Change Color & Font Size"
        android:layout_below="@id/textView"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="16dp"/>

</RelativeLayout>
```

res/values/styles.xml

```
<!-- res/values/styles.xml -->
<resources>
    <style name="AppTheme" parent="'Theme.AppCompat.Light.DarkActionBar">
        <!-- Customize your theme here. -->
    </style>

    <style name="LargeText" parent="android:TextAppearance.Large">
        <item name="android:textSize">30sp</item>
    </style>

    <style name="SmallText" parent="android:TextAppearance.Small">
        <item name="android:textSize">18sp</item>
    </style>

    <color name="colorPrimary">#6200EE</color>
    <color name="colorPrimaryDark">#3700B3</color>
    <color name="colorAccent">#03DAC5</color>
    <color name="customColor">#FF0000</color>
</resources>
```

res/values/colors.xml

```
<!-- res/values/colors.xml -->
<resources>
    <color name="customColor">#FF0000</color>
</resources>
```

MainActivity.java

```
// MainActivity.java
package com.example.colorandfontsize;

import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

    private TextView textView;
    private Button changeButton;

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

        textView = findViewById(R.id.textView);
        changeButton = findViewById(R.id.changeButton);
```

```
        changeButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                // Change text color
                textView.setTextColor(getResources().getColor(R.color.customColor));

                // Change text size
                textView.setTextSize(30); // Change to desired font size
            }
        });
    }
}
```

Q2] Create table Project (id, name, dept, city). Create Application to perform the following operations. (using SQLite database) i] Add at least 5 records. ii] Display all the records.

Solution:

DatabaseHelper.java

```
// DatabaseHelper.java
package com.example.projectdatabase;

import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;

public class DatabaseHelper extends SQLiteOpenHelper {

    private static final String DATABASE_NAME = "project_db";
    private static final int DATABASE_VERSION = 1;

    private static final String TABLE_NAME = "Project";
    private static final String COLUMN_ID = "id";
    private static final String COLUMN_NAME = "name";
    private static final String COLUMN_DEPT = "dept";
    private static final String COLUMN_CITY = "city";

    public DatabaseHelper(Context context) {
        super(context, DATABASE_NAME, null, DATABASE_VERSION);
    }

    @Override
    public void onCreate(SQLiteDatabase db) {
        String createTableQuery = "CREATE TABLE " + TABLE_NAME + " (" +
                COLUMN_ID + " INTEGER PRIMARY KEY AUTOINCREMENT, " +
                COLUMN_NAME + " TEXT, " +
                COLUMN_DEPT + " TEXT, " +
                COLUMN_CITY + " TEXT)";
        db.execSQL(createTableQuery);
    }

    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
        db.execSQL("DROP TABLE IF EXISTS " + TABLE_NAME);
        onCreate(db);
    }

    public boolean addProject(String name, String dept, String city) {
        SQLiteDatabase db = this.getWritableDatabase();
        ContentValues contentValues = new ContentValues();
        contentValues.put(COLUMN_NAME, name);
        contentValues.put(COLUMN_DEPT, dept);
```

```
        contentValues.put(COLUMN_CITY, city);
        long result = db.insert(TABLE_NAME, null, contentValues);
        return result != -1;
    }

    public Cursor getAllProjects() {
        SQLiteDatabase db = this.getWritableDatabase();
        return db.rawQuery("SELECT * FROM " + TABLE_NAME, null);
    }
}
```

MainActivity.java

```
// MainActivity.java
package com.example.projectdatabase;

import androidx.appcompat.app.AppCompatActivity;

import android.database.Cursor;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;

public class MainActivity extends AppCompatActivity {

    private DatabaseHelper databaseHelper;
    private TextView textView;

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

        databaseHelper = new DatabaseHelper(this);
        textView = findViewById(R.id.textView);

        Button addButton = findViewById(R.id.addButton);
        Button displayButton = findViewById(R.id.displayButton);

        addButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                databaseHelper.addProject("Project 1", "IT", "New York");
                databaseHelper.addProject("Project 2", "Finance", "London");
                databaseHelper.addProject("Project 3", "Marketing", "Paris");
                databaseHelper.addProject("Project 4", "HR", "Berlin");
                databaseHelper.addProject("Project 5", "Operations", "Tokyo");
            }
        });

        displayButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                displayProjects();
            }
        });
    }

    private void displayProjects() {
        Cursor cursor = databaseHelper.getAllProjects();
        StringBuilder stringBuilder = new StringBuilder();
        while (cursor.moveToNext()) {
```

```
            int id = cursor.getInt(cursor.getColumnIndex("id"));
            String name = cursor.getString(cursor.getColumnIndex("name"));
            String dept = cursor.getString(cursor.getColumnIndex("dept"));
            String city = cursor.getString(cursor.getColumnIndex("city"));
            stringBuilder.append("ID: ").append(id).append(", Name: ").append(name)
                    .append(", Department: ").append(dept).append(", City: ").append(city)
                    .append("\n");
        }
        textView.setText(stringBuilder.toString());
    }
}
```

activity_main.xml

