11. Consider Practical no.6 for Registration Activity and StudentData Activity.As StudentData activity shows students registration details. Now add four buttons on this activity layout namely “Save”, “Edit”, “Delete” and “View”.Implement SQLite database connection with BookBankApp using SQLiteDatabase and SQLiteOpenHelper classes.Perform below database operation as below:

- While clicking on a button “Save”, store student’s registration detail into database table tblregistration.
- While clicking on a button “Edit”, allow to change student mobile no.
- While clicking on a button “Delete”, remove this student data from database table.
- While clicking on a button “View”, move to next activity called StudentsListActivity.

→StudentDataActivity.java

```
package com.example.book_bank;

import android.content.ContentValues;
import android.content.DialogInterface;
import android.content.Intent;
import android.database.sqlite.SQLiteDatabase;
import android.os.Bundle;
import android.text.InputType;
import android.view.View;
import android.widget.*;
        import androidx.appcompat.app.AlertDialog;
import androidx.appcompat.app.AppCompatActivity;

public class StudentDataActivity extends AppCompatActivity {

    TextView tvStudentData;
    EditText etEditMobile;
    Button btnSave, btnEdit, btnDelete, btnView, btnUpdate;
    DBHelper db;

    String enrollment, name, semester, mobile, email, programme, gender,
interests;

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

        tvStudentData = findViewById(R.id.tvDisplay);
        etEditMobile = findViewById(R.id.etEditMobile);
        btnSave = findViewById(R.id.btnSave);
        btnEdit = findViewById(R.id.btnEdit);
        btnUpdate = findViewById(R.id.btnUpdate);
        btnDelete = findViewById(R.id.btnDelete);
        btnView = findViewById(R.id.btnView);

        db = new DBHelper(this);
```

```
        // Get data from Intent
        Intent i = getIntent();
        enrollment = i.getStringExtra("enrollment");
        name = i.getStringExtra("name");
        semester = i.getStringExtra("semester");
        mobile = i.getStringExtra("mobile");
        email = i.getStringExtra("email");
        programme = i.getStringExtra("programme");
        gender = i.getStringExtra("gender");
        interests = i.getStringExtra("interests");

        // Display data
        tvStudentData.setText(
                "Enrollment: " + enrollment + "\n" +
                        "Name: " + name + "\n" +
                        "Semester: " + semester + "\n" +
                        "Mobile: " + mobile + "\n" +
                        "Email: " + email + "\n" +
                        "Programme: " + programme + "\n" +
                        "Gender: " + gender + "\n" +
                        "Interests: " + interests
        );

        // Save data to DB
        btnSave.setOnClickListener(v -> {
            db.insertStudent(enrollment, name, semester, mobile, email,
programme, gender, interests);
            Toast.makeText(this, "Saved to DB", Toast.LENGTH_SHORT).show();
        });

        btnEdit.setOnClickListener(v -> {
            etEditMobile.setVisibility(View.VISIBLE);
            btnUpdate.setVisibility(View.VISIBLE);
            etEditMobile.setText(mobile);
        });

        // Update mobile number in DB
        btnUpdate.setOnClickListener(v -> {
            String newMobile = etEditMobile.getText().toString().trim();

            if (!newMobile.isEmpty()) {
                db.updateMobile(enrollment, newMobile);
                mobile = newMobile; // update current value
        Toast.makeText(this, "Mobile updated", Toast.LENGTH_SHORT).show();

                // Update displayed data
                tvStudentData.setText(
                        "Enrollment: " + enrollment + "\n" +
                                "Name: " + name + "\n" +
                                "Semester: " + semester + "\n" +
                                "Mobile: " + newMobile + "\n" +
                                "Email: " + email + "\n" +
                                "Programme: " + programme + "\n" +
                                "Gender: " + gender + "\n" +
                                "Interests: " + interests
                );
```

```
                etEditMobile.setVisibility(View.GONE);
                btnUpdate.setVisibility(View.GONE);
            } else {
                Toast.makeText(this, "Enter a valid mobile number",
Toast.LENGTH_SHORT).show();
            }
        });

        // Delete student
        btnDelete.setOnClickListener(v -> {
            db.deleteStudent(enrollment);
            Toast.makeText(this, "Student deleted",
Toast.LENGTH_SHORT).show();
        });

        // View all students
        btnView.setOnClickListener(v -> {
            Intent intent = new Intent(this, StudentsListActivity.class);
            startActivity(intent);
        });
    }
}
```

