

Dr. D.Y. Patil Unitech Society's

Dr. D.Y. Patil Arts, Commerce and Science College Pimpri, Pune 18

Department of Computer Science

2024-2025

Practical Assignment – 4

Class:- T.Y.B.C.A.(Science)

Subject:- Android Programming

Date:-

- 1. In android using java. Create table Company (id, name, address, phno). Create Application for Performing the following operation on the table. i) Insert New Company Details. ii) Show All the Company Details.**

activity_main.xml

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

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="16dp">
```

```
<EditText
    android:id="@+id/etName"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Enter Company Name" />
```

```
<EditText
    android:id="@+id/etAddress"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Enter Address" />
```

```
<EditText
    android:id="@+id/etPhone"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Enter Phone Number"
    android:inputType="phone" />
```

```
<Button
    android:id="@+id/btnInsert"
```

```

        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Insert Company" />

<Button
    android:id="@+id/btnView"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="View All Companies" />
</LinearLayout>

```

DatabaseHelper.java

```

package com.example.companyapp;

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

public class DatabaseHelper extends SQLiteOpenHelper {
    private static final String DATABASE_NAME = "CompanyDB";
    private static final String TABLE_NAME = "Company";
    private static final String COL_ID = "id";
    private static final String COL_NAME = "name";
    private static final String COL_ADDRESS = "address";
    private static final String COL_PHONE = "phno";

    public DatabaseHelper(Context context) {
        super(context, DATABASE_NAME, null, 1);
    }

    @Override
    public void onCreate(SQLiteDatabase db) {
        String createTable = "CREATE TABLE " + TABLE_NAME + " (" +
            COL_ID + " INTEGER PRIMARY KEY AUTOINCREMENT, " +
            COL_NAME + " TEXT, " +
            COL_ADDRESS + " TEXT, " +

```

```

        COL_PHONE + " TEXT)";
        db.execSQL(createTable);
    }

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

    public boolean insertData(String name, String address, String phone) {
        SQLiteDatabase db = this.getWritableDatabase();
        ContentValues contentValues = new ContentValues();
        contentValues.put(COL_NAME, name);
        contentValues.put(COL_ADDRESS, address);
        contentValues.put(COL_PHONE, phone);

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

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

```

MainActivity.java

```

package com.example.companyapp;

import androidx.appcompat.app.AppCompatActivity;
import android.database.Cursor;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

```

```

public class MainActivity extends AppCompatActivity {
    DatabaseHelper myDb;
    EditText etName, etAddress, etPhone;
    Button btnInsert, btnView;

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

        myDb = new DatabaseHelper(this);

        etName = findViewById(R.id.etName);
        etAddress = findViewById(R.id.etAddress);
        etPhone = findViewById(R.id.etPhone);

        btnInsert = findViewById(R.id.btnInsert);
        btnView = findViewById(R.id.btnView);

        addData();
        display();
    }

    public void addData() {
        btnInsert.setOnClickListener(view -> {
            boolean isInserted = myDb.insertData(
                etName.getText().toString(),
                etAddress.getText().toString(),
                etPhone.getText().toString()
            );

            if (isInserted) {
                Toast.makeText(MainActivity.this, "Data Inserted",
                    Toast.LENGTH_LONG).show();
                etName.setText("");
                etAddress.setText("");
                etPhone.setText("");
            } else {
                Toast.makeText(MainActivity.this, "Insertion Failed",
                    Toast.LENGTH_LONG).show();
            }
        });
    }
}

```

```

    }
    });
}

public void display() {
    btnView.setOnClickListener(view -> {
        Cursor res = myDb.getAllData();
        if (res.getCount() == 0) {
            Toast.makeText(MainActivity.this, "No Data Found",
Toast.LENGTH_LONG).show();
            return;
        }

        StringBuilder buffer = new StringBuilder();
        while (res.moveToNext()) {
            buffer.append("ID: ").append(res.getString(0)).append("\n");
            buffer.append("Name: ").append(res.getString(1)).append("\n");
            buffer.append("Address: ").append(res.getString(2)).append("\n");
            buffer.append("Phone: ").append(res.getString(3)).append("\n\n");
        }

        Toast.makeText(MainActivity.this, buffer.toString(),
Toast.LENGTH_LONG).show();
    });
}
}

```

2. Create table Student (sno , s_name,s_class,s_addr) Teacher (tno, t_name, qualification, experience) Student-Teacher has many to many relationship. Using above database Write Application to accept a teacher name from user and display the names of students along with subjects to whom teacher is teaching.

Activity_main.xml

```

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="16dp">

```

```

<EditText
    android:id="@+id/edtTeacherName"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Enter Teacher's Name" />

<Button
    android:id="@+id/btnSearch"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Search Students" />

<ListView
    android:id="@+id/listView"
    android:layout_width="match_parent"
    android:layout_height="wrap_content" />
</LinearLayout>

```

DatabaseHelper.java

```

package com.example.studentteacherapp;

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

public class DatabaseHelper extends SQLiteOpenHelper {

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

    private static final String TABLE_STUDENT = "Student";
    private static final String TABLE_TEACHER = "Teacher";
    private static final String TABLE_STUDENT_TEACHER = "Student_Teacher";

    // Student table columns
    private static final String COLUMN_SNO = "sno";

```

```

private static final String COLUMN_SNAME = "s_name";
private static final String COLUMN_SCLASS = "s_class";
private static final String COLUMN_SADDR = "s_addr";

