LAB PROGRAM 3

Create an application to develop Login window using UI controls.

MainActivity.java file code:

package com.example.labprog;

import android.os.Bundle;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

}

}

MANIFEST file code:

<?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:icon="@mipmap/ic\_launcher"

android:label="@string/app\_name"

android:roundIcon="@mipmap/ic\_launcher\_round"

android:supportsRtl="true">

<activity

android:name=".MainActivity"

android:theme="@style/Theme.AppCompat.Light.NoActionBar"

android:exported="true">

<intent-filter>

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

<category android:name="android.intent.category.LAUNCHER" />

</intent-filter>

</activity>

</application>

</manifest>

LAYOUT File :

<?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/textView"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="This is the main activity"

android:textSize="20sp"

android:layout\_centerInParent="true"/>

</RelativeLayout>

LAB PROGRAM 4

Create an application to implement new activity using explicit intent, implicit intent and content provider.

**Java file**

import android.content.Intent;

import android.database.Cursor;

import android.net.Uri;

import android.os.Bundle;

import android.provider.ContactsContract;

import android.view.View;

import android.widget.Button;

import android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

private static final int PICK\_CONTACT\_REQUEST = 1;

private TextView contactTextView;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

contactTextView = findViewById(R.id.contact\_text\_view);

Button explicitButton = findViewById(R.id.explicit\_button);

explicitButton.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

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

startActivity(explicitIntent);

}

});

Button implicitButton = findViewById(R.id.implicit\_button);

implicitButton.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

Intent implicitIntent = new Intent(Intent.ACTION\_VIEW, Uri.parse("https://www.example.com"));

startActivity(implicitIntent);

}

});

Button contentProviderButton = findViewById(R.id.content\_provider\_button);

contentProviderButton.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

openContacts();

}

});

}

private void openContacts() {

Intent pickContactIntent = new Intent(Intent.ACTION\_PICK, ContactsContract.Contacts.CONTENT\_URI);

startActivityForResult(pickContactIntent, PICK\_CONTACT\_REQUEST);

}

@Override

protected void onActivityResult(int requestCode, int resultCode, Intent data) {

super.onActivityResult(requestCode, resultCode, data);

if (requestCode == PICK\_CONTACT\_REQUEST && resultCode == RESULT\_OK) {

Uri contactUri = data.getData();

Cursor cursor = getContentResolver().query(contactUri, null, null, null, null);

if (cursor != null && cursor.moveToFirst()) {

int nameIndex = cursor.getColumnIndex(ContactsContract.Contacts.DISPLAY\_NAME);

String contactName = cursor.getString(nameIndex);

contactTextView.setText("Contact Name: " + contactName);

cursor.close();

}

}

}

}

**Layout.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/explicit\_button"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Explicit Intent"

android:layout\_marginTop="16dp"/>

<Button

android:id="@+id/implicit\_button"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Implicit Intent"

android:layout\_below="@id/explicit\_button"

android:layout\_marginTop="16dp"/>

<Button

android:id="@+id/content\_provider\_button"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Content Provider"

android:layout\_below="@id/implicit\_button"

android:layout\_marginTop="16dp"/>

<TextView

android:id="@+id/contact\_text\_view"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_below="@id/content\_provider\_button"

android:layout\_marginTop="16dp"/>

</RelativeLayout>

**Create new java class file name SecondActivity.java**

import android.os.Bundle;

import androidx.appcompat.app.AppCompatActivity;

public class SecondActivity extends AppCompatActivity {

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_second);

}

}

**Create new activity main .xml file under layout folder name it as activity\_second.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=".SecondActivity">

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Second Activity"

android:textSize="24sp"

android:layout\_centerInParent="true"/>

</RelativeLayout>

LAB PROGRAM 5

**Create an application that displays custom designed Opening Screen.**

Java file :

package com.example.myapp;

import android.os.Bundle;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.opening\_screen\_layout);

}

}

AndroidManifest.xml

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

package="com.example.myapp">

<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 "

android:theme="@style/Theme.AppCompat.Light.NoActionBar">

<intent-filter>

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

<category android:name="android.intent.category.LAUNCHER" />

</intent-filter>

</activity>

</application>

</manifest>

Layout file :

