

GURU GOBIND SINGH INDRAPRASTHA UNIVERSITY

(University School Of Information Communication & Technology)



MOBILE NETWORK PROGRAMMING LAB IT-765

SUBMITTED TO

Dr. M. Bala Krishna
Assistant Professor

SUBMITTED BY

ISHA
Enroll No. 01216404521
MCA(SE)
3rd Semester

TABLE OF CONTENTS

S.No.	Name of Experiment	Date	Teacher's Sign
1	Layouts: Linear, Relative, Table, Grid View Identify User Centric Design Process Features	19/10/22	
2	Asynchronous tasks in Android. Program to Illustrate Multithreading Identify Collaborative UCD process features	21/10/22	
3	Simple Calculator Application using Android Identify Multimodality features	22/10/22	
4	Program to Illustrate: Privacy, Trust and Group Communication Identify Group Awareness Function	23/10/22	
5	Android Native Application that uses GPS Location Info. Identify SOA for Mobile Services	25/10/22	
6	Program for Phone Dialer with CALL and SAVE Buttons. Identify the Personalization features.	26/10/22	
7	Program to Create Address Book and Contacts. Identify Context Management Framework features	28/10/22	
8	Program to Illustrate: Send SMS, Receive SMS, Send e-mail and Receive e-mail. Identify Service Discovery Framework	29/10/22	
9	Program to Register for the App using SQLite & MySQL Database. Program to Login to The App using Databases. Identify Mobile Web Services	30/10/22	
10	Create Student or Employee Database. WAP to Read, Write & Delete From SQLite Database. Identify Agent Security Development	30/10/22	
11	Program to Show How to Make a SOCKET Connection from a J2ME Phone. Show How to Make a SOCKET Connection from the Phone To Port 80.	09/12/22	
12	Program to Illustrate Mobile Agents & Services Offered by Agents.	15/12/22	

Ques 1.Layouts: Linear, Relative, Table, Grid View & Identify User Centric Design Process Features

LINEAR 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:paddingLeft="20dp"
    android:paddingRight="20dp"
    android:orientation="vertical" >
    <EditText android:id="@+id/txtTo"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="To"/>
    <EditText android:id="@+id/txtSub"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Subject"/>
    <EditText android:id="@+id/txtMsg"
        android:layout_width="match_parent"
        android:layout_height="0dp"
        android:layout_weight="1"
        android:gravity="top"
        android:hint="Message"/>
    <Button android:layout_width="100dp"
        android:layout_height="wrap_content"
        android:layout_gravity="right"
        android:text="Send"/>
</LinearLayout>
```

MainActivity.java

```
package com.tutlane.linearlayout;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
public class MainActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }
}
```

RELATIVE VIEW

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingLeft="10dp"
    android:paddingRight="10dp">
    <Button android:id="@+id/btn1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentLeft="true"
        android:text="Button1" />
    <Button android:id="@+id/btn2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
```

```

        android:layout_alignParentRight="true"
        android:layout_centerVertical="true"
        android:text="Button2" />
<Button android:id="@+id/btn3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentLeft="true"
        android:layout_centerVertical="true"
        android:text="Button3" />
<Button android:id="@+id/btn4"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_alignParentBottom="true"
        android:text="Button4" />
<Button android:id="@+id/btn5"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignBottom="@+id/btn2"
        android:layout_centerHorizontal="true"
        android:text="Button5" />
<Button android:id="@+id/btn6"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_above="@+id/btn4"
        android:layout_centerHorizontal="true"
        android:text="Button6" />
<Button android:id="@+id/btn7"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_toEndOf="@+id/btn1"
        android:layout_toRightOf="@+id/btn1"
        android:layout_alignParentRight="true"
        android:text="Button7" />
</RelativeLayout>

```

MainActivity.java

```

package com.tutlane.linearlayout;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
public class MainActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }
}

```

TABLE VIEW

activity_main.xml

```

<?xml version="1.0" encoding="utf-8"?>
<TableLayout xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_marginTop="100dp"
        android:paddingLeft="10dp"
        android:paddingRight="10dp" >
    <TableRow android:background="#0079D6" android:padding="5dp">
        <TextView android:layout_width="wrap_content"
            android:layout_height="wrap_content"android:layout_weight="1"
            android:text="UserId" />
    </TableRow>
</TableLayout>

```

```

<TextView android:layout_width="wrap_content"
    android:layout_height="wrap_content"android:layout_weight="1"
    android:text="User Name" />
<TextView android:layout_width="wrap_content"
    android:layout_height="wrap_content"android:layout_weight="1"
    android:text="Location" />
</TableRow>
<TableRow android:background="#DAE8FC" android:padding="5dp">
    <TextView android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:text="1" />
    <TextView android:layout_width="wrap_content"
        android:layout_height="wrap_content" android:layout_weight="1"
        android:text="Suresh Dasari" />
    <TextView android:layout_width="wrap_content"
        android:layout_height="wrap_content"android:layout_weight="1"
        android:text="Hyderabad" />
</TableRow>
<TableRow android:background="#DAE8FC" android:padding="5dp">
    <TextView android:layout_width="wrap_content"
        android:layout_height="wrap_content" android:layout_weight="1"
        android:text="2" />
    <TextView android:layout_width="wrap_content"
        android:layout_height="wrap_content"android:layout_weight="1"
        android:text="Rohini Alavala" />
    <TextView android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_weight="1"android:text="Guntur" />
</TableRow>
<TableRow android:background="#DAE8FC" android:padding="5dp">
    <TextView android:layout_width="wrap_content"
        android:layout_height="wrap_content"android:layout_weight="1"
        android:text="3" />
    <TextView android:layout_width="wrap_content"
        android:layout_height="wrap_content"android:layout_weight="1"
        android:text="Trishika Dasari" />
    <TextView android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:text="Guntur" />
</TableRow>
</TableLayout>

```

MainActivity.java

```

package com.tutlane.linearlayout;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
public class MainActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }
}

```

GRID VIEW

activity_main.xml

```

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

```

```

android:id="@+id/gridview" android:layout_width="match_parent"
android:layout_height="match_parent"android:columnWidth="110dp"
android:numColumns="auto_fit"android:verticalSpacing="10dp"
android:horizontalSpacing="10dp"android:stretchMode="columnWidth"
android:gravity="center" />

```

ImageAdapter.java

```

package com.tutlane.gridview;
import android.content.Context;
import android.view.View;
import android.view.ViewGroup;
import android.widget.BaseAdapter;
import android.widget.GridView;
import android.widget.ImageView;
public class ImageAdapter extends BaseAdapter {
    private Context mContext;
    public ImageAdapter(Context c) {
        mContext = c;
    }
    public int getCount() {
        return thumbImages.length;
    }
    public Object getItem(int position) {
        return null;
    }
    public long getItemId(int position) {
        return 0;
    }
    // create a new ImageView for each item referenced by the Adapter
    public View getView(int position, View convertView, ViewGroup parent) {
        ImageView imageView = new ImageView(mContext);
        imageView.setLayoutParams(new GridView.LayoutParams(200, 200));
        imageView.setScaleType(ImageView.ScaleType.CENTER_CROP);
        imageView.setPadding(8, 8, 8, 8);
        imageView.setImageResource(thumbImages[position]);
        return imageView;
    }
    // Add all our images to arraylist
    public Integer[] thumbImages = {
        R.drawable.img1, R.drawable.img2,
        R.drawable.img3, R.drawable.img4,
        R.drawable.img5, R.drawable.img6,
        R.drawable.img7, R.drawable.img8,
        R.drawable.img1, R.drawable.img2,
        R.drawable.img3, R.drawable.img4,
        R.drawable.img5, R.drawable.img6,
        R.drawable.img7, R.drawable.img8,
        R.drawable.img1, R.drawable.img2,
        R.drawable.img3, R.drawable.img4,
        R.drawable.img5
    };
}

```

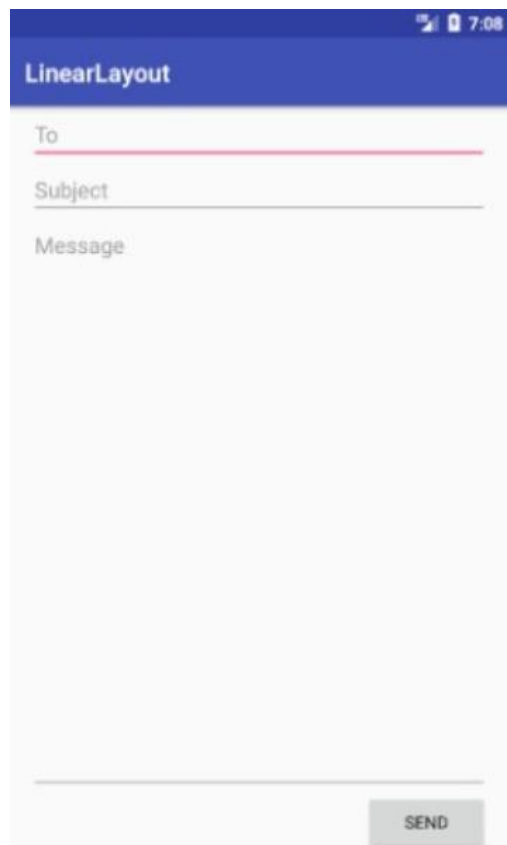
MainActivity.java

```

package com.tutlane.gridview
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.AdapterView;
import android.widget.GridView;
import android.widget.Toast;
public class MainActivity extends AppCompatActivity { @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        GridView gv = (GridView) findViewById(R.id.gvDetails);
        gv.setAdapter(new ImageAdapter(this));
        gv.setOnItemClickListener(new AdapterView.OnItemClickListener() {
            public void onItemClick(AdapterView<?> parent, View v, int position, long id) {
                Toast.makeText(MainActivity.this, "Image Position: " + position, Toast.LENGTH_SHORT).show();
            }
        });
    }
}

```

}
OUTPUT:



Ques 2. Asynchronous tasks in Android. Program to Illustrate Multithreading, Identify Collaborative UCD process features

Asynchronous tasks in Android

AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.legendblogs.asyncntask">
    <application
        android:allowBackup="true" android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name" android:supportRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
</manifest>
```

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:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context="com.legendblogs.asyncntask.MainActivity">
    <TextView android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Welcome to Legend Blogs Exmpale"
        android:layout_alignParentTop="true" android:layout_centerHorizontal="true"
        android:textStyle="bold" android:textSize="20dp"
        android:id="@+id/textView2" />
    <TextView android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textAppearance="?android:attr/textAppearanceLarge"
        android:text="Please enter a waiting Time" android:id="@+id/textView"
        android:layout_below="@+id/textView2" android:layout_centerHorizontal="true"
        android:layout_marginTop="64dp" />
    <EditText android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/editText" android:layout_below="@+id/textView"
        android:layout_marginTop="56dp"
        android:layout_alignEnd="@+id/textView" android:layout_alignStart="@+id/textView" />
    <Button android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Run AsyncTask" android:id="@+id/button"
        android:layout_below="@+id/editText" android:layout_centerHorizontal="true"
        android:layout_marginTop="62dp" />
</RelativeLayout>
```

MainActivity.java

```
package com.legendblogs.asyncntask;
import android.content.Context;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
public class MainActivity extends AppCompatActivity {
    Context context; EditText editText; Button button;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main); this.context = this;
```



```

editText = (EditText) findViewById(R.id.editText);
button = (Button) findViewById(R.id.button);
button.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        if(editText != null && editText.length() > 0)
            new MyAsyncTask(context).execute(editText.getText().toString()); } });}}

