**Mobile application Lab Manual**

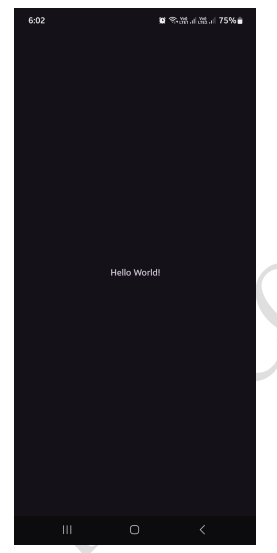
1. **Create a simple Android application to display a "Hello M S Ramaiah" message.**
2. **Design an Android application to demonstrate the use of LinearLayout, RelativeLayout, and ConstraintLayout.**
3. **Develop an app to implement UI components like Buttons, TextViews, Checkboxes, and Radio Buttons.**
4. **Implement event handling for various UI components such as clicks and toggle switches.**
5. **Create an application to demonstrate Explicit and Implicit Intents with data transfer between Activities.**
6. **Create a sample application with login module (check user name and password) on successful login change Textview “Login Successful”. On login fail alert using Toast “login fail”**
7. **Build an application to send an SMS or make a phone call using Native Actions with Intents.**
8. Develop an application to showcase the use of Fragments and Fragment Lifecycle.
9. Create a multi-screen application with communication between Fragments and Activities.
10. Design an application to integrate Google Maps and display the user's current location.
11. Create an Android application to play and record audio or video using Android Media APIs.
12. Create an app to persist data using SQLite by creating a table, inserting, updating, and deleting records.
13. Build an application to save and retrieve data from internal and external storage.
14. Implement Mini project based on the concept learnt in Theory
15. Create a simple Android application to display a "Hello M S Ramaiah" message.

Solution:

activity\_main.xml

<?xml version="1.0" encoding="utf-8"?>  
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:id="@+id/main"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".MainActivity">  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Hello M S Ramaiah"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
</androidx.constraintlayout.widget.ConstraintLayout>

**Output:**



1. Design an Android application to demonstrate the use of LinearLayout, RelativeLayout, and ConstraintLayout.

### Solution:Project Structure:

We’ll create **three activities**:

1.MainActivity.java – LinearLayout

2. RelativeActivity.java – RelativeLayout

3. ConstraintActivity.java – ConstraintLayout

Each activity will show a basic layout example and a button to move to the next layout.

1. **MainActivity.java** – LinearLayout

activity\_main.xml

<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:orientation="vertical"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:padding="16dp">  
  
 <TextView  
 android:id="@+id/linear\_text"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="This is LinearLayout"  
 android:textSize="20sp" />  
  
 <Button  
 android:id="@+id/button\_linear"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Go to RelativeLayout" />  
</LinearLayout>

MainActivity.java

package com.example.prog2;  
import android.content.Intent;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
import androidx.appcompat.app.AppCompatActivity;  
  
public class MainActivity extends AppCompatActivity {  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_main);  
  
 Button btnLinear = findViewById(R.id.button\_linear);  
 btnLinear.setOnClickListener(new View.OnClickListener() {  
 private View v;  
  
 @Override  
 public void onClick(View v) {  
 this.v = v;  
 Intent intent = new Intent(MainActivity.this, RelativeActivity.class);  
 startActivity(intent);  
 }  
 });  
 }  
}

### 2. ****RelativeActivity.java**** – RelativeLayout

#### ✅ activity\_relative.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/relative\_text"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="This is RelativeLayout"  
 android:textSize="20sp" />  
  
 <Button  
 android:id="@+id/button\_relative"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_below="@id/relative\_text"  
 android:layout\_marginTop="20dp"  
 android:text="Go to ConstraintLayout" />  
</RelativeLayout>

**✅ RelativeActivity.java**

package com.example.prog2;  
import android.content.Intent;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
import androidx.appcompat.app.AppCompatActivity;  
  
public class RelativeActivity extends AppCompatActivity {  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_relative);  
  
