1. SQL JOINS

1. Retrieve all employees and their assigned projects using an **INNER JOIN**.

– Return: EmployeeID, FirstName, LastName, Department, Salary, ProjectID, ProjectName, Budget, Status.

```
-- SQL JOINS

-- 1. Retrieve all employees and their assigned projects using an INNER JOIN.Return: EmployeeID, FirstName, LastName, Department, Salary,ProjectID, ProjectName, Budget, Status.

SELECT e.ID, e.first_name, e.last_name, e.Department, e.Salary, p.ProjectID, p.ProjectName, p.Budget, p.Status
FROM "EMPLOYEES"."EMPLOYEES_SCHEMA"."EMPLOYEES" AS e
INNER JOIN
"EMPLOYEES"."EMPLOYEES_SCHEMA"."PROJECTS" AS P ON e.id = P.EMPLOYEEID;
```

Results | Chart

	ID	FIRST_NAME	LAST_NAME	DEPARTMENT	SALARY	PROJECTID	PROJECTNAME	BUDGET	STATUS
1	1	John	Doe	IT	55000	101	AI Development	100000	Completed
2	1	John	Doe	IT	55000	103	Cybersecurity Audit	75000	Pending
3	2	Jane	Smith	HR	48000	102	Employee Training	50000	Ongoing
4	3	Mike	Johnson	Finance	60000	104	Financial Analysis	90000	Ongoing