```

MyAsyncTask.java

```

package com.legendblogs.asynctask;
import android.app.ProgressDialog;
import android.content.Context;
import android.os.AsyncTask;
import android.util.Log;
public class MyAsyncTask extends AsyncTask<String, String, String> {
    Context context; String result;
    ProgressDialog progressDialog;
    public MyAsyncTask (Context context) {
        this.context = context; } @Override
    protected void onPreExecute() {
        super.onPreExecute();
        progressDialog = ProgressDialog.show(context,
        "Progress Dialog", null); } @Override
    protected String doInBackground(String... args) {
        int value = Integer.parseInt(args[0]);
        for(int i = 0; i <= value ;i++){
            try { Thread.sleep(1000); } catch (Exception e) {
                Log.v("Error: ", e.toString());}
            result = "Please wait for " + (value - i ) + " seconds";
            publishProgress(result); }
        return null; } @Override
    protected void onProgressUpdate(String... text) {
        progressDialog.setMessage(text[0]); }
    protected void onPostExecute(String result) {
        progressDialog.dismiss(); }}

```

Multithreading

```

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"android:layout_height="match_parent"
    android:orientation="vertical"android:id="@+id/info" >
    <Button android:id="@+id/button1"android:layout_width="match_parent"
        android:layout_height="wrap_content"android:onClick="fetchData"
        android:text="Start MULTITHREAD" />
    <TextView android:id="@+id/textView1"android:layout_width="wrap_content"
        android:layout_height="wrap_content"android:text="Main thread" />
</LinearLayout>

```

```

package multi.threading;
import android.app.Activity;import android.os.Bundle;
import android.os.Handler;
import android.view.View;
import android.widget.TextView;
public class MultiThreadingActivity extends Activity {
    private TextView tvOutput;
    private static final int t1 = 1; private static final int t2 = 2;
    private static final int t3 = 3;
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
        tvOutput = (TextView) findViewById(R.id.textView1); }
    public void fetchData(View v) {
        tvOutput.setText("Main thread");
        thread1.start();thread2.start();thread3.start(); }
    Thread thread1 = new Thread(new Runnable() {
        @Override public void run() {
            for (int i = 0; i < 5; i++) {
                try { Thread.sleep(1000); } catch (InterruptedException e) {

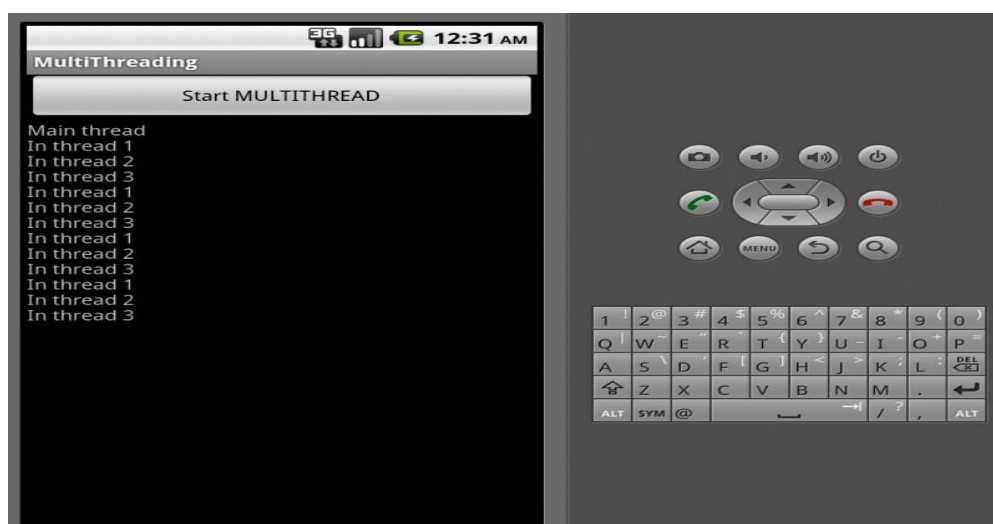
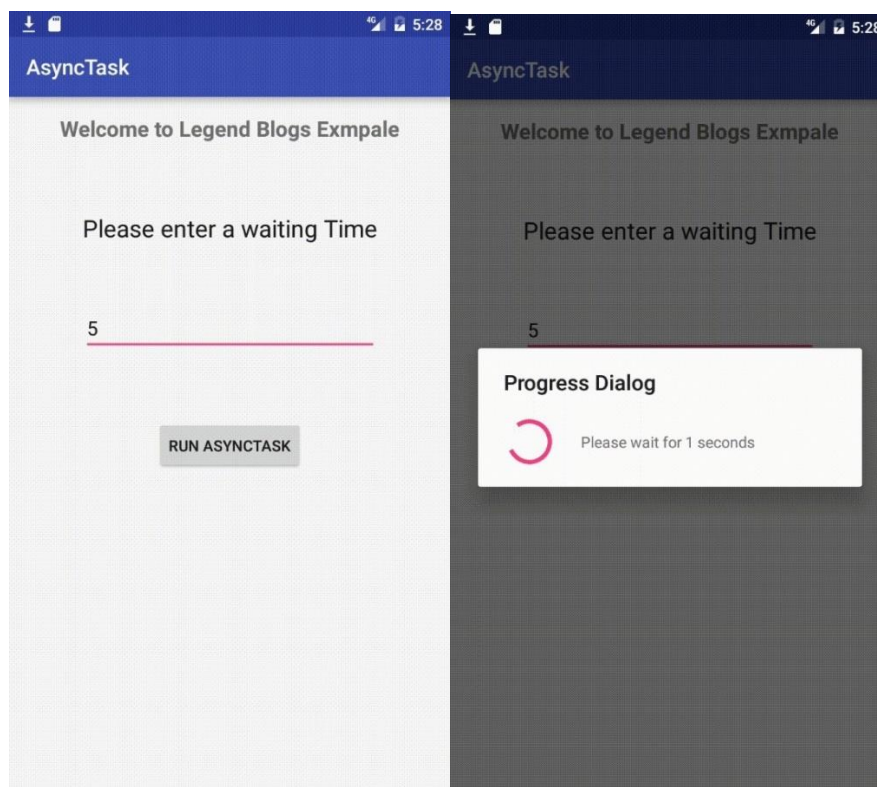
```

```

e.printStackTrace(); }
handler.sendMessage(t1); } } });
Thread thread2 = new Thread(new Runnable() { @Override
public void run() { for (int i = 0; i < 5; i++) {
try { Thread.sleep(1000);
}catch (InterruptedException e) {
e.printStackTrace(); }
handler.sendMessage(t2);} } });
Thread thread3 = new Thread(new Runnable() { @Override
public void run() { for (int i = 0; i < 5; i++) {
try {Thread.sleep(1000);} catch (InterruptedException e) {
e.printStackTrace(); }
handler.sendMessage(t3);} } });
Handler handler = new Handler() {
public void handleMessage(android.os.Message msg) {
if(msg.what == t1) {tvOutput.append("\nIn thread 1");}
if(msg.what == t2) {tvOutput.append("\nIn thread 2");}
if(msg.what == t3) {tvOutput.append("\nIn thread 3");} } }; }

```

OUTPUT:



Ques 3. Simple Calculator Application using Android,Identify Multimodality features

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="fill_parent"
android:layout_height="fill_parent">
<LinearLayout android:layout_width="match_parent"
android:layout_height="wrap_content"android:id="@+id/linearLayout1"
android:layout_marginLeft="10pt"android:layout_marginRight="10pt"
android:layout_marginTop="3pt">
<EditText android:layout_weight="1"
android:layout_height="wrap_content"
android:layout_marginRight="5pt"android:id="@+id/etNum1"
android:layout_width="match_parent"
android:inputType="numberDecimal">
</EditText>
<EditText android:layout_height="wrap_content"
android:layout_weight="1"android:layout_marginLeft="5pt"
android:id="@+id/etNum2"android:layout_width="match_parent"
android:inputType="numberDecimal">
</EditText></LinearLayout>
<LinearLayoutandroid:layout_width="match_parent"
android:layout_height="wrap_content"android:id="@+id/linearLayout2"
android:layout_marginTop="3pt"android:layout_marginLeft="5pt"
android:layout_marginRight="5pt">
<Button android:layout_height="wrap_content"
android:layout_width="match_parent"android:layout_weight="1"
android:text="+"android:textSize="15pt"
android:id="@+id/btnAdd"></Button>
<Button android:layout_height="wrap_content"
android:layout_width="match_parent"android:layout_weight="1"
android:text="-"android:textSize="15pt"android:id="@+id/btnSub">
</Button>
<Button android:layout_height="wrap_content"
android:layout_width="match_parent"android:layout_weight="1"
android:text="*"android:textSize="15pt"android:id="@+id/btnMult">
</Button>
<Button android:layout_height="wrap_content"
android:layout_width="match_parent"android:layout_weight="1"
android:text="/"android:textSize="15pt"android:id="@+id/btnDiv">
</Button>
</LinearLayout>
<TextView android:layout_height="wrap_content"
android:layout_width="match_parent"
android:layout_marginLeft="5pt"android:layout_marginRight="5pt"
android:textSize="12pt"android:layout_marginTop="3pt"android:id="@+id/tvResult"
android:gravity="center_horizontal">
</TextView>
</LinearLayout>
```

MainActivity.java

```
package CALCU.CALU;
import android.app.Activity;
import android.os.Bundle;
import android.text.TextUtils;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
```

```

public class CALCULATORActivity extends Activity implements OnClickListener {
    EditText input1; EditText input2;
    Button addition; Button subtraction;
    Button multiplication; Button division;
    TextView tvResult; String oper = "";
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
        input1 = (EditText) findViewById(R.id.etNum1);
        input2 = (EditText) findViewById(R.id.etNum2);
        addition = (Button) findViewById(R.id.btnAdd);
        subtraction = (Button) findViewById(R.id.btnSub);
        multiplication = (Button) findViewById(R.id.btnMult);
        division = (Button) findViewById(R.id.btnDiv);
        tvResult = (TextView) findViewById(R.id.tvResult);
        // set a listener
        addition.setOnClickListener(this);
        subtraction.setOnClickListener(this);
        multiplication.setOnClickListener(this);
        division.setOnClickListener(this); }
    @Override
    public void onClick(View v) {
        // TODO Auto-generated method stub
        float num1 = 0; float num2 = 0; float result = 0;
        // check if the fields are empty
        if (TextUtils.isEmpty(input1.getText().toString())
            || TextUtils.isEmpty(input2.getText().toString())) {return; }
        // read EditText and fill variables with numbers
        num1 = Float.parseFloat(input1.getText().toString());
        num2 = Float.parseFloat(input2.getText().toString());
        // defines the button that has been clicked and performs the corresponding operation
        // write operation into oper, we will use it later for output
        switch (v.getId()) {
            case R.id.btnAdd:
                oper = "+"; result = num1 + num2; break;
            case R.id.btnSub: oper = "-";
                result = num1 - num2; break;
            case R.id.btnMult: oper = "*";
                result = num1 * num2; break;
            case R.id.btnDiv: oper = "/";
                result = num1 / num2; break;
            default: break; }
        // form the output line
        tvResult.setText(num1 + " " + oper + " " + num2 + " = " + result); } }

```

OUTPUT:



Ques 4. Program to Illustrate: Privacy, Trust and Group Communication Identify Group Awareness Function