```
<!-- activity_main.xml -->
<?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/addButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Add Records"
        android:layout_marginTop="16dp"
        android:layout_centerHorizontal="true" />

    <Button
        android:id="@+id/displayButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Display Records"
        android:layout_below="@id/addButton"
        android:layout_marginTop="16dp"
        android:layout_centerHorizontal="true" />

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text=""
        android:textSize="18sp"
        android:layout_below="@id/displayButton"
        android:layout_marginTop="32dp"
        android:layout_centerHorizontal="true" />

</RelativeLayout>
```

Slip 19

Q1] Write an Android Program to Change the Image Displayed on the Screen.

Solution:

activity_main.xml

```
<!-- activity_main.xml -->
<?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">

    <ImageView
        android:id="@+id/imageView"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:src="@drawable/image1"
        android:scaleType="fitCenter"
        android:layout_centerInParent="true"
        android:adjustViewBounds="true" />

    <Button
        android:id="@+id/changeImageButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Change Image"
        android:layout_below="@id/imageView"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="16dp" />

</RelativeLayout>
```

MainActivity.java

```
// MainActivity.java
package com.example.imagedisplay;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.ImageView;

public class MainActivity extends AppCompatActivity {

    private ImageView imageView;
    private Button changeImageButton;

    private int[] imageIds = {R.drawable.image1, R.drawable.image2, R.drawable.image3};
    private int currentImageIndex = 0;

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

        imageView = findViewById(R.id.imageView);
        changeImageButton = findViewById(R.id.changeImageButton);

        // Set initial image
        imageView.setImageResource(imageIds[currentImageIndex]);

        // Change image on button click
        changeImageButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                currentImageIndex = (currentImageIndex + 1) % imageIds.length;
                imageView.setImageResource(imageIds[currentImageIndex]);
            }
```

```
        });
    }
}
```

Q2] Construct an Android Application to create two option menu as Find Factorial and Find Sum of Digits. Accept a number and calculate Factorial and Sum of Digits of a given number by clicking Menu.

Solution:

activity_main.xml

```
<!-- activity_main.xml -->
<?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/numberEditText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter a number"
        android:inputType="number"
        android:layout_margin="16dp"
        android:layout_centerHorizontal="true" />

    <TextView
        android:id="@+id/resultTextView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text=""
        android:textSize="18sp"
        android:layout_below="@id/numberEditText"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="16dp" />

</RelativeLayout>
```

MainActivity.java

```
// MainActivity.java
package com.example.calculator;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.widget.EditText;
import android.widget.TextView;

public class MainActivity extends AppCompatActivity {

    private EditText numberEditText;
    private TextView resultTextView;

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

        numberEditText = findViewById(R.id.numberEditText);
        resultTextView = findViewById(R.id.resultTextView);
```

```
    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        getMenuInflater().inflate(R.menu.main_menu, menu);
        return true;
    }

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        switch (item.getItemId()) {
            case R.id.menuFactorial:
                calculateFactorial();
                return true;
            case R.id.menuSumOfDigits:
                calculateSumOfDigits();
                return true;
            default:
                return super.onOptionsItemSelected(item);
        }
    }

    private void calculateFactorial() {
        String numberStr = numberEditText.getText().toString();
        if (!numberStr.isEmpty()) {
            int number = Integer.parseInt(numberStr);
            int factorial = 1;
            for (int i = 1; i <= number; i++) {
                factorial *= i;
            }
            resultTextView.setText("Factorial: " + factorial);
        } else {
            resultTextView.setText("Please enter a number.");
        }
    }

    private void calculateSumOfDigits() {
        String numberStr = numberEditText.getText().toString();
        if (!numberStr.isEmpty()) {
            int number = Integer.parseInt(numberStr);
            int sum = 0;
            while (number != 0) {
                sum += number % 10;
                number /= 10;
            }
            resultTextView.setText("Sum of Digits: " + sum);
        } else {
            resultTextView.setText("Please enter a number.");
        }
    }
}
```

res/menu/main_menu.xml

```
<!-- res/menu/main_menu.xml -->
<menu xmlns:android="http://schemas.android.com/apk/res/android">
    <item
        android:id="@+id/menuFactorial"
        android:title="Find Factorial" />
    <item
        android:id="@+id/menuSumOfDigits"
        android:title="Find Sum of Digits" />
</menu>
```

Slip 20

Q1] Write an application to accept two numbers from the user and displays them. But Reject input if both numbers are greater than 20 and asks for two new numbers.

Solution:

activity_main.xml

```
<!-- activity_main.xml -->
<?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"
    android:padding="16dp"
    tools:context=".MainActivity">

    <EditText
        android:id="@+id/firstNumberEditText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter first number"
        android:inputType="number"
        android:layout_marginBottom="16dp" />

    <EditText
        android:id="@+id/secondNumberEditText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter second number"
        android:inputType="number"
        android:layout_below="@id/firstNumberEditText"
        android:layout_marginBottom="16dp" />