5	5	David	White	Marketing	52000	105	Market Expansion	65000	Completed
6	6	Emily	Davis	IT	62000	106	Risk Management	80000	Pending

Query Details
Query duration 138ms
Rows 6
Query ID 01bc9a59-0001-0cd5-0...
Show more

2. Retrieve all employees and their assigned projects, including employees who have no projects using a **LEFT JOIN**.

– Return: EmployeeID, FirstName, LastName, Department, Salary, ProjectID, ProjectName, Budget, Status.

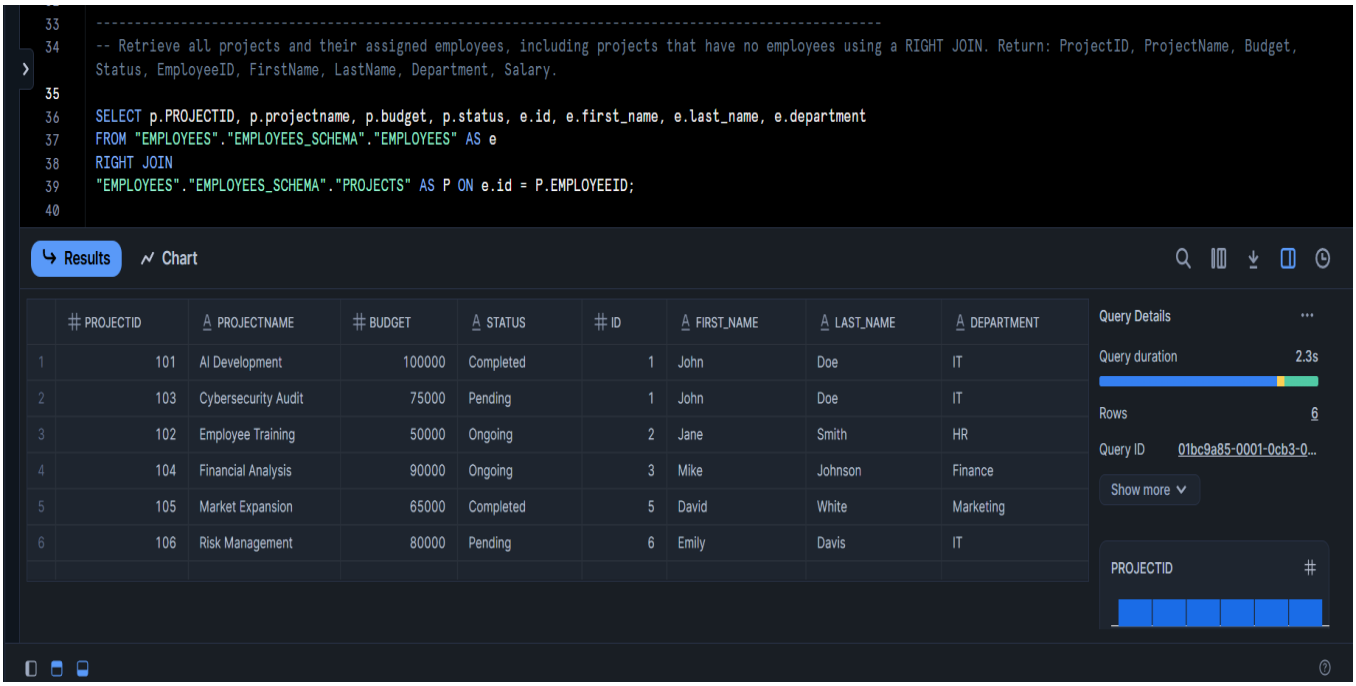
```
--2. Retrieve all employees and their assigned projects, including employees who have no projects using a LEFT JOIN.

SELECT e.ID, e.first_name, e.last_name, e.Department, e.Salary, p.ProjectID, p.ProjectName, p.Budget, p.Status
FROM "EMPLOYEES"."EMPLOYEES_SCHEMA"."EMPLOYEES" AS e
LEFT JOIN
"EMPLOYEES"."EMPLOYEES_SCHEMA"."PROJECTS" AS P ON e.id = P.EMPLOYEEID;

-----------------------------------------------------------------------------------------------
```