// Teacher table columns
private static final String COLUMN_TNO = "tno";
private static final String COLUMN_TNAME = "t_name";
private static final String COLUMN_QUALIFICATION = "qualification";
private static final String COLUMN_EXPERIENCE = "experience";

// Student-Teacher table columns
private static final String COLUMN_SUBJECT = "subject";

// Create Student Table
private static final String CREATE_STUDENT_TABLE = "CREATE TABLE " +
TABLE_STUDENT + "(" +
    COLUMN_SNO + " INTEGER PRIMARY KEY AUTOINCREMENT, " +
    COLUMN_SNAME + " TEXT, " +
    COLUMN_SCLASS + " TEXT, " +
    COLUMN_SADDR + " TEXT);";

// Create Teacher Table
private static final String CREATE_TEACHER_TABLE = "CREATE TABLE " +
TABLE_TEACHER + "(" +
    COLUMN_TNO + " INTEGER PRIMARY KEY AUTOINCREMENT, " +
    COLUMN_TNAME + " TEXT, " +
    COLUMN_QUALIFICATION + " TEXT, " +
    COLUMN_EXPERIENCE + " TEXT);";

// Create Student-Teacher Relationship Table
private static final String CREATE_STUDENT_TEACHER_TABLE = "CREATE
TABLE " + TABLE_STUDENT_TEACHER + "(" +
    COLUMN_SNO + " INTEGER, " +
    COLUMN_TNO + " INTEGER, " +
    COLUMN_SUBJECT + " TEXT, " +
    "FOREIGN KEY (" + COLUMN_SNO + ") REFERENCES " +
TABLE_STUDENT + "(" + COLUMN_SNO + ")", " +
    "FOREIGN KEY (" + COLUMN_TNO + ") REFERENCES " +
TABLE_TEACHER + "(" + COLUMN_TNO + "));";

```

```

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

@Override
public void onCreate(SQLiteDatabase db) {
    db.execSQL(CREATE_STUDENT_TABLE);
    db.execSQL(CREATE_TEACHER_TABLE);
    db.execSQL(CREATE_STUDENT_TEACHER_TABLE);
}

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

// Insert sample data for testing
public void insertSampleData() {
    SQLiteDatabase db = this.getWritableDatabase();

    ContentValues teacherValues = new ContentValues();
    teacherValues.put(COLUMN_TNAME, "John Doe");
    teacherValues.put(COLUMN_QUALIFICATION, "M.Sc");
    teacherValues.put(COLUMN_EXPERIENCE, "5 Years");
    long tno = db.insert(TABLE_TEACHER, null, teacherValues);

    ContentValues studentValues = new ContentValues();
    studentValues.put(COLUMN_SNAME, "Alice Smith");
    studentValues.put(COLUMN_SCLASS, "10th");
    studentValues.put(COLUMN_SADDR, "New York");
    long sno = db.insert(TABLE_STUDENT, null, studentValues);

    ContentValues mappingValues = new ContentValues();
    mappingValues.put(COLUMN_SNO, sno);
    mappingValues.put(COLUMN_TNO, tno);
    mappingValues.put(COLUMN_SUBJECT, "Mathematics");
    db.insert(TABLE_STUDENT_TEACHER, null, mappingValues);
}

```



```

    }

    public Cursor getStudentsByTeacher(String teacherName) {
        SQLiteDatabase db = this.getReadableDatabase();
        String query = "SELECT s.s_name, st.subject FROM " + TABLE_STUDENT + " s
" +
            "JOIN " + TABLE_STUDENT_TEACHER + " st ON s.sno = st.sno " +
            "JOIN " + TABLE_TEACHER + " t ON t.tno = st.tno " +
            "WHERE t.t_name = ?";
        return db.rawQuery(query, new String[] {teacherName});
    }
}

```

MainActivity.java

```

package com.example.studentteacherapp;

import android.database.Cursor;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ListView;
import android.widget.SimpleCursorAdapter;
import android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

    EditText edtTeacherName;
    Button btnSearch;
    ListView listView;
    DatabaseHelper dbHelper;

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

        edtTeacherName = findViewById(R.id.edtTeacherName);
        btnSearch = findViewById(R.id.btnSearch);
        listView = findViewById(R.id.listView);
    }
}

```

```

        dbHelper = new DatabaseHelper(this);

        // Insert sample data (Remove in production)
        dbHelper.insertSampleData();

        btnSearch.setOnClickListener(view -> searchStudents());
    }

    private void searchStudents() {
        String teacherName = edtTeacherName.getText().toString().trim();
        if (teacherName.isEmpty()) {
            Toast.makeText(this, "Please enter a teacher's name",
                Toast.LENGTH_SHORT).show();
            return;
        }

        Cursor cursor = dbHelper.getStudentsByTeacher(teacherName);
        if (cursor.getCount() == 0) {
            Toast.makeText(this, "No students found for this teacher",
                Toast.LENGTH_SHORT).show();
            return;
        }

        String[] from = {"s_name", "subject"};
        int[] to = {R.id.txtStudentName, R.id.txtSubject};

        SimpleCursorAdapter adapter = new SimpleCursorAdapter(
            this, R.layout.list_item, cursor, from, to, 0);
        listView.setAdapter(adapter);
    }
}

```

3. **Create Following Table: Emp (emp_no,emp_name,address,phone,salary) Dept (dept_no,dept_name,location) Emp-Dept is related with one-many relationship. Create application for performing the following Operation on the table 1) Add Records into Emp and Dept table. 2) Accept Department name from User and delete employee information which belongs to that department.**

DatabaseHelper.java