<!-- 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/textView"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="This is the main activity"

android:textSize="20sp"

android:layout\_centerInParent="true"/>

</RelativeLayout>

LAB PROGRAM 6

**Create an UI with all views**.

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">

<TextView

android:id="@+id/textView"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="TextView"

android:textSize="18sp"

android:textStyle="bold"

android:layout\_gravity="center\_horizontal"

android:layout\_marginBottom="16dp"/>

<EditText

android:id="@+id/editText"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:hint="EditText"

android:layout\_marginBottom="16dp"/>

<Button

android:id="@+id/button"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Button"

android:layout\_gravity="center\_horizontal"

android:layout\_marginBottom="16dp"/>

<ImageView

android:id="@+id/imageView"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:src="@drawable/ic\_launcher\_foreground"

android:layout\_marginBottom="16dp"/>

<CheckBox

android:id="@+id/checkBox"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="CheckBox"

android:layout\_marginBottom="16dp"/>

<RadioButton

android:id="@+id/radioButton"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="RadioButton"

android:layout\_marginBottom="16dp"/>

<ToggleButton

android:id="@+id/toggleButton"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="ToggleButton"

android:layout\_gravity="center\_horizontal"

android:layout\_marginBottom="16dp"/>

<SeekBar

android:id="@+id/seekBar"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginBottom="16dp"/>

<RatingBar

android:id="@+id/ratingBar"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_gravity="center\_horizontal"

android:layout\_marginBottom="16dp"/>

<ProgressBar

android:id="@+id/progressBar"

style="?android:attr/progressBarStyleHorizontal"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginBottom="16dp"/>

<Switch

android:id="@+id/switch"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Switch"

android:layout\_gravity="center\_horizontal"

android:layout\_marginBottom="16dp"/>

<Spinner

android:id="@+id/spinner"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginBottom="16dp"/>

</LinearLayout>

**LAB PROGRAM 8**

Create menu in Application

**Create new project , Select Basic Views Activity option**

**The program will be auto generated**

**LAB PROGRAM 9**

Read/ write the Local data.

**Java file**

import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import android.widget.EditText;

import android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;

import java.io.BufferedReader;

import java.io.FileOutputStream;

import java.io.FileReader;

import java.io.IOException;

public class MainActivity extends AppCompatActivity {

private EditText editTextInput;

private TextView textContent;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

editTextInput = findViewById(R.id.edit\_text\_input);

textContent = findViewById(R.id.text\_content);

Button buttonWrite = findViewById(R.id.button\_write);

buttonWrite.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

String inputText = editTextInput.getText().toString();

if (!inputText.isEmpty()) {

writeFile("example.txt", inputText);

editTextInput.setText("");

displayFileContent("example.txt");

}

}

});

// Display initial content of the file

displayFileContent("example.txt");

}

private void writeFile(String filename, String data) {

try {

FileOutputStream fos = openFileOutput(filename, MODE\_PRIVATE);

fos.write(data.getBytes());

fos.close();

} catch (IOException e) {

e.printStackTrace();

}

}

private void displayFileContent(String filename) {

StringBuilder content = new StringBuilder();

try {

BufferedReader br = new BufferedReader(new FileReader(getFilesDir() + "/" + filename));

String line;

while ((line = br.readLine()) != null) {

content.append(line);

}

br.close();

} catch (IOException e) {

e.printStackTrace();

}

textContent.setText(content.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">

<EditText

android:id="@+id/edit\_text\_input"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:hint="Enter text to write to file"/>

<Button

android:id="@+id/button\_write"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_below="@id/edit\_text\_input"

android:layout\_marginTop="8dp"

android:text="Write to File"/>

<TextView

android:id="@+id/text\_content"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_below="@id/button\_write"

android:layout\_marginTop="16dp"

android:textSize="18sp"

