SAVE STUDENT DETAILS IN ROOM DATABASE

Steps:

1. Add Room Dependencies
2. Create Entity Class(POJO, Modal, Table)
3. Create DAO Interface(CRUD OPERATION)
4. Create Room Database Class(Connection establishment)
5. Design the UI (Activity, Fragment) XML
6. Implement Activity, Fragment(.class)
7. Use Adapter For show list of student(need→Adapter java class, layout design)

Code

Steps1: Add Dependency gradle app level

```
implementation "androidx.room:room-runtime:2.6.1"
annotationProcessor "androidx.room:room-compiler:2.6.1"
```

Step2: Create Entity Class(POJO, Modal, Table)

```
import androidx.room.Entity;
import androidx.room.PrimaryKey;

@Entity
public class Student {
    @PrimaryKey(autoGenerate = true)
    public int id;

    public String name;
    public int age;

    public Student(String name, int age) {
        this.name = name;
        this.age = age;
    }
}
```

Step3: Create DAO Interface(CRUD OPERATION)

```
import androidx.room.*;
import java.util.List;
@Dao
public interface StudentDao {
    @Insert
    void insert(Student student);

    @Update
    void update(Student student);

    @Delete
    void delete(Student student);

    @Query("SELECT * FROM Student")
    List<Student> getAllStudents();
}
```

Step4: Create Room Database Class(Connection establishment)

```
import androidx.room.Database;
import androidx.room.Room;
import androidx.room.RoomDatabase;
import android.content.Context;

@Database(entities = {Student.class}, version = 1)
public abstract class StudentDatabase extends RoomDatabase {
  private static StudentDatabase instance;

  public abstract StudentDao studentDao();

  public static synchronized StudentDatabase getInstance(Context context) {
    if (instance == null) {
      instance = Room.databaseBuilder(context.getApplicationContext(),
          StudentDatabase.class, "student_database")
          .fallbackToDestructiveMigration()
          .allowMainThreadQueries() // For simplicity; not recommended for large apps
          .build();
    }
    return instance;
  }
}
```

Step5: Design the UI (Activity, Fragment) XML

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

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

  <EditText
    android:id="@+id/ageInput"
    android:hint="Age"
    android:inputType="number"
    android:layout_width="match_parent"
    android:layout_height="wrap_content" />

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

  <Button
    android:id="@+id/updateButton"
    android:text="Update"
    android:layout_width="match_parent"
    android:layout_height="wrap_content" />

  <Button
    android:id="@+id/deleteButton"
    android:text="Delete"
    android:layout_width="match_parent"
    android:layout_height="wrap_content" />

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

<androidx.recyclerview.widget.RecyclerView
  android:id="@+id/recyclerView"
  android:layout_width="match_parent"
  android:layout_height="match_parent"
  android:layout_marginTop="16dp"/>
</LinearLayout>
```

Step6: Implement Activity, Fragment(.class)

```
package com.example.studentmanager;

import android.os.Bundle;
import android.view.View;
import android.widget.*;
import androidx.appcompat.app.AppCompatActivity;
import androidx.recyclerview.widget.LinearLayoutManager;
import androidx.recyclerview.widget.RecyclerView;

import java.util.List;

public class MainActivity extends AppCompatActivity {

    EditText nameInput, ageInput;
    Button addButton, updateButton, deleteButton, viewButton;
    RecyclerView recyclerView;

    StudentAdapter adapter;
    List<Student> studentList;
    Student selectedStudent = null;

    StudentDatabase db;

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

        // Initialize database
        db = StudentDatabase.getInstance(this);

        // Bind views
        nameInput = findViewById(R.id.nameInput);
        ageInput = findViewById(R.id.ageInput);
        addButton = findViewById(R.id.addButton);
        updateButton = findViewById(R.id.updateButton);
        deleteButton = findViewById(R.id.deleteButton);
        viewButton = findViewById(R.id.viewButton);
        recyclerView = findViewById(R.id.recyclerView);

