

InternshipProgram:Soulvibe.Tech

Higher Education Course Analysis SQL”

Data Analyst Internship – Batch 10



Introduction



Overview of the main objectives

The goal of this task is to evaluate the ability to query, manipulate, and analyze relational data using SQL. We will use the same "Higher Education Course" dataset to write queries and extract meaningful insights from it.

1

Find top5 districts with highest no.of colleges offering prof. courses

The screenshot shows the DB Browser for SQLite interface. The SQL query in the editor window is:

```
1 SELECT District, COUNT(DISTINCT "College Name") AS CollegeCount
2 FROM CollegeCourses
3 WHERE "Is Professional"= 'Professional Course'
4 GROUP BY District
5 ORDER BY CollegeCount DESC
6 LIMIT 5;
```

The results are displayed in a table:

	District	CollegeCount
1	Pune	623
2	Nagpur	309
3	Nashik	206
4	Ahmednagar	204
5	Aurangabad	195

The status bar at the bottom shows system information including weather, battery level, and system time.

2 Calculate the average course duration (in months) for each Course Type and sort them in descending order.

The screenshot shows the DB Browser for SQLite interface. The SQL tab contains the following query:

```
1 SELECT [Course Type], AVG([Course Duration (In months)]) AS AvgDuration
2 FROM CollegeCourses
3 GROUP BY [Course Type]
4 ORDER BY AvgDuration DESC;
```

The results pane displays a table with two columns: "Course Type" and "AvgDuration". The data is as follows:

	Course Type	AvgDuration
1	DUAL Degree	54.9836065573771
2	Ph.D	52.4307692307692
3	Under Graduate Course	38.6023807621635
4	Diploma Course	32.1285225177772
5	Post Graduate Course	24.8091580502216
6	Post Graduate Diploma Course	21.1926605504587
7	Vocational Course	19.763075400099

The status bar at the bottom shows system information including weather, battery level, and system time.

3

Count how many unique College Names offer each Course Category.

DB Browser for SQLite - C:\Users\DELL\CollegeCourses.db

File Edit View Tools Help

New Database Open Database Write Changes Revert Changes Undo Open Project Save Project Attach Database Close Database

Database Structure Browse Data Edit Pragmas Execute SQL

SQL 1*

```
1 SELECT [Course Category], COUNT(DISTINCT [College Name]) AS CollegeCount
2 FROM CollegeCourses
3 GROUP BY [Course Category];
```

	Course Category	CollegeCount
1	ARCHITECTURE AND TOWN PLANNING	99
2	Agriculture	138
3	Arts	7696
4	Commerce	4076
5	Education	1239
6	Engineering	834
7	Health Science	360

No cell active.
Type: NULL; Size: 0 bytes

Remote

Identity Select an identity to connect

DBHub.io Local Current Database

Name Last modified Size

Execution finished without errors.
Result: 18 rows returned in 426ms
At line 1:
SELECT [Course Category], COUNT(DISTINCT [College Name]) AS CollegeCount
FROM CollegeCourses
GROUP BY [Course Category];

SQL Log Plot DB Schema Remote

UTF-8

80°F Mostly cloudy

Search

10:16 PM 6/23/2025

4

Find the names of colleges offering both Post Graduate and Under Graduate courses.

DB Browser for SQLite - C:\Users\DELL\CollegeCourses.db

File Edit View Tools Help

New Database Open Database Write Changes Revert Changes Undo Open Project Save Project Attach Database Close Database

Database Structure Browse Data Edit Pragmas Execute SQL

SQL 1*

```
1 SELECT [College Name]
2 FROM CollegeCourses
3 WHERE [Course Type] IN ('Under Graduate Course', 'Post Graduate Course')
4 GROUP BY [College Name]
5 HAVING COUNT(DISTINCT [Course Type]) = 2;
```

	College Name
1	A.T.S.P. MANDAL'S Arts, Commerce & ...
2	A.V.COLLEGE ARTS, K.M. COLLEGE OF ...
3	ADARSH COLLEGE OF ARTS & COMMERCE
4	ALI YAVAR JUNG NATIONAL INSTITUTE
5	ALL INDIA INSTITUTE OF PHYSICAL ...
6	ANANTRAO PAWAR COLLEGE OF ENGINEERIN...
7	ANIKET SOCIAL WORK COLLEGE, DESAIGAN...