android:text="File Content will be displayed here" />

</RelativeLayout>

LAB PROGRAM 11

**Create an application to send SMS and receive SMS**

**JAVA FILE**

**import android.Manifest;**

**import android.content.BroadcastReceiver;**

**import android.content.Context;**

**import android.content.Intent;**

**import android.content.IntentFilter;**

**import android.content.pm.PackageManager;**

**import android.os.Bundle;**

**import android.telephony.SmsManager;**

**import android.telephony.SmsMessage;**

**import android.util.Log;**

**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;**

**import androidx.core.app.ActivityCompat;**

**import androidx.core.content.ContextCompat;**

**public class MainActivity extends AppCompatActivity {**

**private static final int MY\_PERMISSIONS\_REQUEST\_SEND\_SMS = 1;**

**private static final String SMS\_RECEIVED = "android.provider.Telephony.SMS\_RECEIVED";**

**private EditText editTextPhoneNumber;**

**private EditText editTextMessage;**

**private TextView textViewReceivedMessage;**

**private BroadcastReceiver receiver;**

**@Override**

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

**super.onCreate(savedInstanceState);**

**setContentView(R.layout.activity\_main);**

**editTextPhoneNumber = findViewById(R.id.edit\_text\_phone\_number);**

**editTextMessage = findViewById(R.id.edit\_text\_message);**

**textViewReceivedMessage = findViewById(R.id.text\_view\_received\_message);**

**Button buttonSend = findViewById(R.id.button\_send);**

**buttonSend.setOnClickListener(new View.OnClickListener() {**

**@Override**

**public void onClick(View v) {**

**sendSMS();**

**}**

**});**

**if (ContextCompat.checkSelfPermission(this, Manifest.permission.RECEIVE\_SMS) != PackageManager.PERMISSION\_GRANTED) {**

**ActivityCompat.requestPermissions(this, new String[]{Manifest.permission.RECEIVE\_SMS}, MY\_PERMISSIONS\_REQUEST\_SEND\_SMS);**

**}**

**receiver = new BroadcastReceiver() {**

**@Override**

**public void onReceive(Context context, Intent intent) {**

**if (intent.getAction() != null && intent.getAction().equals(SMS\_RECEIVED)) {**

**Bundle bundle = intent.getExtras();**

**if (bundle != null) {**

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

**if (pdus != null) {**

**for (Object pdu : pdus) {**

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

**String sender = smsMessage.getDisplayOriginatingAddress();**

**String message = smsMessage.getMessageBody();**

**textViewReceivedMessage.setText("From: " + sender + "\nMessage: " + message);**

**}**

**}**

**}**

**}**

**}**

**};**

**registerReceiver(receiver, new IntentFilter(SMS\_RECEIVED));**

**}**

**@Override**

**protected void onDestroy() {**

**super.onDestroy();**

**unregisterReceiver(receiver);**

**}**

**private void sendSMS() {**

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

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

**if (ContextCompat.checkSelfPermission(this, Manifest.permission.SEND\_SMS) != PackageManager.PERMISSION\_GRANTED) {**

**ActivityCompat.requestPermissions(this, new String[]{Manifest.permission.SEND\_SMS}, MY\_PERMISSIONS\_REQUEST\_SEND\_SMS);**

**} else {**

**try {**

**SmsManager smsManager = SmsManager.getDefault();**

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

**Toast.makeText(this, "SMS sent successfully", Toast.LENGTH\_SHORT).show();**

**} catch (Exception e) {**

**Log.e("Send SMS", "Sending SMS failed", e);**

**Toast.makeText(this, "SMS sending failed", Toast.LENGTH\_SHORT).show();**

**}**

**}**

**}**

**}**

**ACTIVITY\_MANIFEST.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.INTERNET" />

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

<uses-permission android:name="android.permission.WRITE\_EXTERNAL\_STORAGE" />

<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.Lab9"

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>

**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"

android:padding="16dp">

<EditText

android:id="@+id/edit\_text\_to"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:hint="To" />

<EditText

android:id="@+id/edit\_text\_subject"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="8dp"

android:hint="Subject" />

<EditText

android:id="@+id/edit\_text\_message"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="8dp"

android:hint="Message"

android:inputType="textMultiLine"

android:minLines="4" />

<Button

android:id="@+id/button\_send"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_gravity="end"

android:layout\_marginTop="16dp"

android:text="Send Email" />

</LinearLayout>

LAB PROGRAM 12

Create an application to send an e-mail.