Results | Chart

	ID	FIRST_NAME	LAST_NAME	DEPARTMENT	SALARY	PROJECTID	PROJECTNAME	BUDGET	STATUS
1	1	John	Doe	IT	55000	101	AI Development	100000	Completed
2	1	John	Doe	IT	55000	103	Cybersecurity Audit	75000	Pending
3	2	Jane	Smith	HR	48000	102	Employee Training	50000	Ongoing
4	3	Mike	Johnson	Finance	60000	104	Financial Analysis	90000	Ongoing
5	4	Sarah	Brown	IT	53000	null	null	null	null
6	5	David	White	Marketing	52000	105	Market Expansion	65000	Completed
7	6	Emily	Davis	IT	62000	106	Risk Management	80000	Pending
8	7	Robert	Wilson	Finance	59000	null	null	null	null

Query Details
Query duration 42ms
Rows 11
Query ID 01bc9a70-0001-0b38-0...
Show more
ID

3. Retrieve all projects and their assigned employees, including projects that have no employees using a **RIGHT JOIN.**
   - Return: ProjectID, ProjectName, Budget, Status, EmployeeID, FirstName, LastName, Department, Salary.



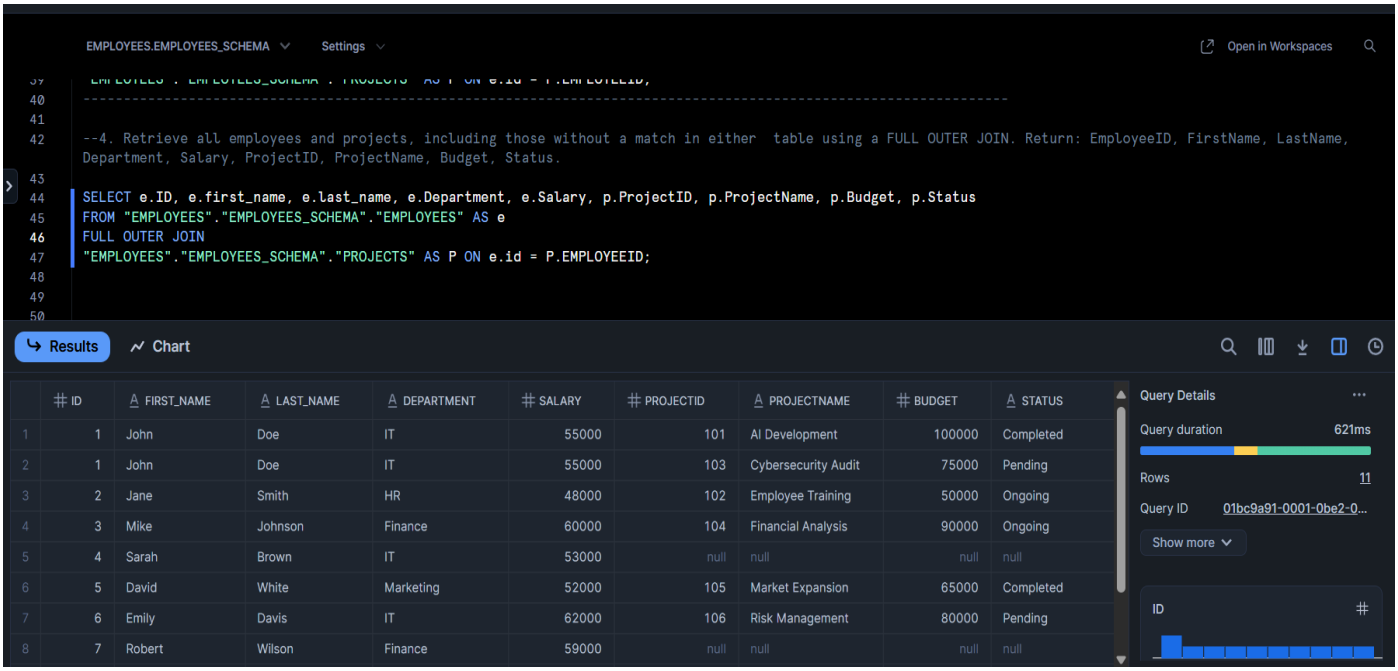
```
33  ----------------------------------------------------------------------------------------------
34  -- Retrieve all projects and their assigned employees, including projects that have no employees using a RIGHT JOIN. Return: ProjectID, ProjectName, Budget,
    Status, EmployeeID, FirstName, LastName, Department, Salary.
35
36  SELECT p.PROJECTID, p.projectname, p.budget, p.status, e.id, e.first_name, e.last_name, e.department
37  FROM "EMPLOYEES"."EMPLOYEES_SCHEMA"."EMPLOYEES" AS e
38  RIGHT JOIN
39  "EMPLOYEES"."EMPLOYEES_SCHEMA"."PROJECTS" AS P ON e.id = P.EMPLOYEEID;
40
```

Results | Chart

	PROJECTID	PROJECTNAME	BUDGET	STATUS	ID	FIRST_NAME	LAST_NAME	DEPARTMENT
1	101	AI Development	100000	Completed	1	John	Doe	IT
2	103	Cybersecurity Audit	75000	Pending	1	John	Doe	IT
3	102	Employee Training	50000	Ongoing	2	Jane	Smith	HR
4	104	Financial Analysis	90000	Ongoing	3	Mike	Johnson	Finance
5	105	Market Expansion	65000	Completed	5	David	White	Marketing
6	106	Risk Management	80000	Pending	6	Emily	Davis	IT