```
public class RegistrationActivity extends AppCompatActivity {
    private TextInputEditText mobile;
    private TextInputEditText name;
    private MaterialButton createUserBtn;
    private ProgressBar progressBar;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_login);
        progressBar = findViewById(R.id.createUser_pb);
        mobile = findViewById(R.id.etMobile);
        name = findViewById(R.id.etName);
        createUserBtn = findViewById(R.id.create_user_btn);
        createUserBtn.setTextColor(getResources().getColor(R.color.textColorWhite));
        createUserBtn.setOnClickListener(v ->
            signUpTapped()
        ); }
    private void signUpTapped(){ User user = new User();
        user.setUid(mobile.getText().toString());
        user.setName(name.getText().toString());
        registerUser(user); }
    private void registerUser(User user) {
        progressBar.setVisibility(View.VISIBLE);
        CometChat.createUser(user, AppConfig.AppDetails.API_KEY, new CometChat.CallbackListener<User>() {
            @Override
            public void onSuccess(User user) {
                progressBar.setVisibility(View.GONE);
                login(user); }
            @Override
            public void onError(CometChatException e) {
                progressBar.setVisibility(View.GONE);
                createUserBtn.setClickable(true);
                Toast.makeText(RegistrationActivity.this, "Failed to create user", Toast.LENGTH_LONG).show();
            } }
        ); }
    private void login(User user) {
        progressBar.setVisibility(View.VISIBLE);
        CometChat.login(user.getUid(), AppConfig.AppDetails.API_KEY, new CometChat.CallbackListener<User>() {
            @Override
            public void onSuccess(User user) {
                progressBar.setVisibility(View.GONE);
                startActivity(new Intent(RegistrationActivity.this, ConversationsActivity.class)); }
            @Override
            public void onError(CometChatException e) {
                progressBar.setVisibility(View.GONE);
                Toast.makeText(RegistrationActivity.this, e.getLocalizedMessage(), Toast.LENGTH_LONG).show();
            } }
        ); }
}

public class LoginActivity extends AppCompatActivity {
    private TextInputEditText mobile;
    private MaterialButton loginBtn;
    private MaterialButton signupBtn;
    private ProgressBar progressBar;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
```

```

        setContentView(R.layout.activity_login);
        progressBar = findViewById(R.id.createUser_pb);
        mobile = findViewById(R.id.etMobile);
        signupBtn = findViewById(R.id.createUser_btn);
        loginBtn = findViewById(R.id.login_user_btn);
        loginBtn.setTextColor(getResources().getColor(R.color.textColorWhite));
        loginBtn.setOnClickListener(v ->
            signInTapped());
        signupBtn.setOnClickListener(v ->
            startActivity(new Intent(LoginActivity.this, RegistrationActivity.class))
        ); }
private void signInTapped(){
    User user = new User();
    user.setUid(mobile.getText().toString());
    login(user); }
private void login(User user) {
    progressBar.setVisibility(View.VISIBLE);
    CometChat.login(user.getUid(), AppConfig.AppDetails.API_KEY, new CometChat.CallbackListener<User>() {
        @Override
        public void onSuccess(User user) {
            progressBar.setVisibility(View.GONE);
            startActivity(new Intent(LoginActivity.this, ConversationsActivity.class)); }
        @Override
        public void onError(CometChatException e) {
            progressBar.setVisibility(View.GONE);
        }
    });}

```

ConversationsActivity.

```

public class ConversationsActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_conversations); }
    CometChatConversationListScreen.setItemClickListener(new OnItemClickListener<Conversation>() {
        @Override
        public void OnItemClick(Conversation var, int position) {
            User user = (User)var.getConversationWith();
            Intent intent = new Intent(ConversationsActivity.this, CometChatMessageScreen.class);
            intent.putExtra(StringContract.IntentStrings.UID, user.getUid());
            intent.putExtra(StringContract.IntentStrings.NAME, user.getName());
            intent.putExtra(StringContract.IntentStrings.AVATAR, user.getAvatar());
            intent.putExtra(StringContract.IntentStrings.TYPE, CometChatConstants.RECEIVER_TYPE_USER);
            startActivity(intent); }
        @Override
        public void OnItemLongClick(Conversation var, int position) {
            super.OnItemLongClick(var, position);
        }
    });}

```

ConversationsActivity.java

```

public class ConversationsActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_conversations);
        CometChatConversationListScreen.setItemClickListener(new OnItemClickListener<Conversation>() {
            @Override
            public void OnItemClick(Conversation var, int position) {
                User user = User.fromJson(var.getConversationWith().toString());
                Intent intent = new Intent(ConversationsActivity.this, CometChatMessageScreen.class);
                intent.putExtra(StringContract.IntentStrings.UID, user.getUid());
                intent.putExtra(StringContract.IntentStrings.NAME, user.getName());
                intent.putExtra(StringContract.IntentStrings.AVATAR, user.getAvatar());
                intent.putExtra(StringContract.IntentStrings.TYPE, CometChatConstants.RECEIVER_TYPE_USER);
                startActivity(intent); }
            @Override
            public void OnItemLongClick(Conversation var, int position) {

```

```

        super.OnItemLongClick(var, position);
    });}
    public void newChatTapped(View view) {
        startActivity(new Intent(ConversationsActivity.this, ContactsActivity.class)); }

```

activity_conversations.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=".ConversationsActivity">
    <fragment
        android:id="@+id/conversationList"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        class="screen.CometChatConversationListScreen" />
    <com.google.android.material.floatingactionbutton.FloatingActionButton
        android:id="@+id/fab"
        android:layout_marginEnd="32dp"
        android:layout_marginBottom="32dp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintBottom_toBottomOf="parent"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:src="@android:drawable/ic_menu_add"
        android:layout_gravity="bottom|end"
        app:elevation="6dp"
        android:onClick="newChatTapped"
        app:pressedTranslationZ="12dp"/>
</androidx.constraintlayout.widget.ConstraintLayout>

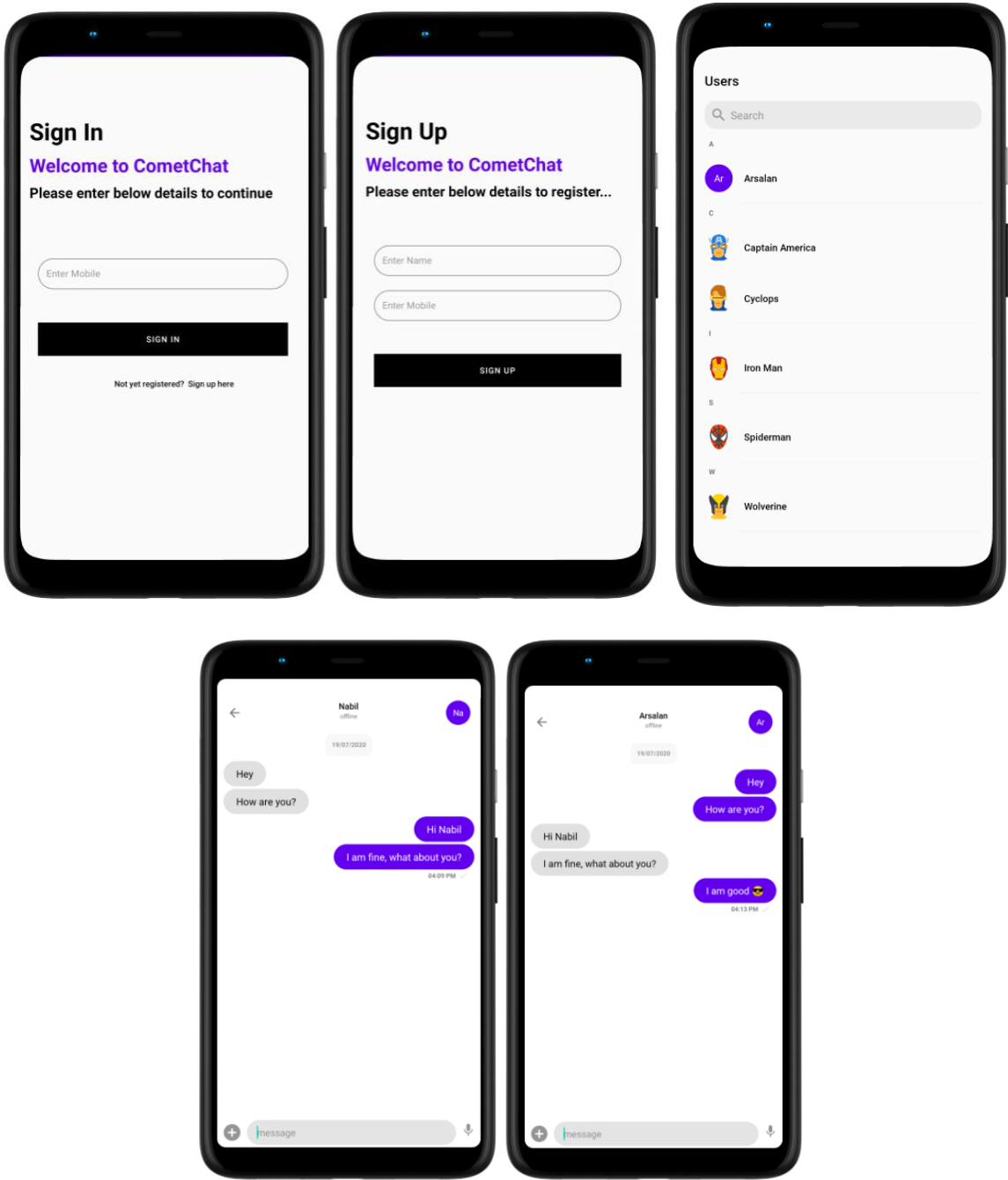
```

```

CometChatUserListScreen.setItemClickListener(new OnItemClickListener<User>() {
    @Override
    public void OnItemClick(User var, int position) {
        Intent intent = new Intent(ContactsActivity.this, CometChatMessageScreen.class);
        intent.putExtra(StringContract.IntentStrings.UID, var.getId());
        intent.putExtra(StringContract.IntentStrings.NAME, var.getName());
        intent.putExtra(StringContract.IntentStrings.AVATAR, var.getAvatar());
        intent.putExtra(StringContract.IntentStrings.TYPE, CometChatConstants.RECEIVER_TYPE_USER);
        startActivity(intent);
    }
    @Override
    public void OnItemLongClick(User var, int position) {
        super.OnItemLongClick(var, position);
    }
});
Intent intent = new Intent(ContactsActivity.this, CometChatMessageScreen.class);
intent.putExtra(StringContract.IntentStrings.UID, var.getId());
intent.putExtra(StringContract.IntentStrings.NAME, var.getName());
intent.putExtra(StringContract.IntentStrings.AVATAR, var.getAvatar());
intent.putExtra(StringContract.IntentStrings.TYPE, CometChatConstants.RECEIVER_TYPE_USER);
startActivity(intent);

```

OUTPUT:



Ques 5. Android Native Application that uses GPS Location Info. Identify SOA for Mobile Services

MainActivity.java.

```
package com.example.tutorialspoint7.myapplication;
import android.Manifest;
import android.app.Activity;
import android.os.Bundle;
import android.support.v4.app.ActivityCompat;
import android.test.mock.MockPackageManager;
import android.view.View;
import android.widget.Button;
import android.widget.Toast;
public class MainActivity extends Activity {
    Button btnShowLocation;
    private static final int REQUEST_CODE_PERMISSION = 2;
    String mPermission = Manifest.permission.ACCESS_FINE_LOCATION;
    // GPSTracker class
    GPSTracker gps; @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        try {
            if (ActivityCompat.checkSelfPermission(this, mPermission)
                != MockPackageManager.PERMISSION_GRANTED) {
                ActivityCompat.requestPermissions(this, new String[]{mPermission},
                    REQUEST_CODE_PERMISSION);
                // If any permission above not allowed by user, this condition will
                // execute every time, else your else part will work}
            } catch (Exception e) { e.printStackTrace(); }
            btnShowLocation = (Button) findViewById(R.id.button);
            // show location button click event
            btnShowLocation.setOnClickListener(new View.OnClickListener() {
                @Override
                public void onClick(View arg0) {
                    gps = new GPSTracker(MainActivity.this);
                    // check if GPS enabled
                    if(gps.canGetLocation()){
                        double latitude = gps.getLatitude();
                        double longitude = gps.getLongitude();
                        Toast.makeText(getApplicationContext(), "Your Location is - \nLat: "
                            + latitude + "\nLong: " + longitude, Toast.LENGTH_LONG).show();
                    }else{
                        gps.showSettingsAlert(); } } }); } }
```