No cell active.
Type: NULL; Size: 0 bytes

Remote

Identity Select an identity to connect

DBHub.io Local Current Database

Name Last modified Size

Execution finished without errors.
Result: 1633 rows returned in 177ms
At line 1:
SELECT [College Name]
FROM CollegeCourses
WHERE [Course Type] IN ('Under Graduate Course', 'Post Graduate Course')
GROUP BY [College Name]
HAVING COUNT(DISTINCT [Course Type]) = 2;

SQL Log Plot DB Schema Remote

UTF-8

3 80°F Mostly cloudy

Search

10:16 PM 6/23/2025

ENG IN

5

List all universities that have more than 10 unaided courses that are not professional.

The screenshot shows the DB Browser for SQLite interface. The SQL editor contains the following query:

```
1 SELECT "University"
2 FROM CollegeCourses
3 WHERE TRIM("Course_Aided__Unaided") LIKE "%Unaided%"
4 AND TRIM("Is_Professional") LIKE "%Non%"
5 GROUP BY "University"
6 HAVING COUNT(*) > 10;
```

The results pane displays a table with one column, "University", listing the following universities:

University
AUTONOMOUS INSTITUTE
DR. B. A. MARATHWADA UNIVERSITY
GONDWANA UNIVERSITY
KAVI KULGURU SANSKRIT ...
MAHARASHTRA STATE BOARD OF SECONDARY...
MUMBAI UNIVERSITY
NCVT

Below the results, the SQL log shows the executed query and its execution details:

```
Execution finished without errors.
Result: 18 rows returned in 49ms
At line 1:
SELECT "University"
FROM CollegeCourses
WHERE TRIM("Course_Aided__Unaided") LIKE "%Unaided%"
AND TRIM("Is_Professional") LIKE "%Non%"
GROUP BY "University"
HAVING COUNT(*) > 10;
```

The status bar at the bottom indicates the system is at 80°F, mostly cloudy, and shows the date and time as 6/23/2025, 10:27 PM.

6

Display colleges from the "Engineering" category that have at least one course with a duration greater than the category's average.

The screenshot shows the DB Browser for SQLite interface. The SQL tab contains the following query:

```
1 WITH AvgDur AS (
2     SELECT AVG("Course_Duration_In_months") AS AvgEnggDur
3     FROM CollegeCourses
4     WHERE "Course_Category" = 'Engineering'
5 )
6 SELECT DISTINCT "College_Name"
7 FROM CollegeCourses, AvgDur
8 WHERE "Course_Category" = 'Engineering'
9 AND "Course_Duration_In_months" > AvgEnggDur;
10
```

The results pane displays a list of college names:

College_Name
Government College of Engineering, ...
Sant Gadge Baba Amravati ...
Government College of Engineering ...
Shri Sant Gajanan Maharaj College of...
Prof. Ram Meghe Institute of ...
P. R. Pote (Patil) Education & ...
P.R. Patil College of Engineering & ...

Execution details at the bottom:

```
Execution finished without errors.
Result: 397 rows returned in 72ms
At line 1:
WITH AvgDur AS (
    SELECT AVG("Course_Duration_In_months") AS AvgEnggDur
    FROM CollegeCourses
    WHERE "Course_Category" = 'Engineering'
)
SELECT DISTINCT "College_Name"
FROM CollegeCourses, AvgDur
WHERE "Course_Category" = 'Engineering'
AND "Course_Duration_In_months" > AvgEnggDur;
```

The status bar at the bottom shows: 80°F Mostly cloudy, Search, ENG IN, 10:29 PM, 6/23/2025.

7

Assign a rank to each course within a College Name based on course duration, longest first.

