Program No:	**18**
Roll No :	**1545**
Title of Program :	**Sqlite Database**
Objective :	**Create and Add Data to SQLite Database in Android.**

SQLite is a lightweight, self-contained, serverless, and highly portable relational database management system. Its primary uses revolve around scenarios where a full-fledged client-server database is overkill or impractical.
Key uses of SQLite include:

- Mobile and Desktop Applications:
  It serves as a local data storage solution for mobile apps (Android, iOS) and desktop applications, managing user preferences, cached content, and application-specific data.
- Embedded Systems:
  Its small footprint and efficiency make it suitable for embedded devices, IoT applications, and various hardware where resources are limited.
- Web Browsers:
  Major web browsers utilize SQLite for storing browsing history, cookies, and other local data.
- Prototyping and Testing:
  Developers frequently employ SQLite during the development and testing phases of projects due to its ease of setup and quick iteration capabilities.
- File Formats:
  SQLite can be used as an application file format, as seen in the Navigation Data Standard, providing a structured way to store and access data within a single file.
- Lightweight Server-side Applications:
  While not designed for high-concurrency, multi-user environments, SQLite can be used in small-scale web applications or scripts where a simple, embedded database is sufficient.

Source Code:

activity_main.xml

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

   <!--Edit text to enter course name-->
   <EditText
       android:id="@+id/idEdtCourseName"
       android:layout_width="match_parent"
       android:layout_height="wrap_content"
       android:layout_margin="10dp"
       android:hint="Enter course Name" />

   <!--edit text to enter course duration-->
   <EditText
       android:id="@+id/idEdtCourseDuration"
       android:layout_width="match_parent"
       android:layout_height="wrap_content"
       android:layout_margin="10dp"
       android:hint="Enter Course Duration" />

   <!--edit text to display course tracks-->
   <EditText
       android:id="@+id/idEdtCourseTracks"
       android:layout_width="match_parent"
       android:layout_height="wrap_content"
       android:layout_margin="10dp"
       android:hint="Enter Course Tracks" />

   <!--edit text for course description-->
   <EditText
       android:id="@+id/idEdtCourseDescription"
       android:layout_width="match_parent"
       android:layout_height="wrap_content"
       android:layout_margin="10dp"
       android:hint="Enter Course Description" />

   <!--button for adding new course-->
   <Button
       android:id="@+id/idBtnAddCourse"
       android:layout_width="match_parent"
       android:layout_height="wrap_content"
```

```
        android:layout_margin="10dp"
        android:text="Add Course"
        android:textAllCaps="false" />

</LinearLayout>
```

DBHandler.java

```
package com.example.sqlitedatabase;

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

public class DBHandler extends SQLiteOpenHelper {

    // creating a constant variables for our database.
    // below variable is for our database name.
    private static final String DB_NAME = "coursedb";

    // below int is our database version
    private static final int DB_VERSION = 1;

    // below variable is for our table name.
    private static final String TABLE_NAME = "mycourses";

    // below variable is for our id column.
    private static final String ID_COL = "id";

    // below variable is for our course name column
    private static final String NAME_COL = "name";

    // below variable id for our course duration column.
    private static final String DURATION_COL = "duration";

    // below variable for our course description column.
    private static final String DESCRIPTION_COL = "description";

    // below variable is for our course tracks column.
    private static final String TRACKS_COL = "tracks";

    // creating a constructor for our database handler.
    public DBHandler(Context context) {
        super(context, DB_NAME, null, DB_VERSION);
    }

    // below method is for creating a database by running a sqlite query
    @Override
```

```
    public void onCreate(SQLiteDatabase db) {
        // on below line we are creating
        // an sqlite query and we are
        // setting our column names
        // along with their data types.
        String query = "CREATE TABLE " + TABLE_NAME + " ("
                + ID_COL + " INTEGER PRIMARY KEY AUTOINCREMENT, "
                + NAME_COL + " TEXT,"
                + DURATION_COL + " TEXT,"
                + DESCRIPTION_COL + " TEXT,"
                + TRACKS_COL + " TEXT)";

        // at last we are calling a exec sql
        // method to execute above sql query
        db.execSQL(query);
    }

    // this method is use to add new course to our sqlite database.
    public void addNewCourse(String courseName, String courseDuration, String
courseDescription, String courseTracks) {

        // on below line we are creating a variable for
        // our sqlite database and calling writable method
        // as we are writing data in our database.
        SQLiteDatabase db = this.getWritableDatabase();

        // on below line we are creating a
        // variable for content values.
        ContentValues values = new ContentValues();

        // on below line we are passing all values
        // along with its key and value pair.
        values.put(NAME_COL, courseName);
        values.put(DURATION_COL, courseDuration);
        values.put(DESCRIPTION_COL, courseDescription);
        values.put(TRACKS_COL, courseTracks);