```
package com.example.empdeptapp;
```

```

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

public class DatabaseHelper extends SQLiteOpenHelper {
    private static final String DATABASE_NAME = "CompanyDB";
    private static final int DATABASE_VERSION = 1;

    // Table Names
    private static final String TABLE_EMP = "Emp";
    private static final String TABLE_DEPT = "Dept";

    // Employee Columns
    private static final String EMP_NO = "emp_no";
    private static final String EMP_NAME = "emp_name";
    private static final String ADDRESS = "address";
    private static final String PHONE = "phone";
    private static final String SALARY = "salary";
    private static final String DEPT_NO_FK = "dept_no";

    // Department Columns
    private static final String DEPT_NO = "dept_no";
    private static final String DEPT_NAME = "dept_name";
    private static final String LOCATION = "location";

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

    @Override
    public void onCreate(SQLiteDatabase db) {
        // Create Department Table
        String createDeptTable = "CREATE TABLE " + TABLE_DEPT + " (" +
            DEPT_NO + " INTEGER PRIMARY KEY AUTOINCREMENT, " +
            DEPT_NAME + " TEXT UNIQUE, " +
            LOCATION + " TEXT)";
        db.execSQL(createDeptTable);
    }

```

```

// Create Employee Table
String createEmpTable = "CREATE TABLE " + TABLE_EMP + " (" +
    EMP_NO + " INTEGER PRIMARY KEY AUTOINCREMENT, " +
    EMP_NAME + " TEXT, " +
    ADDRESS + " TEXT, " +
    PHONE + " TEXT, " +
    SALARY + " REAL, " +
    DEPT_NO_FK + " INTEGER, " +
    "FOREIGN KEY (" + DEPT_NO_FK + ") REFERENCES " + TABLE_DEPT
+ "(" + DEPT_NO + ")");
db.execSQL(createEmpTable);
}

```

@Override

```

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

```

// Insert Department

```

public boolean insertDepartment(String deptName, String location) {
    SQLiteDatabase db = this.getWritableDatabase();
    ContentValues values = new ContentValues();
    values.put(DEPT_NAME, deptName);
    values.put(LOCATION, location);

    long result = db.insert(TABLE_DEPT, null, values);
    return result != -1;
}

```

// Insert Employee

```

public boolean insertEmployee(String name, String address, String phone, double
salary, int deptNo) {
    SQLiteDatabase db = this.getWritableDatabase();
    ContentValues values = new ContentValues();
    values.put(EMP_NAME, name);
    values.put(ADDRESS, address);
    values.put(PHONE, phone);
    values.put(SALARY, salary);
}

```

```

        values.put(DEPT_NO_FK, deptNo);

        long result = db.insert(TABLE_EMP, null, values);
        return result != -1;
    }

    // Delete Employees by Department Name
    public int deleteEmployeesByDept(String deptName) {
        SQLiteDatabase db = this.getWritableDatabase();
        int deptId = getDeptIdByName(deptName);
        if (deptId == -1) {
            return 0;
        }
        return db.delete(TABLE_EMP, DEPT_NO_FK + "=?", new
String[]{String.valueOf(deptId)});
    }

    // Get Department ID by Name
    public int getDeptIdByName(String deptName) {
        SQLiteDatabase db = this.getReadableDatabase();
        Cursor cursor = db.rawQuery("SELECT " + DEPT_NO + " FROM " +
TABLE_DEPT + " WHERE " + DEPT_NAME + "=?", new String[]{deptName});
        if (cursor.moveToFirst()) {
            int id = cursor.getInt(0);
            cursor.close();
            return id;
        }
        cursor.close();
        return -1;
    }
}

```

Activity_main.xml

```

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="16dp">

    <EditText
        android:id="@+id/etDeptName"

```

```
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Department Name" />
```

```
<EditText
    android:id="@+id/etDeptLocation"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Department Location" />
```

```
<Button
    android:id="@+id/btnAddDept"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Add Department" />
```

```
<EditText
    android:id="@+id/etEmpName"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Employee Name" />
```

```
<EditText
    android:id="@+id/etEmpAddress"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Employee Address" />
```

```
<EditText
    android:id="@+id/etEmpPhone"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Employee Phone" />
```

```
<EditText
    android:id="@+id/etEmpSalary"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Employee Salary" />
```

```

<EditText
    android:id="@+id/etEmpDept"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Employee Dept No" />

<Button
    android:id="@+id/btnAddEmp"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Add Employee" />

<EditText
    android:id="@+id/etDeleteDept"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Department to Delete Employees" />

<Button
    android:id="@+id/btnDeleteEmp"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Delete Employees by Dept" />

</LinearLayout>

```

MainActivity.java

```

package com.example.empdeptapp;

import android.os.Bundle;
import android.view.View;
import android.widget.EditText;
import android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {
    DatabaseHelper db;
    EditText etDeptName, etDeptLocation, etEmpName, etEmpAddress, etEmpPhone,
    etEmpSalary, etEmpDept, etDeleteDept;

```

```

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

    db = new DatabaseHelper(this);

    etDeptName = findViewById(R.id.etDeptName);
    etDeptLocation = findViewById(R.id.etDeptLocation);
    etEmpName = findViewById(R.id.etEmpName);
    etEmpAddress = findViewById(R.id.etEmpAddress);
    etEmpPhone = findViewById(R.id.etEmpPhone);
    etEmpSalary = findViewById(R.id.etEmpSalary);
    etEmpDept = findViewById(R.id.etEmpDept);
    etDeleteDept = findViewById(R.id.etDeleteDept);

    findViewById(R.id.btnAddDept).setOnClickListener(v -> {
        db.insertDepartment(etDeptName.getText().toString(),
            etDeptLocation.getText().toString());
        Toast.makeText(this, "Department Added", Toast.LENGTH_SHORT).show();
    });

    findViewById(R.id.btnDeleteEmp).setOnClickListener(v -> {
        db.deleteEmployeesByDept(etDeleteDept.getText().toString());
        Toast.makeText(this, "Employees Deleted", Toast.LENGTH_SHORT).show();
    });
}
}

```

4. Create application to send and receive messages using SMSManager.

5. Create application to send email.

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">