GPSTracker.java.

```
package com.example.tutorialspoint7.myapplication;
import android.app.AlertDialog;
import android.app.Service;
import android.content.Context;
import android.content.DialogInterface;
import android.content.Intent;
import android.location.Location;
import android.location.LocationListener;
import android.location.LocationManager;
import android.os.Bundle;
import android.os.IBinder;
```

```

import android.provider.Settings;
import android.util.Log;
public class GPSTracker extends Service implements LocationListener {
    private final Context mContext;
    boolean isGPSEnabled = false;
    boolean isNetworkEnabled = false;
    boolean canGetLocation = false;
    Location location; // location
    double latitude; // latitude
    double longitude; // longitude
    private static final long MIN_DISTANCE_CHANGE_FOR_UPDATES = 10; // 10 meters
    private static final long MIN_TIME_BW_UPDATES = 1000 * 60 * 1; // 1 minute
    protected LocationManager locationManager;
    public GPSTracker(Context context) {
        this.mContext = context; getLocation(); }
    public Location getLocation() {
        try {
            locationManager = (LocationManager) mContext.getSystemService(LOCATION_SERVICE);
            isGPSEnabled = locationManager.isProviderEnabled(LocationManager.GPS_PROVIDER);
            isNetworkEnabled = locationManager
                .isProviderEnabled(LocationManager.NETWORK_PROVIDER);
            if (!isGPSEnabled && !isNetworkEnabled) {
                // no network provider is enabled } else { this.canGetLocation = true;
                // First get location from Network Provider if (isNetworkEnabled) {
                locationManager.requestLocationUpdates(
                    LocationManager.NETWORK_PROVIDER,
                    MIN_TIME_BW_UPDATES,
                    MIN_DISTANCE_CHANGE_FOR_UPDATES, this);
                Log.d("Network", "Network"); if (locationManager != null) {
                    location = locationManager
                        .getLastKnownLocation(LocationManager.NETWORK_PROVIDER);
                    if (location != null) { latitude = location.getLatitude();
                        longitude = location.getLongitude(); } } }
                if (isGPSEnabled) { if (location == null) {
                    locationManager.requestLocationUpdates(
                        LocationManager.GPS_PROVIDER,
                        MIN_TIME_BW_UPDATES, MIN_DISTANCE_CHANGE_FOR_UPDATES, this);
                    Log.d("GPS Enabled", "GPS Enabled");
                    if (locationManager != null) { location = locationManager
                        .getLastKnownLocation(LocationManager.GPS_PROVIDER);
                        if (location != null) { latitude = location.getLatitude();
                            longitude = location.getLongitude(); } } } } }
            } catch (Exception e) { e.printStackTrace(); } return location; }
    public void stopUsingGPS(){if(locationManager != null){
        locationManager.removeUpdates(GPSTracker.this); } }
    public double getLatitude(){ if(location != null){
        latitude = location.getLatitude(); } return latitude; }
    public double getLongitude(){if(location != null){
        longitude = location.getLongitude(); }return longitude; }
    public boolean canGetLocation() {return this.canGetLocation; }
    public void showSettingsAlert(){
        AlertDialog.Builder alertDialog = new AlertDialog.Builder(mContext);
        alertDialog.setTitle("GPS is settings");
        alertDialog.setMessage("GPS is not enabled. Do you want to go to settings menu?");
        alertDialog.setPositiveButton("Settings", new DialogInterface.OnClickListener() {
            public void onClick(DialogInterface dialog,int which) {
                Intent intent = new Intent(Settings.ACTION_LOCATION_SOURCE_SETTINGS);
                mContext.startActivity(intent); } });
        alertDialog.setNegativeButton("Cancel", new DialogInterface.OnClickListener() {
            public void onClick(DialogInterface dialog, int which) {
                dialog.cancel(); } });
        alertDialog.show(); }
    @Override

```

```

public void onLocationChanged(Location location) {
} @Override
public void onProviderDisabled(String provider) {
} @Override
public void onProviderEnabled(String provider) {
} @Override
public void onStatusChanged(String provider, int status, Bundle extras) {
} @Override
public IBinder onBind(Intent arg0) { return null; }}

```

res/layout/activity_main.xml file –

```

<?xml version = "1.0" encoding = "utf-8"?>
<LinearLayout xmlns:android = "http://schemas.android.com/apk/res/android"
    android:layout_width = "fill_parent" android:layout_height = "fill_parent"
    android:orientation = "vertical" >
    <Button android:id = "@+id/button" android:layout_width = "fill_parent"
        android:layout_height = "wrap_content" android:text = "getlocation"/>
</LinearLayout>

```

res/values/strings.xml

```

<?xml version = "1.0" encoding = "utf-8"?><resources>
    <string name = "app_name">Tutorialspoint</string></resources>

```

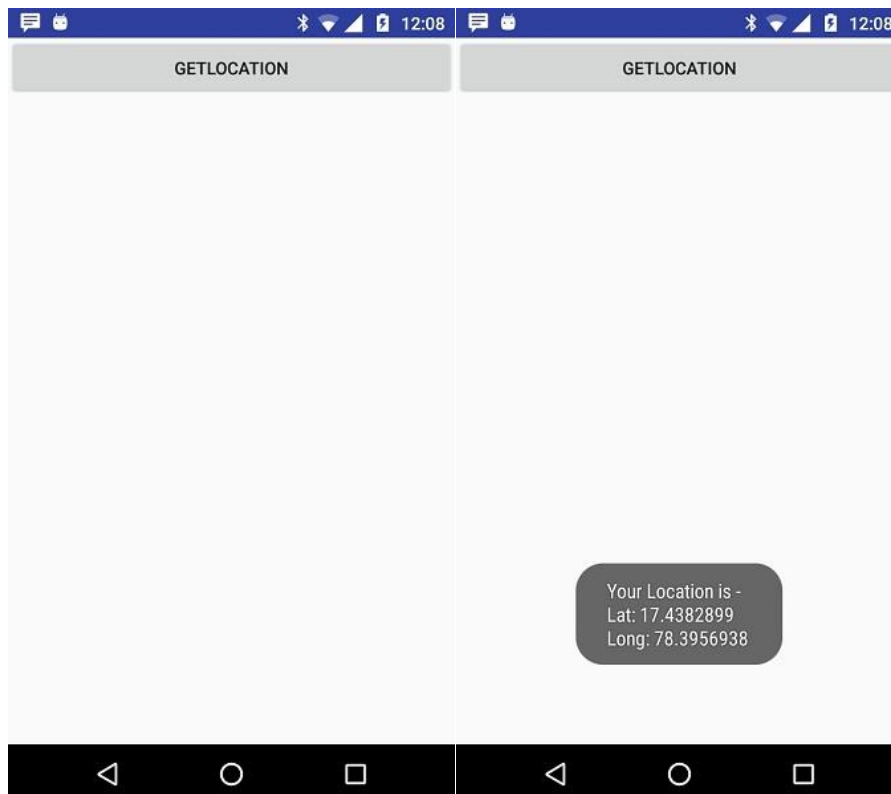
AndroidManifest.xml –

```

<?xml version = "1.0" encoding = "utf-8"?>
<manifest xmlns:android = "http://schemas.android.com/apk/res/android"
    package = "com.example.tutorialspoint7.myapplication">
    <uses-permission android:name = "android.permission.ACCESS_FINE_LOCATION" />
    <uses-permission android:name = "android.permission.INTERNET" />
    <application android:allowBackup = "true"
        android:icon = "@mipmap/ic_launcher" android:label = "@string/app_name"
        android:supportsRtl = "true" android:theme = "@style/AppTheme">
        <activity android:name = ".MainActivity"><intent-filter>
            <action android:name = "android.intent.action.MAIN" />
            <category android:name = "android.intent.category.LAUNCHER" />
        </intent-filter></activity> </application></manifest>

```

OUTPUT



Ques 6. Program for Phone Dialer with CALL and SAVE Buttons. Identify the Personalization features

open res/values/strings.xml and edit it to read

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <string name="app_name">SimpleDialer</string>
    <string name="action_settings">Settings</string>
    <string name="hello_world">Hello world!</string>
    <string name="hello">SimpleDialer</string>
    <string name="main_label">My Friends</string></resources>
```

Then edit res/layout/activity_main.xml to read

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent" android:layout_height="fill_parent"
    android:orientation="vertical" android:padding="15dip" >
    <TextView android:layout_width="wrap_content"
        android:layout_height="wrap_content" android:layout_gravity="center"
        android:layout_marginBottom="25dip" android:text="@string/main_label"
        android:textColor="@color/colorPrimary" android:textSize="22sp" />
    <Button android:id="@+id/button1" android:layout_width="fill_parent"
        android:layout_height="wrap_content" android:textSize="18sp" />
    <Button android:id="@+id/button2" android:layout_width="fill_parent"
        android:layout_height="wrap_content" android:textSize="18sp" />
    <Button android:id="@+id/button3" android:layout_width="fill_parent"
        android:layout_height="wrap_content" android:textSize="18sp" />
    <Button android:id="@+id/button4" android:layout_width="fill_parent"
        android:layout_height="wrap_content" android:textSize="18sp" />
    <Button android:id="@+id/button5" android:layout_width="fill_parent"
        android:layout_height="wrap_content" android:textSize="18sp" />
    <Button android:id="@+id/button6" android:layout_width="fill_parent"
        android:layout_height="wrap_content" android:textSize="18sp" />
</LinearLayout>
```

Finally, open src/<YourNamespace>.simpledialer/MainActivity.java and edit it to read

```
package <YourNamespace>.simpledialer;
import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.content.Intent; import android.net.Uri;
import android.view.View; import android.view.View.OnClickListener;
import android.widget.Button;
public class MainActivity extends AppCompatActivity implements OnClickListener {
    private int entries = 6;
    private String phoneNum[]; private String buttonLabels[];
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        phoneNum = new String[entries];
        buttonLabels = new String[entries];
        populateArrays();
        // Set up buttons and attach click listeners
        Button button1 = (Button)findViewById(R.id.button1);
        button1.setText(buttonLabels[0]);
        button1.setOnClickListener(this);
        Button button2 = (Button)findViewById(R.id.button2);
        button2.setText(buttonLabels[1]);
        button2.setOnClickListener(this);
        Button button3 = (Button)findViewById(R.id.button3);
        button3.setText(buttonLabels[2]);
        button3.setOnClickListener(this);
        Button button4 = (Button)findViewById(R.id.button4);
```

```

        button4.setText(buttonLabels[3]);
        button4.setOnClickListener(this);
        Button button5 = (Button)findViewById(R.id.button5);
        button5.setText(buttonLabels[4]);
        button5.setOnClickListener(this);
        Button button6 = (Button)findViewById(R.id.button6);
        button6.setText(buttonLabels[5]);
        button6.setOnClickListener(this);}

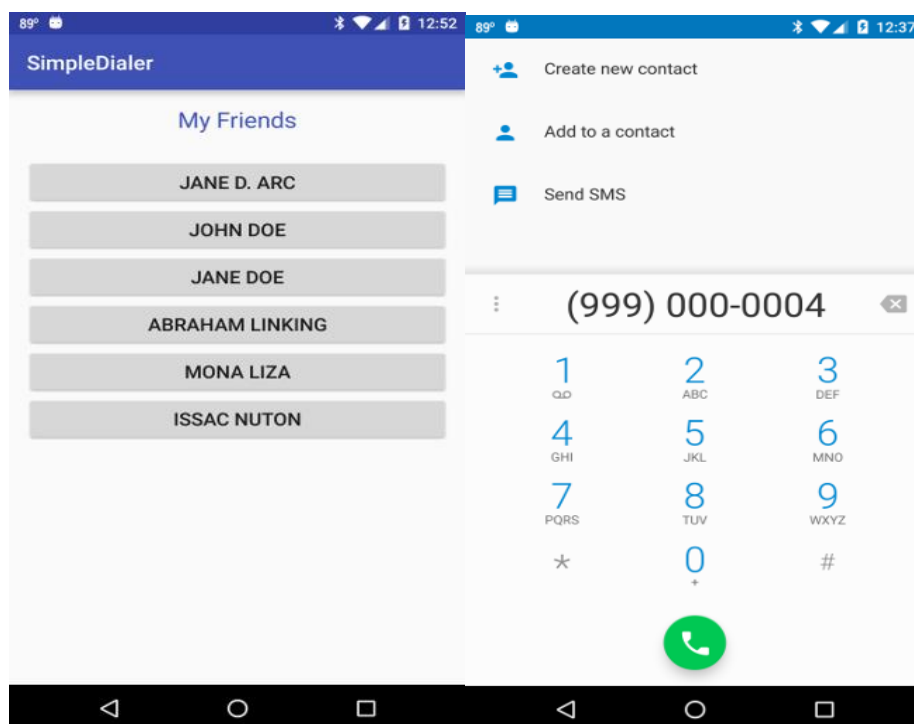
// Launch the phone dialer
public void launchDialer(String number){
    String numberToDial = "tel:"+number;
    startActivity(new Intent(Intent.ACTION_DIAL, Uri.parse(numberToDial)));}

/** Method to populate the data arrays */
public void populateArrays(){
    phoneNum[0] = "999-000-0001"; phoneNum[1] = "999-000-0002";
    phoneNum[2] = "999-000-0003"; phoneNum[3] = "999-000-0004";
    phoneNum[4] = "999-000-0005"; phoneNum[5] = "999-000-0006";
    buttonLabels[0] = "Jane D. Arc"; buttonLabels[1] = "John Doe";
    buttonLabels[2] = "Jane Doe"; buttonLabels[3] = "Abraham Linking";
    buttonLabels[4] = "Mona Liza"; buttonLabels[5] = "Issac Nuton"; }

/** Process button events */ @Override
public void onClick(View v) {
    switch (v.getId()) {
        case R.id.button1:
            launchDialer(phoneNum[0]); break;
        case R.id.button2:
            launchDialer(phoneNum[1]); break;
        case R.id.button3:
            launchDialer(phoneNum[2]); break;
        case R.id.button4:
            launchDialer(phoneNum[3]); break;
        case R.id.button5:
            launchDialer(phoneNum[4]); break;
        case R.id.button6:
            launchDialer(phoneNum[5]);
            break; } }

