

Android End Sem Lab Examination

Name: Abhishek Sharma

Roll No: CYS21001

Task assigned : To create a notes app

Functionalities in the app:

1. Create an Empty Notes
2. Add Notes
3. Read the entered notes
4. Edit the notes
5. Delete the notes

Code and implementation :

AndroidManifest.xml

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

    <application
        android:allowBackup="true"
        android:dataExtractionRules="@xml/data_extraction_rules"
        android:fullBackupContent="@xml/backup_rules"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.NotesApp"
        tools:targetApi="31">
        <activity
            android:name=".NotesActivity"
            android:exported="false" />
        <activity
            android:name=".MainActivity"
            android:exported="true">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>
```



MainActivity.java

```
package com.example.notesapp;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

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

```

        // Navigate to NotesActivity when the button is clicked
        findViewById(R.id.nextButton).setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent intent = new Intent(MainActivity.this, NotesActivity.class);
                startActivity(intent);
            }
        });
    }
}

```

DatabaseHelper.java

```

package com.example.notesapp;

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 {
    // Database Name and Version
    private static final String DATABASE_NAME = "notes.db";
    private static final int DATABASE_VERSION = 1;

    // Table and Columns
    private static final String TABLE_NAME = "notes";
    private static final String COL_ID = "id";
    private static final String COL_NOTE = "note";

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

    // Create Table
    @Override
    public void onCreate(SQLiteDatabase db) {
        String createTable = "CREATE TABLE " + TABLE_NAME + " (" +
            COL_ID + " INTEGER PRIMARY KEY AUTOINCREMENT, " +
            COL_NOTE + " TEXT)";
        db.execSQL(createTable);
    }

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

```

```

// Insert a new note
public boolean addNote(String note) {
    SQLiteDatabase db = this.getWritableDatabase();
    ContentValues values = new ContentValues();
    values.put(COL_NOTE, note);
    long result = db.insert(TABLE_NAME, null, values);
    return result != -1;
}

// Fetch all notes
public Cursor getAllNotes() {
    SQLiteDatabase db = this.getReadableDatabase();
    return db.rawQuery("SELECT * FROM " + TABLE_NAME, null);
}

// Update a note by ID
public boolean updateNote(int id, String newNote) {
    SQLiteDatabase db = this.getWritableDatabase();
    ContentValues values = new ContentValues();
    values.put(COL_NOTE, newNote);
    return db.update(TABLE_NAME, values, COL_ID + " = ?", new
String[]{String.valueOf(id)}) > 0;
}

// Delete a note by ID
public boolean deleteNoteById(int id) {
    SQLiteDatabase db = this.getWritableDatabase();
    return db.delete(TABLE_NAME, COL_ID + " = ?", new String[]{String.valueOf(id)}) > 0;
}
}

```

NotesHelper.java

```

package com.example.notesapp;

import android.content.DialogInterface;
import android.database.Cursor;
import android.os.Bundle;
import android.text.InputType;
import android.view.View;
import android.widget.EditText;
import android.widget.Toast;
import androidx.appcompat.app.AlertDialog;
import androidx.appcompat.app.AppCompatActivity;

public class NotesActivity extends AppCompatActivity {

    DatabaseHelper dbHelper;

    @Override

```

```

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

    dbHelper = new DatabaseHelper(this);

    findViewById(R.id.createNoteButton).setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            createEmptyNote();
        }
    });

    findViewById(R.id.addNoteButton).setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            addNewNote();
        }
    });

    findViewById(R.id.readNoteButton).setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            showAllNotes();
        }
    });

    findViewById(R.id.editNoteButton).setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            editNote();
        }
    });

    findViewById(R.id.deleteNoteButton).setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            deleteNote();
        }
    });
}

private void createEmptyNote() {
    boolean inserted = dbHelper.addNote("");
    if (inserted) {
        Toast.makeText(this, "Empty Note Created!", Toast.LENGTH_SHORT).show();
    } else {
        Toast.makeText(this, "Error Creating Note!", Toast.LENGTH_SHORT).show();
    }
}