        // after adding all values we are passing
        // content values to our table.
        db.insert(TABLE_NAME, null, values);

        // at last we are closing our
        // database after adding database.
        db.close();
    }

    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
        // this method is called to check if the table exists already.
        db.execSQL("DROP TABLE IF EXISTS " + TABLE_NAME);
        onCreate(db);
```

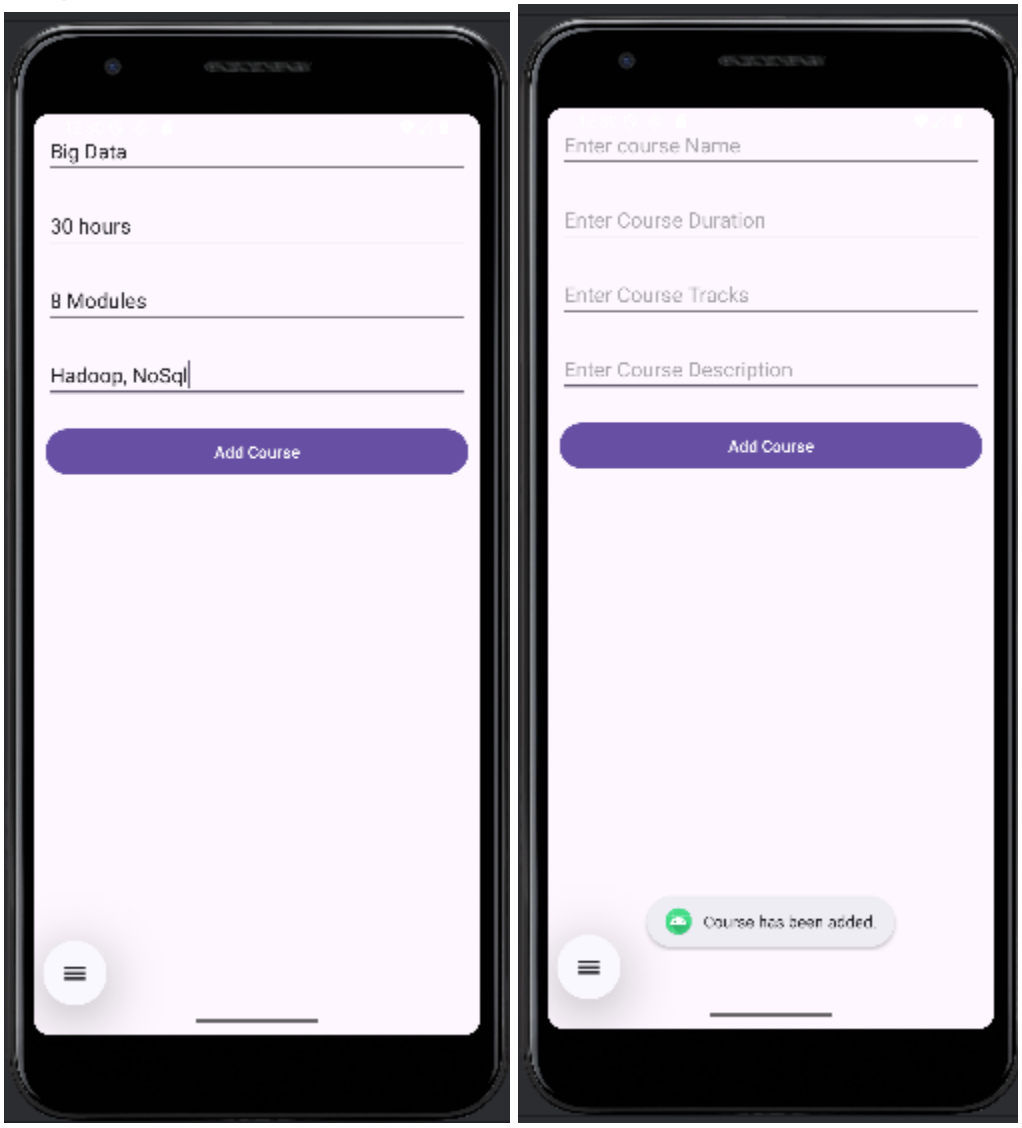
```
    }
}
```

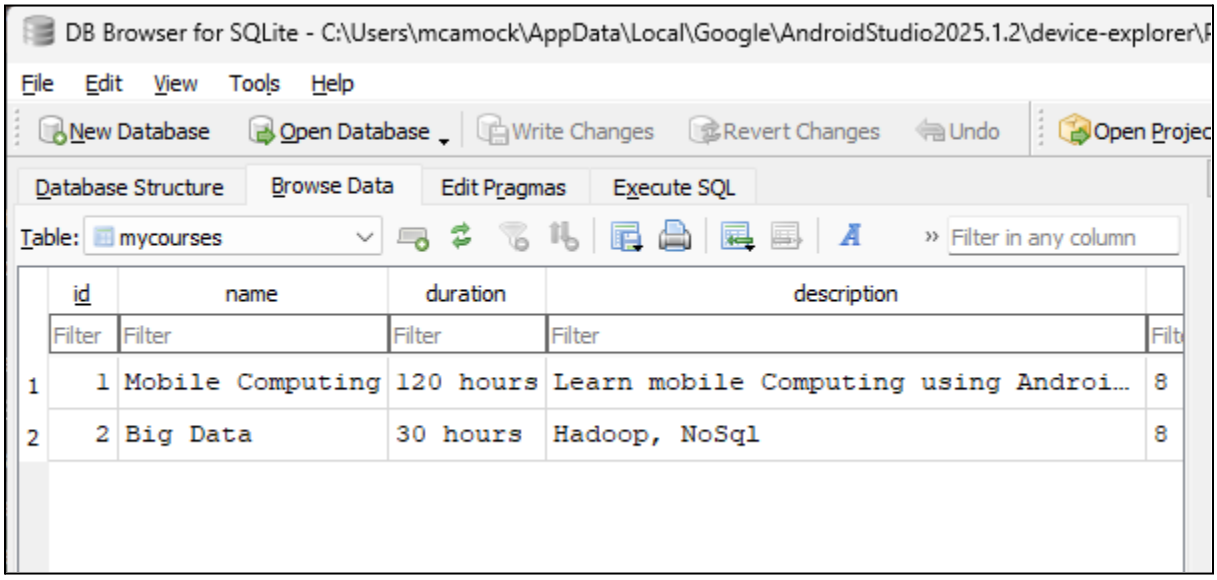
MainActivity.java