DB Browser for SQLite - C:\Users\DELL\CollegeCourses.db

File Edit View Tools Help

New Database Open Database Write Changes Revert Changes Undo Open Project Save Project Attach Database Close Database

Database Structure Browse Data Edit Pragmas Execute SQL

SQL 1*

```
1 SELECT
2     "College_Name",
3     "Course_Name",
4     "Course_Duration_In_months",
5     RANK() OVER (
6         PARTITION BY "College_Name"
7         ORDER BY "Course_Duration_In_months" DESC
8     ) AS DurationRank
9 FROM CollegeCourses;
10
11
12
13
```

	College_Name	Course_Name	Course_Duration_In_months	DurationRank
1	(BHARAT RATNA) KHAN ABDUL GAFFAR ...	11th Arts	12	1
2	(BHARAT RATNA) KHAN ABDUL GAFFAR ...	12th Arts	12	1
3	(S.E.S.) S.D.OCHANI HIGH SCHOOL & JR...	12th Composite	12	1
4	(S.E.S.) S.D.OCHANI HIGH SCHOOL & JR...	11th Composite	12	1
5	A A DESAI HIGH SCHOOL BHAISHT	11th Composite	12	1
6	A A DESAI HIGH SCHOOL BHAISHT	12th Composite	12	1
7	A A PADHYE MADHYMIK ENGLISH MEDI ...	12th Commerce	12	1

