Lab 1:

Demonstrate the setup and installation of android project with java.

Objective

- Learn to install and setup android project.

Steps to setup and install the project

Step 1: Download the latest version of Android studio from the official website

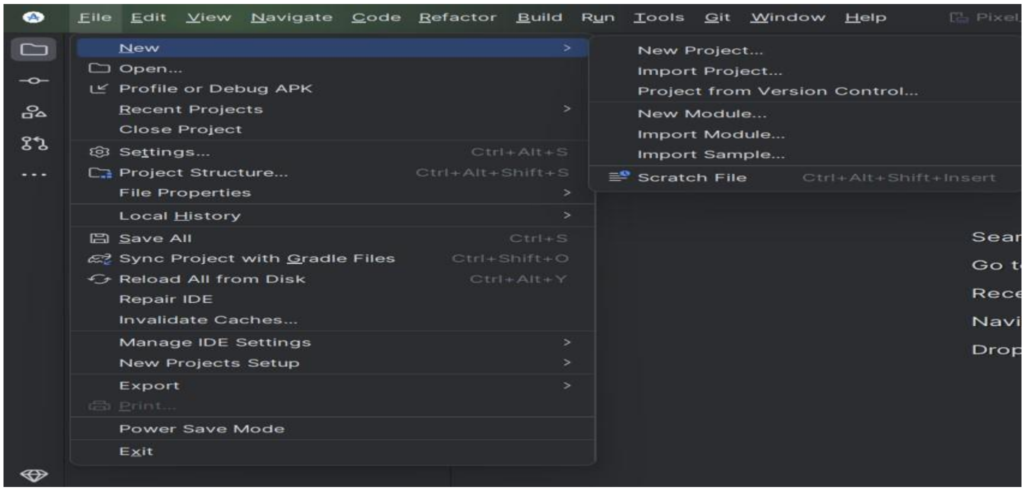
Step 2: Install Android Studio and select Android SDK, Android Virtual Device (AVD).

Step 3: Setting up the project

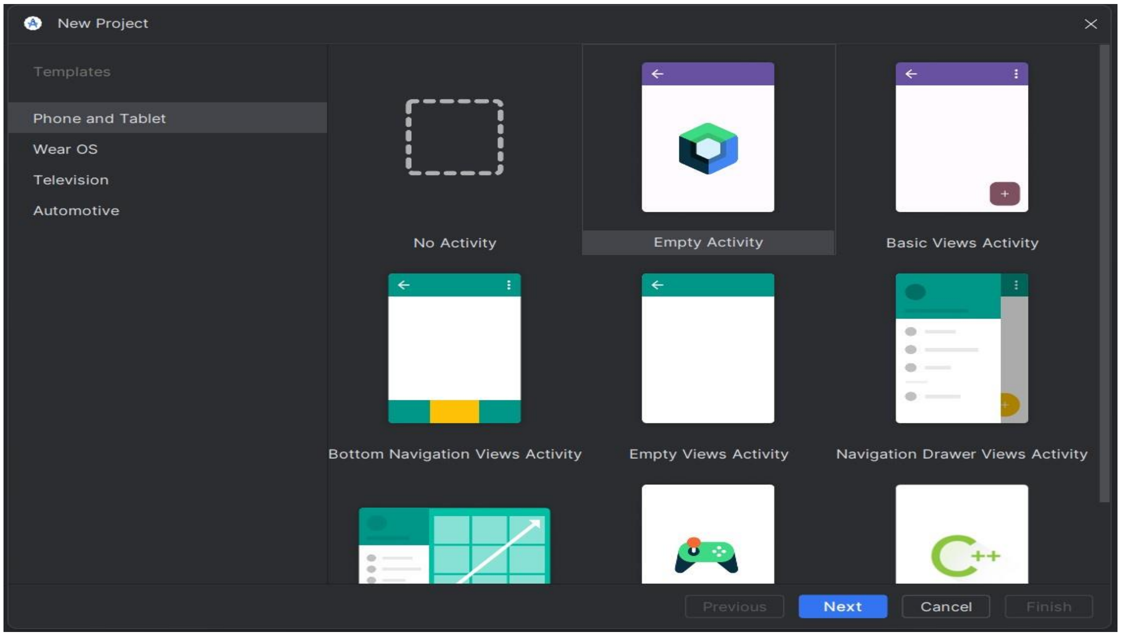
- Open Android Studio
- Click "Start a new Android Studio project."
- In the "New Project" window:

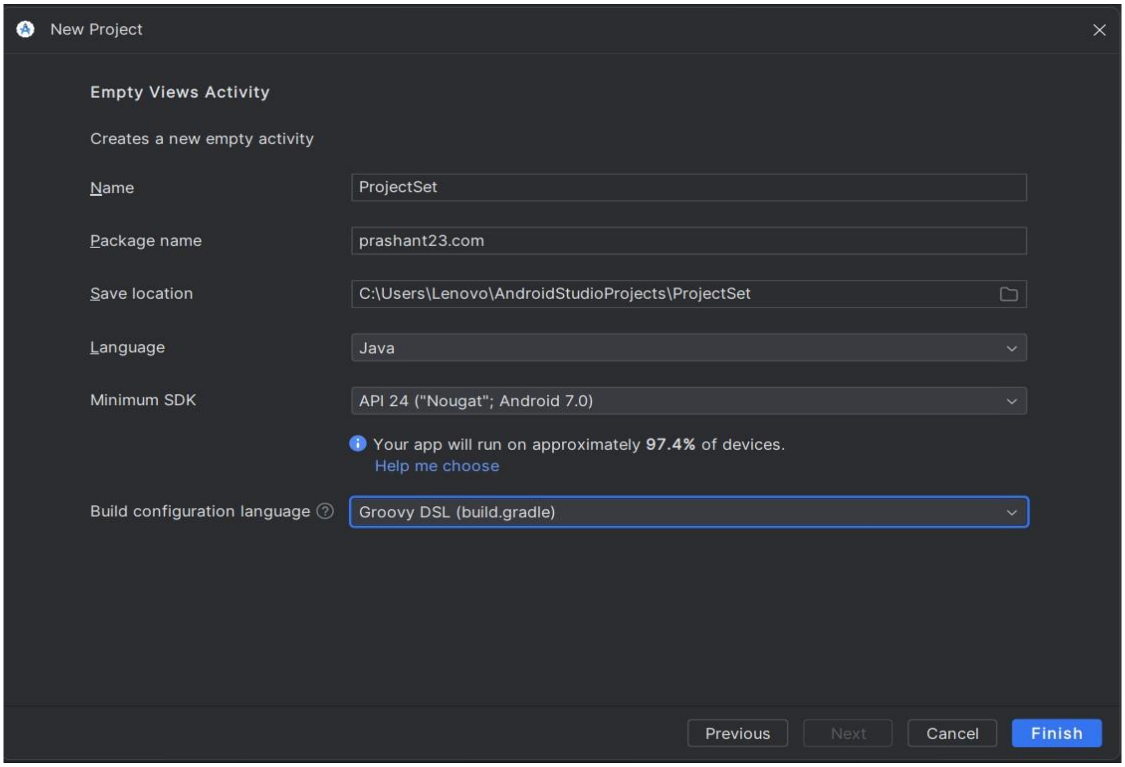
○ **Project Name:** Enter a descriptive name for your project.
○ **Package Name:** Choose a unique package name. This acts as a namespace for your app's code.
○ **Save Location:** Select a location on your computer to save your project files.
○ **Minimum SDK:** Choose the minimum SDK level which support the maximum feature. ○ We use API 24 (“Nougat”; Android 7.0) ○ **Language:** Select "Java" as the development language.
○ **Build configuration language:** choose Groovy DSL(build.gradle) ○ Click “Finish”.