```



```

<EditText
    android:id="@+id/recipientEmail"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Recipient Email"
    android:inputType="textEmailAddress"/>

<EditText
    android:id="@+id/subject"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Subject"
    android:layout_below="@id/recipientEmail"/>

<EditText
    android:id="@+id/emailBody"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Email Body"
    android:layout_below="@id/subject"
    android:inputType="textMultiLine"
    android:minHeight="100dp"/>

<Button
    android:id="@+id/sendButton"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Send Email"
    android:layout_below="@id/emailBody"
    android:layout_centerHorizontal="true"/>
</RelativeLayout>

```

MainActivity.java

```

package com.example.sendemailapp;

import android.content.Intent;
import android.os.Bundle;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

```

```

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

    EditText recipientEmailEditText, subjectEditText, emailBodyEditText;
    Button sendButton;

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

        recipientEmailEditText = findViewById(R.id.recipientEmail);
        subjectEditText = findViewById(R.id.subject);
        emailBodyEditText = findViewById(R.id.emailBody);
        sendButton = findViewById(R.id.sendButton);

        sendButton.setOnClickListener(v -> sendEmail());
    }

    private void sendEmail() {
        String recipient = recipientEmailEditText.getText().toString();
        String subject = subjectEditText.getText().toString();
        String body = emailBodyEditText.getText().toString();

        // Check if fields are not empty
        if (!recipient.isEmpty() && !subject.isEmpty() && !body.isEmpty()) {
            // Create an email intent
            Intent emailIntent = new Intent(Intent.ACTION_SEND);
            emailIntent.setType("message/rfc822");

            // Add email recipient, subject, and body to the intent
            emailIntent.putExtra(Intent.EXTRA_EMAIL, new String[]{recipient});
            emailIntent.putExtra(Intent.EXTRA_SUBJECT, subject);
            emailIntent.putExtra(Intent.EXTRA_TEXT, body);

            try {
                // Launch the email client
                startActivity(emailIntent);//OR
                startActivity(Intent.createChooser(emailIntent, "Send Email"));
            }
        }
    }
}

```

```

        } catch (android.content.ActivityNotFoundException ex) {
            // If no email client is found
            Toast.makeText(MainActivity.this, "No email client installed.",
Toast.LENGTH_SHORT).show();
        }
        } else {
            Toast.makeText(MainActivity.this, "Please fill all fields.",
Toast.LENGTH_SHORT).show();
        }
    }
}

```

6. Create application to send message. After sending message display delivery report of message.

AndroidManifest.xml

```

<uses-permission android:name="android.permission.SEND_SMS"/>
<uses-permission android:name="android.permission.RECEIVE_SMS"/>
<uses-permission android:name="android.permission.READ_SMS"/>

```

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">

    <EditText
        android:id="@+id/phoneNumber"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Phone Number"
        android:inputType="phone"/>

    <EditText
        android:id="@+id/messageText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Message"
        android:layout_below="@id/phoneNumber"
        android:inputType="text"/>

```

```

<Button
    android:id="@+id/sendButton"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Send SMS"
    android:layout_below="@id/messageText"
    android:layout_centerHorizontal="true"/>

<TextView
    android:id="@+id/statusTextView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@id/sendButton"
    android:text="Status will appear here"
    android:layout_centerHorizontal="true"/>
</RelativeLayout>

```

MainActivity.java

```

package com.example.smsdeliveryreport;

import android.content.BroadcastReceiver;
import android.content.Context;
import android.content.Intent;
import android.content.IntentFilter;
import android.os.Bundle;
import android.telephony.SmsManager;
import android.telephony.SmsMessage;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

    EditText phoneNumberEditText, messageEditText;
    Button sendButton;
    TextView statusTextView;

    @Override

```

```

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

    phoneNumberEditText = findViewById(R.id.phoneNumber);
    messageEditText = findViewById(R.id.messageText);
    sendButton = findViewById(R.id.sendButton);
    statusTextView = findViewById(R.id.statusTextView);

    sendButton.setOnClickListener(v -> sendSMS());
}

private void sendSMS() {
    String phoneNumber = phoneNumberEditText.getText().toString();
    String message = messageEditText.getText().toString();

    if (!phoneNumber.isEmpty() && !message.isEmpty()) {
        SmsManager smsManager = SmsManager.getDefault();
        smsManager.sendTextMessage(phoneNumber, null, message, null, null);

        // Show a message that the SMS is sent
        statusTextView.setText("Message Sent. Waiting for delivery report...");
    }
}

@Override
protected void onStart() {
    super.onStart();
    // Register the broadcast receiver for delivery reports
    IntentFilter filter = new
IntentFilter("android.intent.action.SMS_DELIVER");
    registerReceiver(smsReceiver, filter);
}

@Override
protected void onStop() {
    super.onStop();
    // Unregister the receiver when the activity is stopped
    unregisterReceiver(smsReceiver);
}

```