Java File

import android.content.Intent;

import android.net.Uri;

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 {

private EditText editTextTo;

private EditText editTextSubject;

private EditText editTextMessage;

private Button buttonSend;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

editTextTo = findViewById(R.id.edit\_text\_to);

editTextSubject = findViewById(R.id.edit\_text\_subject);

editTextMessage = findViewById(R.id.edit\_text\_message);

buttonSend = findViewById(R.id.button\_send);

buttonSend.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

sendEmail();

}

});

}

private void sendEmail() {

String recipientEmail = editTextTo.getText().toString().trim();

String subject = editTextSubject.getText().toString().trim();

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

Intent intent = new Intent(Intent.ACTION\_SENDTO);

intent.setData(Uri.parse("mailto:"));

intent.putExtra(Intent.EXTRA\_EMAIL, new String[]{recipientEmail});

intent.putExtra(Intent.EXTRA\_SUBJECT, subject);

intent.putExtra(Intent.EXTRA\_TEXT, message);

try {

startActivity(Intent.createChooser(intent, "Send mail..."));

} catch (android.content.ActivityNotFoundException ex) {

Toast.makeText(MainActivity.this, "There are no email clients installed.", Toast.LENGTH\_SHORT).show();

}

}

}

ACTIVITY\_MANIFEST.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.INTERNET" />

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

<uses-permission android:name="android.permission.WRITE\_EXTERNAL\_STORAGE" />

<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.Lab9"

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>

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"

android:padding="16dp">

<EditText

android:id="@+id/edit\_text\_to"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:hint="To" />

<EditText

android:id="@+id/edit\_text\_subject"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="8dp"

android:hint="Subject" />

<EditText

android:id="@+id/edit\_text\_message"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="8dp"

android:hint="Message"

android:inputType="textMultiLine"

android:minLines="4" />

<Button

android:id="@+id/button\_send"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_gravity="end"

android:layout\_marginTop="16dp"

android:text="Send Email" />

</LinearLayout>

LAB PROGRAM 14

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”

**JAVA FILE**

import android.os.Bundle;

import android.support.v7.app.AppCompatActivity;

import android.view.View;

import android.widget.Button;

import android.widget.EditText;

import android.widget.TextView;

import android.widget.Toast;

public class LoginActivity extends AppCompatActivity {

private EditText usernameEditText;

private EditText passwordEditText;

private Button loginButton;

private TextView statusTextView;

// Sample username and password, replace with your authentication mechanism

private static final String CORRECT\_USERNAME = "user";

private static final String CORRECT\_PASSWORD = "password";

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_login);

usernameEditText = findViewById(R.id.usernameEditText);

passwordEditText = findViewById(R.id.passwordEditText);

loginButton = findViewById(R.id.loginButton);

statusTextView = findViewById(R.id.statusTextView);

loginButton.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

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

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

if (username.equals(CORRECT\_USERNAME) && password.equals(CORRECT\_PASSWORD)) {

// Successful login

statusTextView.setText("Login Successful");

} else {

// Login failed

Toast.makeText(LoginActivity.this, "Login Failed", Toast.LENGTH\_SHORT).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=".LoginActivity">

<EditText

android:id="@+id/usernameEditText"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:hint="Username"

android:layout\_margin="16dp"/>

<EditText

android:id="@+id/passwordEditText"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

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

android:hint="Password"

android:inputType="textPassword"

android:layout\_margin="16dp"/>

<Button

android:id="@+id/loginButton"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

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

android:layout\_centerHorizontal="true"

android:text="Login"

android:layout\_marginTop="16dp"/>

<TextView

android:id="@+id/statusTextView"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

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

android:layout\_centerHorizontal="true"

android:text=""

android:textSize="20sp"

android:layout\_marginTop="16dp"/>

</RelativeLayout>

LAB PROGRAM 13

Display Map based on the Current/given location.