○ Click on File => New => New Project



Click on Empty Views Activity





Empty Views Activity

Creates a new empty activity

Name: ProjectSet

Package name: prashant23.com

Save location: C:\Users\Lenovo\AndroidStudioProjects\ProjectSet

Language: Java

Minimum SDK: API 24 ("Nougat"; Android 7.0)

Your app will run on approximately **97.4%** of devices.
Help me choose

Build configuration language: Groovy DSL (build.gradle)

Lab 2:

Develop an android application that prints "hello world" on the bottom of the page.

Objective:

- Learn to develop and run the android project

Lab work:

MainActivity.java

```
package prashant.com;

import android.os.Bundle; import
androidx.activity.EdgeToEdge; import
androidx.appcompat.app.AppCompatActivity; import
androidx.core.graphics.Insets; import androidx.core.view.ViewCompat;
import androidx.core.view.WindowInsetsCompat;

public class MainActivity extends AppCompatActivity {

   @Override     protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);        EdgeToEdge.enable(this);
setContentView(R.layout.activity_main);
       ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main), (v, insets) -> {
           Insets systemBars = insets.getInsets(WindowInsetsCompat.Type.systemBars());
           v.setPadding(systemBars.left, systemBars.top, systemBars.right, systemBars.bottom);
           return insets;
       });
   } }
```

activity_main.xml

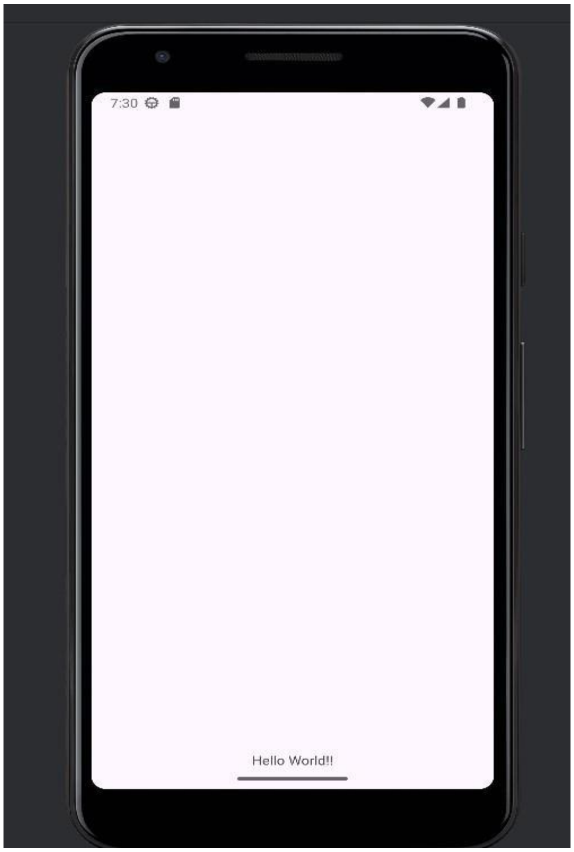
```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:id="@+id/main"     android:layout_width="match_parent"
android:layout_height="match_parent"     android:gravity="bottom|center_horizontal"
tools:context=".MainActivity">
```

```
 <TextView      android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="@string/hello"
    />
</LinearLayout>
```

strings.xml

```
<resources>
  <string name="app_name">ProfileApp</string>
  <string name="hello">Hello World!!</string>
</resources>
```

Output:



Hello World!!

Discussion and Conclusion:

In this lab, we create a simple App to print “Hello world!!” in the bottom of the page layout. We set layout gravity bottom and center_horizontal and in string file we write the string value as “hello world” as set its name as hello and in activity_main.xml file we return the value of string in text as @string/hello.

Lab 3:

Develop an android application with two activities, mainactivity and greetactivity .mainactivity should contain a textbox and button with label " submit'. when clicked on submit greetactivity should open with message hello {name} where name is submitted from mainactivity.

Objective:

- Learn how to deal with two activities

Lab work:

MainActivity.java

```
package prashant.com;

import android.content.Intent; import
android.os.Bundle; import android.view.View; import
android.widget.Button; import android.widget.EditText;
import androidx.activity.EdgeToEdge; import
androidx.appcompat.app.AppCompatActivity; import
androidx.core.graphics.Insets; import
androidx.core.view.ViewCompat; import
androidx.core.view.WindowInsetsCompat;

public class MainActivity extends AppCompatActivity { private
Button submitButton;
private EditText editText;

@Override
protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);         EdgeToEdge.enable(this);
setContentView(R.layout.activity_main);
ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main), (v, insets) -> {
Insets systemBars = insets.getInsets(WindowInsetsCompat.Type.systemBars());
v.setPadding(systemBars.left, systemBars.top, systemBars.right, systemBars.bottom); return insets;
}); submitButton = findViewById(R.id.submit_btn);
String  editText = findViewById(R.id.editText);
submitButton.setOnClickListener(new
View.OnClickListener() {
@Override
public void onClick(View view) {
String text = editText.getText().toString();
Intent i = new Intent(MainActivity.this,GreetActivity.class);
i.putExtra("text",text);
startActivity(i);
}
});
}
}
```

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:id="@+id/main" android:layout_width="match_parent"
android:layout_height="match_parent"
android:orientation="vertical" android:gravity="center"
tools:context=".MainActivity">

<LinearLayout  android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:orientation="horizontal"
>
<EditText android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:gravity="center_vertical"
android:hint="@string/hello"
android:id="@+id/editText"
/>
</LinearLayout>
<Button android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="@string/submit"
android:id="@+id/submit_btn"
        />
</LinearLayout>
```

strings.xml