No cell active.
Type: NULL; Size: 0 bytes

Remote Identity Select an identity to connect Upload

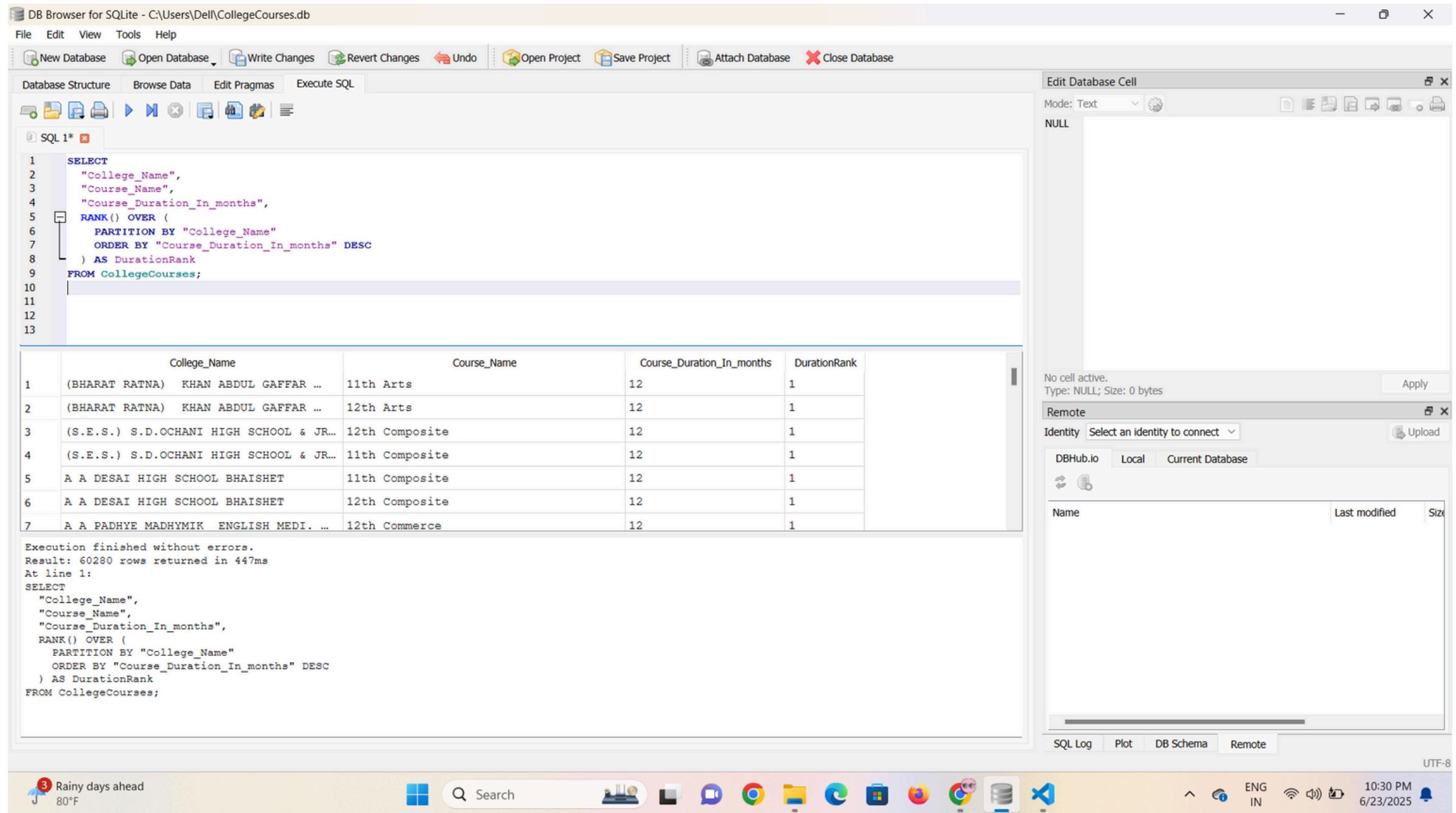
DBHub.io Local Current Database

Name Last modified Size

Execution finished without errors.
Result: 60280 rows returned in 447ms
At line 1:
SELECT
 "College_Name",
 "Course_Name",
 "Course_Duration_In_months",
 RANK() OVER (
 PARTITION BY "College_Name"
 ORDER BY "Course_Duration_In_months" DESC
) AS DurationRank
FROM CollegeCourses;

SQL Log Plot DB Schema Remote UTF-8

Rainy days ahead 80°F ENG IN 10:30 PM 6/23/2025



8 Find colleges where the longest and shortest course durations are more than 24 months apart.

The screenshot shows the DB Browser for SQLite interface. The SQL editor contains the following query:

```
1 SELECT "College_Name"
2 FROM CollegeCourses
3 GROUP BY "College_Name"
4 HAVING MAX("Course_Duration_In_months") - MIN("Course_Duration_In_months") > 24;
```

The results pane displays a table with one column, "College_Name", listing various college names. The output is as follows:

College_Name
1 ADARSH COLLEGE OF ARTS & COMMERCE
2 Abhinav Law College
3 Adarsh Shikshan Prasarak Mandal's ...
4 Adarsha Comprehensive College Of ...
5 Akhil Bhartiya Maratha Shikshan ...
6 Akola Law College, Akola
7 Anjuman-I-Islam's Kalsekar Technical...

Execution finished without errors.
Result: 166 rows returned in 93ms
At line 1:
SELECT "College_Name"
FROM CollegeCourses
GROUP BY "College_Name"
HAVING MAX("Course_Duration_In_months") - MIN("Course_Duration_In_months") > 24;

9 Show the cumulative number of professional courses offered by each university sorted alphabetically.

The screenshot shows the DB Browser for SQLite interface. The SQL editor contains the following query:

```
1 SELECT "University",
2     COUNT(*) AS ProfessionalCourseCount
3 FROM CollegeCourses
4 WHERE "Is_Professional" = 'Professional Course'
5 GROUP BY "University"
6 ORDER BY "University";
```

The results pane displays a table with two columns: University and ProfessionalCourseCount. The data is as follows:

	University	ProfessionalCourseCount
1	AUTONOMOUS INSTITUTE	264
2	AUTONOMUS INSTITUTE OF GOVERNMENT OF...	59
3	All India Institute of Local Self-...	1
4	BHARATI VIDYAPEETH UNIVERSITY	20
5	CENTRAL GOVERNMENT	3
6	CIPET, Head Office, Chennai	8
7	DATTA MEGHE INSTITUTE OF MEDICAL ...	4

Execution finished without errors.
Result: 43 rows returned in 32ms
At line 1:
SELECT "University",
 COUNT(*) AS ProfessionalCourseCount
FROM CollegeCourses
WHERE "Is_Professional" = 'Professional Course'
GROUP BY "University"
ORDER BY "University";

10

Using a self-join or CTE, find colleges offering more than one course category.