    <Button
        android:id="@+id/submitButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Submit"
        android:layout_below="@id/secondNumberEditText"
        android:layout_centerHorizontal="true" />

</RelativeLayout>
```

MainActivity.java

```
// MainActivity.java
package com.example.numberinput;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {

    private EditText firstNumberEditText;
    private EditText secondNumberEditText;
    private Button submitButton;

    @Override
```

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

        firstNumberEditText = findViewById(R.id.firstNumberEditText);
        secondNumberEditText = findViewById(R.id.secondNumberEditText);
        submitButton = findViewById(R.id.submitButton);

        submitButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                int firstNumber = Integer.parseInt(firstNumberEditText.getText().toString());
                int secondNumber = Integer.parseInt(secondNumberEditText.getText().toString());

                if (firstNumber > 20 && secondNumber > 20) {
                    Toast.makeText(MainActivity.this, "Both numbers are greater than 20. Please enter two new numbers.",
Toast.LENGTH_SHORT).show();
                } else {
                    String message = "First number: " + firstNumber + "\nSecond number: " + secondNumber;
                    Toast.makeText(MainActivity.this, message, Toast.LENGTH_SHORT).show();
                }
            }
        });
    }
}
```

Q2] Java Android Program to send email with attachment.

Solution:

activity_main.xml:

```
<!-- activity_main.xml -->
<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"
    android:padding="16dp"
    tools:context=".MainActivity">

    <Button
        android:id="@+id/sendEmailButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Send Email"
        android:layout_centerInParent="true" />

</RelativeLayout>
```

MainActivity.java:

```
package com.example.emailsender;

import android.os.Bundle;
import android.os.StrictMode;
import android.view.View;
import android.widget.Button;

import androidx.appcompat.app.AppCompatActivity;

import javax.activation.DataHandler;
import javax.activation.DataSource;
import javax.mail.BodyPart;
import javax.mail.Message;
import javax.mail.MessagingException;
import javax.mail.Multipart;
```

```
import javax.mail.PasswordAuthentication;
import javax.mail.Session;
import javax.mail.Transport;
import javax.mail.internet.InternetAddress;
import javax.mail.internet.MimeBodyPart;
import javax.mail.internet.MimeMessage;
import javax.mail.internet.MimeMultipart;
import javax.mail.util.ByteArrayDataSource;

public class MainActivity extends AppCompatActivity {

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

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

    private void sendEmail() {
        // Allow network access on the main thread (for demo purposes only)
        StrictMode.ThreadPolicy policy = new StrictMode.ThreadPolicy.Builder().permitAll().build();
        StrictMode.setThreadPolicy(policy);

        // Sender's credentials
        final String username = "your_email@gmail.com";
        final String password = "your_password";

        // Recipient's email
        String toEmail = "recipient_email@example.com";

        // Sender's email
        String fromEmail = "your_email@gmail.com";

        // Email subject
        String subject = "Test Email with Attachment";

        // Email content
        String body = "Hello,\n\nPlease find the attached file.";

        // File path of the attachment
        String filePath = "path_to_your_attachment";

        Session session = Session.getInstance(getProperties(), new javax.mail.Authenticator() {
            protected PasswordAuthentication getPasswordAuthentication() {
                return new PasswordAuthentication(username, password);
            }
        });

        try {
            Message message = new MimeMessage(session);
            message.setFrom(new InternetAddress(fromEmail));
            message.setRecipients(Message.RecipientType.TO, InternetAddress.parse(toEmail));
            message.setSubject(subject);

            // Create the email body
```

```
            BodyPart messageBodyPart = new MimeBodyPart();
            messageBodyPart.setText(body);

            // Attach the file
            Multipart multipart = new MimeMultipart();
            multipart.addBodyPart(messageBodyPart);
            messageBodyPart = new MimeBodyPart();
            DataSource source = new ByteArrayDataSource(getAssets().open(filePath), "application/octet-stream");
            messageBodyPart.setDataHandler(new DataHandler(source));
            messageBodyPart.setFileName("attachment_name");
            multipart.addBodyPart(messageBodyPart);

            // Set the email content with attachment
            message.setContent(multipart);

            // Send the email
            Transport.send(message);

            // Email sent successfully
        } catch (MessagingException e) {
            e.printStackTrace();
        } catch (Exception e) {
            e.printStackTrace();
        }
    }

    private javax.mail.Session getProperties() {
        java.util.Properties props = new java.util.Properties();
        props.put("mail.smtp.auth", "true");
        props.put("mail.smtp.starttls.enable", "true");
        props.put("mail.smtp.host", "smtp.gmail.com");
        props.put("mail.smtp.port", "587");
        return javax.mail.Session.getInstance(props);
    }
}
```

AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.emailsender">

    <uses-permission android:name="android.permission.INTERNET" />

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>
```

Q.1] Write an Android Program to demonstrate Activity life Cycle

Solution:

activity_main.xml