```
<resources>
    <string name="app_name">Profile</string>
    <string name="submit">Submit</string>
    <string name="hello">Enter your name.. </string>
    <string name="helloText">Hello, </string>
</resources>
```

GreetActivity.java

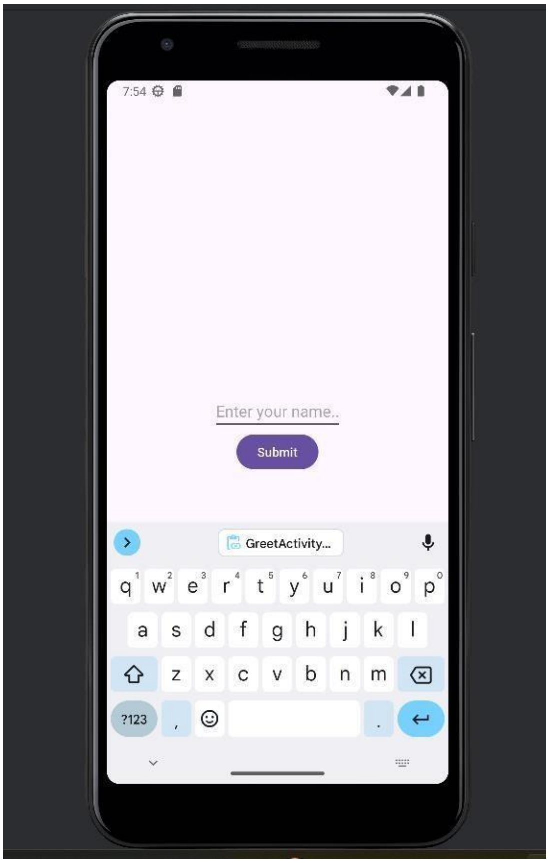
```
package prashant.com;

import android.os.Bundle; import
android.widget.TextView; import androidx.activity.EdgeToEdge;
import androidx.appcompat.app.AppCompatActivity; import
androidx.core.graphics.Insets; import
androidx.core.view.ViewCompat;
import androidx.core.view.WindowInsetsCompat;

public class GreetActivity extends AppCompatActivity {
 @Override    protected void
onCreate(BundlesavedInstanceState) {
super.onCreate(savedInstanceState);         EdgeToEdge.enable(this);
setContentView(R.layout.activity_greeta);
ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main), (v, insets) -> {
Insets systemBars = insets.getInsets(WindowInsetsCompat.Type.systemBars());
v.setPadding(systemBars.left, systemBars.top, systemBars.right, systemBars.bottom); return insets;
});

TextView handleTextView = findViewById(R.id.hello_input);
String  text =getIntent().getStringExtra("text");         handleTextView.setText(text);
}
}
```

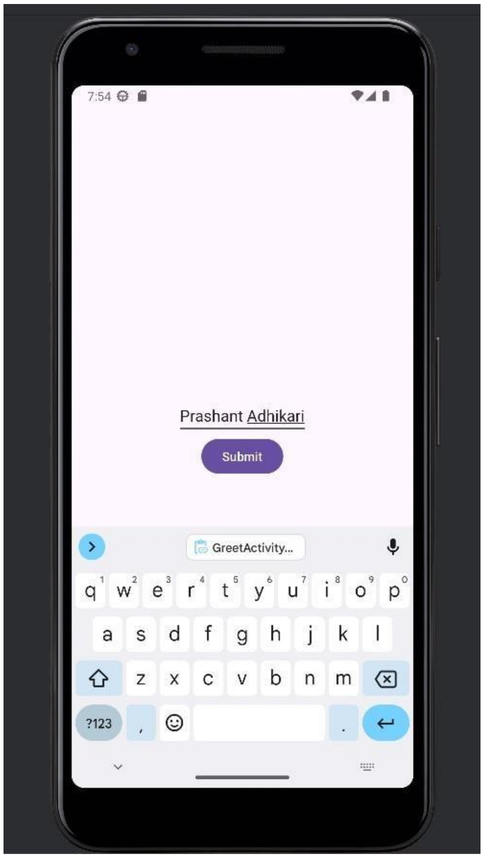
activity_greeta.xml

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

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:id="@+id/main" android:layout_width="match_parent"
android:layout_height="match_parent"
android:gravity="center_vertical|center"
tools:context=".GreetActivity">
<TextView        android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="@string/helloText"
/>
<TextView        android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:id="@+id/hello_input"
/>
</LinearLayout>
```

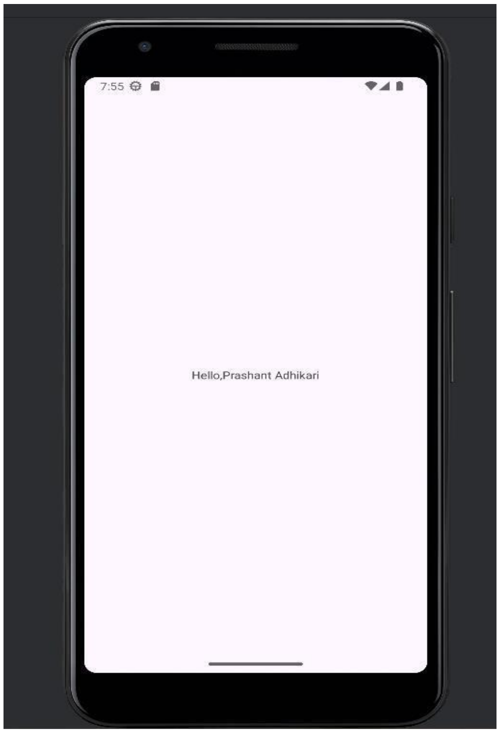
Output:





Prashant Adhikari

After click on submit button the GreetActivity comes into scene.



```
Hello,Prashant Adhikari
```

Discussion and conclusion:

In this lab, we learn and deal with two activities as MainActivity and GreetActivity. The activities are store in stack. The MainActivity contains the Text input field and the submit button after click the submit button the activity is change and goes to GreetActivity which contains the user input value and Hello as default value as its UI.

We use Intent feature to communicate between two activities. We create an object of the Intent and pass the MainActivity and GreetActivity as parameter and use startActivity to run the activity.