```

OUTPUT



Ques 7. Program to Create Address Book and Contacts. Identify Context Management Framework features

Adding Permission in Manifest.xml file-

```
<!--permissions to read contacts-->
<uses-permission android:name="android.permission.READ_CONTACTS" />
<!--permissions to write contacts-->
<uses-permission android:name="android.permission.WRITE_CONTACTS" />
<!--permissions to make a call-->
<uses-permission android:name="android.permission.CALL_PHONE"/>
<!--permissions to send sms-->
<uses-permission android:name="android.permission.SEND_SMS" />
<!--permissions to write sms-->
<uses-permission android:name="android.permission.WRITE_SMS"/>
```

Activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
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"android:orientation="vertical"
tools:context=".MainActivity">
<!--Recycler view for displaying list of contacts-->
<androidx.recyclerview.widget.RecyclerView
android:id="@+id/idRVContacts"android:layout_width="match_parent"
android:layout_height="match_parent" />
<!--progress bar for displaying loading-->
<ProgressBar android:id="@+id/idPBLoading"
android:layout_width="wrap_content"android:layout_height="wrap_content"
android:layout_centerInParent="true" />
<!--fab for adding a new contact-->
<com.google.android.material.floatingactionbutton.FloatingActionButton
android:id="@+id/idFABadd"android:layout_width="wrap_content"
android:layout_height="wrap_content"android:layout_alignParentEnd="true"
android:layout_alignParentBottom="true" android:layout_margin="20dp"
android:src="@drawable/ic_account"app:fabCustomSize="40dp"
app:tint="@color/white" /></RelativeLayout>
```

Creating java class for storing contacts

```
public class ContactsModal {
private String userName;private String contactNumber;
public ContactsModal(String userName, String contactNumber) {
this.userName = userName;this.contactNumber = contactNumber;
}
public String getUserName() { return userName;}
public void setUserName(String userName) {this.userName = userName;}
public String getContactNumber() {return contactNumber;}
public void setContactNumber(String contactNumber) {this.contactNumber = contactNumber;}}
```

Creating a new menu resource file for adding search view in the toolbar

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto">
<!--creating an item for our search view-->
<item android:id="@+id/app_bar_search"android:icon="@drawable/ic_account"
android:title="Search"app:actionViewClass="android.widget.SearchView"
app:showAsAction="ifRoom|withText" /> </menu>
```

MainActivity.java file

```
import android.Manifest;
import android.content.DialogInterface;
import android.content.Intent;
import android.database.Cursor;
import android.net.Uri;
```

```

import android.os.Bundle;
import android.provider.ContactsContract;
import android.provider.Settings;
import android.view.Menu;
import android.view.MenuInflater;
import android.view.MenuItem;
import android.view.View;
import android.widget.ProgressBar;
import android.widget.SearchView;
import androidx.appcompat.app.AlertDialog;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.view.MenuItemCompat;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;
import com.google.android.material.floatingactionbutton.FloatingActionButton;
import com.karumi.dexter.Dexter;
import com.karumi.dexter.MultiplePermissionsReport;
import com.karumi.dexter.PermissionToken;
import com.karumi.dexter.listener.DexterError;
import com.karumi.dexter.listener.PermissionRequest;
import com.karumi.dexter.listener.PermissionRequestErrorListener;
import com.karumi.dexter.listener.multi.MultiplePermissionsListener;
import java.util.ArrayList;
import java.util.List;
public class MainActivity extends AppCompatActivity {
    private ArrayList<ContactsModal> contactsModalArrayList;
    private RecyclerView contactRV;
    private ContactRVAdapter contactRVAdapter;
    private ProgressBar loadingPB;
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState); setContentView(R.layout.activity_main);
        contactsModalArrayList = new ArrayList<>();
        contactRV = findViewById(R.id.idRVContacts);
        FloatingActionButton addNewContactFAB = findViewById(R.id.idFABadd);
        loadingPB = findViewById(R.id.idPBLoding);
        prepareContactRV(); requestPermissions();
        addNewContactFAB.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent i = new Intent(MainActivity.this, CreateNewContactActivity.class);
                startActivity(i);
            }
        });
        public boolean onCreateOptionsMenu(Menu menu) {
            MenuInflater inflater = getMenuInflater();
            inflater.inflate(R.menu.search_menu, menu);
            MenuItem searchViewItem = menu.findItem(R.id.app_bar_search);
            final SearchView searchView = (SearchView) MenuItemCompat.getActionView(searchViewItem);
            searchView.setOnQueryTextListener(new SearchView.OnQueryTextListener() {
                @Override
                public boolean onQueryTextSubmit(String query) {
                    // on query submit we are clearing the focus for our search view.
                    searchView.clearFocus(); return false;
                }
                @Override
                public boolean onQueryTextChange(String newText) {
                    // on changing the text in our search view we are calling
                    // a filter method to filter our array list.
                    filter(newText.toLowerCase()); return false;
                }
            });
            return super.onCreateOptionsMenu(menu);
        }
        private void filter(String text) {
            ArrayList<ContactsModal> filteredlist = new ArrayList<>();
            for (ContactsModal item : contactsModalArrayList) {
                if (item.getUserName().toLowerCase().contains(text.toLowerCase())) {
                    // on below line we are adding item to our filtered array list.
                    filteredlist.add(item);
                }
            }
        }
    }
}

```

```

        if (filteredlist.isEmpty()) {
            Toast.makeText(this, "No Contact Found",
Toast.LENGTH_SHORT).show();
        } else {
            // passing this filtered list to our adapter with filter list method.
            contactRVAdapter.filterList(filteredlist);
        }
    }
    private void prepareContactRV() {
        contactRVAdapter = new ContactRVAdapter(this, contactsModalArrayList);
        contactRV.setLayoutManager(new LinearLayoutManager(this));
        contactRV.setAdapter(contactRVAdapter);
    }
    private void requestPermissions() {
        Dexter.withActivity(this)
            .withPermissions(Manifest.permission.READ_CONTACTS,
                Manifest.permission.CALL_PHONE,
                Manifest.permission.SEND_SMS,
                Manifest.permission.WRITE_CONTACTS)
            .withListener(new MultiplePermissionsListener() {
                @Override
                public void
onPermissionsChecked(MultiplePermissionsReport multiplePermissionsReport) {
                    if
(multiplePermissionsReport.areAllPermissionsGranted()) {
                        getContacts();
                        Toast.makeText(MainActivity.this,
                            "All the permissions are granted..", Toast.LENGTH_SHORT).show();
                    }
                    if
(multiplePermissionsReport.isAnyPermissionPermanentlyDenied()) {
                        showSettingsDialog();
                    }
                }
            })
            .onPermissionRationaleShouldBeShown(List<PermissionRequest> list, PermissionToken permissionToken) {
                permissionToken.continuePermissionRequest();
            })
            .withErrorListener(new PermissionRequestErrorListener() {
                @Override
                public void onError(DexterError error) {
                    Toast.makeText(getApplicationContext(), "Error occurred! ",
Toast.LENGTH_SHORT).show();
                }
            })
            .onSameThread().check();
    }
    private void getContacts() {
        String contactId = "";
        String displayName = "";
        Cursor cursor =
getContentResolver().query(ContactsContract.Contacts.CONTENT_URI, null, null, null,
ContactsContract.CommonDataKinds.Phone.DISPLAY_NAME + " ASC");
        if (cursor.getCount() > 0) {
            while (cursor.moveToNext()) {
                int hasPhoneNumber =
Integer.parseInt(cursor.getString(cursor.getColumnIndex(ContactsContract.Contacts.HAS_PHONE_NUMBER)));
                if (hasPhoneNumber > 0) {
                    contactId =
cursor.getString(cursor.getColumnIndex(ContactsContract.Contacts._ID));
                    displayName =
cursor.getString(cursor.getColumnIndex(ContactsContract.Contacts.DISPLAY_NAME));
                    Cursor phoneCursor = getContentResolver().query(

```



```

ContactsContract.CommonDataKinds.Phone.CONTENT_URI,
                                null,

ContactsContract.CommonDataKinds.Phone.CONTACT_ID + "= ?",
                                new String[]{contactId},
                                null);
                                if (phoneCursor.moveToNext()) {
                                    // on below line we are getting the phone
number for our users and then adding the name along with phone number in array list.
                                    String phoneNumber =
phoneCursor.getString(phoneCursor.getColumnIndex(ContactsContract.CommonDataKinds.Phone.NUMBER));
                                    contactsModalArrayList.add(new
ContactsModal(displayName, phoneNumber));
                                }
                                phoneCursor.close();
                            }
                        }
                    }
                }
            }
        }
        cursor.close();
        loadingPB.setVisibility(View.GONE);
        contactRVAdapter.notifyDataSetChanged();
    }
}

```

Working with the ContactDetailActivity

```

import android.Manifest;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.net.Uri;
import android.os.Bundle;
import android.view.View;
import android.widget.ImageView;
import android.widget.TextView;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.ActivityCompat;

public class ContactDetailActivity extends AppCompatActivity {
    private String contactName, contactNumber;
    private TextView contactTV, nameTV;
    private ImageView contactIV, callIV, messageIV;
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_contact_detail);
        contactName = getIntent().getStringExtra("name");
        contactNumber = getIntent().getStringExtra("contact");
        nameTV = findViewById(R.id.idTVName);
        contactIV = findViewById(R.id.idIVContact);
        contactTV = findViewById(R.id.idTVPhone);
        nameTV.setText(contactName);
        contactTV.setText(contactNumber);
        callIV = findViewById(R.id.idIVCall);
        messageIV = findViewById(R.id.idIVMessage);
        callIV.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                // calling a method to make a call.
                makeCall(contactNumber);
            }
        });
        messageIV.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                // calling a method to send message
                sendMessage(contactNumber);
            }
        });
    }
}