```
<!-- activity_main.xml -->
<?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:id="@+id/container"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:padding="16dp"
    tools:context=".MainActivity">

    <TextView
        android:id="@+id/statusTextView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Activity Status:"
        android:textSize="18sp"
        android:textStyle="bold" />

</RelativeLayout>
```

MainActivity.java

```
package com.example.activitylifecycledemo;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.util.Log;
import android.widget.TextView;

public class MainActivity extends AppCompatActivity {

    private static final String TAG = "MainActivity";
    private TextView statusTextView;

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

        statusTextView = findViewById(R.id.statusTextView);
        statusTextView.append("\nActivity Created");
        Log.d(TAG, "onCreate: Activity Created");
    }

    @Override
    protected void onStart() {
        super.onStart();
        statusTextView.append("\nActivity Started");
        Log.d(TAG, "onStart: Activity Started");
    }

    @Override
    protected void onResume() {
        super.onResume();
        statusTextView.append("\nActivity Resumed");
        Log.d(TAG, "onResume: Activity Resumed");
    }
```

```
    @Override
    protected void onPause() {
        super.onPause();
        statusTextView.append("\nActivity Paused");
        Log.d(TAG, "onPause: Activity Paused");
    }

    @Override
    protected void onStop() {
        super.onStop();
        statusTextView.append("\nActivity Stopped");
        Log.d(TAG, "onStop: Activity Stopped");
    }

    @Override
    protected void onDestroy() {
        super.onDestroy();
        statusTextView.append("\nActivity Destroyed");
        Log.d(TAG, "onDestroy: Activity Destroyed");
    }
}
```

AndroidManifest.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.activitylifecycledemo">

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>
```

Q.2] Create an Android Application that writes data to the SD Card

Solution:

activity_main.xml

```
<!-- activity_main.xml -->
<?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"
    android:padding="16dp"
    tools:context=".MainActivity">

    <Button
        android:id="@+id/writeButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Write to SD Card"
```

```
    android:layout_centerInParent="true" />

</RelativeLayout>
```

MainActivity.java:

```
package com.example.sdcardwrite;

import android.Manifest;
import android.content.pm.PackageManager;
import android.os.Bundle;
import android.os.Environment;
import android.util.Log;
import android.view.View;
import android.widget.Button;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.ActivityCompat;
import androidx.core.content.ContextCompat;

import java.io.File;
import java.io.FileOutputStream;
import java.io.IOException;

public class MainActivity extends AppCompatActivity {

   private static final String TAG = "MainActivity";
   private static final int REQUEST_WRITE_STORAGE_PERMISSION = 100;

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

      Button writeButton = findViewById(R.id.writeButton);
      writeButton.setOnClickListener(new View.OnClickListener() {
         @Override
         public void onClick(View v) {
            if (checkPermission()) {
               writeToSDCard();
            } else {
               requestPermission();
            }
         }
      });
   }

   private boolean checkPermission() {
      return ContextCompat.checkSelfPermission(this, Manifest.permission.WRITE_EXTERNAL_STORAGE)
            == PackageManager.PERMISSION_GRANTED;
   }

   private void requestPermission() {
      ActivityCompat.requestPermissions(this,
            new String[]{Manifest.permission.WRITE_EXTERNAL_STORAGE},
            REQUEST_WRITE_STORAGE_PERMISSION);
   }

   @Override
   public void onRequestPermissionsResult(int requestCode, String[] permissions, int[] grantResults) {
      super.onRequestPermissionsResult(requestCode, permissions, grantResults);
      if (requestCode == REQUEST_WRITE_STORAGE_PERMISSION) {
```

```
        if (grantResults.length > 0 && grantResults[0] == PackageManager.PERMISSION_GRANTED) {
            writeToSDCard();
        } else {
            Toast.makeText(this, "Permission Denied!", Toast.LENGTH_SHORT).show();
        }
    }
  }

  private void writeToSDCard() {
    String fileName = "example.txt";
    String data = "This is a sample text file saved to the SD card.";

    File directory = new File(Environment.getExternalStorageDirectory(), "MyDirectory");
    if (!directory.exists()) {
        directory.mkdirs();
    }

    File file = new File(directory, fileName);
    try {
        FileOutputStream outputStream = new FileOutputStream(file);
        outputStream.write(data.getBytes());
        outputStream.close();
        Log.d(TAG, "writeToSDCard: File saved to " + file.getAbsolutePath());
        Toast.makeText(this, "File saved to " + file.getAbsolutePath(), Toast.LENGTH_SHORT).show();
    } catch (IOException e) {
        e.printStackTrace();
    }
  }
}
```

AndroidManifest.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
  package="com.example.sdcardwrite">

  <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
  <uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE" />

  <application
    android:allowBackup="true"
    android:icon="@mipmap/ic_launcher"
    android:label="@string/app_name"
    android:roundIcon="@mipmap/ic_launcher_round"
    android:supportsRtl="true"
    android:theme="@style/AppTheme">
    <activity android:name=".MainActivity">
      <intent-filter>
        <action android:name="android.intent.action.MAIN" />