Lab 4: Google Map

Objective

➢ To implement google map in android application

Lab work

MainActivity.java

```
package com.example.googlemap;

import android.*;
 import android.app.*;
import android.content.pm.*;
import android.location.*;
import android.os.Bundle;
import androidx.annotation.*;
import androidx.appcompat.app.*;
import androidx.core.content.*
import com.google.android.gms.location.*;
import com.google.android.gms.maps.*;
import com.google.android.gms.tasks.*;

// google map sdk
public class MapActivity extends AppCompatActivity implements OnMapReadyCallback {

   private GoogleMap googleMap;
   private FusedLocationProviderClient locationProviderClient;
   @Override
   protected void onCreate(@Nullable Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
       setContentView(R.layout.activity_map);
       locationProviderClient = LocationServices.getFusedLocationProviderClient(this);
       if (ContextCompat.checkSelfPermission(this,
android.Manifest.permission.ACCESS_FINE_LOCATION)                  !=
PackageManager.PERMISSION_GRANTED) {
ActivityCompat.requestPermissions(this, new
String[]{Manifest.permission.ACCESS_FINE_LOCATION}, 1);
       }
       SupportMapFragment mapFragment = (SupportMapFragment)
getSupportFragmentManager().findFragmentById(R.id.map);
  mapFragment.getMapAsync(this);
    }
   @Override public void onMapReady(@NonNull GoogleMap googleMap) {
 this.googleMap = googleMap;
       UpdateLocation();
   }
   @Override
   public void onRequestPermissionsResult(int requestCode,
@NonNull
String[] permissions,
int[] grantResults) {
        super.onRequestPermissionsResult(requestCode, permissions, grantResults);
  if (requestCode == 1) {
            if (grantResults[0] == PackageManager.PERMISSION_GRANTED) {
                UpdateLocation();
            }
        }     }

    private void UpdateLocation(){
if (ActivityCompat.checkSelfPermission(this, android.Manifest.permission.ACCESS_FINE_LOCATION)
!= PackageManager.PERMISSION_GRANTED  && ActivityCompat.checkSelfPermission(this,
android.Manifest.permission.ACCESS_COARSE_LOCATION) !=
                PackageManager.PERMISSION_GRANTED) {
             return;
        }
        locationProviderClient.getLastLocation().addOnSuccessListener(this, new
OnSuccessListener<Location>() {
            @Override            \
  public void onSuccess(Location location) {if (location == null) {
return;
                }
                LatLng currentLat = new LatLng(location.getLatitude(),location.getLongitude());
googleMap.addMarker(new MarkerOptions().position(currentLat).title("you is here"));
googleMap.moveCamera(CameraUpdateFactory.newLatLngZoom(currentLat,15));
            }         });
    } }
```

activity_main.xml

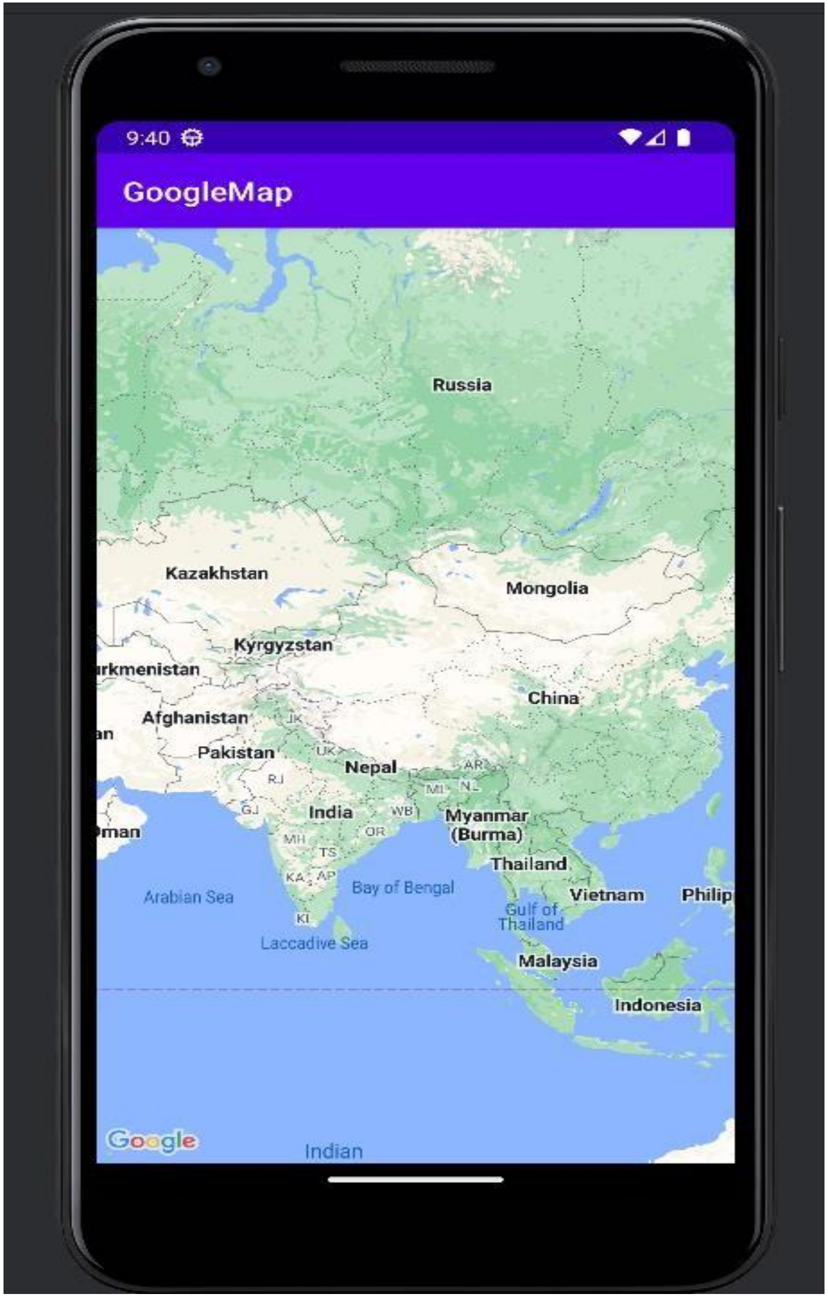
```
<?xml version="1.0" encoding="utf-8"?>

<androidx.fragment.app.FragmentContainerView
android:layout_width="match_parent"
 android:layout_height="match_parent"
android:layout_gravity="center"
android:name="com.google.android.gms.maps.SupportMapFragment"
android:id="@+id/map"
 xmlns:android="http://schemas.android.com/apk/res/android" />
```

AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:tools="http://schemas.android.com/tools">
    <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"/>
  <uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION"/>
```

```
    <application  android:allowBackup="true"
android:dataExtractionRules="@xml/data_extraction_rules"
android:fullBackupContent="@xml/backup_rules"
android:icon="@mipmap/ic_launcher"
 android:label="@string/app_name"
 android:roundIcon="@mipmap/ic_launcher_round"
 android:supportsRtl="true"
 android:theme="@style/Theme.GoogleMap"
tools:targetApi="31">

        <meta-data android:name="com.google.android.geo.API_KEY"
            android:value="AIzaSyBiK29ZZJ9hjSho0c9UhX1hKujvygh8UVY"/>

        <activity android:name=".MapActivity" android:exported="true">
            <intent-filter >
                <action android:name="android.intent.action.MAIN"/>
                <category android:name="android.intent.category.LAUNCHER"/>
            </intent-filter>