→DBHelper.java

```
package com.example.book_bank;

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 {

    public DBHelper(Context context) {
        super(context, "bookbank.db", null, 1);
    }

    @Override
    public void onCreate(SQLiteDatabase db) {
        db.execSQL("CREATE TABLE tblregistration(enrollment TEXT PRIMARY KEY,
name TEXT, semester TEXT, mobile TEXT, email TEXT, programme TEXT, gender
TEXT, interests TEXT)");
    }

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

    public void insertStudent(String enrollment, String name, String
semester, String mobile, String email, String programme, String gender,
String interests) {
```

```
        SQLiteDatabase db = this.getWritableDatabase();
        ContentValues values = new ContentValues();
        values.put("enrollment", enrollment);
        values.put("name", name);
        values.put("semester", semester);
        values.put("mobile", mobile);
        values.put("email", email);
        values.put("programme", programme);
        values.put("gender", gender);
        values.put("interests", interests);
        db.insert("tblregistration", null, values);
    }

    public void updateMobile(String enrollment, String newMobile) {
        SQLiteDatabase db = this.getWritableDatabase();
        ContentValues values = new ContentValues();
        values.put("mobile", newMobile);
        db.update("tblregistration", values, "enrollment=?", new
String[]{enrollment});
    }

    public void deleteStudent(String enrollment) {
        SQLiteDatabase db = this.getWritableDatabase();
        db.delete("tblregistration", "enrollment=?", new
String[]{enrollment});
    }

    public Cursor getAllStudents() {
        SQLiteDatabase db = this.getReadableDatabase();
        return db.rawQuery("SELECT * FROM tblregistration", null);
    }
}
```

12. Consider StudentListActvity created in Practical No.11 and will display all
students' registration details in proper format.
(Note : Students details must be retrieve from tblregistration table)

→

activity_students_list.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">

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

</LinearLayout>
```

StudentsListActivity.java

```
package com.example.book_bank;

import android.annotation.SuppressLint;
import android.database.Cursor;
import android.os.Bundle;
import android.widget.ArrayAdapter;
import android.widget.ListView;
import androidx.appcompat.app.AppCompatActivity;

import java.util.ArrayList;

public class StudentsListActivity extends AppCompatActivity {

    ListView listViewStudents;
    DBHelper db;
    ArrayList<String> studentList;
    ArrayAdapter<String> adapter;

    @SuppressLint("MissingInflatedId")
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_students_list);

        listViewStudents = findViewById(R.id.listView);
        db = new DBHelper(this);
        studentList = new ArrayList<>();

        Cursor c = db.getAllStudents();
        while (c.moveToNext()) {
```

```
            String data = "Enrollment: " + c.getString(0) + "\n" +
                    "Name: " + c.getString(1) + "\n" +
                    "Semester: " + c.getString(2) + "\n" +
                    "Mobile: " + c.getString(3) + "\n" +
                    "Email: " + c.getString(4) + "\n" +
                    "Programme: " + c.getString(5) + "\n" +
                    "Gender: " + c.getString(6) + "\n" +
                    "Interests: " + c.getString(7);
            studentList.add(data);
        }

        adapter = new ArrayAdapter<>(this,
android.R.layout.simple_list_item_1, studentList);
        listViewStudents.setAdapter(adapter);
    }
}
```

13. Consider Practical No. 4 for DashboardActivity and extends its functionality as below:

- **Create an Option menu/Context menu on DashboardActivity with options**
- **such as “Settings”, “Profile” and “Logout”. Display appropriate Toast**
- **message upon selected option from the menu.**

→

Res-->menu->menu_dashboard.xml

menu_dashboard.xml

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android">
    <item
        android:id="@+id/menu_settings"
        android:title="Settings" />
    <item
        android:id="@+id/menu_profile"
        android:title="Profile" />
    <item
        android:id="@+id/menu_logout"
        android:title="Logout" />
</menu>
```

DashBoardActivity.java

```
@Override
public boolean onCreateOptionsMenu(Menu menu) {
    getMenuInflater().inflate(R.menu.menu_dashboard, menu);
    return true;
}

@Override
public boolean onOptionsItemSelected(MenuItem item) {
    int id = item.getItemId();

    if (id == R.id.menu_settings) {
        Toast.makeText(this, "Settings selected", Toast.LENGTH_SHORT).show();
        return true;
    } else if (id == R.id.menu_profile) {
        Toast.makeText(this, "Profile selected", Toast.LENGTH_SHORT).show();
        return true;
    } else if (id == R.id.menu_logout) {
        Toast.makeText(this, "Logout selected", Toast.LENGTH_SHORT).show();
        return true;
    }

    return super.onOptionsItemSelected(item);
}
```