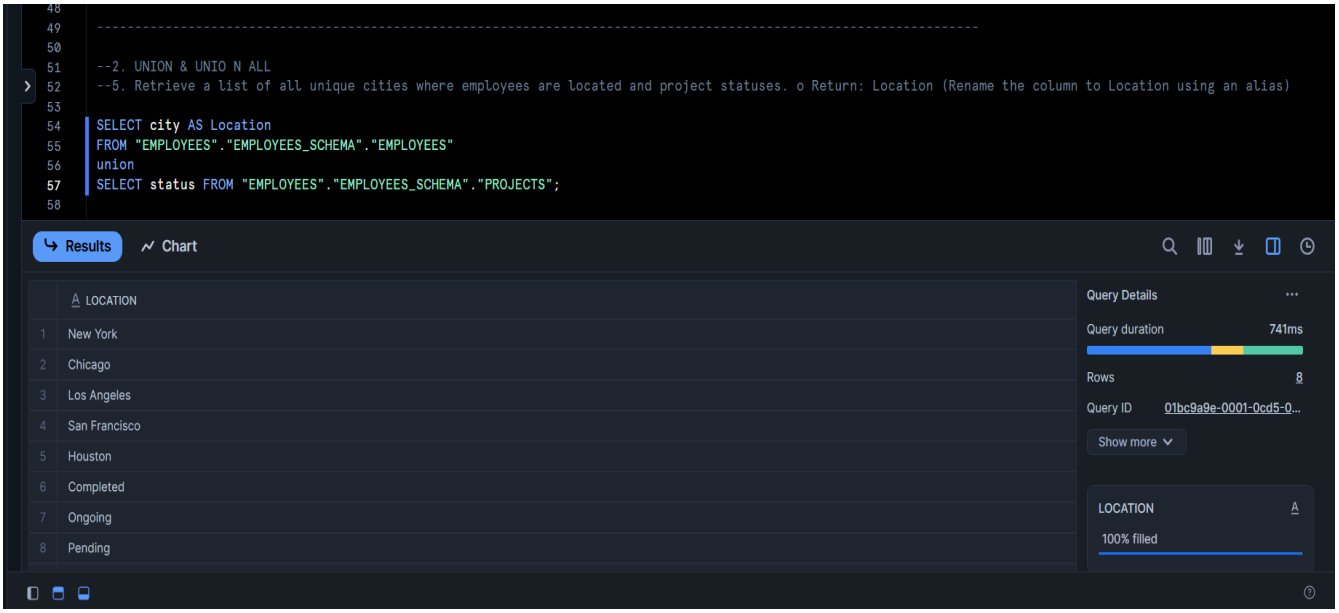
Query Details
Query duration 2.3s
Rows 6
Query ID 01bc9a85-0001-0cb3-0...
Show more
PROJECTID

4. Retrieve all employees and projects, including those without a match in either table using a **FULL OUTER JOIN**.
   - Return: EmployeeID, FirstName, LastName, Department, Salary, ProjectID, ProjectName, Budget, Status.