```

// BroadcastReceiver to receive the delivery report
private final BroadcastReceiver smsReceiver = new BroadcastReceiver() {
    @Override
    public void onReceive(Context context, Intent intent) {
        String status = "SMS not delivered";
        // Check the result code to determine if the message was delivered
        switch (getResultCode()) {
            case Activity.RESULT_OK:
                status = "SMS delivered successfully!";
                break;
            case SmsManager.RESULT_ERROR_GENERIC_FAILURE:
                status = "SMS delivery failed!";
                break;
            case SmsManager.RESULT_ERROR_NO_SERVICE:
                status = "No service available!";
                break;
            case SmsManager.RESULT_ERROR_NULL_PDU:
                status = "Error with the PDU!";
                break;
        }
        statusTextView.setText(status);
    }
};
}

```

7. **Create sample application with login module (Check username and password). On successful login, pass username to next screen And on failing login, alert user using Toast (Hint :Use Login(username, password) Table.)**

DBHelper.java

```

package com.example.loginapp;

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

```

```

public class DBHelper extends SQLiteOpenHelper {

    private static final String DATABASE_NAME = "LoginDB";
    private static final int DATABASE_VERSION = 1;
    private static final String TABLE_NAME = "Login";
    private static final String COLUMN_USERNAME = "username";
    private static final String COLUMN_PASSWORD = "password";

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

    @Override
    public void onCreate(SQLiteDatabase db) {
        String createTable = "CREATE TABLE " + TABLE_NAME + " (" +
            COLUMN_USERNAME + " TEXT PRIMARY KEY, " +
            COLUMN_PASSWORD + " TEXT)";
        db.execSQL(createTable);

        // Insert default users
        ContentValues values = new ContentValues();
        values.put(COLUMN_USERNAME, "admin");
        values.put(COLUMN_PASSWORD, "12345");
        db.insert(TABLE_NAME, null, values);
    }

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

    public boolean checkUser(String username, String password) {
        SQLiteDatabase db = this.getReadableDatabase();
        String query = "SELECT * FROM " + TABLE_NAME + " WHERE " +
            COLUMN_USERNAME + "=? AND " + COLUMN_PASSWORD + "=?";
        Cursor cursor = db.rawQuery(query, new String[]{username, password});
        boolean exists = cursor.getCount() > 0;
        cursor.close();
        db.close();
    }
}

```

```
        return exists;
    }
}
```

Activity_login.xml

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:padding="20dp"
    android:orientation="vertical">

    <EditText
        android:id="@+id/etUsername"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter Username" />

    <EditText
        android:id="@+id/etPassword"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter Password"
        android:inputType="textPassword" />

    <Button
        android:id="@+id/btnLogin"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Login" />
</LinearLayout>
```

LoginActivity.java

```
package com.example.loginapp;

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



```

import androidx.appcompat.app.AppCompatActivity;

public class LoginActivity extends AppCompatActivity {
    EditText etUsername, etPassword;
    Button btnLogin;
    DBHelper dbHelper;

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

        etUsername = findViewById(R.id.etUsername);
        etPassword = findViewById(R.id.etPassword);
        btnLogin = findViewById(R.id.btnLogin);

        dbHelper = new DBHelper(this);

        btnLogin.setOnClickListener(v -> {
            String username = etUsername.getText().toString().trim();
            String password = etPassword.getText().toString().trim();

            if (dbHelper.checkUser(username, password)) {
                // Successful login
                Toast.makeText(LoginActivity.this, "Login Successful",
                    Toast.LENGTH_SHORT).show();
                Intent intent = new Intent(LoginActivity.this, WelcomeActivity.class);
                intent.putExtra("USERNAME", username);
                startActivity(intent);
            } else {
                // Failed login
                Toast.makeText(LoginActivity.this, "Invalid Username or Password",
                    Toast.LENGTH_SHORT).show();
            }
        });
    }
}

```

Activity_welcome.xml

```

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

```

```

        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:padding="20dp"
        android:orientation="vertical">

        <TextView
            android:id="@+id/tvWelcome"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="Welcome!"
            android:textSize="24sp"
            android:textStyle="bold" />
    </LinearLayout>

```

WelcomeActivity.java

```

package com.example.loginapp;

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

public class WelcomeActivity extends AppCompatActivity {
    TextView tvWelcome;

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

        tvWelcome = findViewById(R.id.tvWelcome);

        String username = getIntent().getStringExtra("USERNAME");
        tvWelcome.setText("Welcome, " + username + "!");
    }
}

```

8. **Create Table project (pno, p_name, ptype, duration) and employee (id, e_name, qualification, joindate) Project – employee have many to many relationship. Using**

database perform following operation. 1) Add new record into table. 3. Accept a project name from user and display information of employees working on the project.

DBHelper.java

```
package com.example.projectemployeeapp;
```

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

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

```
public class DBHelper extends SQLiteOpenHelper {
```

```
    private static final String DATABASE_NAME = "CompanyDB";
    private static final int DATABASE_VERSION = 1;
```

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

```
    @Override
```

```
    public void onCreate(SQLiteDatabase db) {
        // Create Project Table
        db.execSQL("CREATE TABLE Project (pno INTEGER PRIMARY KEY, p_name TEXT, ptype TEXT, duration TEXT)");
```

```
        // Create Employee Table
        db.execSQL("CREATE TABLE Employee (id INTEGER PRIMARY KEY, e_name TEXT, qualification TEXT, joindate TEXT)");
```

```
        // Create Mapping Table (Many-to-Many)
        db.execSQL("CREATE TABLE Project_Employee (pno INTEGER, id INTEGER, "
+
        "FOREIGN KEY(pno) REFERENCES Project(pno), " +
        "FOREIGN KEY(id) REFERENCES Employee(id))");
    }
```