```

```

        });
    }
    private void sendMessage(String contactNumber) {
        Intent intent = new Intent(Intent.ACTION_VIEW, Uri.parse("sms:" +
contactNumber));

        intent.putExtra("sms_body", "Enter your message");
        startActivity(intent);
    }
    private void makeCall(String contactNumber) {
        Intent callIntent = new Intent(Intent.ACTION_CALL);
        callIntent.setData(Uri.parse("tel:" + contactNumber));
        if (ActivityCompat.checkSelfPermission(ContactDetailActivity.this,
Manifest.permission.CALL_PHONE) !=
PackageManager.PERMISSION_GRANTED) {

            return;
        }
        startActivity(callIntent);
    }
}

```

Working with the CreateNewContactActivity

```

import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.provider.ContactsContract;
import android.text.TextUtils;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;
import androidx.annotation.Nullable;
import androidx.appcompat.app.AppCompatActivity;
public class CreateNewContactActivity extends AppCompatActivity {
    private EditText nameEdt, phoneEdt, emailEdt;
    private Button addContactEdt;
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_create_new_contact);
        nameEdt = findViewById(R.id.idEdtName);
        phoneEdt = findViewById(R.id.idEdtPhoneNumber);
        emailEdt = findViewById(R.id.idEdtEmail);
        addContactEdt = findViewById(R.id.idBtnAddContact);
        addContactEdt.setOnClickListener(new View.OnClickListener() {
            public void onClick(View v) {
                String name = nameEdt.getText().toString();
                String phone = phoneEdt.getText().toString();
                String email = emailEdt.getText().toString();
                if (TextUtils.isEmpty(name) && TextUtils.isEmpty(email) &&
TextUtils.isEmpty(phone)) {
                    Toast.makeText(CreateNewContactActivity.this, "Please
enter the data in all fields. ", Toast.LENGTH_SHORT).show();
                } else {
                    addContact(name, email, phone);
                }
            }
        });
    }
    private void addContact(String name, String email, String phone) {
        Intent contactIntent = new Intent(ContactsContract.Intents.Insert.ACTION);
        contactIntent.setType(ContactsContract.RawContacts.CONTENT_TYPE);
        contactIntent
            .putExtra(ContactsContract.Intents.Insert.NAME, name)

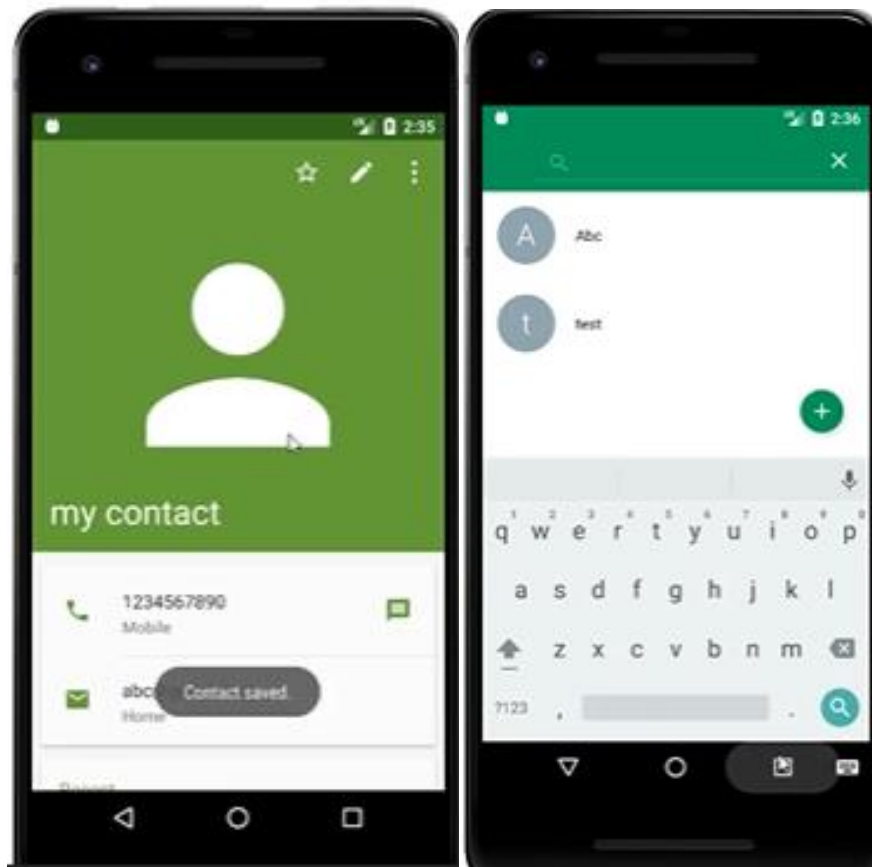
```

```

        .putExtra(ContactsContract.Intents.Insert.PHONE, phone)
        .putExtra(ContactsContract.Intents.Insert.EMAIL, email);
        startActivityForResult(contactIntent, 1);
    }
    protected void onActivityResult(int requestCode, int resultCode, @Nullable Intent data) {
        super.onActivityResult(requestCode, resultCode, data);
        if (requestCode == 1) {
            if (resultCode == Activity.RESULT_OK) {
                Toast.makeText(this, "Contact has been added.",
                    MainActivity.class);
                Intent i = new Intent(CreateNewContactActivity.this,
                    MainActivity.class);
                startActivity(i);
            }
            if (resultCode == Activity.RESULT_CANCELED) {
                Toast.makeText(this, "Cancelled Added Contact",
                    Toast.LENGTH_SHORT).show();
            }
        }
    }
}
}
}
}
}

```

OUTPUT:



Ques 8. Program to Illustrate: Send SMS,Receive SMS,Send e-mail and Receive e-mail.Identify Service Discovery Framework

Send/Receive SMS

AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.tutlane.sendsmsexample">
    <uses-permission android:name="android.permission.SEND_SMS"/>
    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
</manifest>
```

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">
    <TextView android:id="@+id/fstTxt"android:layout_width="wrap_content"
        android:layout_height="wrap_content"android:layout_marginLeft="100dp"
        android:layout_marginTop="150dp"android:text="Mobile No" />
    <EditText android:id="@+id/mbtTxt"android:layout_width="wrap_content"
        android:layout_height="wrap_content"android:layout_marginLeft="100dp"
        android:ems="10"/>
    <TextView android:id="@+id/secTxt"android:layout_width="wrap_content"
        android:layout_height="wrap_content"android:text="Message"
        android:layout_marginLeft="100dp" />
    <EditText android:id="@+id/msgTxt"android:layout_width="wrap_content"
        android:layout_height="wrap_content"android:layout_marginLeft="100dp"
        android:ems="10" />
    <Button android:id="@+id/btnSend"android:layout_width="wrap_content"
        android:layout_height="wrap_content"android:layout_marginLeft="100dp"
        android:text="Send SMS" /></LinearLayout>
```

MainActivity.java

```
package com.tutlane.sendsmsexample;
import android.content.Intent;
import android.net.Uri;
import android.provider.Telephony;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.telephony.SmsManager;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;
public class MainActivity extends AppCompatActivity {
    private EditText txtMobile;
    private EditText txtMessage;
```

```

private Button btnSms;
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    txtMobile = (EditText)findViewById(R.id.mblTxt);
    txtMessage = (EditText)findViewById(R.id.msgTxt);
    btnSms = (Button)findViewById(R.id.btnSend);
    btnSms.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            try{
                SmsManager smgr = SmsManager.getDefault();
                smgr.sendTextMessage(txtMobile.getText().toString(),null,txtMessage.getText().toString(),null,null);
                Toast.makeText(MainActivity.this, "SMS Sent Successfully", Toast.LENGTH_SHORT).show();
            }
            catch (Exception e){
                Toast.makeText(MainActivity.this, "SMS Failed to Send, Please try again", Toast.LENGTH_SHORT).show();
            }
        }
    });
}

```

Send/Receive Email

activity_main.xml file

```

<?xml version="1.0" encoding="utf-8"?>
<!-- Relative Layout -->
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">
    <EditText android:id="@+id/editText1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentTop="true"
        android:layout_alignParentRight="true"
        android:layout_marginTop="18dp"
        android:layout_marginRight="22dp" />
    <EditText android:id="@+id/editText2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@+id/editText1"
        android:layout_alignLeft="@+id/editText1"
        android:layout_marginTop="20dp" />
    <EditText android:id="@+id/editText3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@+id/editText2"
        android:layout_alignLeft="@+id/editText2"
        android:layout_marginTop="30dp" />
    <TextView android:id="@+id/textView1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignBaseline="@+id/editText1"
        android:layout_alignBottom="@+id/editText1"

```

```

        android:layout_alignParentLeft="true"
        android:text="Send To:"
        android:textColor="#0F9D58" />
<TextView android:id="@+id/textView2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignBaseline="@+id/editText2"
        android:layout_alignBottom="@+id/editText2"
        android:layout_alignParentLeft="true"
        android:text="Email Subject:"
        android:textColor="#0F9D58" />
<TextView android:id="@+id/textView3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignBaseline="@+id/editText3"
        android:layout_alignBottom="@+id/editText3"
        android:text="Email Body:"
        android:textColor="#0F9D58" />
<Button android:id="@+id/button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@+id/editText3"
        android:layout_alignLeft="@+id/editText3"
        android:layout_marginLeft="76dp"
        android:layout_marginTop="20dp"
        android:text="Send email!!" />

```

</RelativeLayout>

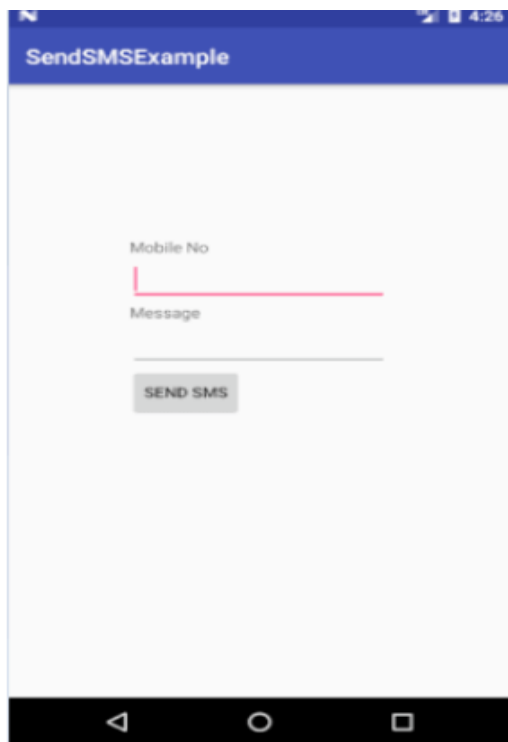
MainActivity.java

```

import android.content.Intent;
import android.os.Bundle;
import android.widget.Button;
import android.widget.EditText;
import androidx.appcompat.app.AppCompatActivity;
public class MainActivity extends AppCompatActivity {
    Button button;
    EditText sendto, subject, body;
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        sendto = findViewById(R.id.editText1);
        subject = findViewById(R.id.editText2);
        body = findViewById(R.id.editText3);
        button = findViewById(R.id.button);
        button.setOnClickListener(view -> {
            String emailsend = sendto.getText().toString();
            String emailsubject = subject.getText().toString();
            String emailbody = body.getText().toString();
            Intent intent = new Intent(Intent.ACTION_SEND);
            intent.putExtra(Intent.EXTRA_EMAIL, new String[]{emailsend});
            intent.putExtra(Intent.EXTRA_SUBJECT, emailsubject);
            intent.putExtra(Intent.EXTRA_TEXT, emailbody);
            intent.setType("message/rfc822");
            startActivity(Intent.createChooser(intent, "Choose an Email client :"));
        });
    }
}