1. Create a new project in Android Studio.
2. Add the Google Maps dependency in your build.gradle file

dependencies {

implementation 'com.google.android.gms:play-services-maps:18.0.2'

}

1. Go to the Google Cloud Console and create a new project or select an existing one.
2. Enable the Maps SDK for Android.
3. Generate an API key and restrict it to your app.
4. Add the API key to your AndroidManifest.xml:

<application>

<!-- Other elements -->

<meta-data

android:name="com.google.android.geo.API\_KEY"

android:value="YOUR\_API\_KEY"/>

</application>

1. Create an XML layout file activity\_maps.xml with a MapFragment:

<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=".MapsActivity">

<fragment

android:id="@+id/map"

android:name="com.google.android.gms.maps.SupportMapFragment"

android:layout\_width="match\_parent"

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

</RelativeLayout>

1. Create a new Activity MapsActivity.java:

package com.example.yourapp;

import android.location.Location;

import android.os.Bundle;

import androidx.annotation.NonNull;

import androidx.annotation.Nullable;

import androidx.core.app.ActivityCompat;

import androidx.fragment.app.FragmentActivity;

import com.google.android.gms.location.FusedLocationProviderClient;

import com.google.android.gms.location.LocationServices;

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;

import com.google.android.gms.tasks.OnCompleteListener;

import com.google.android.gms.tasks.Task;

public class MapsActivity extends FragmentActivity implements OnMapReadyCallback {

private GoogleMap mMap;

private FusedLocationProviderClient fusedLocationProviderClient;

private final LatLng defaultLocation = new LatLng(-33.8523341, 151.2106085);

private static final int DEFAULT\_ZOOM = 15;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_maps);

SupportMapFragment mapFragment = (SupportMapFragment) getSupportFragmentManager()

.findFragmentById(R.id.map);

mapFragment.getMapAsync(this);

fusedLocationProviderClient = LocationServices.getFusedLocationProviderClient(this);

}

@Override

public void onMapReady(GoogleMap googleMap) {

mMap = googleMap;

getLocationPermission();

}

private void getLocationPermission() {

if (ActivityCompat.checkSelfPermission(this, android.Manifest.permission.ACCESS\_FINE\_LOCATION)

!= android.content.pm.PackageManager.PERMISSION\_GRANTED && ActivityCompat.checkSelfPermission(this,

android.Manifest.permission.ACCESS\_COARSE\_LOCATION) != android.content.pm.PackageManager.PERMISSION\_GRANTED) {

ActivityCompat.requestPermissions(this,

new String[]{android.Manifest.permission.ACCESS\_FINE\_LOCATION}, 1);

} else {

getDeviceLocation();

}

}

@Override

public void onRequestPermissionsResult(int requestCode,

@NonNull String[] permissions,

@NonNull int[] grantResults) {

super.onRequestPermissionsResult(requestCode, permissions, grantResults);

if (requestCode == 1) {

if (grantResults.length > 0

&& grantResults[0] == android.content.pm.PackageManager.PERMISSION\_GRANTED) {

getDeviceLocation();

}

}

}

private void getDeviceLocation() {

try {

if (ActivityCompat.checkSelfPermission(this, android.Manifest.permission.ACCESS\_FINE\_LOCATION)

== android.content.pm.PackageManager.PERMISSION\_GRANTED) {

Task<Location> locationResult = fusedLocationProviderClient.getLastLocation();

locationResult.addOnCompleteListener(this, new OnCompleteListener<Location>() {

@Override

public void onComplete(@NonNull Task<Location> task) {

if (task.isSuccessful()) {

Location lastKnownLocation = task.getResult();

if (lastKnownLocation != null) {

LatLng currentLocation = new LatLng(lastKnownLocation.getLatitude(), lastKnownLocation.getLongitude());

mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(currentLocation, DEFAULT\_ZOOM));

mMap.addMarker(new MarkerOptions().position(currentLocation).title("Current Location"));

}

} else {

mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(defaultLocation, DEFAULT\_ZOOM));

mMap.addMarker(new MarkerOptions().position(defaultLocation).title("Default Location"));

}

}

});

}

} catch (SecurityException e) {

e.printStackTrace();