```

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

// Insert Project
public boolean addProject(int pno, String p_name, String ptype, String duration) {
    SQLiteDatabase db = this.getWritableDatabase();
    ContentValues values = new ContentValues();
    values.put("pno", pno);
    values.put("p_name", p_name);
    values.put("ptype", ptype);
    values.put("duration", duration);
    long result = db.insert("Project", null, values);
    return result != -1;
}

// Insert Employee
public boolean addEmployee(int id, String e_name, String qualification, String
joindate) {
    SQLiteDatabase db = this.getWritableDatabase();
    ContentValues values = new ContentValues();
    values.put("id", id);
    values.put("e_name", e_name);
    values.put("qualification", qualification);
    values.put("joindate", joindate);
    long result = db.insert("Employee", null, values);
    return result != -1;
}

// Assign Employee to Project
public boolean assignEmployeeToProject(int pno, int id) {
    SQLiteDatabase db = this.getWritableDatabase();
    ContentValues values = new ContentValues();
    values.put("pno", pno);
    values.put("id", id);

```

```

        long result = db.insert("Project_Employee", null, values);
        return result != -1;
    }

    // Get Employees by Project Name
    public List<String> getEmployeesByProject(String projectName) {
        SQLiteDatabase db = this.getReadableDatabase();
        List<String> employees = new ArrayList<>();
        String query = "SELECT e.e_name, e.qualification, e.joindate FROM Employee e "
+
        "JOIN Project_Employee pe ON e.id = pe.id " +
        "JOIN Project p ON pe.pno = p.pno WHERE p.p_name=?";
        Cursor cursor = db.rawQuery(query, new String[] {projectName});

        if (cursor.moveToFirst()) {
            do {
                String e_name = cursor.getString(0);
                String qualification = cursor.getString(1);
                String joindate = cursor.getString(2);
                employees.add(e_name + " | " + qualification + " | " + joindate);
            } while (cursor.moveToNext());
        }
        cursor.close();
        return employees;
    }
}

```

Activity_main.xml

```

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:padding="20dp"
    android:orientation="vertical">

    <EditText android:id="@+id/etProjectName" android:layout_width="match_parent"
    android:layout_height="wrap_content" android:hint="Project Name" />

    <Button android:id="@+id/btnSearch" android:layout_width="match_parent"
    android:layout_height="wrap_content" android:text="Search Employees" />

```

```
<TextView android:id="@+id/tvEmployees" android:layout_width="match_parent"
android:layout_height="wrap_content" android:text="Employees will appear here" />
</LinearLayout>
```

MainActivity.java

```
package com.example.projectemployeeapp;
```

```
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;
import java.util.List;
```

```
public class MainActivity extends AppCompatActivity {
    EditText etProjectName;
    Button btnSearch;
    TextView tvEmployees;
    DBHelper dbHelper;
```

```
@Override
```

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

```
    etProjectName = findViewById(R.id.etProjectName);
    btnSearch = findViewById(R.id.btnSearch);
    tvEmployees = findViewById(R.id.tvEmployees);
```

```
    dbHelper = new DBHelper(this);
```

```
// Insert Sample Data
```

```
    dbHelper.addProject(101, "Project Alpha", "Software", "6 Months");
    dbHelper.addEmployee(1, "Alice", "B.Tech", "2023-01-10");
    dbHelper.assignEmployeeToProject(101, 1);
```

```
    btnSearch.setOnClickListener(v -> {
        String projectName = etProjectName.getText().toString().trim();
        if (projectName.isEmpty()) {
```

```

        Toast.makeText(MainActivity.this, "Enter project name",
        Toast.LENGTH_SHORT).show();
        return;
    }
    List<String> employees = dbHelper.getEmployeesByProject(projectName);
    if (employees.isEmpty()) {
        tvEmployees.setText("No employees found for this project.");
    } else {
        StringBuilder result = new StringBuilder();
        for (String emp : employees) {
            result.append(emp).append("\n");
        }
        tvEmployees.setText(result.toString());
    }
    });
}
}

```

9. Create application to design login form, validate it. Write and send email with appropriate message.

DBHelper.java

```

package com.example.loginemailapp;

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

public class DBHelper extends SQLiteOpenHelper {
    private static final String DATABASE_NAME = "UserDB";
    private static final int DATABASE_VERSION = 1;

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

    @Override
    public void onCreate(SQLiteDatabase db) {

```

```
db.execSQL("CREATE TABLE Users (id INTEGER PRIMARY KEY  
AUTOINCREMENT, username TEXT, password TEXT, email TEXT)");
```

```
// Insert Sample User  
ContentValues values = new ContentValues();  
values.put("username", "admin");  
values.put("password", "1234");  
values.put("email", "admin@example.com");  
db.insert("Users", null, values);  
}
```

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

```
// Validate login  
public String validateUser(String username, String password) {  
    SQLiteDatabase db = this.getReadableDatabase();  
    Cursor cursor = db.rawQuery("SELECT email FROM Users WHERE username=?  
AND password=?", new String[]{username, password});  
  
    if (cursor.moveToFirst()) {  
        return cursor.getString(0); // Return email if user exists  
    }  
    cursor.close();  
    return null; // Invalid login  
}  
}
```

Activity_main.xml

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
    android:layout_width="match_parent"  
    android:layout_height="match_parent"  
    android:padding="20dp"  
    android:orientation="vertical">  
  
    <EditText android:id="@+id/etUsername" android:layout_width="match_parent"  
        android:layout_height="wrap_content" android:hint="Username"/>
```



```

        <EditText android:id="@+id/etPassword" android:layout_width="match_parent"
        android:layout_height="wrap_content" android:hint="Password"
        android:inputType="textPassword"/>
        <Button android:id="@+id/btnLogin" android:layout_width="match_parent"
        android:layout_height="wrap_content" android:text="Login"/>
    </LinearLayout>