        </activity>

    </application>

</manifest>
```

Output:



Discussion and conclusion:
In this lab, we implement google map using API key. We implement the OnMapReadyCallBack method to add google map feature in the application. In this lab we try to pin points on our own current location and add marker there as You are here.

Lab 5: Simple Sqlite operations on android

1. Create a sqlite database named “noteApp.db”.
2. Create a table named “notes” with following columns:
    a. _id (autoincrement primary key)
    b. UUID string
    c. Title string
    d. Description string
3. Create two buttons in MainActivity
    a. Insert
    b. List
4. When clicked on insert, make a database operation to insert dummy data on notes table.

a. Insert at least 5 dummy notes

5. When clicked on List, make a database operation to query all data on notes table and show it in a listView just below the buttons.

MainActivity.java

```
package com.example.noteapp;

import android.database.*;
 import android.os.*;
import android.view.*;
import android.widget.*;
import androidx.appcompat.app.*;
import java.util.*;

public class MainActivity extends AppCompatActivity {
   private NoteDbHelper dbHelper;
   private ListView listView;

   @Override protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
       setContentView(R.layout.activity_main);
       dbHelper = new NoteDbHelper(this);
       listView = findViewById(R.id.listView);
       Button insertButton = findViewById(R.id.insertButton);
       Button listButton = findViewById(R.id.listButton);
       // Insert dummy notes when clicking the insert button
insertButton.setOnClickListener(new View.OnClickListener() {
           @Override

public void onClick(View v) {
dbHelper.insertDummyData();
            }
  });
```

```
        // List all notes when clicking the list button
        listButton.setOnClickListener(new View.OnClickListener() {
       @Override
            public void onClick(View v) {
                List<String> notes = getAllNotes();
                ArrayAdapter<String> adapter = new ArrayAdapter<>(MainActivity.this,
android.R.layout.simple_list_item_1, notes);
listView.setAdapter(adapter);
            }
        });
  }

    // Retrieve all notes from the database and convert to a list
 private List<String> getAllNotes() {
        List<String> notesList = new ArrayList<>();
        Cursor cursor = dbHelper.getAllNotes();

        if (cursor.moveToFirst()) {
 do {
                String title = cursor.getString(cursor.getColumnIndex("title"));
                String description = cursor.getString(cursor.getColumnIndex("description"));
notesList.add(title + ": " + description);
            } while (cursor.moveToNext());
        }
        cursor.close();
return notesList;
    } }
```

NoteDbHelper.java

```
package com.example.noteapp;

import android.content.*;
import android.database.*;
import android.database.sqlite.*;
import android.widget.*;
import java.util.UUID;

public class NoteDbHelper extends SQLiteOpenHelper {
private static final String DATABASE_NAME = "noteApp.db";
private static final int DATABASE_VERSION = 1;
    private static final String TABLE_NOTES = "notes";

    // Columns
private static final String COLUMN_ID = "_id";
private static final String COLUMN_UUID = "uuid";
private static final String COLUMN_TITLE = "title";
```

```
 private static final String COLUMN_DESCRIPTION = "description";

    private final Context context;
    public NoteDbHelper(Context context) {
super(context, DATABASE_NAME, null, DATABASE_VERSION);
this.context = context;
    }
    @Override
    public void onCreate(SQLiteDatabase db) {
        String createTable = "CREATE TABLE " + TABLE_NOTES + " (" +
                COLUMN_ID + " INTEGER PRIMARY KEY AUTOINCREMENT, " +
                COLUMN_UUID + " TEXT, " +
                COLUMN_TITLE + " TEXT, " +
                COLUMN_DESCRIPTION + " TEXT)";
        db.execSQL(createTable);
    }
    @Override
public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
db.execSQL("DROP TABLE IF EXISTS " + TABLE_NOTES);
onCreate(db);
    }
    @Override
 public void onOpen(SQLiteDatabase db) {
 super.onOpen(db);
 // Clear all previous data
        db.execSQL("DELETE FROM " + TABLE_NOTES);
    }
    // Insert Dummy Data with unique descriptions
public void insertDummyData() {
        SQLiteDatabase db = this.getWritableDatabase();
        ContentValues contentValues = new ContentValues();
        // Array of unique descriptions
        String[] descriptions = {
                "Advance Java Programming Note",
                "Mobile Programming Note",
                "Network Programming Note",
                "Applied Economics Note",
                "Distributed System Note"
        };

        for (int i = 1; i <= descriptions.length; i++) {
 contentValues.put(COLUMN_UUID, UUID.randomUUID().toString());
contentValues.put(COLUMN_TITLE, "Note " + i);
            contentValues.put(COLUMN_DESCRIPTION, descriptions[i - 1]);
// Assign unique description
 db.insert(TABLE_NOTES, null, contentValues);
        }
        // Use the context to show a Toast message
        Toast.makeText(context, "Data inserted", Toast.LENGTH_SHORT).show();
    }

    // Insert Dummy Data with dynamic descriptions (optional)
public void insertDynamicDummyData() {

SQLiteDatabase db = this.getWritableDatabase();
 ContentValues contentValues = new ContentValues();

        for (int i = 1; i <= 5; i++) {
            String uniqueDescription = "CodyNoteApp " + i + " - " + System.currentTimeMillis();
contentValues.put(COLUMN_UUID, UUID.randomUUID().toString());
contentValues.put(COLUMN_TITLE, "Note " + i);
contentValues.put(COLUMN_DESCRIPTION, uniqueDescription);
 db.insert(TABLE_NOTES, null, contentValues);
        }

        Toast.makeText(context, "Dynamic data inserted", Toast.LENGTH_SHORT).show();
    }
    // Query All Notes
    public Cursor getAllNotes() {
        SQLiteDatabase db = this.getReadableDatabase();
return db.rawQuery("SELECT * FROM " + TABLE_NOTES, null);
    }
}
```

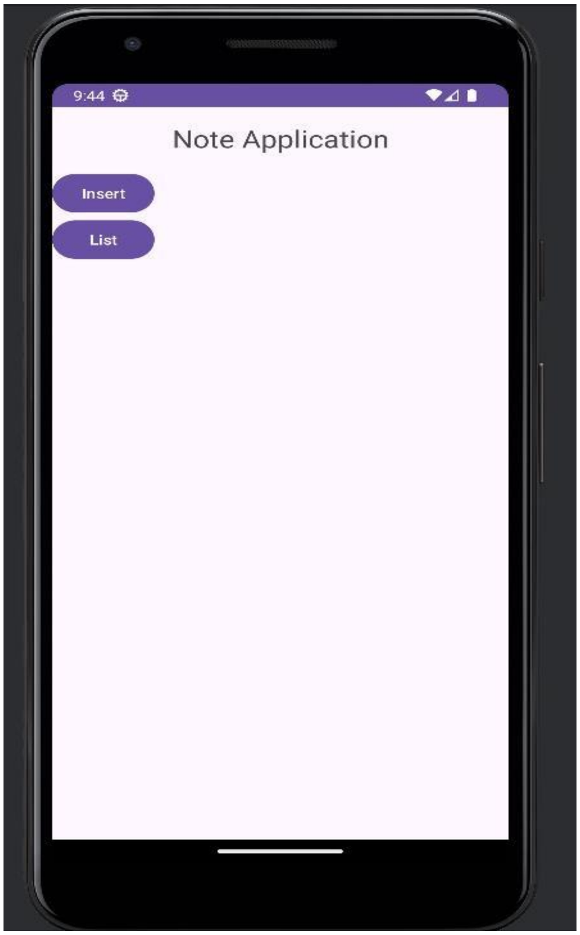
activity_main.xml