 Button btnRelative = findViewById(R.id.button\_relative);  
 btnRelative.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 Intent intent = new Intent(RelativeActivity.this, ConstraintActivity.class);  
 startActivity(intent);  
 }  
 });  
 }  
}

### 3. ****ConstraintActivity.java**** – ConstraintLayout

#### ✅ activity\_constraint.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"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:padding="16dp">  
  
 <TextView  
 android:id="@+id/constraint\_text"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="This is ConstraintLayout"  
 android:textSize="20sp"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintLeft\_toLeftOf="parent"  
 app:layout\_constraintRight\_toRightOf="parent"  
 android:layout\_marginTop="100dp"/>  
  
 <Button  
 android:id="@+id/button\_constraint"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Back to LinearLayout"  
 app:layout\_constraintTop\_toBottomOf="@id/constraint\_text"  
 app:layout\_constraintLeft\_toLeftOf="parent"  
 app:layout\_constraintRight\_toRightOf="parent"  
 android:layout\_marginTop="20dp"/>  
</androidx.constraintlayout.widget.ConstraintLayout>

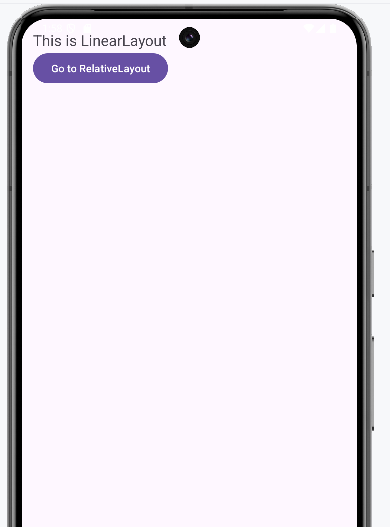
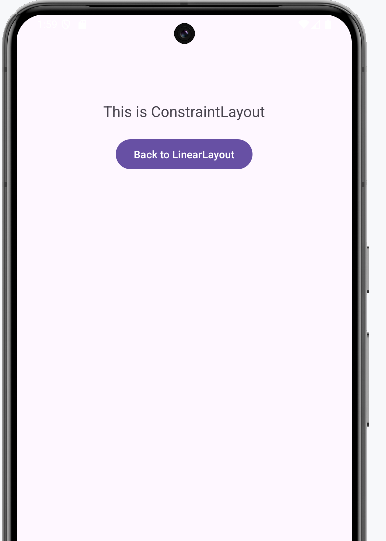
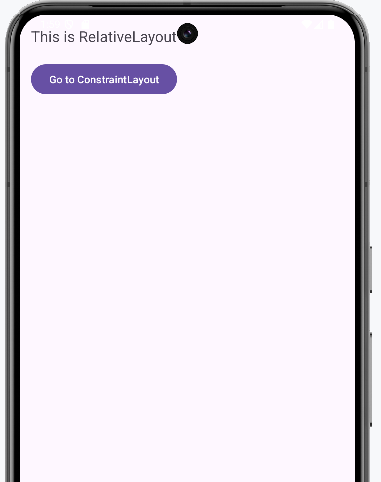
#### ✅ RelativeActivity.java

package com.example.prog2;  
import android.content.Intent;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
import androidx.appcompat.app.AppCompatActivity;  
  
public class RelativeActivity extends AppCompatActivity {  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_relative);  
  
 Button btnRelative = findViewById(R.id.button\_relative);  
 btnRelative.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 Intent intent = new Intent(RelativeActivity.this, ConstraintActivity.class);  
 startActivity(intent);  
 }  
 });  
 }  
}

### ****AndroidManifest.xml:**** Make sure you register all three activities:

<?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.Prog2"  
 tools:targetApi="31">  
 <activity android:name=".ConstraintActivity"/>  
 <activity android:name=".RelativeActivity"/>  
 <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:

**3. Develop an app to implement UI components like Buttons, TextViews, Checkboxes, and Radio Buttons.**

**Solution:**

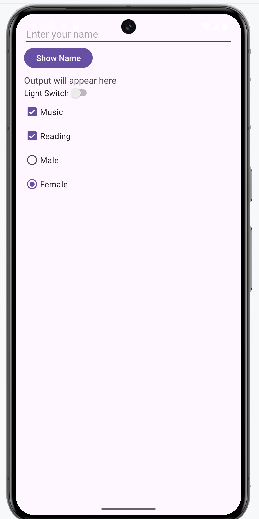
**MainActivity.Java**

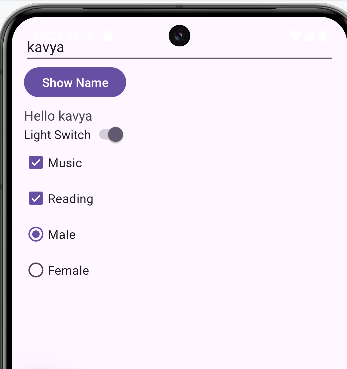
package com.example.prog4;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.\*;  
import androidx.appcompat.app.AppCompatActivity;  
  
public class MainActivity extends AppCompatActivity {  
  
 EditText etName;  
 Button btnShow;  
 TextView tvOutput;  
 Switch switchLight;  
 CheckBox cbMusic, cbReading;  
 RadioGroup rgGender;  
 RadioButton rbMale, rbFemale;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_main);  
  
 // Initialize UI components  
 etName = findViewById(R.id.etName);  
 btnShow = findViewById(R.id.btnShow);  
 tvOutput = findViewById(R.id.tvOutput);  
 switchLight = findViewById(R.id.switchLight);  
 cbMusic = findViewById(R.id.cbMusic);  
 cbReading = findViewById(R.id.cbReading);  
 rgGender = findViewById(R.id.rgGender);  
 rbMale = findViewById(R.id.rbMale);  
 rbFemale = findViewById(R.id.rbFemale);  
  
 // Handle Button click  
 btnShow.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View view) {  
 String name = etName.getText().toString();  
 tvOutput.setText("Hello " + name);  
 }  
 });  
  
 // Handle Switch toggle  
 switchLight.setOnCheckedChangeListener((compoundButton, isChecked) -> {  
 if (isChecked) {  
 Toast.makeText(this, "Light is ON", Toast.LENGTH\_SHORT).show();  
 } else {  
 Toast.makeText(this, "Light is OFF", Toast.LENGTH\_SHORT).show();  
 }  
 });  
  
 // Handle CheckBox events  
 cbMusic.setOnCheckedChangeListener((buttonView, isChecked) -> {  
 if (isChecked) {  
 Toast.makeText(this, "You like Music", Toast.LENGTH\_SHORT).show();  
 }  
 });  
  
 cbReading.setOnCheckedChangeListener((buttonView, isChecked) -> {  
 if (isChecked) {  
 Toast.makeText(this, "You like Reading", Toast.LENGTH\_SHORT).show();  
 }  
 });  
  
 // Handle RadioGroup (Gender) selection  
 rgGender.setOnCheckedChangeListener((group, checkedId) -> {  
 RadioButton selected = findViewById(checkedId);  
 Toast.makeText(this, "Gender: " + selected.getText(), Toast.LENGTH\_SHORT).show();  
 });  
 }  
}

**activity\_main.xml**

<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout  
 xmlns:android="http://schemas.android.com/apk/res/android"  
 android:orientation="vertical"  
 android:padding="16dp"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent">  
  
 <EditText  
 android:id="@+id/etName"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:hint="Enter your name" />  
  
 <Button  
 android:id="@+id/btnShow"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Show Name" />  
  
 <TextView  
 android:id="@+id/tvOutput"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Output will appear here"  
 android:textSize="16sp"  
 android:paddingTop="8dp" />  
  
 <Switch  
 android:id="@+id/switchLight"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Light Switch" />  
  
 <CheckBox  
 android:id="@+id/cbMusic"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Music" />  
  
 <CheckBox  
 android:id="@+id/cbReading"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Reading" />  
  
 <RadioGroup  
 android:id="@+id/rgGender"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content">  
  
 <RadioButton  
 android:id="@+id/rbMale"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Male" />  
  