```

MainActivity.java

```
package com.example.loginemailapp;
```

```

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;

```

```

public class MainActivity extends AppCompatActivity {
    EditText etUsername, etPassword;
    Button btnLogin;
    DBHelper dbHelper;

```

```
@Override
```

```

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

```

```

        etUsername = findViewById(R.id.etUsername);
        etPassword = findViewById(R.id.etPassword);
        btnLogin = findViewById(R.id.btnLogin);
        dbHelper = new DBHelper(this);

```

```

        btnLogin.setOnClickListener(v -> {
            String username = etUsername.getText().toString().trim();
            String password = etPassword.getText().toString().trim();

```

```

            if (username.isEmpty() || password.isEmpty()) {
                Toast.makeText(MainActivity.this, "Enter Username and Password",
                Toast.LENGTH_SHORT).show();
            }

```

```

        return;
    }

    String email = dbHelper.validateUser(username, password);
    if (email != null) {
        Intent intent = new Intent(MainActivity.this, EmailActivity.class);
        intent.putExtra("username", username);
        intent.putExtra("email", email);
        startActivity(intent);
    } else {
        Toast.makeText(MainActivity.this, "Invalid Credentials!",
            Toast.LENGTH_SHORT).show();
    }
});
}
}

```

Activity_email.xml

```

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:padding="20dp"
    android:orientation="vertical">

    <TextView android:id="@+id/tvWelcome" android:layout_width="match_parent"
    android:layout_height="wrap_content" android:textSize="18sp"/>

    <Button android:id="@+id/btnSendEmail" android:layout_width="match_parent"
    android:layout_height="wrap_content" android:text="Send Welcome Email"/>

</LinearLayout>

```

EmailActivity.java

```

package com.example.loginemailapp;

import android.os.AsyncTask;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;
import android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;

```

```

import java.util.Properties;
import javax.mail.Authenticator;
import javax.mail.Message;
import javax.mail.MessagingException;
import javax.mail.PasswordAuthentication;
import javax.mail.Session;
import javax.mail.Transport;
import javax.mail.internet.InternetAddress;
import javax.mail.internet.MimeMessage;

public class EmailActivity extends AppCompatActivity {
    TextView tvWelcome;
    Button btnSendEmail;
    String email, username;

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

        tvWelcome = findViewById(R.id.tvWelcome);
        btnSendEmail = findViewById(R.id.btnSendEmail);

        username = getIntent().getStringExtra("username");
        email = getIntent().getStringExtra("email");

        tvWelcome.setText("Welcome, " + username + "!");

        btnSendEmail.setOnClickListener(v -> {
            String subject = "Login Successful!";
            String message = "Hello " + username + ",\n\nYour login was\nsuccessful.\n\nThank you!";
            new SendEmailTask().execute(email, subject, message);
        });
    }

    private class SendEmailTask extends AsyncTask<String, Void, Boolean> {
        @Override
        protected Boolean doInBackground(String... params) {

```

```

try {
    String toEmail = params[0];
    String subject = params[1];
    String message = params[2];

    final String senderEmail = "your-email@gmail.com";
    final String senderPassword = "your-email-password";

    Properties props = new Properties();
    props.put("mail.smtp.host", "smtp.gmail.com");
    props.put("mail.smtp.socketFactory.port", "465");
    props.put("mail.smtp.socketFactory.class", "javax.net.ssl.SSLSocketFactory");
    props.put("mail.smtp.auth", "true");
    props.put("mail.smtp.port", "465");

    Session session = Session.getInstance(props, new Authenticator() {
        protected PasswordAuthentication getPasswordAuthentication() {
            return new PasswordAuthentication(senderEmail, senderPassword);
        }
    });

    Message mimeTypeMessage = new MimeMessage(session);
    mimeTypeMessage.setFrom(new InternetAddress(senderEmail));
    mimeTypeMessage.setRecipients(Message.RecipientType.TO,
InternetAddress.parse(toEmail));
    mimeTypeMessage.setSubject(subject);
    mimeTypeMessage.setText(message);
    Transport.send(mimeTypeMessage);
    return true;
} catch (MessagingException e) {
    e.printStackTrace();
    return false;
}
}

@Override
protected void onPostExecute(Boolean result) {
    if (result) {
        Toast.makeText(EmailActivity.this, "Email Sent Successfully!",
Toast.LENGTH_LONG).show();
    }
}

```

```

        } else {
            Toast.makeText(EmailActivity.this, "Failed to send email",
                Toast.LENGTH_LONG).show();
        }
    }
}
}
}

```

10. Create application to send SMS with image and contact as attachment.

AndroidManifest.xml

```

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

```

Activity_main.xml

```

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:padding="20dp"
    android:orientation="vertical">

    <EditText android:id="@+id/etPhoneNumber"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter Phone Number"
        android:inputType="phone"/>

    <EditText android:id="@+id/etMessage"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Enter Message"/>

    <Button android:id="@+id/btnSendSMS"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Send SMS"/>

```

```

<Button android:id="@+id/btnPickImage"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Pick Image"/>

<Button android:id="@+id/btnPickContact"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Pick Contact"/>

<Button android:id="@+id/btnSendMMS"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Send MMS"/>
</LinearLayout>