```
<?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:id="@+id/main"
android:layout_width="match_parent"
android:layout_height="match_parent"
tools:context=".MainActivity"
android:orientation="vertical">

    <TextView
android:layout_width="match_parent"
 android:layout_height="wrap_content"
android:text="Note Application"
android:textSize="25dp"
 android:textAlignment="center"
 android:layout_margin="16dp"
        />
    <Button
```

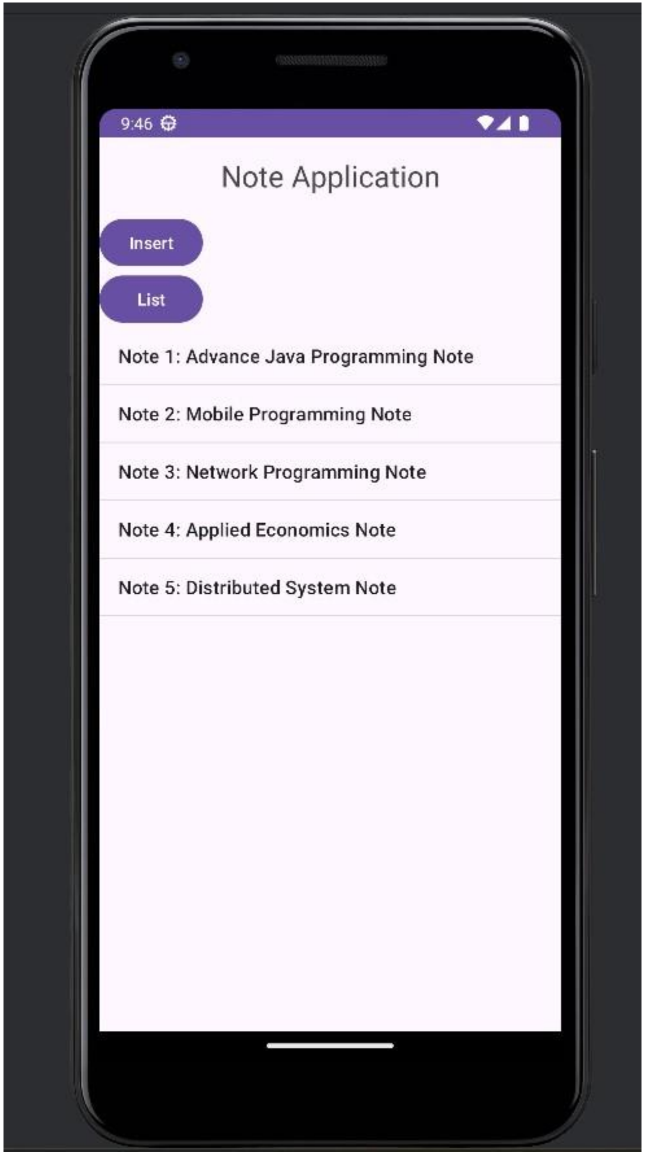
```
android:id="@+id/insertButton"
 android:layout_width="wrap_content"
 android:layout_height="wrap_content"
android:text="Insert" />

    <Button
 android:id="@+id/listButton"
 android:layout_width="wrap_content"
 android:layout_height="wrap_content"
 android:text="List"
        />
    <ListView
 android:id="@+id/listView"
android:layout_width="match_parent"
 android:layout_height="match_parent"
        />
  </LinearLayout>
```

Output:

First view of the app



After clicking on List button we can see list of an item like this



Note 1: Advance Java Programming Note

Note 2: Mobile Programming Note

Note 3: Network Programming Note

Note 4: Applied Economics Note

Note 5: Distributed System Note

Conclusion:

In this lab, we learn about the database SQlite to store data. We insert dummy data in the database and show that data in list form. We create two button as Insert and List. The insert button function as it help to insert data in the database and List button function as to show the list of item store in the database.

Lab:6
Develop an android application to input your Name, Age, gender, email address, phone number and a submit button. When clicked on the button, show this information on another activity. Perform given validation on the following input fields:

Name:	Must not be empty, must satisfy the following expression: “FirstName LastName”.
Email	Must not be empty, must be a valid email address
Phone Number	Must not be empty
Gender	One gender must be selected

Note: You can use Google’s material UI for better looking Input fields and error messages.

MainActivity.java

```
package com.example.personalinfoapp;

import android.content.Intent;
 import android.os.Bundle;
import android.text.TextUtils;
import android.util.Patterns;
import android.view.View;
import android.widget.*;

import androidx.appcompat.app.AppCompatActivity;
import com.google.android.material.textfield.TextInputEditText;

public class MainActivity extends AppCompatActivity {
private TextInputEditText nameInput, ageInput, emailInput, phoneInput;
private RadioGroup genderRadioGroup;
private Button submitButton;

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

        // Initialize Views
nameInput = findViewById(R.id.inputName);
```

```
 ageInput = findViewById(R.id.inputAge);
emailInput = findViewById(R.id.inputEmail);
 phoneInput = findViewById(R.id.inputPhone);
 genderRadioGroup = findViewById(R.id.genderRadioGroup);
submitButton = findViewById(R.id.submitButton);

        submitButton.setOnClickListener(new View.OnClickListener() {
            @Override
 public void onClick(View view) {
 if (validateInputs()) {
 // Get selected gender
                    int genderId = genderRadioGroup.getCheckedRadioButtonId();
                    RadioButton selectedGender = findViewById(genderId);

                    // Create intent to pass data to SecondActivity
 Intent intent = new Intent(MainActivity.this, SecondActivity.class);
 intent.putExtra("name", nameInput.getText().toString());
intent.putExtra("age", ageInput.getText().toString());
 intent.putExtra("email", emailInput.getText().toString());
 intent.putExtra("phone", phoneInput.getText().toString());
intent.putExtra("gender", selectedGender.getText().toString());
 startActivity(intent);
                }
            }
        });
  }

    private boolean validateInputs() {
 boolean isValid = true;

        // Name validation
        String namePattern = "^[A-Za-z]+\\s[A-Za-z]+$";
 if (TextUtils.isEmpty(nameInput.getText()) || !nameInput.getText().toString().matches(namePattern)) {
 nameInput.setError("Enter valid full name (FirstName LastName)");
isValid = false;
        }