        <category android:name="android.intent.category.LAUNCHER" />
      </intent-filter>
    </activity>
  </application>

</manifest>
```

Slip 22

Q.1] Write an Java Android Program to Change the Image on the Screen.
Solution:

activity_main.xml

```
<!-- activity_main.xml -->
<?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:id="@+id/container"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:padding="16dp"
    tools:context=".MainActivity">

    <ImageView
        android:id="@+id/imageView"
        android:layout_width="200dp"
        android:layout_height="200dp"
        android:layout_centerInParent="true"
        android:src="@drawable/image1" />

    <Button
        android:id="@+id/changeImageButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Change Image"
        android:layout_below="@id/imageView"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="16dp" />

</RelativeLayout>
```

MainActivity.java

```
package com.example.imagechange;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.ImageView;

public class MainActivity extends AppCompatActivity {

    private ImageView imageView;
    private Button changeImageButton;

    private int currentImageIndex = 1; // Starting image index

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

        imageView = findViewById(R.id.imageView);
        changeImageButton = findViewById(R.id.changeImageButton);

        changeImageButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                changeImage();
            }
        });
    }
```

```
    private void changeImage() {
        // Change image based on current index
        switch (currentImageIndex) {
            case 1:
                imageView.setImageResource(R.drawable.image2);
                currentImageIndex = 2;
                break;
            case 2:
                imageView.setImageResource(R.drawable.image3);
                currentImageIndex = 3;
                break;
            case 3:
                imageView.setImageResource(R.drawable.image1);
                currentImageIndex = 1;
                break;
        }
    }
}
```

Q.2] Perform following numeric operation according to user selection of radio button.

Solution:

activity_main.xml

```
<!-- activity_main.xml -->
<?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"
    android:padding="16dp"
    tools:context=".MainActivity">

    <RadioGroup
        android:id="@+id/radioGroup"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerHorizontal="true"
        android:orientation="horizontal">

        <RadioButton
            android:id="@+id/additionRadioButton"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Addition" />

        <RadioButton
            android:id="@+id/subtractionRadioButton"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Subtraction"
            android:layout_marginStart="20dp" />

        <RadioButton
            android:id="@+id/multiplicationRadioButton"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Multiplication"
            android:layout_marginStart="20dp" />

        <RadioButton
```

```
        android:id="@+id/divisionRadioButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Division"
        android:layout_marginStart="20dp" />

    </RadioGroup>

    <EditText
        android:id="@+id/firstNumberEditText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@id/radioGroup"
        android:layout_marginTop="20dp"
        android:hint="Enter first number"
        android:inputType="numberDecimal"
        android:padding="10dp" />

    <EditText
        android:id="@+id/secondNumberEditText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@id/firstNumberEditText"
        android:layout_marginTop="10dp"
        android:hint="Enter second number"
        android:inputType="numberDecimal"
        android:padding="10dp" />

    <Button
        android:id="@+id/calculateButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/secondNumberEditText"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="20dp"
        android:text="Calculate" />

    <TextView
        android:id="@+id/resultTextView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/calculateButton"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="20dp"
        android:text=""
        android:textSize="18sp"
        android:textStyle="bold" />

</RelativeLayout>
```

MainActivity.java

```
package com.example.numericoperations;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.RadioButton;
import android.widget.RadioGroup;
import android.widget.TextView;
```

```
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {

    private EditText firstNumberEditText, secondNumberEditText;
    private Button calculateButton;
    private TextView resultTextView;
    private RadioGroup radioGroup;

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

        firstNumberEditText = findViewById(R.id.firstNumberEditText);
        secondNumberEditText = findViewById(R.id.secondNumberEditText);
        calculateButton = findViewById(R.id.calculateButton);
        resultTextView = findViewById(R.id.resultTextView);
        radioGroup = findViewById(R.id.radioGroup);

        calculateButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                performOperation();
            }
        });
    }

    private void performOperation() {
        int selectedId = radioGroup.getCheckedRadioButtonId();
        if (selectedId == -1) {
            Toast.makeText(this, "Select an operation", Toast.LENGTH_SHORT).show();
            return;
        }

        RadioButton radioButton = findViewById(selectedId);
        String operation = radioButton.getText().toString();

        double firstNumber = Double.parseDouble(firstNumberEditText.getText().toString());
        double secondNumber = Double.parseDouble(secondNumberEditText.getText().toString());
        double result = 0;

        switch (operation) {
            case "Addition":
                result = firstNumber + secondNumber;
                break;
            case "Subtraction":
                result = firstNumber - secondNumber;
                break;
            case "Multiplication":
                result = firstNumber * secondNumber;
                break;
            case "Division":
                if (secondNumber == 0) {
                    Toast.makeText(this, "Cannot divide by zero", Toast.LENGTH_SHORT).show();
                    return;
                }
                result = firstNumber / secondNumber;
                break;
        }

        resultTextView.setText("Result: " + result);
```

```
    }
}
```

Slip23

Q. 1] Write a Java android program to demonstrate implicit intent.

Solution:

MainActivity.java