 <RadioButton  
 android:id="@+id/rbFemale"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Female" />  
 </RadioGroup>  
  
</LinearLayout>

Output:





**4.Implement event handling for various UI components such as clicks and toggle switches.**

Solution:

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.Pro3"  
 tools:targetApi="31">  
 <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>

**mainActivity.java**

package com.example.pro3;  
import android.os.Bundle;  
import androidx.appcompat.app.AppCompatActivity;  
import android.view.View;  
import android.widget.\*;  
  
  
public class MainActivity extends AppCompatActivity {  
  
 TextView tvResult;  
 RadioGroup rgGender;  
 RadioButton rbMale, rbFemale;  
 CheckBox cbJava, cbPython, cbAndroid;  
 Button btnSubmit;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_main);  
  
 tvResult = findViewById(R.id.tvResult);  
 rgGender = findViewById(R.id.rgGender);  
 rbMale = findViewById(R.id.rbMale);  
 rbFemale = findViewById(R.id.rbFemale);  
  
 cbJava = findViewById(R.id.cbJava);  
 cbPython = findViewById(R.id.cbPython);  
 cbAndroid = findViewById(R.id.cbAndroid);  
  
 btnSubmit = findViewById(R.id.btnSubmit);  
  
 btnSubmit.setOnClickListener(new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
  
 StringBuilder result = new StringBuilder("You selected:\n");  
  
 // Gender  
 int selectedId = rgGender.getCheckedRadioButtonId();  
 if (selectedId != -1) {  
 RadioButton selectedRadio = findViewById(selectedId);  
 result.append("Gender: ").append(selectedRadio.getText()).append("\n");  
 }  
  
 // Skills  
 result.append("Skills: ");  
 if (cbJava.isChecked()) result.append("Java ");  
 if (cbPython.isChecked()) result.append("Python ");  
 if (cbAndroid.isChecked()) result.append("Android ");  
  
 tvResult.setText(result.toString());  
 }  
 });  
 }  
}

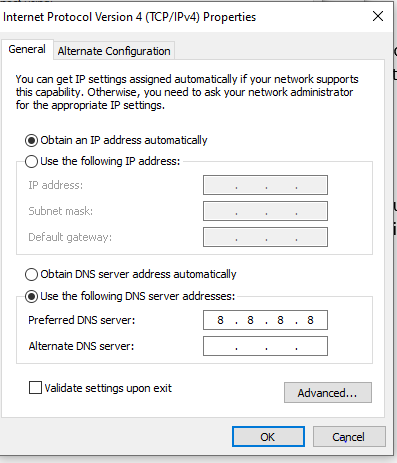
activity\_main.xml

<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:orientation="vertical">  
  
  
 <!-- TextView -->  
 <TextView  
 android:id="@+id/tvResult"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:paddingBottom="10dp"  
 android:text="Your Selection:"  
 android:textAlignment="center"  
 android:textSize="18sp" />  
  
 <!-- RadioGroup for Gender -->  
 <RadioGroup  
 android:id="@+id/rgGender"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content">  
  
 <RadioButton  
 android:id="@+id/rbMale"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Male"/>  
  
 <RadioButton  
 android:id="@+id/rbFemale"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Female"/>  
 </RadioGroup>  
  
 <!-- CheckBoxes for Skills -->  
 <CheckBox  
 android:id="@+id/cbJava"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Java"/>  
  
 <CheckBox  
 android:id="@+id/cbPython"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Python"/>  
  
 <CheckBox  
 android:id="@+id/cbAndroid"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Android"/>  
  
 <!-- Button -->  
 <Button  
 android:id="@+id/btnSubmit"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Submit"  
 android:layout\_marginTop="16dp"/>  
</LinearLayout>

**5.**Create an application to demonstrate Explicit and Implicit Intents with data transfer between Activities.

/\* Explicit Intent: This involves navigating from one activity to another within the same application. Implicit Intent: This involves triggering an action that can be handled by another application. \*/

NOTE:.Goto device manager->select the emulator and wipe out data) and set DNS server IP as 8.8.8.8