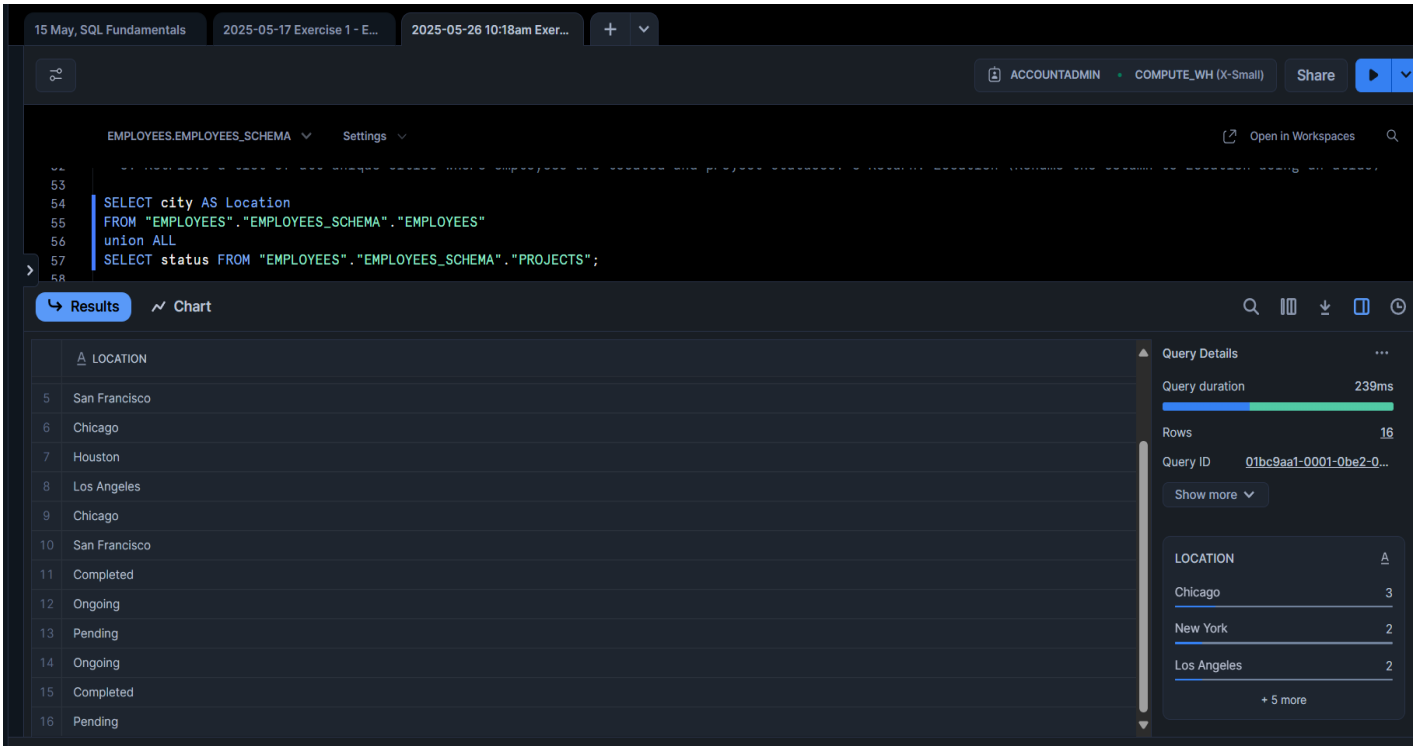


2. UNION & UNION ALL

5. Retrieve a list of all unique cities where employees are located and project statuses. o Return: Location (Rename the column to Location using an alias).



6. Retrieve a list of all cities where employees are located and project statuses, allowing duplicates. o Return: Location (Rename the column to Location using an alias).



3. Filtering Statements

7. Retrieve employees who earn more than 70,000. o Return: EmployeeID, FirstName, LastName, Department, Salary.

NO RESULTS

8. Retrieve employees working in either IT or Finance departments. o Return: EmployeeID, FirstName, LastName, Department, Salary.

```sql
-- 8. Retrieve employees working in either IT or Finance departments. Return: EmployeeID, FirstName, LastName, Department, Salary.

SELECT id, first_name, last_name, department, salary
FROM "EMPLOYEES"."EMPLOYEES_SCHEMA"."EMPLOYEES"
WHERE department IN ('IT','Finance');
```

Results | Chart

	ID	FIRST_NAME	LAST_NAME	DEPARTMENT	SALARY
1	1	John	Doe	IT	55000
2	3	Mike	Johnson	Finance	60000
3	4	Sarah	Brown	IT	53000
4	6	Emily	Davis	IT	62000
5	7	Robert	Wilson	Finance	59000

Query Details
Query duration 747ms
Rows 6
Query ID 01bc9be1-0001-0b38-0...
Show more

9. Retrieve projects that are not yet completed. o Return: ProjectID, ProjectName, Budget, Status.

```sql
SELECT PROJECTID, PROJECTNAME, BUDGET, STATUS
FROM "EMPLOYEES"."EMPLOYEES_SCHEMA"."PROJECTS"
WHERE status != 'Completed';
```