        // Setup RecyclerView
        recyclerView.setLayoutManager(new LinearLayoutManager(this));
        adapter = new StudentAdapter(null, student -> {
            selectedStudent = student;
            nameInput.setText(student.name);
            ageInput.setText(String.valueOf(student.age));
        });
        recyclerView.setAdapter(adapter);

        // Add Student
        addButton.setOnClickListener(v -> {
            String name = nameInput.getText().toString().trim();
            String ageStr = ageInput.getText().toString().trim();

            if (name.isEmpty() || ageStr.isEmpty()) {
                Toast.makeText(this, "Please enter both name and age", Toast.LENGTH_SHORT).show();
                return;
            }

            int age = Integer.parseInt(ageStr);
            Student student = new Student(name, age);
            db.studentDao().insert(student);
            Toast.makeText(this, "Student added", Toast.LENGTH_SHORT).show();
            clearInput();
            loadStudents();
        });

        // Update Student
        updateButton.setOnClickListener(v -> {
            if (selectedStudent != null) {
                String name = nameInput.getText().toString().trim();
                String ageStr = ageInput.getText().toString().trim();

                if (name.isEmpty() || ageStr.isEmpty()) {
                    Toast.makeText(this, "Please enter both name and age", Toast.LENGTH_SHORT).show();
                    return;
                }

                selectedStudent.name = name;
                selectedStudent.age = Integer.parseInt(ageStr);
                db.studentDao().update(selectedStudent);
                Toast.makeText(this, "Student updated", Toast.LENGTH_SHORT).show();
                clearInput();
                loadStudents();
            } else {
                Toast.makeText(this, "Select a student to update", Toast.LENGTH_SHORT).show();
            }
        });

        // Delete Student
        deleteButton.setOnClickListener(v -> {
            if (selectedStudent != null) {
                db.studentDao().delete(selectedStudent);
                Toast.makeText(this, "Student deleted", Toast.LENGTH_SHORT).show();
                clearInput();
                loadStudents();
            } else {
                Toast.makeText(this, "Select a student to delete", Toast.LENGTH_SHORT).show();
            }
        });

        // View All Students
        viewButton.setOnClickListener(v -> loadStudents());
      }

      // Load all students into RecyclerView
      private void loadStudents() {
        studentList = db.studentDao().getAllStudents();
        adapter.setStudents(studentList);
      }

      // Clear input fields and selection
      private void clearInput() {
        nameInput.setText("");
        ageInput.setText("");
        selectedStudent = null;
      }
    }
```

Step7: Use Adapter For show list of student(need→Adapter java class, layout design)

7.A Create student_item.xml

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
  android:orientation="vertical"
  android:padding="12dp"
  android:layout_width="match_parent"
  android:layout_height="wrap_content">

  <TextView
    android:id="@+id/nameText"
    android:textStyle="bold"
    android:textSize="18sp"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />

  <TextView
    android:id="@+id/ageText"
    android:textSize="14sp"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" />
</LinearLayout>
```

7.B Create StudentAdapter

```
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 StudentAdapter extends RecyclerView.Adapter<StudentAdapter.StudentViewHolder> {

    private List<Student> studentList;
    private OnItemClickListener listener;

    public interface OnItemClickListener {
        void onItemClick(Student student);
    }

    public StudentAdapter(List<Student> studentList, OnItemClickListener listener) {
        this.studentList = studentList;
        this.listener = listener;
    }

    public void setStudents(List<Student> students) {
        this.studentList = students;
        notifyDataSetChanged();
    }

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

    @Override
    public void onBindViewHolder(@NonNull StudentViewHolder holder, int position) {
        Student student = studentList.get(position);
        holder.nameText.setText(student.name);
        holder.ageText.setText("Age: " + student.age);

        holder.itemView.setOnClickListener(v -> listener.onItemClick(student));
    }

    @Override
    public int getItemCount() {
        return studentList == null ? 0 : studentList.size();
    }

    static class StudentViewHolder extends RecyclerView.ViewHolder {
        TextView nameText, ageText;

        public StudentViewHolder(@NonNull View itemView) {
            super(itemView);
            nameText = itemView.findViewById(R.id.nameText);
            ageText = itemView.findViewById(R.id.ageText);
        }
    }
}
```