```
package com.example.implicitintent;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;

public class MainActivity extends AppCompatActivity {

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

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

    private void openWebsite() {
        // Create an intent with ACTION_VIEW action and the URL of the website to be opened
        String websiteUrl = "https://www.example.com";
        Intent intent = new Intent(Intent.ACTION_VIEW, Uri.parse(websiteUrl));

        // Verify that the intent will resolve to an activity
        if (intent.resolveActivity(getPackageManager()) != null) {
            // Start the activity with the intent
            startActivity(intent);
        }
    }
}
```

activity_main.xml

```
<!-- activity_main.xml -->
<?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"
    android:padding="16dp"
    tools:context=".MainActivity">

    <Button
        android:id="@+id/openWebsiteButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
```

```
    android:text="Open Website"
    android:layout_centerInParent="true" />

</RelativeLayout>
```

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.

Solution:

activity_main.xml

```
<!-- activity_main.xml -->
<?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"
   android:padding="16dp"
   tools:context=".MainActivity">

   <EditText
      android:id="@+id/nameEditText"
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      android:hint="Enter your name"
      android:inputType="text" />

   <RadioGroup
      android:id="@+id/formatRadioGroup"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:layout_below="@id/nameEditText"
      android:layout_marginTop="16dp"
      android:orientation="horizontal">

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

      <RadioButton
         android:id="@+id/italicRadioButton"
         android:layout_width="wrap_content"
         android:layout_height="wrap_content"
         android:text="Italic"
         android:layout_marginStart="16dp" />

      <RadioButton
         android:id="@+id/underlineRadioButton"
         android:layout_width="wrap_content"
         android:layout_height="wrap_content"
         android:text="Underline"
         android:layout_marginStart="16dp" />
   </RadioGroup>

   <CheckBox
      android:id="@+id/changeColorCheckBox"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:text="Change Color"
```

```
        android:layout_below="@id/formatRadioGroup"
        android:layout_marginTop="16dp" />

    <Button
        android:id="@+id/displayButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Display"
        android:layout_below="@id/changeColorCheckBox"
        android:layout_marginTop="16dp"
        android:layout_centerHorizontal="true" />

    <Button
        android:id="@+id/clearButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Clear"
        android:layout_below="@id/displayButton"
        android:layout_marginTop="16dp"
        android:layout_marginEnd="8dp"
        android:layout_toStartOf="@+id/exitButton" />

    <Button
        android:id="@+id/exitButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Exit"
        android:layout_below="@id/displayButton"
        android:layout_marginTop="16dp"
        android:layout_alignParentEnd="true" />

    <TextView
        android:id="@+id/resultTextView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text=""
        android:layout_below="@id/clearButton"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="16dp" />

</RelativeLayout>
```

MainActivity.java:

```
package com.example.userinput;

import androidx.appcompat.app.AppCompatActivity;

import android.graphics.Color;
import android.os.Bundle;
import android.text.Html;
import android.view.View;
import android.widget.Button;
import android.widget.CheckBox;
import android.widget.EditText;
import android.widget.RadioButton;
import android.widget.RadioGroup;
import android.widget.TextView;

public class MainActivity extends AppCompatActivity {

    private EditText nameEditText;
    private RadioButton boldRadioButton, italicRadioButton, underlineRadioButton;
```

```
private CheckBox changeColorCheckBox;
private Button displayButton, clearButton, exitButton;
private TextView resultTextView;

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

    nameEditText = findViewById(R.id.nameEditText);
    boldRadioButton = findViewById(R.id.boldRadioButton);
    italicRadioButton = findViewById(R.id.italicRadioButton);
    underlineRadioButton = findViewById(R.id.underlineRadioButton);
    changeColorCheckBox = findViewById(R.id.changeColorCheckBox);
    displayButton = findViewById(R.id.displayButton);
    clearButton = findViewById(R.id.clearButton);
    exitButton = findViewById(R.id.exitButton);
    resultTextView = findViewById(R.id.resultTextView);

    displayButton.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            displayMessage();
        }
    });

    clearButton.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            clearFields();
        }
    });

    exitButton.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            finish();
        }
    });
}

private void displayMessage() {
    String name = nameEditText.getText().toString();
    StringBuilder message = new StringBuilder();

    // Apply formatting based on selected options
    if (boldRadioButton.isChecked()) {
        message.append("<b>");
    }
    if (italicRadioButton.isChecked()) {
        message.append("<i>");
    }
    if (underlineRadioButton.isChecked()) {
        message.append("<u>");
    }

    message.append(name);

    // Apply closing tags for formatting
    if (underlineRadioButton.isChecked()) {
        message.append("</u>");
    }
```

```
        if (italicRadioButton.isChecked()) {
            message.append("</i>");
        }
        if (boldRadioButton.isChecked()) {
            message.append("</b>");
        }