14. Implement RecyclerView and cardView for displaying books details suchas Book ID, Title and Price.

→

res/layout/item_book.xml

->

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.cardview.widget.CardView
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:card_view="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    card_view:cardElevation="4dp"
    card_view:cardCornerRadius="8dp"
    android:layout_margin="8dp">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical"
        android:padding="16dp">

        <TextView
            android:id="@+id/tvBookID"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Book ID: 101"
            android:textStyle="bold"
            android:textSize="16sp" />

        <TextView
            android:id="@+id/tvTitle"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Title: Introduction to Android"
            android:textSize="14sp"
            android:layout_marginTop="4dp" />

        <TextView
            android:id="@+id/tvPrice"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Price: $499"
            android:textSize="14sp"
            android:layout_marginTop="4dp" />

    </LinearLayout>

</androidx.cardview.widget.CardView>
```

res/layout/activity_books_list.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">

    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/recyclerBooks"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
       />

</LinearLayout>
```

Create Book Model class : Book.java

```
package com.example.book_bank;

public class Book {
    private String bookID;
    private String title;
    private String price;

    public Book(String bookID, String title, String price) {
        this.bookID = bookID;
        this.title = title;
        this.price = price;
    }

    public String getBookID() {
        return bookID;
    }

    public String getTitle() {
        return title;
    }

    public String getPrice() {
        return price;
    }
}
```

Create RecyclerView Adapter

Create: `BooksAdapter.java`

```
package com.example.book_bank;

import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.TextView;

import androidx.annotation.NonNull;
import androidx.recyclerview.widget.RecyclerView;

import java.util.List;

public class BookAdapter extends
RecyclerView.Adapter<BookAdapter.BookViewHolder> {

    private List<Book> bookList;

    public BookAdapter(List<Book> bookList) {
        this.bookList = bookList;
    }

    @NonNull
    @Override
    public BookViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int
viewType) {
        View view = LayoutInflater.from(parent.getContext())
                .inflate(R.layout.item_book, parent, false);
        return new BookViewHolder(view);
    }

    @Override
    public void onBindViewHolder(@NonNull BookViewHolder holder, int
position) {
        Book book = bookList.get(position);
        holder.tvBookID.setText("Book ID: " + book.getBookID());
        holder.tvTitle.setText("Title: " + book.getTitle());
        holder.tvPrice.setText("Price: " + book.getPrice());
    }

    @Override
    public int getItemCount() {
        return bookList.size();
    }

    public static class BookViewHolder extends RecyclerView.ViewHolder {
        TextView tvBookID, tvTitle, tvPrice;

        public BookViewHolder(@NonNull View itemView) {
            super(itemView);
            tvBookID = itemView.findViewById(R.id.tvBookID);
            tvTitle = itemView.findViewById(R.id.tvTitle);
            tvPrice = itemView.findViewById(R.id.tvPrice);
        }
    }
}
```

BooksListActivity.java
→

```
package com.example.book_bank;

import android.content.SharedPreferences;
import android.os.Bundle;

import androidx.appcompat.app.AppCompatActivity;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;

import java.util.ArrayList;

public class BooksListActivity extends AppCompatActivity {

    RecyclerView recyclerBooks;
    BookAdapter adapter;
    ArrayList<Book> booksList;

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

        recyclerBooks = findViewById(R.id.recyclerBooks);
        recyclerBooks.setLayoutManager(new LinearLayoutManager(this));

        booksList = new ArrayList<>();

        // Retrieve book data from SharedPreferences
        SharedPreferences sharedPreferences =
getSharedPreferences("BookPrefs", MODE_PRIVATE);
        String id = sharedPreferences.getString("BookID", null);
        String title = sharedPreferences.getString("Title", null);
        String price = sharedPreferences.getString("Price", null);

        // Check if data exists and add to list
        if (id != null && title != null && price != null) {
            booksList.add(new Book(id, title, price));
        }

        // Setup adapter and RecyclerView
        adapter = new BookAdapter(booksList);
        recyclerBooks.setAdapter(adapter);
    }
}
```