**Mainactivity.java**

package com.example.program4;

import android.content.Intent;

import android.net.Uri;

import android.os.Bundle;

import android.view.View;

import androidx.activity.EdgeToEdge;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

EdgeToEdge.enable(this);

setContentView(R.layout.activity\_main);

}

public void onImplicitButtonClicked(View view)

{

Uri url=Uri.parse("https://www.google.com");

Intent i=new Intent(Intent.ACTION\_VIEW, url);

startActivity(i);

}

public void onExplicitButtonClicked(View view )

{

Intent i=new Intent(MainActivity.this, SecondActivity.class);

startActivity(i);

}

}

**SecondActivity.java**

package com.example.program4;

import android.os.Bundle;

import android.widget.Button;

import androidx.appcompat.app.AppCompatActivity;

public class SecondActivity extends AppCompatActivity {

Button btnImplicitContent;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_second);

}

}

**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="center"

tools:context=".MainActivity">

<Button

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="@string/implicit\_intent"

android:onClick="onImplicitButtonClicked"

style="?android:attr/buttonBarButtonStyle" />

<Button

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="@string/explicit\_intent"

android:onClick="onExplicitButtonClicked"

style="?android:attr/buttonBarButtonStyle" />

</LinearLayout>

**Activity\_second.xml**

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

<androidx.constraintlayout.widget.ConstraintLayout

xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:id="@+id/main"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

tools:context=".SecondActivity">

<TextView

android:id="@+id/textView"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="@string/welcome\_to\_explicit\_intent"

android:textSize="28sp"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="parent" />

</androidx.constraintlayout.widget.ConstraintLayout>

**Manifest file**

<?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.PRogram4"

tools:targetApi="31">

<activity

android:name=".SecondActivity"

android:exported="false">

</activity>

<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>

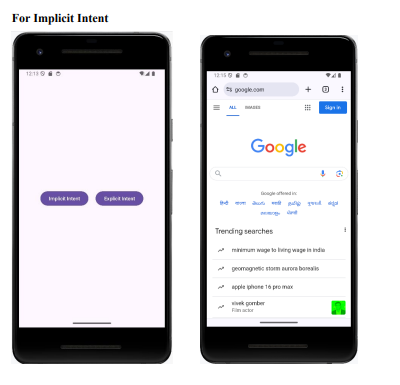
</activity>

</application>

</manifest>

**Output**:

On click of implicit intent button appears below screen



On click of Explicit intent button appears below screen



1. **Create a sample application with login module (check user name and password) on successful login change Textview “Login Successful”. On login fail alert using Toast “login fail”**
2. **XML Layout (activity\_main.xml)**

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

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

android:orientation="vertical"

android:padding="20dp"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent">

<EditText

android:id="@+id/etUsername"

android:hint="Username"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content" />

<EditText

android:id="@+id/etPassword"

android:hint="Password"

android:inputType="textPassword"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="10dp" />

<Button

android:id="@+id/btnLogin"

android:text="Login"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginTop="20dp" />

<TextView

android:id="@+id/tvStatus"

android:text=""

android:textSize="18sp"

android:textColor="#008000"

android:layout\_marginTop="30dp"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content" />

</LinearLayout>

1. **Java Code (MainActivity.java)**

public class MainActivity extends AppCompatActivity {

EditText etUsername, etPassword;

Button btnLogin;

TextView tvStatus;

final String validUsername = "admin";

final String validPassword = "1234";

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

etUsername = findViewById(R.id.etUsername);

etPassword = findViewById(R.id.etPassword);

btnLogin = findViewById(R.id.btnLogin);

tvStatus = findViewById(R.id.tvStatus);

btnLogin.setOnClickListener(v -> {

String username = etUsername.getText().toString();

String password = etPassword.getText().toString();

if (username.equals(validUsername) && password.equals(validPassword)) {

tvStatus.setText("Login Successful");

Toast.makeText(this, "Welcome " + username, Toast.LENGTH\_SHORT).show();