}

}

}

Lab Program 10

**Create / Read / Write data with database (SQLite).**

**1.Create a Database Helper Class**

**Right-click on your package directory (usually under app/src/main/java/com/example/yourapp).**

**Select New > Java Class.**

**Name it DatabaseHelper.**

**DatabaseHelper class code .**

**package com.example.yourapp;**

**import android.content.Context;**

**import android.database.sqlite.SQLiteDatabase;**

**import android.database.sqlite.SQLiteOpenHelper;**

**import android.database.Cursor;**

**import android.content.ContentValues;**

**public class DatabaseHelper extends SQLiteOpenHelper {**

**// Database constants**

**private static final String DATABASE\_NAME = "example.db";**

**private static final int DATABASE\_VERSION = 1;**

**// Table and columns**

**private static final String TABLE\_NAME = "messages";**

**private static final String COLUMN\_ID = "id";**

**private static final String COLUMN\_NAME = "name";**

**private static final String COLUMN\_MESSAGE = "message";**

**// Constructor**

**public DatabaseHelper(Context context) {**

**super(context, DATABASE\_NAME, null, DATABASE\_VERSION);**

**}**

**// onCreate method to create the table**

**@Override**

**public void onCreate(SQLiteDatabase db) {**

**String CREATE\_TABLE = "CREATE TABLE " + TABLE\_NAME + " ("**

**+ COLUMN\_ID + " INTEGER PRIMARY KEY AUTOINCREMENT, "**

**+ COLUMN\_NAME + " TEXT, "**

**+ COLUMN\_MESSAGE + " TEXT)";**

**db.execSQL(CREATE\_TABLE);**

**}**

**// onUpgrade method to handle database version upgrades**

**@Override**

**public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {**

**db.execSQL("DROP TABLE IF EXISTS " + TABLE\_NAME);**

**onCreate(db);**

**}**

**// Insert data into the database**

**public boolean insertData(String name, String message) {**

**SQLiteDatabase db = this.getWritableDatabase();**

**ContentValues contentValues = new ContentValues();**

**contentValues.put(COLUMN\_NAME, name);**

**contentValues.put(COLUMN\_MESSAGE, message);**

**long result = db.insert(TABLE\_NAME, null, contentValues);**

**db.close();**

**return result != -1; // returns false if data is not inserted**

**}**

**// Retrieve data from the database**

**public Cursor getData() {**

**SQLiteDatabase db = this.getReadableDatabase();**

**String query = "SELECT \* FROM " + TABLE\_NAME;**

**Cursor cursor = db.rawQuery(query, null);**

**return cursor;**

**}**

**}**

**Activity main. XML Layout (res/layout/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"**

**android:padding="16dp">**

**<EditText**

**android:id="@+id/name"**

**android:layout\_width="match\_parent"**

**android:layout\_height="wrap\_content"**

**android:hint="Your Name" />**

**<EditText**

**android:id="@+id/message"**

**android:layout\_width="match\_parent"**

**android:layout\_height="wrap\_content"**

**android:hint="Your Message" />**

**<Button**

**android:id="@+id/submit"**

**android:layout\_width="wrap\_content"**

**android:layout\_height="wrap\_content"**

**android:text="Send" />**

**<Button**

**android:id="@+id/view\_messages"**

**android:layout\_width="wrap\_content"**

**android:layout\_height="wrap\_content"**

**android:text="View Messages" />**

**</LinearLayout>**

**MAINACTIVITY.java**

**package com.example.yourapp;**

**import android.database.Cursor;**

**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 {**

**DatabaseHelper dbHelper;**

**EditText nameEditText, messageEditText;**

**Button submitButton, viewMessagesButton;**

**@Override**

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

**super.onCreate(savedInstanceState);**

**setContentView(R.layout.activity\_main);**

**dbHelper = new DatabaseHelper(this);**

**nameEditText = findViewById(R.id.name);**

**messageEditText = findViewById(R.id.message);**

**submitButton = findViewById(R.id.submit);**

**viewMessagesButton = findViewById(R.id.view\_messages);**

**submitButton.setOnClickListener(new View.OnClickListener() {**

**@Override**

**public void onClick(View v) {**

**String name = nameEditText.getText().toString();**

**String message = messageEditText.getText().toString();**

**if (name.isEmpty() || message.isEmpty()) {**

**Toast.makeText(MainActivity.this, "Please fill all fields", Toast.LENGTH\_SHORT).show();**

**} else {**

**boolean isInserted = dbHelper.insertData(name, message);**

**if (isInserted) {**

**Toast.makeText(MainActivity.this, "Message Sent", Toast.LENGTH\_SHORT).show();**

**} else {**

**Toast.makeText(MainActivity.this, "Message Sending Failed", Toast.LENGTH\_SHORT).show();**

**}**

**}**

**}**

**});**

**viewMessagesButton.setOnClickListener(new View.OnClickListener() {**

**@Override**

**public void onClick(View v) {**

**Cursor cursor = dbHelper.getData();**

**if (cursor.getCount() == 0) {**

**Toast.makeText(MainActivity.this, "No Messages Found", Toast.LENGTH\_SHORT).show();**

**return;**

**}**

**StringBuilder messages = new StringBuilder();**

**while (cursor.moveToNext()) {**

**messages.append("ID: ").append(cursor.getInt(0)).append("\n");**

**messages.append("Name: ").append(cursor.getString(1)).append("\n");**

**messages.append("Message: ").append(cursor.getString(2)).append("\n\n");**

**}**

**Toast.makeText(MainActivity.this, messages.toString(), Toast.LENGTH\_LONG).show();**

**}**

**});**

**}**

**}**

**Labprogram 15**

**Learn to deploy Android applications.