The screenshot shows the DB Browser for SQLite interface. The SQL tab contains the following query:

```
1 SELECT DISTINCT a."College_Name"
2 FROM CollegeCourses a
3 JOIN CollegeCourses b
4 ON a."College_Name" = b."College_Name"
5 WHERE a."Course_Category" <> b."Course_Category";
```

The Results tab displays a table with the column "College_Name" containing the following data:

College_Name
Government College of Engineering, ...
Sant Gadge Baba Amravati ...
Government Polytechnic, Amravati
Shri Sant Gajanan Maharaj College of...
Vidya Bharati Mahavidyalaya ...
Prof. Ram Meghe Institute of ...
P. R. Pote (Patil) Education & ...

The status bar at the bottom shows system information including weather, search, taskbar icons, and system status.

Create a temporary table (CTE) that includes average duration of courses by district and use it to list talukas where the average course duration is above the district average.

11

The screenshot shows the DB Browser for SQLite interface. The SQL tab contains the following query:

```
3   FROM CollegeCourses
4   GROUP BY "District"
5   ),
6   TalukaAvg AS (
7       SELECT "District", "Taluka", AVG("Course_Duration_In_months") AS TalukaAvgDuration
8       FROM CollegeCourses
9       GROUP BY "District", "Taluka"
10  )
11  SELECT t."District", t."Taluka", t.TalukaAvgDuration
12  FROM TalukaAvg t
13  JOIN DistrictAvg d ON t."District" = d."District"
14  WHERE t.TalukaAvgDuration > d.DistrictAvgDuration;
15
```

The results pane displays a table with three columns: District, Taluka, and TalukaAvgDuration. The data is as follows:

	District	Taluka	TalukaAvgDuration
1	Ahmednagar	Aurangabad	48.0
2	Ahmednagar	Kopargaon	23.3717277486911
3	Ahmednagar	Nagar	22.9211009174312
4	Ahmednagar	Rahta	22.007722007722
5	Ahmednagar	Sangamner	21.6398891966759
6	Ahmednagar	Shrigonda	22.1965317919075
7	Akola	Akola	20.52

The status bar at the bottom indicates: 1 inch of rain Tuesday, Search, ENG IN, 10:33 PM, 6/23/2025, and UTF-8.

Create a new column classifying course duration as:

- Short (< 12 months)**
- Medium (12-36 months)**
- Long (> 36 months)**

Then count the number of each duration type per course category.

The screenshot shows the DB Browser for SQLite interface. The top menu bar includes File, Edit, View, Tools, Help, New Database, Open Database, Write Changes, Revert Changes, Undo, Open Project, Save Project, Attach Database, and Close Database. Below the menu is a toolbar with icons for Database Structure, Browse Data, Edit Pragmas, Execute SQL, and others. A large central window contains an SQL editor tab labeled "SQL 1*" with the following query:

```
1  SELECT "Course_Category",
2      CASE
3          WHEN "Course_Duration_In_months" < 12 THEN 'Short'
4          WHEN "Course_Duration_In_months" BETWEEN 12 AND 36 THEN 'Medium'
5          ELSE 'Long'
6      END AS DurationType,
7      COUNT(*) AS CourseCount
8  FROM CollegeCourses
9  GROUP BY "Course_Category", DurationType;
```

Below the SQL editor is a results grid showing the output of the query:

	Course_Category	DurationType	CourseCount
1	ARCHITECTURE AND TOWN PLANNING	Long	103
2	ARCHITECTURE AND TOWN PLANNING	Medium	40
3	Agriculture	Long	103
4	Agriculture	Medium	78
5	Arts	Long	100
6	Arts	Medium	16815
7	Commerce	Long	2

At the bottom of the results grid, the message "Execution finished without errors." is displayed, followed by the execution details: "Result: 32 rows returned in 72ms" and the original query text again.

The bottom status bar shows system information: "Hot days ahead 80°F", a search bar, taskbar icons for various applications like File Explorer, Edge, and File Manager, and system status indicators including battery level, signal strength, and date/time (10:34 PM, 6/23/2025).