        // Change text color if checkbox is checked
        if (changeColorCheckBox.isChecked()) {
            resultTextView.setTextColor(Color.RED);
        } else {
            resultTextView.setTextColor(Color.BLACK);
        }

        // Display the formatted message
        resultTextView.setText(Html.fromHtml(message.toString()));
    }

    private void clearFields() {
        nameEditText.setText("");
        boldRadioButton.setChecked(false);
        italicRadioButton.setChecked(false);
        underlineRadioButton.setChecked(false);
        changeColorCheckBox.setChecked(false);
        resultTextView.setText("");
    }
}
```

Slip 24

Q.1] Write an application to accept a string from the user. With two buttons to display the string in Uppercase and Lowercase using the toast message.

Solution:

activity_main.xml

```
<!-- activity_main.xml -->
<?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"
    android:padding="16dp"
    tools:context=".MainActivity">

    <EditText
        android:id="@+id/inputEditText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter a string"
        android:inputType="text" />

    <Button
        android:id="@+id/uppercaseButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Uppercase"
        android:layout_below="@id/inputEditText"
```

```
        android:layout_marginTop="16dp" />

    <Button
        android:id="@+id/lowercaseButton"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Lowercase"
        android:layout_below="@id/uppercaseButton"
        android:layout_marginTop="16dp" />

</RelativeLayout>
```

MainActivity.java

```
package com.example.stringcaseconverter;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {

    private EditText inputEditText;
    private Button uppercaseButton, lowercaseButton;

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

        inputEditText = findViewById(R.id.inputEditText);
        uppercaseButton = findViewById(R.id.uppercaseButton);
        lowercaseButton = findViewById(R.id.lowercaseButton);

        uppercaseButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                convertToUppercase();
            }
        });

        lowercaseButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                convertToLowercase();
            }
        });
    }

    private void convertToUppercase() {
        String input = inputEditText.getText().toString();
        String result = input.toUpperCase();
        showToast("Uppercase: " + result);
    }

    private void convertToLowercase() {
        String input = inputEditText.getText().toString();
        String result = input.toLowerCase();
        showToast("Lowercase: " + result);
```

```
    }

    private void showToast(String message) {
        Toast.makeText(this, message, Toast.LENGTH_SHORT).show();
    }
}
```

Q.2] Create table Car (id, name, type, color). Create Java Android Application for performing the following operation on the table. (Using SQLite database) i) Insert 5 New Car Details. ii) Show All the Car Details

Solution:

Car.java:

```
package com.example.carapp;

public class Car {
    private int id;
    private String name;
    private String type;
    private String color;

    public Car() {
    }

    public Car(String name, String type, String color) {
        this.name = name;
        this.type = type;
        this.color = color;
    }

    public Car(int id, String name, String type, String color) {
        this.id = id;
        this.name = name;
        this.type = type;
        this.color = color;
    }

    public int getId() {
        return id;
    }

    public void setId(int id) {
        this.id = id;
    }

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public String getType() {
        return type;
    }

    public void setType(String type) {
        this.type = type;
    }

    public String getColor() {
        return color;
    }
```

```
    public void setColor(String color) {
        this.color = color;
    }
}
```

DatabaseHelper.java

```
package com.example.carapp;

import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;
import java.util.ArrayList;
import java.util.List;

public class DatabaseHelper extends SQLiteOpenHelper {
    private static final int DATABASE_VERSION = 1;
    private static final String DATABASE_NAME = "carManager";
    private static final String TABLE_CAR = "car";
    private static final String KEY_ID = "id";
    private static final String KEY_NAME = "name";
    private static final String KEY_TYPE = "type";
    private static final String KEY_COLOR = "color";

    public DatabaseHelper(Context context) {
        super(context, DATABASE_NAME, null, DATABASE_VERSION);
    }

    @Override
    public void onCreate(SQLiteDatabase db) {
        String CREATE_CAR_TABLE = "CREATE TABLE " + TABLE_CAR + "("
                + KEY_ID + " INTEGER PRIMARY KEY,"
                + KEY_NAME + " TEXT,"
                + KEY_TYPE + " TEXT,"
                + KEY_COLOR + " TEXT" + ")";
        db.execSQL(CREATE_CAR_TABLE);
    }

    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
        db.execSQL("DROP TABLE IF EXISTS " + TABLE_CAR);
        onCreate(db);
    }

    public void addCar(Car car) {
        SQLiteDatabase db = this.getWritableDatabase();
        ContentValues values = new ContentValues();
        values.put(KEY_NAME, car.getName());
        values.put(KEY_TYPE, car.getType());
        values.put(KEY_COLOR, car.getColor());
        db.insert(TABLE_CAR, null, values);
        db.close();
    }

    public List<Car> getAllCars() {
        List<Car> carList = new ArrayList<>();
        String selectQuery = "SELECT * FROM " + TABLE_CAR;
        SQLiteDatabase db = this.getWritableDatabase();
        Cursor cursor = db.rawQuery(selectQuery, null);

        if (cursor.moveToFirst()) {
```

```
            do {
                Car car = new Car();
                car.setId(Integer.parseInt(cursor.getString(0)));
                car.setName(cursor.getString(1));
                car.setType(cursor.getString(2));
                car.setColor(cursor.getString(3));
                carList.add(car);
            } while (cursor.moveToNext());
        }
        cursor.close();
        return carList;
    }
}
```

MainActivity.java