        // Email validation
 if (TextUtils.isEmpty(emailInput.getText()) ||
!Patterns.EMAIL_ADDRESS.matcher(emailInput.getText()).matches()) {
emailInput.setError("Enter a valid email");
isValid = false;
        }

        // Gender validation
        if (genderRadioGroup.getCheckedRadioButtonId() == -1) {
            Toast.makeText(this, "Please select a gender", Toast.LENGTH_SHORT).show();
 isValid = false;
        }

        return isValid;
    } }
```

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
 android:layout_width="match_parent"
android:layout_height="match_parent"
   tools:context=".MainActivity">

    <com.google.android.material.textfield.TextInputLayout
 android:id="@+id/inputNameLayout"
android:layout_width="0dp"
 android:layout_height="wrap_content"
android:hint="Name"
app:layout_constraintTop_toTopOf="parent"
 app:layout_constraintStart_toStartOf="parent"
app:layout_constraintEnd_toEndOf="parent">

        <com.google.android.material.textfield.TextInputEditText
 android:id="@+id/inputName"
android:layout_width="match_parent"
 android:layout_height="wrap_content"/>
</com.google.android.material.textfield.TextInputLayout>

    <com.google.android.material.textfield.TextInputLayout
 android:id="@+id/inputAgeLayout"
 android:layout_width="0dp"
 android:layout_height="wrap_content"
android:hint="Age"
app:layout_constraintTop_toBottomOf="@id/inputNameLayout"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintEnd_toEndOf="parent">

        <com.google.android.material.textfield.TextInputEditText
android:id="@+id/inputAge"
android:inputType="number"
 android:layout_width="match_parent"
 android:layout_height="wrap_content"/>
 </com.google.android.material.textfield.TextInputLayout>
```

```
   <com.google.android.material.textfield.TextInputLayout
 android:id="@+id/inputEmailLayout"
 android:layout_width="0dp"
android:layout_height="wrap_content"
 android:hint="Email"
app:layout_constraintTop_toBottomOf="@id/inputAgeLayout"
app:layout_constraintStart_toStartOf="parent"

app:layout_constraintEnd_toEndOf="parent">

        <com.google.android.material.textfield.TextInputEditText
android:id="@+id/inputEmail"
android:inputType="textEmailAddress"
 android:layout_width="match_parent"
android:layout_height="wrap_content"/>
 </com.google.android.material.textfield.TextInputLayout>

    <com.google.android.material.textfield.TextInputLayout
 android:id="@+id/inputPhoneLayout"
 android:layout_width="0dp"
 android:layout_height="wrap_content"
android:hint="Phone Number"
app:layout_constraintTop_toBottomOf="@id/inputEmailLayout"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintEnd_toEndOf="parent">

        <com.google.android.material.textfield.TextInputEditText
 android:id="@+id/inputPhone"
 android:inputType="phone"
 android:layout_width="match_parent"
 android:layout_height="wrap_content"/>
</com.google.android.material.textfield.TextInputLayout>
    <RadioGroup
android:id="@+id/genderRadioGroup"
 android:layout_width="wrap_content"
 android:layout_height="wrap_content"
app:layout_constraintTop_toBottomOf="@id/inputPhoneLayout"
app:layout_constraintStart_toStartOf="parent"
 app:layout_constraintEnd_toEndOf="parent">

        <RadioButton
android:id="@+id/maleRadioButton"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Male"/>

        <RadioButton
```

```
android:id="@+id/femaleRadioButton"
 android:layout_width="wrap_content"
android:layout_height="wrap_content"
 android:text="Female"/>
    </RadioGroup>

    <Button
android:id="@+id/submitButton"
 android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Submit"
app:layout_constraintTop_toBottomOf="@id/genderRadioGroup"
app:layout_constraintStart_toStartOf="parent"
 app:layout_constraintEnd_toEndOf="parent"/>

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

SecondActivity.java

```
package com.example.personalinfoapp;

import android.os.Bundle;
import android.widget.TextView;
import androidx.appcompat.app.AppCompatActivity;

public class SecondActivity extends AppCompatActivity {
private TextView nameTextView, ageTextView, emailTextView, phoneTextView, genderTextView;

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

        // Initialize Views
        nameTextView = findViewById(R.id.nameTextView);
 ageTextView = findViewById(R.id.ageTextView);
emailTextView = findViewById(R.id.emailTextView);
phoneTextView = findViewById(R.id.phoneTextView);
genderTextView = findViewById(R.id.genderTextView);

        // Get data from intent
        String name = getIntent().getStringExtra("name");
        String age = getIntent().getStringExtra("age");
        String email = getIntent().getStringExtra("email");
        String phone = getIntent().getStringExtra("phone");
        String gender = getIntent().getStringExtra("gender");