```
package com.example.sqlitedatabase;

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 {

    // creating variables for our edittext, button and dbhandler
    private EditText courseNameEdt, courseTracksEdt, courseDurationEdt,
courseDescriptionEdt;
    private Button addCourseBtn;
    private DBHandler dbHandler;

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

        // initializing all our variables.
        courseNameEdt = findViewById(R.id.idEdtCourseName);
        courseTracksEdt = findViewById(R.id.idEdtCourseTracks);
        courseDurationEdt = findViewById(R.id.idEdtCourseDuration);
        courseDescriptionEdt = findViewById(R.id.idEdtCourseDescription);
        addCourseBtn = findViewById(R.id.idBtnAddCourse);

        // creating a new dbhandler class
        // and passing our context to it.
        dbHandler = new DBHandler(MainActivity.this);

        // below line is to add on click listener for our add course button.
        addCourseBtn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {

                // below line is to get data from all edit text fields.
                String courseName = courseNameEdt.getText().toString();
                String courseTracks = courseTracksEdt.getText().toString();
                String courseDuration = courseDurationEdt.getText().toString();
                String courseDescription =
```

```
courseDescriptionEdt.getText().toString();

                // validating if the text fields are empty or not.
                if (courseName.isEmpty() && courseTracks.isEmpty() &&
courseDuration.isEmpty() && courseDescription.isEmpty()) {
                    Toast.makeText(MainActivity.this, "Please enter all the
data..", Toast.LENGTH_SHORT).show();
                    return;
                }

                // on below line we are calling a method to add new
                // course to sqlite data and pass all our values to it.
                dbHandler.addNewCourse(courseName, courseDuration,
courseDescription, courseTracks);

                // after adding the data we are displaying a toast message.
                Toast.makeText(MainActivity.this, "Course has been added.",
Toast.LENGTH_SHORT).show();
                courseNameEdt.setText("");
                courseDurationEdt.setText("");
                courseTracksEdt.setText("");
                courseDescriptionEdt.setText("");
            }
        });
    }
}
```

Output:


Big Data
30 hours
8 Modules
Hadoop, NoSql
Add Course
Enter course Name
Enter Course Duration
Enter Course Tracks
Enter Course Description
Add Course
Course has been added.


DB Browser for SQLite - C:\Users\mcamock\AppData\Local\Google\AndroidStudio2025.1.2\device-explorer\
File Edit View Tools Help
New Database
Open Database
Write Changes
Revert Changes
Undo
Open Projec
Database Structure
Browse Data
Edit Pragmas
Execute SQL
Table: mycourses
Filter in any column
id
name
duration
description
Filter
1 1 Mobile Computing 120 hours Learn mobile Computing using Androi… 8
2 2 Big Data 30 hours Hadoop, NoSql 8