13

Extract only the course specialization from Course Name. (e.g., from "Bachelor of Engineering (B. E.) - Electrical", extract "Electrical")

The screenshot shows the DB Browser for SQLite interface. The SQL tab contains the following query:

```
1 SELECT "Course_Name",
2       TRIM(SUBSTR("Course_Name", INSTR("Course_Name", '-') + 1)) AS Specialization
3 FROM CollegeCourses
4 WHERE "Course_Name" LIKE '%-%';
5
```

The results pane displays a table with two columns: Course_Name and Specialization. The data is as follows:

	Course_Name	Specialization
1	Master of Engineering (M. E.) - ...	Computer Science and Engineering
2	Master of Engineering (M. E.) - ...	Electrical Power System
3	Master of Engineering (M. E.) - ...	Thermal Power Engineering
4	Bachelor of Engineering (B. E.) - ...	Information Technology
5	Bachelor of Engineering (B. E.) - ...	Electrical Engineering
6	Bachelor of Engineering (B. E.) - ...	Instrumentation Engineering
7	Master of Engineering (M. E.) - Geo ...	Geo Technical Engineering

The status bar at the bottom shows system information including weather, search, and network status.

14

Count how many courses include the word Engineering in the name.

The screenshot shows the DB Browser for SQLite interface. The main window displays the following SQL query:

```
1 SELECT COUNT(*) AS EngineeringCourseCount
2 FROM CollegeCourses
3 WHERE "Course_Name" LIKE '%Engineering%';
4
```

The results pane shows a single row with the heading "EngineeringCourseCount" and the value "5771". Below the results, the SQL log shows the executed query and its execution details:

```
Execution finished without errors.
Result: 1 rows returned in 33ms
At line 1:
SELECT COUNT(*) AS EngineeringCourseCount
FROM CollegeCourses
WHERE "Course_Name" LIKE '%Engineering%';
```

The status bar at the bottom indicates "Hot days ahead" and "80°F".

15

List all unique combinations of Course Name, Course Type, and Course Category.

The screenshot shows the DB Browser for SQLite interface. The SQL tab contains the following query:

```
1 SELECT DISTINCT "Course_Name", "Course_Type", "Course_Category"
2 FROM CollegeCourses;
```

The results pane displays a table with three columns: Course_Name, Course_Type, and Course_Category. The data shows various engineering courses, all categorized under Engineering:

	Course_Name	Course_Type	Course_Category
1	Master of Engineering (M. E.) - ...	Post Graduate Course	Engineering
2	Master of Engineering (M. E.) - ...	Post Graduate Course	Engineering
3	Master of Engineering (M. E.) - ...	Post Graduate Course	Engineering
4	Bachelor of Engineering (B. E.) - ...	Under Graduate Course	Engineering
5	Bachelor of Engineering (B. E.) - ...	Under Graduate Course	Engineering
6	Bachelor of Engineering (B. E.) - ...	Under Graduate Course	Engineering
7	Master of Engineering (M. E.) - Geo ...	Post Graduate Course	Engineering

Execution details at the bottom of the results pane:

```
Execution finished without errors.
Result: 1259 rows returned in 62ms
At line 1:
SELECT DISTINCT "Course_Name", "Course_Type", "Course_Category"
FROM CollegeCourses;
```

The status bar at the bottom shows system information and a notification for "Hot days ahead 80°F".

16

Write a query to get all courses that are not offered by any Government college.