```

```

private void addNewNote() {
    AlertDialog.Builder builder = new AlertDialog.Builder(this);
    builder.setTitle("Enter Note");

    final EditText input = new EditText(this);
    input.setInputType(InputType.TYPE_CLASS_TEXT);
    builder.setView(input);

    builder.setPositiveButton("Add", new DialogInterface.OnClickListener() {
        @Override
        public void onClick(DialogInterface dialog, int which) {
            String noteText = input.getText().toString();
            if (dbHelper.addNote(noteText)) {
                Toast.makeText(NotesActivity.this, "Note Added!",
                    Toast.LENGTH_SHORT).show();
            } else {
                Toast.makeText(NotesActivity.this, "Error Adding Note!",
                    Toast.LENGTH_SHORT).show();
            }
        }
    });

    builder.setNegativeButton("Cancel", new DialogInterface.OnClickListener() {
        @Override
        public void onClick(DialogInterface dialog, int which) {
            dialog.cancel();
        }
    });

    builder.show();
}

private void showAllNotes() {
    Cursor cursor = dbHelper.getAllNotes();
    if (cursor.getCount() == 0) {
        Toast.makeText(this, "No Notes Available", Toast.LENGTH_SHORT).show();
        return;
    }

    StringBuilder notes = new StringBuilder();
    while (cursor.moveToNext()) {
        notes.append("ID: ").append(cursor.getInt(0))
            .append("\nNote: ").append(cursor.getString(1))
            .append("\n\n");
    }

    AlertDialog.Builder builder = new AlertDialog.Builder(this);
    builder.setTitle("All Notes");
    builder.setMessage(notes.toString());
    builder.setPositiveButton("OK", null);
    builder.show();
}

```

```

}

private void editNote() {
    AlertDialog.Builder builder = new AlertDialog.Builder(this);
    builder.setTitle("Enter Note ID to Edit");

    final EditText input = new EditText(this);
    input.setInputType(InputType.TYPE_CLASS_NUMBER);
    builder.setView(input);

    builder.setPositiveButton("Next", new DialogInterface.OnClickListener() {
        @Override
        public void onClick(DialogInterface dialog, int which) {
            int notelId = Integer.parseInt(input.getText().toString());
            showEditNoteDialog(notelId);
        }
    });

    builder.setNegativeButton("Cancel", new DialogInterface.OnClickListener() {
        @Override
        public void onClick(DialogInterface dialog, int which) {
            dialog.cancel();
        }
    });

    builder.show();
}

private void showEditNoteDialog(final int notelId) {
    AlertDialog.Builder builder = new AlertDialog.Builder(this);
    builder.setTitle("Edit Note");

    final EditText input = new EditText(this);
    input.setInputType(InputType.TYPE_CLASS_TEXT);
    builder.setView(input);

    builder.setPositiveButton("Update", new DialogInterface.OnClickListener() {
        @Override
        public void onClick(DialogInterface dialog, int which) {
            String newNote = input.getText().toString();
            if (dbHelper.updateNote(notelId, newNote)) {
                Toast.makeText(NotesActivity.this, "Note Updated!",
                    Toast.LENGTH_SHORT).show();
            } else {
                Toast.makeText(NotesActivity.this, "Error Updating Note!",
                    Toast.LENGTH_SHORT).show();
            }
        }
    });

    builder.setNegativeButton("Cancel", new DialogInterface.OnClickListener() {

```

```

        @Override
        public void onClick(DialogInterface dialog, int which) {
            dialog.cancel();
        }
    });

    builder.show();
}

private void deleteNote() {
    AlertDialog.Builder builder = new AlertDialog.Builder(this);
    builder.setTitle("Enter Note ID to Delete");

    final EditText input = new EditText(this);
    input.setInputType(InputType.TYPE_CLASS_NUMBER);
    builder.setView(input);

    builder.setPositiveButton("Delete", new DialogInterface.OnClickListener() {
        @Override
        public void onClick(DialogInterface dialog, int which) {
            int noteId = Integer.parseInt(input.getText().toString());
            if (dbHelper.deleteNoteById(noteId)) {
                Toast.makeText(NotesActivity.this, "Note Deleted!",
                    Toast.LENGTH_SHORT).show();
            } else {
                Toast.makeText(NotesActivity.this, "Error Deleting Note!",
                    Toast.LENGTH_SHORT).show();
            }
        }
    });

    builder.setNegativeButton("Cancel", new DialogInterface.OnClickListener() {
        @Override
        public void onClick(DialogInterface dialog, int which) {
            dialog.cancel();
        }
    });

    builder.show();
}
}