        // Set text to TextViews
nameTextView.setText("Name: " + name);
 ageTextView.setText("Age: " + age);
emailTextView.setText("Email: " + email);
 phoneTextView.setText("Phone: " + phone);
genderTextView.setText("Gender: " + gender);
    }
}
```

second_activity.xml

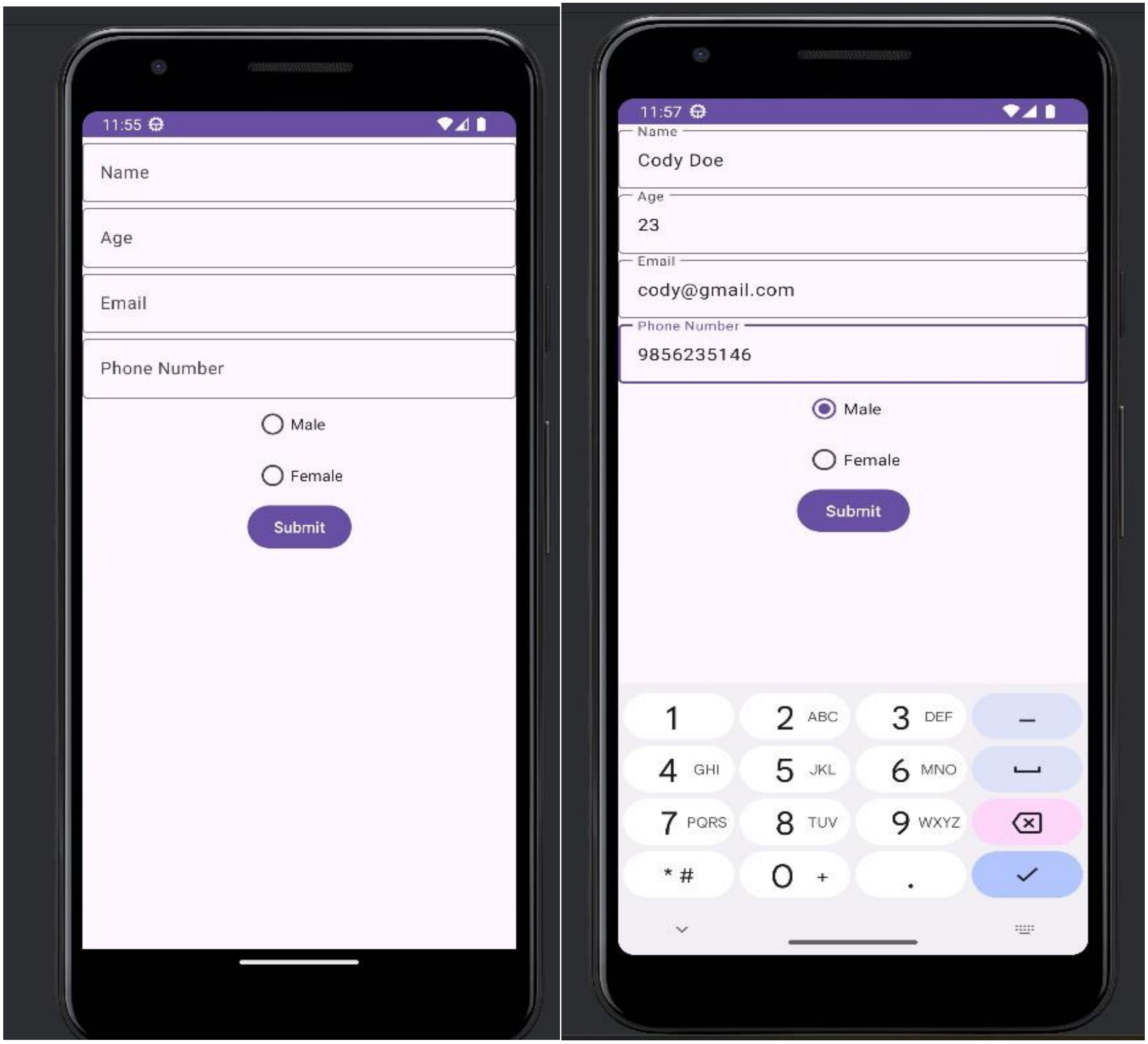
```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
 android:layout_height="match_parent"
tools:context=".SecondActivity"
    >
    <TextView
android:id="@+id/nameTextView"
android:layout_width="wrap_content"
 android:layout_height="wrap_content"
 android:text="Name"
 app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintStart_toStartOf="parent"/>

    <TextView
android:id="@+id/ageTextView"
 android:layout_width="wrap_content"
 android:layout_height="wrap_content" android:text="Age"
app:layout_constraintTop_toBottomOf="@id/nameTextView"
        app:layout_constraintStart_toStartOf="parent"/>

    <TextView
android:id="@+id/emailTextView"
 android:layout_width="wrap_content"
android:layout_height="wrap_content"
 android:text="Email"
app:layout_constraintTop_toBottomOf="@id/ageTextView"
app:layout_constraintStart_toStartOf="parent"/>

    <TextView
 android:id="@+id/phoneTextView"
 android:layout_width="wrap_content"
  android:layout_height="wrap_content"
 android:text="Phone"
```

```
app:layout_constraintTop_toBottomOf="@id/emailTextView"
app:layout_constraintStart_toStartOf="parent"/>

    <TextView
 android:id="@+id/genderTextView"
 android:layout_width="wrap_content"
 android:layout_height="wrap_content"
 android:text="Gender"
 app:layout_constraintTop_toBottomOf="@id/phoneTextView"
app:layout_constraintStart_toStartOf="parent"/>

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

AndroidManifest.xml

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

    <application
android:allowBackup="true"
 android:dataExtractionRules="@xml/data_extraction_rules"
android:fullBackupContent="@xml/backup_rules"
 android:icon="@mipmap/ic_launcher"
android:label="@string/app_name"
 android:roundIcon="@mipmap/ic_launcher_round"
android:supportsRtl="true" android:theme="@style/Theme.PersonalInfoApp" tools:targetApi="31">
        <activity
android:name=".SecondActivity"
 android:exported="false" />
        <activity
android:name=".MainActivity"
 android:exported="true">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

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

</manifest>
```

Output:
This is the initial view of mainActivty create a simple form



11:55

Name

Age

Email

Phone Number

○ Male

○ Female

Submit

11:57

Name: Cody Doe

Age: 23

Email: cody@gmail.com

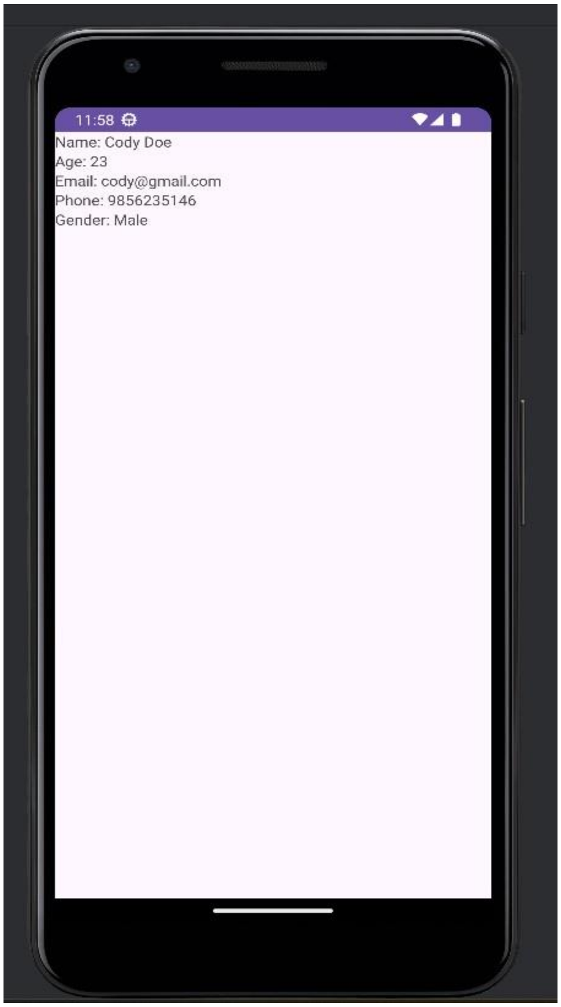
Phone Number: 9856235146

● Male

○ Female

Submit

After clicking on submit we goto secondActivity and that activity show the information inputed in the form



```
Name: Cody Doe
Age: 23
Email: cody@gmail.com
Phone: 9856235146
Gender: Male
```

Conclusion:

In this lab, we learn about the multiple activity and how to handle them. we create a simple form and validate them and that form information is submitted and show in another activity.

Intent class is used to build connection between to activity. Main activity has simple form and second activity is used to show the details of form input.