} else {

Toast.makeText(this, "Login Failed", Toast.LENGTH\_SHORT).show();

tvStatus.setText("");

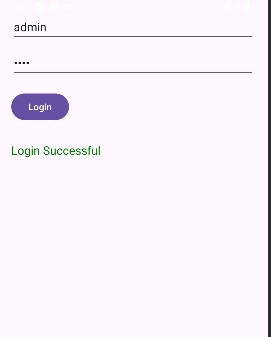
}

});

}

}

Output:



#### 7.Build an application to send an SMS or make a phone call using Native Actions with Intents.

Solution:

**Mainactivity.java**

package com.example.sms;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

import androidx.core.app.ActivityCompat;

import androidx.core.content.ContextCompat;

import android.content.BroadcastReceiver;

import android.content.Context;

import android.content.Intent;

import android.content.IntentFilter;

import android.content.pm.PackageManager;

import android.telephony.SmsManager;

import android.telephony.SmsMessage;

import android.view.View;

import android.widget.EditText;

import android.widget.TextView;

import android.widget.Toast;

import android.Manifest;

public class MainActivity extends AppCompatActivity {

private static final int SMS\_PERMISSION\_CODE = 101;

private EditText editTextPhoneNumber;

private EditText editTextMessage;

private TextView textViewReceivedMessages;

**@Override**

**protected void onCreate(Bundle savedInstanceState)** {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

editTextPhoneNumber = findViewById(R.id.editTextPhoneNumber);

editTextMessage = findViewById(R.id.editTextMessage);

textViewReceivedMessages = findViewById(R.id.textViewReceivedMessages);

// Request SMS permissions if not granted

if (!checkSMSPermission()) {

requestSMSPermission();

}

// Register a SMS **BroadcastReceiver** to listen for incoming SMS messages.

IntentFilter intentFilter = new IntentFilter();

intentFilter.addAction("android.provider.Telephony.SMS\_RECEIVED");

registerReceiver(smsReceiver, intentFilter);

}

**@Override**

**protected void onDestroy()** {

super.onDestroy();

unregisterReceiver(smsReceiver);

}

// Button click listener for sending SMS

**public void sendMessage(View view)** {

String phoneNumber =

editTextPhoneNumber.getText().toString().trim();

String message = editTextMessage.getText().toString();

**// Validates phone number.**

if (phoneNumber.isEmpty()) {

Toast.makeText(this, "Please enter a valid phone number",

Toast.LENGTH\_SHORT).show();

return;

}

try {

**//Sends SMS using Android's SmsManager is an** Android class that handles sending SMS messages

SmsManager smsManager = SmsManager.getDefault();

smsManager.sendTextMessage(phoneNumber, null, message, null, null);

Toast.makeText(this, "Message sent", Toast.LENGTH\_SHORT).show();

} catch (IllegalArgumentException e) {

Toast.makeText(this, "Invalid phone number format",

Toast.LENGTH\_SHORT).show();

} catch (Exception e) {

Toast.makeText(this, "Failed to send message",

Toast.LENGTH\_SHORT).show();

e.printStackTrace();

}

}

**// Check if SMS permission is granted**

**private boolean checkSMSPermission()** {

return ContextCompat.checkSelfPermission(this,

Manifest.permission.SEND\_SMS) == PackageManager.PERMISSION\_GRANTED;

}

**// Request SMS permission**

**private void requestSMSPermission()** {

ActivityCompat.requestPermissions(this, new

String[]{Manifest.permission.SEND\_SMS}, SMS\_PERMISSION\_CODE);

}

**// This receiver listens for SMS\_RECEIVED broadcasts**

**private final BroadcastReceiver smsReceiver = new BroadcastReceiver()** {

@Override

public void onReceive(Context context, Intent intent) {

**//Extracts SMS messages from the intent.**

Bundle bundle = intent.getExtras();

if (bundle != null) {

Object[] pdus = (Object[]) bundle.get("pdus");

if (pdus != null) {

for (Object pdu : pdus) {

**//Reads the sender's phone number and message content.**

SmsMessage smsMessage = SmsMessage.createFromPdu((byte[]) pdu);