The screenshot shows the DB Browser for SQLite interface. In the top-left corner, the title bar reads "DB Browser for SQLite - C:\Users\...CollegeCourses.db". The menu bar includes File, Edit, View, Tools, Help, and several database-related options like New Database, Open Database, Write Changes, Undo, Open Project, Save Project, Attach Database, and Close Database. Below the menu is a toolbar with icons for Database Structure, Browse Data, Edit Pragmas, and Execute SQL. A sub-toolbar under Execute SQL contains icons for SQL, CSV, XML, JSON, and other formats. A central SQL editor window titled "SQL 1*" contains the following SQL code:

```
1 SELECT DISTINCT "Course_Name"
2 FROM CollegeCourses
3 WHERE "College_Type" <> 'Government';
```

Below the SQL editor is a results grid titled "Course_Name" with the following data:

Course_Name
1 Master of Engineering (M. E.) - ...
2 Bachelor of Technology (B. Tech.) - ...
3 M. B. A.
4 Master in Computer Application
5 Post S.S.C. Diploma in Engineering ...
6 Post S.S.C. Diploma in Engineering ...
7 Post S.S.C. Diploma in Engineering ...

At the bottom of the results grid, the message "Execution finished without errors. Result: 1137 rows returned in 54ms At line 1:" is displayed, followed by the executed SQL query again.

To the right of the results grid is a "Edit Database Cell" panel titled "Edit Database Cell" with "Mode: Text" set to "NULL". Below it is a "Remote" connection configuration panel with "Identity" set to "Select an identity to connect". The bottom of the interface shows tabs for SQL Log, Plot, DB Schema, and Remote, along with a status bar indicating "UTF-8" encoding, system icons, and the date/time "6/23/2025 10:36 PM".

17 Find the university that has the second-highest number of aided courses.

The screenshot shows the DB Browser for SQLite interface with the following details:

- Title Bar:** DB Browser for SQLite - C:\Users\...CollegeCourses.db
- Menu Bar:** File, Edit, View, Tools, Help
- Toolbar:** New Database, Open Database, Write Changes, Revert Changes, Undo, Open Project, Save Project, Attach Database, Close Database
- Tab Bar:** Database Structure, Browse Data, Edit Pragmas, Execute SQL
- SQL Editor:** SQL 1* contains the following SQL query:

```
3     FROM CollegeCourses
4     WHERE "Course_Aided__Unaided" = 'Aided'
5     GROUP BY "University"
6   ),
7   Ranked AS (
8     SELECT "University", AidedCourses,
9        RANK() OVER (ORDER BY AidedCourses DESC) AS RankNum
10    FROM AidedCount
11  )
12  SELECT "University"
13  FROM Ranked
14  WHERE RankNum = 2
15  LIMIT 1;
```
- Results Table:** University column shows NCVT.
- Message Bar:** Execution finished without errors. Result: 1 rows returned in 49ms
- Log Area:** Shows the executed SQL statements and their results.
- Database List:** Shows a single database entry: NCVT.
- System Tray:** Shows a notification for 3 sports headlines, the user's name (Isaiah Hartenstein), and the system status (ENG IN, 10:36 PM, 6/23/2025).

18

Show courses whose durations are above the median course duration.

The screenshot shows the DB Browser for SQLite interface. The SQL tab contains the following query:

```
4   ORDER BY "Course_Duration_In_months"
5   ),
6   Counted AS (
7     SELECT COUNT(*) AS Total FROM SortedCourses
8   ),
9   MedianCourse AS (
10    SELECT "Course_Duration_In_months"
11    FROM SortedCourses
12    LIMIT 1 OFFSET (SELECT Total/2 FROM Counted)
13  )
14  SELECT *
15  FROM CollegeCourses
16  WHERE "Course_Duration_In_months" > (SELECT "Course_Duration_In_months" FROM MedianCourse);
```

The results pane displays a table with 6 rows, showing data for Amravati district and Taluka. The columns are Sr.No, District, Taluka, College_Name, University, and College_Type. All entries show Government College of Engineering, SANT GADGE BABA AMRAVATI UNIVERSITY, and Government as the type.