Results | Chart

	PROJECTID	PROJECTNAME	BUDGET	STATUS
1	102	Employee Training	50000	Ongoing
2	103	Cybersecurity Audit	75000	Pending
3	104	Financial Analysis	90000	Ongoing
4	106	Risk Management	80000	Pending

Query Details
Query duration 226ms
Rows 4
Query ID 01bc9be4-0001-0cb3-0...
Show more

10. Retrieve projects that have a budget greater than 70,000 and are not completed. • Return: ProjectID, ProjectName, Budget, Status.

```sql
--10. Retrieve projects that have a budget greater than 70,000 and are not completed. · Return: ProjectID, ProjectName, Budget, Status.

SELECT PROJECTID, PROJECTNAME, BUDGET, STATUS
FROM "EMPLOYEES"."EMPLOYEES_SCHEMA"."PROJECTS"
WHERE status != 'Completed' AND BUDGET > 70000;
```

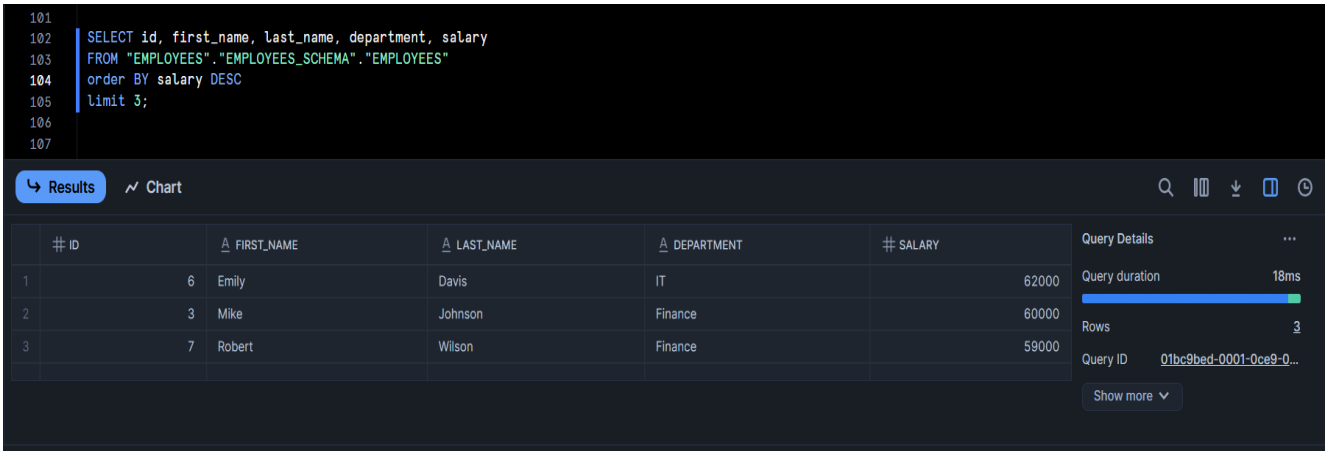
Results | Chart

	PROJECTID	PROJECTNAME	BUDGET	STATUS
1	103	Cybersecurity Audit	75000	Pending
2	104	Financial Analysis	90000	Ongoing
3	106	Risk Management	80000	Pending

Query Details
Query duration 311ms
Rows 3
Query ID 01bc9be8-0001-0ce9-0...
Show more

11. Retrieve employees from New York OR Toronto, ordered by salary in descending order.
• Return: EmployeeID, FirstName, LastName, Department, Salary, City.