```

OUTPUT

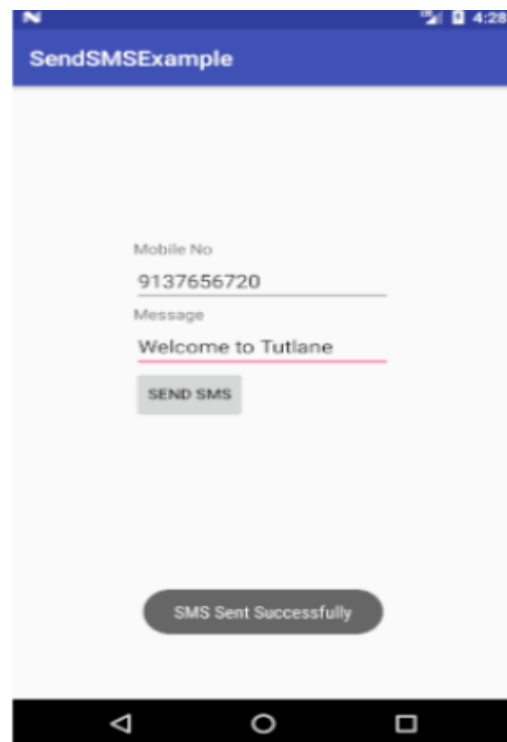


SendSMSExample

Mobile No

Message

SEND SMS



SendSMSExample

Mobile No
9137656720

Message
Welcome to Tutlane

SEND SMS

SMS Sent Successfully

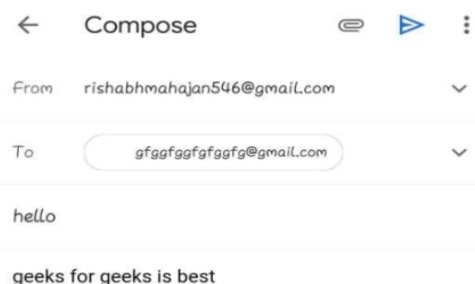


Send To: gfggfgfgfgfgfg@gmail.com

Email Subject: hello

Email Body: geeks for geeks is best

SEND EMAIL!!



Compose

From: rishabhmahajan546@gmail.com

To: gfggfgfgfgfgfg@gmail.com

hello

geeks for geeks is best

Ques 9. Program to Register for the App using SQLite & MySQL Database. Program to Login to The App using Databases. Identify Mobile Web Services

Android Manifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.vamsi.login">
    <uses-permission android:name="android.permission.INTERNET" />
    <application android:allowBackup="true"
        android:icon="@mipmap/ic_launcher" android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportRtl="true" android:theme="@style/AppTheme">
        <activity android:name=".MainActivity"
            android:windowSoftInputMode="stateHidden">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter></activity>
        <activity android:name=".RegisterActivity"
            android:label="Registration" android:windowSoftInputMode="stateHidden" />
        <activity android:name=".LoginSucess"></activity>
    </application></manifest>
```

MainActivity.java

```
package com.example.vamsi.login;
import android.content.Intent;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;
public class MainActivity extends AppCompatActivity {
    private TextView tvRegister;
    private EditText etLoginGmail, etLoginPassword;
    private Button loginButton;
    private SQLiteDatabase db;
    private SQLiteOpenHelper openHelper;
    private Cursor cursor;
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        openHelper = new DatabaseHelper(this);
        db = openHelper.getReadableDatabase();
        tvRegister = findViewById(R.id.tvRegister);
        etLoginGmail = findViewById(R.id.etLogGmail);
        etLoginPassword = findViewById(R.id.etLoginPassword);
        loginButton = findViewById(R.id.btnLogin);
        loginButton.setOnClickListener(new View.OnClickListener() {
            public void onClick(View v) {
                String email = etLoginGmail.getText().toString().trim();
                String password = etLoginPassword.getText().toString().trim();
                if (email.isEmpty() || password.isEmpty()) {
                    Toast.makeText(MainActivity.this, "Enter your Email and Password to login", Toast.LENGTH_SHORT).show();
                } else {
                    cursor = db.rawQuery("SELECT *FROM " + DatabaseHelper.TABLE_NAME + " WHERE " +
DatabaseHelper.COL_4 + "=? AND " + DatabaseHelper.COL_5 + "=?", new String[]{email, password});
                    if (cursor != null) {
                        if (cursor.getCount() > 0) {
                            startActivity(new Intent(MainActivity.this, LoginSucess.class));
                            Toast.makeText(getApplicationContext(), "Login sucess", Toast.LENGTH_SHORT).show();
                        } else {

```



```

Toast.makeText(getApplicationContext(), "Login error", Toast.LENGTH_SHORT).show();
        } } }
    });
    tvRegister.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            startActivity(new Intent(MainActivity.this, RegisterActivity.class));
            finish();
        }
    });
}
}
}
ConfigurDatabase.java
package com.example.vamsi.login;
import android.content.Context;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;
public class DatabaseHelper extends SQLiteOpenHelper {
    public static final String DATABASE_NAME="register.db";
    public static final String TABLE_NAME="registration";
    public static final String COL_1="ID";
    public static final String COL_2="Name";
    public static final String COL_3="Phone";
    public static final String COL_4="Gmail";
    public static final String COL_5="Password";
    public DatabaseHelper(Context context) {
        super(context, DATABASE_NAME, null, 1);
    }
    public void onCreate(SQLiteDatabase db) {
        db.execSQL("CREATE TABLE " + TABLE_NAME + " (ID INTEGER PRIMARY KEY AUTOINCREMENT,Name
TEXT,Phone TEXT,Gmail TEXT,Password TEXT)");
    }
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
        db.execSQL("DROP TABLE IF EXISTS " +TABLE_NAME);
        onCreate(db);
    }
}
}
Login.java
package com.example.vamsi.login;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
public class LoginSucess extends AppCompatActivity {
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_login_sucess);
    }
}
Registration.java
package com.example.vamsi.login;
import android.content.ContentValues;
import android.content.Intent;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;
public class RegisterActivity extends AppCompatActivity {
    private Button registerBtn, gotoLoginBtn;
    private SQLiteOpenHelper openHelper;
    private SQLiteDatabase db;
    private EditText regName, regPhone, regGmail, regPassword;
    protected void onCreate(Bundle savedInstanceState) {

```

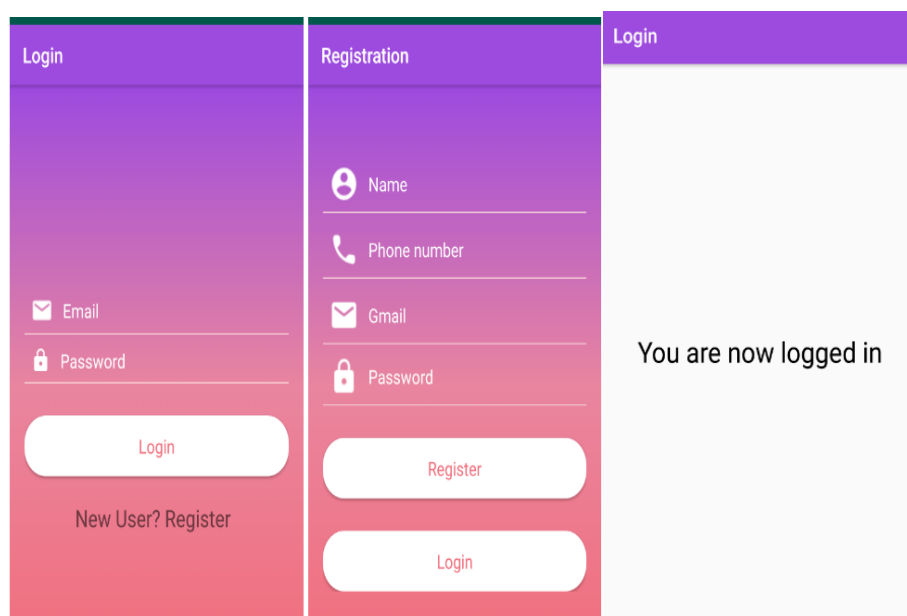
```

super.onCreate(savedInstanceState);
setContentView(R.layout.activity_register);
openHelper = new DatabaseHelper(this);
registerBtn = findViewById(R.id.btnRegLogin);
gotoLoginBtn = findViewById(R.id.btnGotoLogin);
regName = findViewById(R.id.etRegName);
regPhone = findViewById(R.id.etRegPhone);
regGmail = findViewById(R.id.etRegGmail);
regPassword = findViewById(R.id.etRegPassword);
registerBtn.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {
        db = openHelper.getWritableDatabase();
        String fname = regName.getText().toString().trim();
        String fPhone = regPhone.getText().toString().trim();
        String fGmail = regGmail.getText().toString().trim();
        String fPassword = regPassword.getText().toString().trim();
        if (fname.isEmpty() || fPassword.isEmpty() || fGmail.isEmpty() || fPhone.isEmpty()) {
            Toast.makeText(RegisterActivity.this, "Please fill all the details", Toast.LENGTH_SHORT).show();
        } else {
            insertData(fname,fPhone,fGmail,fPassword);
            Toast.makeText(RegisterActivity.this, "Registration Successful", Toast.LENGTH_SHORT).show();
        }
    }
});
gotoLoginBtn.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        startActivity(new Intent(RegisterActivity.this,MainActivity.class));
        finish();
    }
});
}

public void insertData(String fname,String fPhone,String fGmail,String fPassword){
    ContentValues contentValues = new ContentValues();
    contentValues.put(DatabaseHelper.COL_2,fname);
    contentValues.put(DatabaseHelper.COL_3,fPhone);
    contentValues.put(DatabaseHelper.COL_4,fGmail);
    contentValues.put(DatabaseHelper.COL_5,fPassword);
    long id = db.insert(DatabaseHelper.TABLE_NAME,null,contentValues);
}
}

```

OUTPUT:



Ques 10. Create Student or Employee Database. WAP to Read, Write & Delete From SQLite Database. Identify Agent Security Development

ActivityMain.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout 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="net.simplifiedlearning.sqlitedcrudexample.MainActivity">
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_centerVertical="true"
        android:orientation="vertical"
        android:padding="16dp">
        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginBottom="12dp"
            android:text="Add a new Employee"
            android:textAlignment="center"
            android:textAppearance="@style/Base.TextAppearance.AppCompat.Large" />
        <EditText
            android:id="@+id/editTextName"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:hint="Enter Employee Name" />
        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginTop="10dp"
            android:paddingLeft="6dp"
            android:text="Select Department" />
        <Spinner
            android:id="@+id/spinnerDepartment"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:entries="@array/departments" />
        <EditText
            android:id="@+id/editTextSalary"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:digits="0123456789"
            android:hint="Enter Employee Salary"
            android:inputType="number" />
        <Button
            android:id="@+id/buttonAddEmployee"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="Add Employee" />
        <TextView
            android:id="@+id/textViewViewEmployees"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:padding="16dp"
            android:text="View Employees"
            android:textAlignment="center"
            android:textAppearance="@style/Base.TextAppearance.AppCompat.Medium"
            android:textStyle="bold" />
    </LinearLayout>
</RelativeLayout>
MainActivity.java
```

```

package net.simplifiedlearning.sqlitedcrudexample;
import android.content.Intent;
import android.database.sqlite.SQLiteDatabase;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.EditText;
import android.widget.Spinner;
import android.widget.TextView;
public class MainActivity extends AppCompatActivity implements View.OnClickListener {
    public static final String DATABASE_NAME = "myemployeedatabase";
    TextView textViewViewEmployees;
    EditText editTextName, editTextSalary;
    Spinner spinnerDepartment;
    SQLiteDatabase mDatabase;
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        textViewViewEmployees = (TextView) findViewById(R.id.textViewViewEmployees);
        editTextName = (EditText) findViewById(R.id.editTextName);
        editTextSalary = (EditText) findViewById(R.id.editTextSalary);
        spinnerDepartment = (Spinner) findViewById(R.id.spinnerDepartment);
        findViewById(R.id.buttonAddEmployee).setOnClickListener(this);
        textViewViewEmployees.setOnClickListener(this);
        mDatabase = openOrCreateDatabase(DATABASE_NAME, MODE_PRIVATE, null);
    }
    private boolean inputsAreCorrect(String name, String salary) {
        if (name.isEmpty()) {
            editTextName.setError("Please enter a name");
            editTextName.requestFocus();
            return false;
        }
        if (salary.isEmpty() || Integer.parseInt(salary) <= 0) {
            editTextSalary.setError("Please enter salary");
            editTextSalary.requestFocus();
            return false;
        }
        return true;
    }
    public void onClick(View view) {
        switch (view.getId()) {
            case R.id.buttonAddEmployee:

                addEmployee();

                break;
            case R.id.textViewViewEmployees:

                startActivity(new Intent(this, EmployeeActivity.class));

                break;
        }
    }
    private void createEmployeeTable() {
        mDatabase.execSQL(
            "CREATE TABLE IF NOT EXISTS employees (\n" +
            "    id int NOT NULL CONSTRAINT employees_pk PRIMARY KEY,\n" +
            "    name varchar(200) NOT NULL,\n" +
            "    department varchar(200) NOT NULL,\n" +
            "    joiningdate datetime NOT NULL,\n" +
            "    salary double NOT NULL\n" +
            ");"
        );
    }
}