**

**Step 1: Make a Developer Account**

**A developer account is must be needed to upload an app on the Google Play Store, and the process is very simple. Just go through Google Play Store and do as instructed.**

**The account can be created in four simple steps:**

**Sign-In with Your Google Account**

**Accept Terms**

**Pay Registration Fee of $25.**

**Complete Your Account Details**

**Step 2: After you completed step 1 you will be redirected to this page where you have to click on the CREATE APPLICATION button.**

**Once you click on it a pop up will be shown like this where you have to choose your Default language and Title of your app. Then click on the CREATE button.**

**Step 3: Store listing**

**After you completed step 2 you will be redirected to this page where you have to provide the Short description and Full description of your App.**

**Then you scroll down the page and now you have to add the Hi-res icon of your app.**

**Then you have to provide the Screenshots of your app.**

**Ant next thing you have to provide is the Feature Graphic of your app. Note that this graphic is then used everywhere your app is featured on Google Play.**

**Then come to Categorization part where you have to provide your Application type and Category of your app.**

**Then come to Contact details part where you have to provide your Website(if any), email, and Phone of yours.**

**And finally when you click on SAVE DRAFT button you can see that Store listing tab is now become turned to green and you are done for Store listing.**

**Step 4: App release**

**After completing step 3 go to App releases then scroll down to Production track and click on MANAGE button.**

**After redirecting to the next page click on the CREATE RELEASE button.**

**After that on the next page, you have to upload your APK file in Android App Bundles and APKs to add section.**

**After that simply click on the SAVE button.**

**Step 5: Content rating**

**Now after completing step 4 go to Content rating and click on CONTINUE button.**

**And then Select your app category.**

**After selecting your app category make sure that you read all of these and answer them correctly.**

**And after answering them correctly don’t forget to click on SAVE QUESTIONNAIRE button.**

**Once you saved all those things then click on CALCULATE RATING button.**

**When you redirected to another page scroll down and click on APPLY RATING button. And you are done for Content rating section. Don’t forget to notice that Content rating section is now become turned to green.**

**Step 6: Pricing & distribution**

**Then go to the Pricing & distribution section. Then select the country in which you want to available your app.**

**Then go down and down and check out the Content guidelines and US export laws section by marking them tick mark. And click on the SAVE DRAFT button. Don’t forget to notice that Pricing & distribution section is now become turned to green tick.**

**Step 7: App content**

**Then come to the App content section. And in the Privacy policy section click on the Start button.**

**And then provide a valid Privacy policy URL. Note that google will check this.**

**Then go back and continue further steps by clicking start button in Ads section.**

**Then select does your app contain ads or not? And click on SAVE button.**

**Then again go back and continue further steps by clicking start button in Target audience and content section.**

**In the next page select the Target age group and scroll down and click on the Next button.**

**Then check the Appeal to children section. And click on the Next button.**

**On the next page click on the Save button and you are done for App content section.**

**Step 8: App releases**

**Again go back to the App releases section. And in the Production track click on the EDIT RELEASE button.**

**Then on the next page go down and down and click on the REVIEW button.**

**And finally, on the next page click on the START ROLLOUT TO PRODUCTION button to send your app to review. And you are finally done.**

**After usually 4 to 5 days they will review your app and let you know to either approve or reject your app.**