String senderPhoneNumber = smsMessage.getDisplayOriginatingAddress();

String messageBody = smsMessage.getMessageBody();

**// Appends the received message to the TextView.**

textViewReceivedMessages.append("From: " +

senderPhoneNumber + "\n");

textViewReceivedMessages.append("Message: " +

messageBody + "\n\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:background="@color/white"

tools:context=".MainActivity">

<EditText

android:id="@+id/editTextPhoneNumber"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginStart="16dp"

android:layout\_marginTop="16dp"

android:layout\_marginEnd="16dp"

android:layout\_marginBottom="16dp"

android:autofillHints="Enter phone number"

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/editTextPhoneNumber"

android:layout\_marginStart="16dp"

android:layout\_marginTop="16dp"

android:layout\_marginEnd="16dp"

android:layout\_marginBottom="16dp"

android:autofillHints="Enter Message"

android:hint="Enter Message" />

<Button

android:id="@+id/buttonSend"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Send"

android:layout\_below="@id/editTextMessage"

android:layout\_alignParentEnd="true"

android:layout\_marginEnd="16dp"

android:onClick="sendMessage"

tools:ignore="HardcodedText,UsingOnClickInXml" />

<TextView

android:id="@+id/textViewReceivedMessages"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_below="@id/buttonSend"

android:layout\_marginStart="16dp"

android:layout\_marginTop="16dp"

android:layout\_marginEnd="16dp"

android:layout\_marginBottom="16dp"

android:textColor="@color/black" />

</RelativeLayout>

AndroidManifest.xml File

<?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-feature

android:name="android.hardware.telephony"

android:required="false" />

<uses-permission android:name="android.permission.SEND\_SMS"/>

<uses-permission android:name="android.permission.RECEIVE\_SMS"/>

<uses-permission android:name="android.permission.READ\_SMS"/>

<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.SMS"

tools:targetApi="31">

<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>

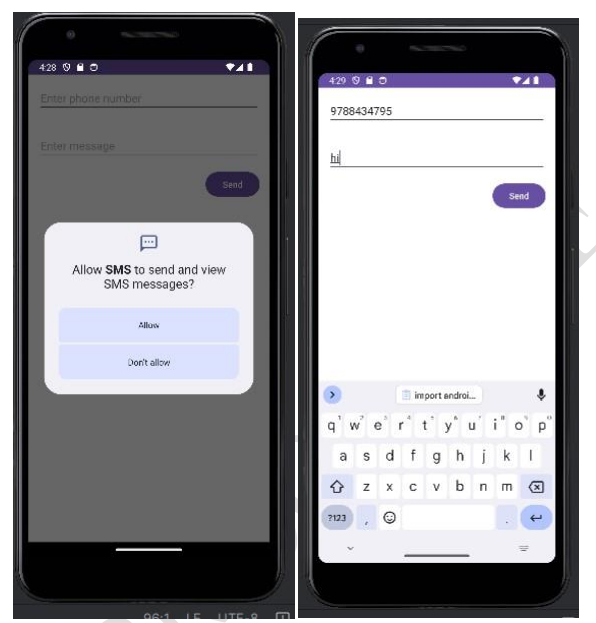
</manifest>

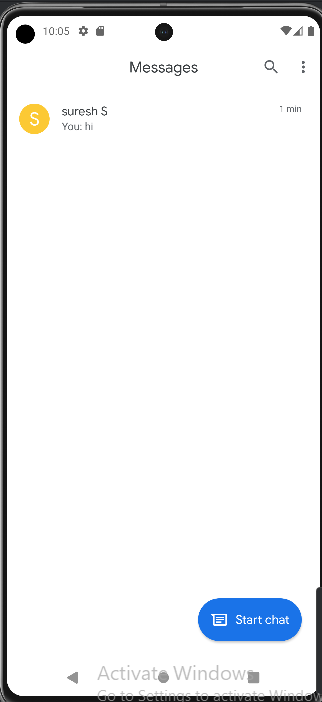
**Output:**

1.Goto Emulator Contact -> Store some contacts

2. Send sms to those contacts

3. Goto message in emulator and check sms





Program 14

9.Create a multi-screen application with communication between Fragments and Activities.