```

```

private void addEmployee() {
    String name = editTextName.getText().toString().trim();
    String salary = editTextSalary.getText().toString().trim();
    String dept = spinnerDepartment.getSelectedItem().toString();
    Calendar cal = Calendar.getInstance();
    SimpleDateFormat sdf = new SimpleDateFormat("yyyy-mm-dd hh:mm:ss");
    String joiningDate = sdf.format(cal.getTime());
    if (inputsAreCorrect(name, salary)) {
        String insertSQL = "INSERT INTO employees \n" +
            "(name, department, joiningdate, salary)\n" +
            "VALUES \n" +
            "(?, ?, ?, ?)";
        mDatabase.execSQL(insertSQL, new String[]{ name, dept, joiningDate, salary });
        Toast.makeText(this, "Employee Added Successfully", Toast.LENGTH_SHORT).show();
    }
}

```

EmployeeAdapter.java

```

package net.simplifiedlearning.sqlitedcrudexample;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.support.annotation.NonNull;
import android.support.annotation.Nullable;
import android.support.v7.app.AlertDialog;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Spinner;
import android.widget.TextView;
import android.widget.Toast;
import java.util.List;
public class EmployeeAdapter extends ArrayAdapter<Employee> {
    Context mContext;
    int listLayoutRes;
    List<Employee> employeeList;
    SQLiteDatabase mDatabase;
    public EmployeeAdapter(Context mContext, int listLayoutRes, List<Employee> employeeList, SQLiteDatabase mDatabase) {
        super(mContext, listLayoutRes, employeeList);
        this.mContext = mContext;
        this.listLayoutRes = listLayoutRes;
        this.employeeList = employeeList;
        this.mDatabase = mDatabase;
    }
    public View getView(int position, @Nullable View convertView, @NonNull ViewGroup parent) {
        LayoutInflater inflater = LayoutInflater.from(mContext);
        View view = inflater.inflate(listLayoutRes, null);
        Employee employee = employeeList.get(position);
        TextView textViewName = view.findViewById(R.id.textViewName);
        TextView textViewDept = view.findViewById(R.id.textViewDepartment);
        TextView textViewSalary = view.findViewById(R.id.textViewSalary);
        TextView textViewJoiningDate = view.findViewById(R.id.textViewJoiningDate);
        textViewName.setText(employee.getName());
        textViewDept.setText(employee.getDept());
        textViewSalary.setText(String.valueOf(employee.getSalary()));
        textViewJoiningDate.setText(employee.getJoiningDate());
        Button buttonDelete = view.findViewById(R.id.buttonDeleteEmployee);
        Button buttonEdit = view.findViewById(R.id.buttonEditEmployee);
        return view;
    }
    public View getView(int position, @Nullable View convertView, @NonNull ViewGroup parent) {
        LayoutInflater inflater = LayoutInflater.from(mContext);

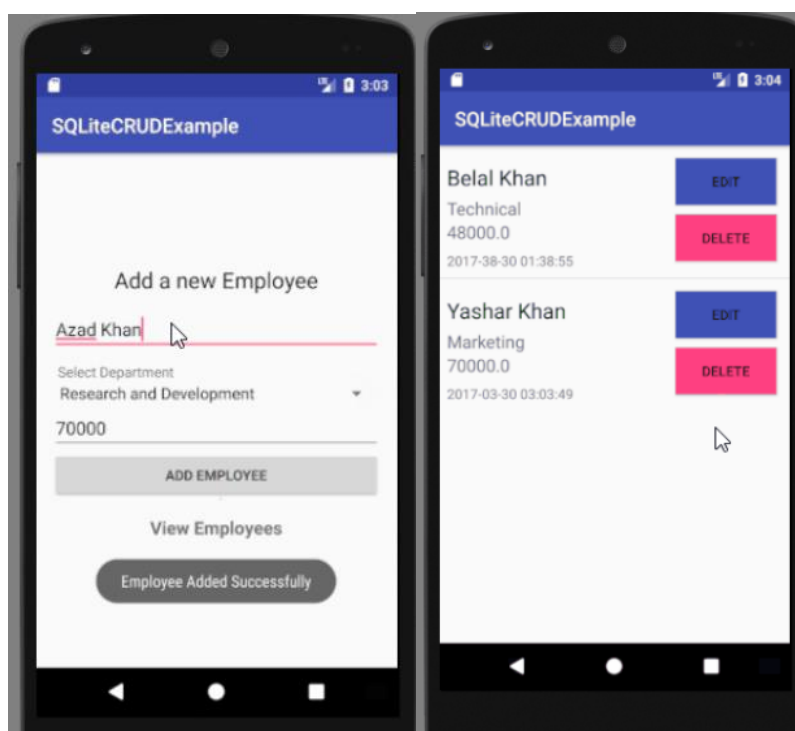
```

```

View view = inflater.inflate(listLayoutRes, null);
final Employee employee = employeeList.get(position);
TextView textViewName = view.findViewById(R.id.textViewName);
TextView textViewDept = view.findViewById(R.id.textViewDepartment);
TextView textViewSalary = view.findViewById(R.id.textViewSalary);
TextView textViewJoiningDate = view.findViewById(R.id.textViewJoiningDate);
textViewName.setText(employee.getName());
textViewDept.setText(employee.getDept());
textViewSalary.setText(String.valueOf(employee.getSalary()));
textViewJoiningDate.setText(employee.getJoiningDate());
Button buttonDelete = view.findViewById(R.id.buttonDeleteEmployee);
Button buttonEdit = view.findViewById(R.id.buttonEditEmployee);
buttonEdit.setOnClickListener(new View.OnClickListener() {
    public void onClick(View view) {
        updateEmployee(employee);
    }
});
buttonDelete.setOnClickListener(new View.OnClickListener() {
    public void onClick(View view) {
        AlertDialog.Builder builder = new AlertDialog.Builder(mCtx);
        builder.setTitle("Are you sure?");
        builder.setPositiveButton("Yes", new DialogInterface.OnClickListener() {
            public void onClick(DialogInterface dialogInterface, int i) {
                String sql = "DELETE FROM employees WHERE id = ?";
                mDatabase.execSQL(sql, new Integer[] {employee.getId()});
                reloadEmployeesFromDatabase();
            }
        });
        builder.setNegativeButton("Cancel", new DialogInterface.OnClickListener() {
            public void onClick(DialogInterface dialogInterface, int i) {
            }
        });
        AlertDialog dialog = builder.create();
        dialog.show();
    }
});
return view;
}
}

```

OUTPUT



Ques 11. Program to Show How to Make a SOCKET Connection from a J2ME Phone. Show How to Make a SOCKET Connection from the Phone To Port 80.

```
import javax.microedition.midlet.*;
import javax.microedition.io.*;
import javax.microedition.lcdui.*;
import java.io.*;

public class socket extends MIDlet {
    // StreamConnection allows bidirectional communication
    private StreamConnection streamConnection = null;
    // use OutputStream to send requests
    private OutputStream outputStream = null;
    private DataOutputStream dataOutputStream = null;
    // use InputStream to receive responses from Web server
    private InputStream inputStream = null;
    private DataInputStream dataInputStream = null;
    // specify the connect string
    private String connectString = "socket://www.java-samples.com:80";
    // use a StringBuffer to store the retrieved page contents
    private StringBuffer results;
    // define GUI components
    private Display myDisplay = null;
    private Form resultScreen;
    private StringItem resultField;
    public socket() {
        // initializing GUI display
        results = new StringBuffer();
        myDisplay = Display.getDisplay(this);
        resultScreen = new Form("Page Content:");
    }
    public void startApp() {
        try {
            // establish a socket connection with remote server
            streamConnection = (StreamConnection) Connector.open(connectString);
            // create DataOutputStream on top of the socket connection
            outputStream = streamConnection.openOutputStream();
            dataOutputStream = new DataOutputStream(outputStream);
            // send the HTTP request
            dataOutputStream.writeChars("GET /index.htm HTTP/1.0 \n");
            dataOutputStream.flush();
            // create DataInputStream on top of the socket connection
            inputStream = streamConnection.openInputStream();
            dataInputStream = new DataInputStream(inputStream);
            // retrieve the contents of the requested page from Web server
            int inputChar;
            while ( (inputChar = dataInputStream.read()) != -1) {
                results.append((char) inputChar);
            }
            // display the page contents on the phone screen
            resultField = new StringItem(null, results.toString());
            resultScreen.append(resultField);
            myDisplay.setCurrent(resultScreen);
        } catch (IOException e) {
            System.err.println("Exception caught: " + e);
        }
        finally {
            // free up I/O streams and close the socket connection
            try {
                if (dataInputStream != null)
                    dataInputStream.close();
            } catch (Exception ignored) {}
            try {
                if (dataOutputStream != null)
                    dataOutputStream.close();
            } catch (Exception ignored) {}
        }
    }
}
```

```

if (outputStream != null)
outputStream.close();
} catch (Exception ignored) {}
try {
if (inputStream != null)
inputStream.close();
} catch (Exception ignored) {}
try {
if (streamConnection != null)
streamConnection.close();
} catch (Exception ignored) {}
} }
public void pauseApp() { }
public void destroyApp(boolean unconditional) { } }

```

OUTPUT



Ques 12. Program to Illustrate Mobile Agents & Services Offered by Agents.

GoSNMP:

```
// Default is a pointer to a GoSNMP struct that contains sensible defaults
// eg port 161, community public, etc
g.Default.Target = "192.168.1.10"
err := g.Default.Connect()
if err != nil {
    log.Fatalf("Connect() err: %v", err)
}
defer g.Default.Conn.Close()

oids := []string{ "1.3.6.1.2.1.1.4.0", "1.3.6.1.2.1.1.7.0" }
result, err2 := g.Default.Get(oids) // Get() accepts up to g.MAX_OIDS
if err2 != nil {
    log.Fatalf("Get() err: %v", err2)
}

for i, variable := range result.Variables {
    fmt.Printf("%d: oid: %s ", i, variable.Name)

    // the Value of each variable returned by Get() implements
    // interface{ }. You could do a type switch...
    switch variable.Type {
    case g.OctetString:
        bytes := variable.Value.([]byte)
        fmt.Printf("string: %s\n", string(bytes))
    default:
        // ... or often you're just interested in numeric values.
        // ToBigInt() will return the Value as a BigInt, for plugging
        // into your calculations.
        fmt.Printf("number: %d\n", g.ToBigInt(variable.Value))
    }
}
```

Running this example gives the following output (from my printer):

```
% go run example.go
0: oid: 1.3.6.1.2.1.1.4.0 string: Administrator
1: oid: 1.3.6.1.2.1.1.7.0 number: 104
```