```

MainActivity.java

```

package com.example.smsmmsapp;

import android.Manifest;
import android.content.ContentResolver;
import android.content.Intent;
import android.database.Cursor;
import android.net.Uri;
import android.os.Bundle;
import android.provider.ContactsContract;
import android.provider.MediaStore;
import android.telephony.SmsManager;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;
import androidx.activity.result.ActivityResultLauncher;
import androidx.activity.result.contract.ActivityResultContracts;
import androidx.annotation.Nullable;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.ActivityCompat;

public class MainActivity extends AppCompatActivity {
    EditText etPhoneNumber, etMessage;

```

```

Button btnSendSMS, btnPickImage, btnPickContact, btnSendMMS;
Uri selectedImageUri = null;
String selectedContactNumber = "";

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

    etPhoneNumber = findViewById(R.id.etPhoneNumber);
    etMessage = findViewById(R.id.etMessage);
    btnSendSMS = findViewById(R.id.btnSendSMS);
    btnPickImage = findViewById(R.id.btnPickImage);
    btnPickContact = findViewById(R.id.btnPickContact);
    btnSendMMS = findViewById(R.id.btnSendMMS);

    // Request Permissions
    ActivityCompat.requestPermissions(this, new
String[] {Manifest.permission.SEND_SMS, Manifest.permission.READ_CONTACTS,
Manifest.permission.READ_EXTERNAL_STORAGE}, 1);

    // Send SMS
    btnSendSMS.setOnClickListener(v -> sendSMS());

    // Pick Image
    btnPickImage.setOnClickListener(v -> pickImage());

    // Pick Contact
    btnPickContact.setOnClickListener(v -> pickContact());

    // Send MMS
    btnSendMMS.setOnClickListener(v -> sendMMS());
}

// 📌 1. Send SMS using SmsManager
private void sendSMS() {
    String phoneNumber = etPhoneNumber.getText().toString().trim();
    String message = etMessage.getText().toString().trim();

    if (phoneNumber.isEmpty() || message.isEmpty()) {

```

```

        Toast.makeText(this, "Enter phone number and message",
Toast.LENGTH_SHORT).show();
        return;
    }

    try {
        SmsManager smsManager = SmsManager.getDefault();
        smsManager.sendTextMessage(phoneNumber, null, message, null, null);
        Toast.makeText(this, "SMS Sent Successfully!",
Toast.LENGTH_SHORT).show();
    } catch (Exception e) {
        Toast.makeText(this, "Failed to send SMS", Toast.LENGTH_SHORT).show();
        e.printStackTrace();
    }
}

```

```

// 📌 2. Pick Image from Gallery
private void pickImage() {
    Intent intent = new Intent(Intent.ACTION_PICK,
MediaStore.Images.Media.EXTERNAL_CONTENT_URI);
    pickImageLauncher.launch(intent);
}

private final ActivityResultLauncher<Intent> pickImageLauncher =
registerForActivityResult(
    new ActivityResultContracts.StartActivityForResult(),
    result -> {
        if (result.getResultCode() == RESULT_OK && result.getData() != null) {
            selectedImageUri = result.getData().getData();
            Toast.makeText(this, "Image Selected", Toast.LENGTH_SHORT).show();
        }
    }
);

```

```

// 📌 3. Pick Contact from Phonebook
private void pickContact() {
    Intent intent = new Intent(Intent.ACTION_PICK,
ContactsContract.CommonDataKinds.Phone.CONTENT_URI);
    pickContactLauncher.launch(intent);
}

```



```

        private final ActivityResultLauncher<Intent> pickContactLauncher =
registerForActivityResult(
    new ActivityResultContracts.StartActivityForResult(),
    result -> {
        if (result.getResultCode() == RESULT_OK && result.getData() != null) {
            Uri contactUri = result.getData().getData();
            retrieveContactNumber(contactUri);
        }
    }
);

```

```

private void retrieveContactNumber(Uri contactUri) {
    ContentResolver contentResolver = getContentResolver();
    Cursor cursor = contentResolver.query(contactUri, null, null, null, null);
    if (cursor != null && cursor.moveToFirst()) {
        int columnIndex =
cursor.getColumnIndex(ContactsContract.CommonDataKinds.Phone.NUMBER);
        selectedContactNumber = cursor.getString(columnIndex);
        etPhoneNumber.setText(selectedContactNumber);
        cursor.close();
        Toast.makeText(this, "Contact Selected: " + selectedContactNumber,
Toast.LENGTH_SHORT).show();
    }
}

```

// 📧 4. Send MMS with Image and Contact

```

private void sendMMS() {
    String phoneNumber = etPhoneNumber.getText().toString().trim();
    String message = etMessage.getText().toString().trim();

    if (phoneNumber.isEmpty()) {
        Toast.makeText(this, "Enter phone number", Toast.LENGTH_SHORT).show();
        return;
    }

    Intent intent = new Intent(Intent.ACTION_SEND);
    intent.setType("image/*");
    intent.putExtra("address", phoneNumber);
    intent.putExtra(Intent.EXTRA_TEXT, message);
}

```

```
// Attach Image
if (selectedImageUri != null) {
    intent.putExtra(Intent.EXTRA_STREAM, selectedImageUri);
}

// Attach Contact
if (!selectedContactNumber.isEmpty()) {
    intent.putExtra(Intent.EXTRA_PHONE_NUMBER, selectedContactNumber);
}

startActivity(Intent.createChooser(intent, "Send MMS via"));
}
}
```