```

Activity_main.xml

```

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

<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"

```



```
android:layout_height="match_parent"
tools:context=".MainActivity"
>
```

```
<androidx.constraintlayout.widget.ConstraintLayout
    android:layout_width="413dp"
    android:layout_height="85dp"
    android:background="#A2123F"
    app:layout_constraintBottom_toTopOf="@+id/constraintLayout"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent">
```

```
<TextView
    android:id="@+id/textView3"
    android:layout_width="201dp"
    android:layout_height="41dp"
    android:text="My Notes App"
    android:textAlignment="center"
    android:textColor="@color/white"
    android:textColorLink="@color/white"
    android:textSize="25dp"
    android:textStyle="bold"
    android:fontFamily=""
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.498"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.938" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

```
<androidx.constraintlayout.widget.ConstraintLayout
    android:id="@+id/constraintLayout"
    android:layout_width="409dp"
    android:layout_height="646dp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent">
```

```
<ImageView
    android:id="@+id/profileImage"
    android:layout_width="405dp"
    android:layout_height="389dp"
    android:layout_marginTop="28dp"
    android:scaleType="centerCrop"
    android:src="@drawable/your_image"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
```

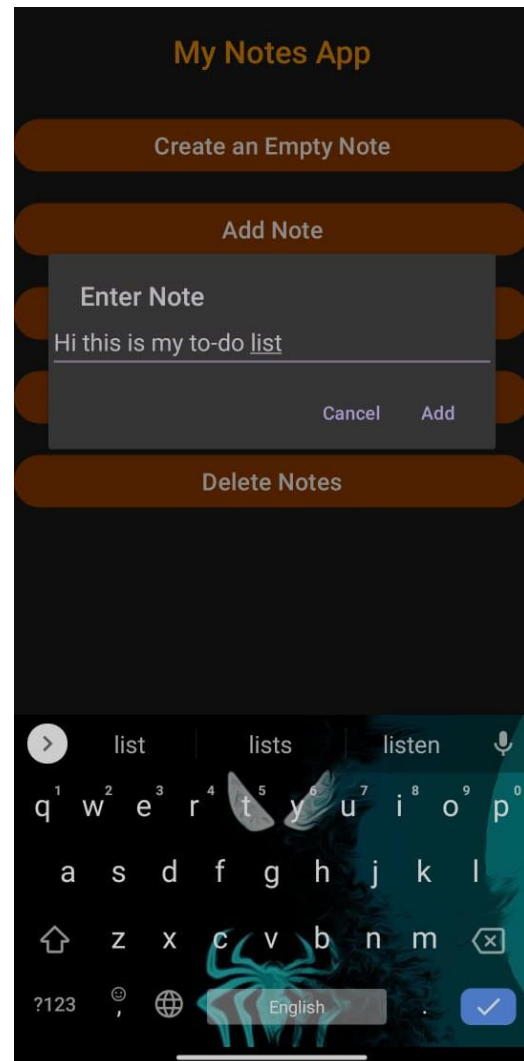
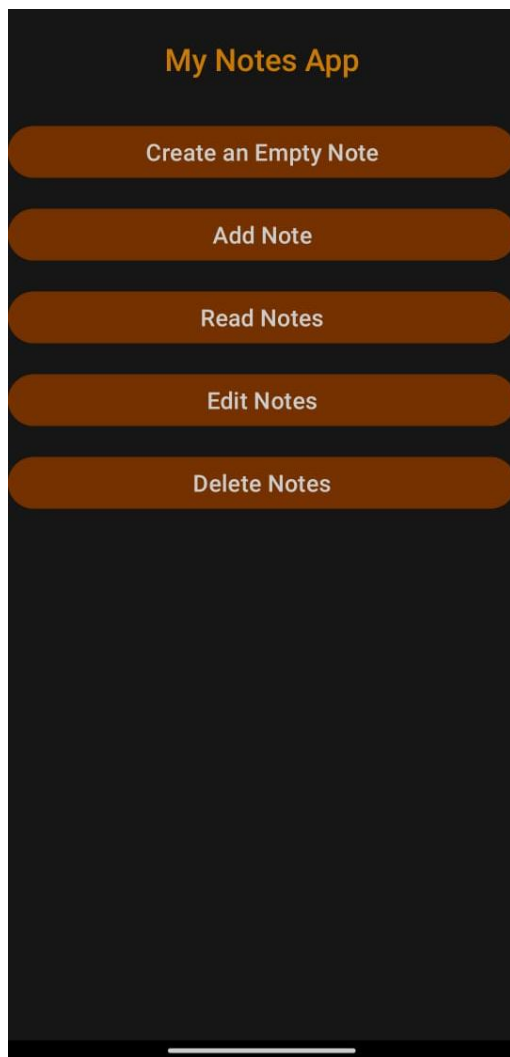
```
app:layout_constraintTop_toTopOf="parent"
/>
```