Sr.No	District	Taluka	College_Name	University	College_Type
1	Amravati	Amravati	Government College of Engineering, ...	SANT GADGE BABA AMRAVATI UNIVERSITY	Government
2	Amravati	Amravati	Government College of Engineering, ...	SANT GADGE BABA AMRAVATI UNIVERSITY	Government
3	Amravati	Amravati	Government College of Engineering, ...	SANT GADGE BABA AMRAVATI UNIVERSITY	Government
4	Amravati	Amravati	Government College of Engineering, ...	SANT GADGE BABA AMRAVATI UNIVERSITY	Government
5	Amravati	Amravati	Government College of Engineering, ...	SANT GADGE BABA AMRAVATI UNIVERSITY	Government
6	Amravati	Amravati	Government College of Engineering, ...	SANT GADGE BABA AMRAVATI UNIVERSITY	Government

Execution finished without errors.
Result: 27298 rows returned in 247ms
At line 1:
WITH SortedCourses AS (
 SELECT "Course_Name", "Course_Duration_In_months"
 FROM CollegeCourses
 ORDER BY "Course_Duration_In_months"
)
,
Counted AS (
 SELECT COUNT(*) AS Total FROM SortedCourses
)
,
MedianCourse AS (
 SELECT "Course_Duration_In_months"
 FROM SortedCourses
 LIMIT 1 OFFSET (SELECT Total/2 FROM Counted)

19

For each University, find the percentage of unaided courses that are professional.

The screenshot shows the DB Browser for SQLite interface. The SQL tab contains the following query:

```
1 SELECT "University",
2     ROUND(100.0 * SUM(CASE WHEN TRIM("Is_Professional") LIKE '%Professional%' THEN 1 ELSE 0 END)
3         / COUNT(*), 2) AS Professional_Percentage
4 FROM CollegeCourses
5 WHERE TRIM("Course_Aided__Unaided") LIKE '%Unaided%'
6 GROUP BY "University";
7
```

The Results tab displays a table with two columns: University and Professional_Percentage. All rows show a value of 100.0. The table has 7 rows, corresponding to the 7 universities listed in the query results.

	University	Professional_Percentage
1	AUTONOMOUS INSTITUTE	100.0
2	AUTONOMUS INSTITUTE OF GOVERNMENT OF...	100.0
3	BHARATI VIDYAPEETH UNIVERSITY	100.0
4	CBSE	100.0
5	CENTRAL GOVERNMENT	100.0
6	CIPET, Head Office, Chennai	100.0
7	DATTA MEGHE INSTITUTE OF MEDICAL ...	100.0

Execution finished without errors.
Result: 44 rows returned in 67ms
At line 1:
SELECT "University",
 ROUND(100.0 * SUM(CASE WHEN TRIM("Is_Professional") LIKE '%Professional%' THEN 1 ELSE 0 END)
 / COUNT(*), 2) AS Professional_Percentage
FROM CollegeCourses
WHERE TRIM("Course_Aided__Unaided") LIKE '%Unaided%'
GROUP BY "University";

20. Determine which Course Category has the highest average course duration and display the top 3.

The screenshot shows the DB Browser for SQLite interface with the following details:

- SQL Editor:** Contains the following SQL query:

```
1 SELECT "Course_Category",
2     AVG("Course_Duration_In_months") AS AvgDuration
3 FROM CollegeCourses
4 GROUP BY "Course_Category"
5 ORDER BY AvgDuration DESC
6 LIMIT 3;
```
- Results Table:** Displays the results of the query in a tabular format:

Course_Category	AvgDuration
1 ARCHITECTURE AND TOWN PLANNING	49.9300699300699
2 Health Science	46.0826709062003
3 Agriculture	40.707182320442
- Message Bar:** Shows the message "Execution finished without errors. Result: 3 rows returned in 52ms".
- System Tray:** Shows the date and time as 10:38 PM on 6/23/2025, along with other system icons.

Conclusion

Conclusion:

This SQL project helped me explore the CollegeCourses dataset and gain practical skills in filtering, grouping, and analyzing educational data. I was able to:

- Find top districts and universities by course count
 - Compare UG, PG, and diploma course durations
- Understand patterns in professional vs. non-professional and aided vs. unaided courses
 - Identify colleges offering multiple course categories
- Use SQL functions like CTEs and window functions for deeper insights

Key Takeaways:

- SQL is a powerful tool for exploring and analyzing real-world data
- This project improved my query writing and data interpretation skills
- The insights gained can support educational planning and reporting
- It also builds a strong base for using tools like Power BI and Tableau



Thank You