**Solution:**

The app has:

* **Fragment A** with an EditText and Button.
* When the Button is clicked, the entered text is sent to **MainActivity**.
* MainActivity receives the data and passes it to **Fragment B** to display it.

**Project Structure**

* MainActivity.java
* SenderFragment.java (Fragment A)
* ReceiverFragment.java (Fragment B)
* XML Layouts for each

**Key Concepts Demonstrated:**

| **Concept** | **Use** |
| --- | --- |
| **Fragments** | Modular screens inside activity |
| **Interface Callback** | To communicate Fragment → Activity |
| **Activity Logic** | Transfers data between fragments |

MainActivity.Java

package com.example.fra;

import android.os.Bundle;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity implements SenderFragment.OnDataPass {

ReceiverFragment receiverFragment;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

// Add SenderFragment

getSupportFragmentManager().beginTransaction()

.replace(R.id.fragmentContainerA, new SenderFragment())

.commit();

// Add ReceiverFragment

receiverFragment = new ReceiverFragment();

getSupportFragmentManager().beginTransaction()

.replace(R.id.fragmentContainerB, receiverFragment)

.commit();

}

// Interface method called from SenderFragment

@Override

public void onDataPass(String data) {

receiverFragment.updateText(data);

}

}

**ReceiverFragment.java**

package com.example.fra;

import android.os.Bundle;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

import android.widget.\*;

import androidx.annotation.NonNull;

import androidx.annotation.Nullable;

import androidx.fragment.app.Fragment;

public class ReceiverFragment extends Fragment {

TextView textView;

@Nullable

@Override

public View onCreateView(@NonNull LayoutInflater inflater,

@Nullable ViewGroup container,

@Nullable Bundle savedInstanceState) {

View view = inflater.inflate(R.layout.fragment\_receiver, container, false);

textView = view.findViewById(R.id.tvResult);

return view;

}

public void updateText(String data) {

if (textView != null) {

textView.setText(data);

}

}

}

**SenderFragment.Java**

package com.example.fra;

import android.content.Context;

import android.os.Bundle;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

import android.widget.\*;

import androidx.annotation.NonNull;

import androidx.annotation.Nullable;

import androidx.fragment.app.Fragment;

public class SenderFragment extends Fragment {

EditText editText;

Button button;

OnDataPass dataPasser;

public interface OnDataPass {

void onDataPass(String data);

}

@Override

public void onAttach(@NonNull Context context) {

super.onAttach(context);

dataPasser = (OnDataPass) context;

}

@Nullable

@Override

public View onCreateView(@NonNull LayoutInflater inflater,

@Nullable ViewGroup container,

@Nullable Bundle savedInstanceState) {

View view = inflater.inflate(R.layout.fragment\_sender, container, false);

editText = view.findViewById(R.id.etInput);

button = view.findViewById(R.id.btnSend);

button.setOnClickListener(v -> {

String text = editText.getText().toString();

dataPasser.onDataPass(text);

});

return view;

}

}

**activity\_main.xml**

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

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:orientation="horizontal">

<FrameLayout

android:id="@+id/fragmentContainerA"

android:layout\_width="0dp"

android:layout\_weight="1"

android:layout\_height="match\_parent"/>

<FrameLayout

android:id="@+id/fragmentContainerB"

android:layout\_width="0dp"

android:layout\_weight="1"

android:layout\_height="match\_parent"/>

</LinearLayout>

**fragment\_receiver.xml**

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

<FrameLayout 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/tvResult"

android:text="Waiting for data..."

android:layout\_gravity="center"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:textSize="18sp"/>

</FrameLayout>

**fragment\_sender.xml**

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

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:orientation="vertical"

android:padding="16dp">

<EditText

android:id="@+id/etInput"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:hint="Enter something" />

<Button

android:id="@+id/btnSend"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Send to Fragment B" />

</LinearLayout>