```
package com.example.carapp;

import androidx.appcompat.app.AppCompatActivity;
import android.os.Bundle;
import android.util.Log;
import java.util.List;

public class MainActivity extends AppCompatActivity {

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

        DatabaseHelper db = new DatabaseHelper(this);

        // Insert 5 new car details
        db.addCar(new Car("Toyota", "SUV", "Red"));
        db.addCar(new Car("Honda", "Sedan", "Black"));
        db.addCar(new Car("Ford", "Truck", "Blue"));
        db.addCar(new Car("Chevrolet", "SUV", "Silver"));
        db.addCar(new Car("BMW", "Coupe", "White"));

        // Show all car details
        List<Car> cars = db.getAllCars();
        for (Car car : cars) {
            Log.d("CarDetails", "ID: " + car.getId() + ", Name: " + car.getName() + ", Type: " + car.getType() + ", Color: " +
car.getColor());
        }
    }
}
```

Slip 25

Q.1] Create an android application for SMS activity.

Solution:

activity_main.xml:

```
<!-- activity_main.xml -->
<?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"
    android:padding="16dp"
    tools:context=".MainActivity">

    <EditText
        android:id="@+id/editTextPhone"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter phone number"
        android:inputType="phone" />

    <EditText
        android:id="@+id/editTextMessage"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@id/editTextPhone"
        android:layout_marginTop="16dp"
        android:hint="Enter message" />

    <Button
        android:id="@+id/buttonSend"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/editTextMessage"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="16dp"
        android:text="Send" />

</RelativeLayout>
```

MainActivity.java

```
package com.example.smsapp;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.telephony.SmsManager;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {

    private EditText editTextPhone, editTextMessage;
    private Button buttonSend;

    @Override
```

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

        editTextPhone = findViewById(R.id.editTextPhone);
        editTextMessage = findViewById(R.id.editTextMessage);
        buttonSend = findViewById(R.id.buttonSend);

        buttonSend.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String phone = editTextPhone.getText().toString().trim();
                String message = editTextMessage.getText().toString().trim();

                if (!phone.isEmpty() && !message.isEmpty()) {
                    sendSMS(phone, message);
                } else {
                    Toast.makeText(MainActivity.this, "Please enter phone number and message",
Toast.LENGTH_SHORT).show();
                }
            }
        });
    }

    private void sendSMS(String phone, String message) {
        try {
            SmsManager smsManager = SmsManager.getDefault();
            smsManager.sendTextMessage(phone, null, message, null, null);
            Toast.makeText(this, "SMS sent successfully", Toast.LENGTH_SHORT).show();
        } catch (Exception e) {
            Toast.makeText(this, "Failed to send SMS", Toast.LENGTH_SHORT).show();
            e.printStackTrace();
        }
    }
}
```

Q.2] Create an Android application, which show Login Form in table layout. After clicking LOGIN button display the "Login Successful…" message if username and password is same else display "Invalid Login" message in Toast Control.

Solution:

activity_main.xml:

```
<!-- activity_main.xml -->
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout 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"
    android:orientation="vertical"
    android:padding="16dp"
    tools:context=".MainActivity">

    <TableLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:stretchColumns="*">

        <TableRow>
            <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="Username:" />
```

```
        <EditText
            android:id="@+id/editTextUsername"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:inputType="text" />
    </TableRow>

    <TableRow>
        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Password:" />
        <EditText
            android:id="@+id/editTextPassword"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:inputType="textPassword" />
    </TableRow>

    <TableRow>
        <Button
            android:id="@+id/buttonLogin"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Login" />
    </TableRow>

  </TableLayout>

</LinearLayout>
```

MainActivity.java:

```
package com.example.loginapp;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {

  private EditText editTextUsername, editTextPassword;
  private Button buttonLogin;

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

    editTextUsername = findViewById(R.id.editTextUsername);
    editTextPassword = findViewById(R.id.editTextPassword);
    buttonLogin = findViewById(R.id.buttonLogin);

    buttonLogin.setOnClickListener(new View.OnClickListener() {
      @Override
      public void onClick(View v) {
        login();
      }
    });
```

```
    }

    private void login() {
        String username = editTextUsername.getText().toString().trim();
        String password = editTextPassword.getText().toString().trim();

        // Sample username and password for demonstration
        String validUsername = "user";
        String validPassword = "password";

        if (username.equals(validUsername) && password.equals(validPassword)) {
            showToast("Login Successful");
        } else {
            showToast("Invalid Login");
        }
    }

    private void showToast(String message) {
        Toast.makeText(this, message, Toast.LENGTH_SHORT).show();
    }
}
```