```
<TextView
    android:id="@+id/nameTextView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="56dp"
    android:text="Abhsihek Sharma"
    android:textColor="@color/black"
    android:textSize="24sp"
    android:textStyle="bold"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@id/profileImage" />
```

```
<TextView
    android:id="@+id/rollNumberTextView"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="CH.EN.U4CYS21001"
    android:textSize="20sp"
    android:textColor="@color/black"
    app:layout_constraintTop_toBottomOf="@id/nameTextView"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintEnd_toEndOf="parent" />
```

```
<Button
    android:id="@+id/nextButton"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="36dp"
    android:backgroundTint="#A2123F"
    android:text="Go to Notes"
    android:textColor="@color/white"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.891"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@id/rollNumberTextView" />
```

```
</androidx.constraintlayout.widget.ConstraintLayout>
</androidx.constraintlayout.widget.ConstraintLayout>
```



Activity_notes.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"
    android:background="@color/white"
    tools:context=".NotesActivity">

    <TextView
        android:id="@+id/titleTextView"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:text="My Notes App"
        android:textSize="24sp"
        android:textColor="@color/orange"
        android:layout_marginTop="32dp"
        android:layout_marginBottom="16dp"
        android:fontFamily="sans-serif-medium">
```

```
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintEnd_toEndOf="parent"
android:gravity="center"/>
```

```
<Button
    android:id="@+id/createNoteButton"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_marginTop="32dp"
    android:backgroundTint="@color/orange"
    android:text="Create an Empty Note"
    android:textColor="@android:color/white"
    android:textSize="18sp"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@id/titleTextView"
    app:layout_constraintHorizontal_bias="0.5"/>
```

```
<Button
    android:id="@+id/addNoteButton"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_marginTop="16dp"
    android:backgroundTint="@color/orange"
    android:text="Add Note"
    android:textColor="@android:color/white"
    android:textSize="18sp"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@id/createNoteButton"
    app:layout_constraintHorizontal_bias="0.5"/>
```

```
<Button
    android:id="@+id/readNoteButton"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_marginTop="16dp"
    android:backgroundTint="@color/orange"
    android:text="Read Notes"
    android:textColor="@android:color/white"
    android:textSize="18sp"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@id/addNoteButton"
    app:layout_constraintHorizontal_bias="0.5"/>
```

```
<Button
    android:id="@+id/editNoteButton"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
```

```

        android:layout_marginTop="16dp"
        android:backgroundTint="@color/orange"
        android:text="Edit Notes"
        android:textColor="@android:color/white"
        android:textSize="18sp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@id/readNoteButton"
        app:layout_constraintHorizontal_bias="0.5"/>
<Button
    android:id="@+id/deleteNoteButton"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_marginTop="16dp"
    android:backgroundTint="@color/orange"
    android:text="Delete Notes"
    android:textColor="@android:color/white"
    android:textSize="18sp"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@id/editNoteButton"
    app:layout_constraintHorizontal_bias="0.5"/>

</androidx.constraintlayout.widget.ConstraintLayout>

```