12. Retrieve the top 3 highest-paid employees. • Return: EmployeeID, FirstName, LastName, Department, Salary.



4. Aggregate Functions with GROUP BY & HAVING

13. Find the total salary per department, sorted in descending order. • Return: Department, TotalSalary (Rename SUM(Salary) as TotalSalary).

```
108
109  -- 14. Aggregate Functions with GROUP BY & HAVING
110  -- 13. Find the total salary per department, sorted in descending order. · Return: Department, TotalSalary (Rename SUM(Salary) as TotalSalary).
111
112  SELECT DEPARTMENT, SUM(SALARY) AS TotalSalary
113  FROM "EMPLOYEES"."EMPLOYEES_SCHEMA"."EMPLOYEES"
114  GROUP BY DEPARTMENT;
115
```

Results | Chart

	DEPARTMENT	TOTALSALARY
1	IT	220000
2	HR	99000
3	Finance	119000
4	Marketing	105000

Query Details
Query duration 169ms
Rows 4
Query ID 01bc9bf9-0001-0ce9-0...
Show more

14. Find the average salary per city, but only include cities where the average salary is greater than 65,000. • Return: City, AverageSalary (Rename AVG(Salary) as AverageSalary).

```
116  ----------------------------------------------------------------------------------------------------------------
117
118  -- 14. Find the average salary per city, but only include cities where the average salary is greater than 65,000. · Return: City, AverageSalary (Rename
     AVG(Salary) as AverageSalary).
119
120  SELECT CITY, AVG(SALARY) AS AverageSalary
121  FROM "EMPLOYEES"."EMPLOYEES_SCHEMA"."EMPLOYEES"
122  GROUP BY CITY
123  HAVING AverageSalary > 65000;
124
```

Results | Chart

CITY	AVERAGESALARY

Query produced no results

Query Details
Query duration 51ms
Rows 0
Query ID 01bc9bfe-0001-0ce9-0...
Show more

15. Count the number of employees per department, including only departments with more than 1 employee. • Return: Department, EmployeeCount (Rename COUNT(EmployeeID) as EmployeeCount).

```
125 ----------------------------------------------------------------------------------------------------------------------------
126 -- 15. Count the number of employees per department, including only departments with more than 1 employee. · Return: Department, EmployeeCount (Rename
    COUNT(EmployeeID) as EmployeeCount).
127
128 SELECT Department, COUNT(ID) AS EmployeeCount
129 FROM "EMPLOYEES"."EMPLOYEES_SCHEMA"."EMPLOYEES"
130 GROUP BY DEPARTMENT
131 HAVING EmployeeCount > 1;
132
133
134
```

Results | Chart

	DEPARTMENT	EMPLOYEECOUNT
1	IT	4
2	HR	2
3	Finance	2
4	Marketing	2

Query Details
Query duration 266ms
Rows 4
Query ID 01bc9c01-0001-0b38-0...
Show more

16. Retrieve the number of projects per status, but only include statuses with at least 2 projects. • Return: Status, ProjectCount (Rename COUNT(ProjectID) as ProjectCount).

```
133 ----------------------------------------------------------------------------------------------------------------------------
134 --16. Retrieve the number of projects per status, but only include statuses with at least 2 projects. · Return: Status, ProjectCount (Rename COUNT(ProjectID)
    as ProjectCount).
135
136 SELECT STATUS, COUNT(PROJECTID) AS ProjectCount
137 FROM "EMPLOYEES"."EMPLOYEES_SCHEMA"."PROJECTS"
138 GROUP BY STATUS
139 HAVING ProjectCount >= 2;
140
141
```

Results | Chart

	STATUS	PROJECTCOUNT
1	Completed	2
2	Ongoing	2
3	Pending	2

Query Details
Query duration 384ms
Rows 3
Query ID 01bc9c06-0001-0d09-0...
Show more

17. Retrieve the total project budget per employee, but only for employees who are managing projects worth more than 150,000. • Return: EmployeeID, FirstName, LastName, TotalProjectBudget (Rename SUM(Budget) as TotalProjectBudget).

```
141  ----------------------------------------------------------------------------------------------------------------------------
142
143  -- 17. Retrieve the total project budget per employee, but only for employees who are managing projects worth more than 150,000. · Return: EmployeeID,
     FirstName, LastName, TotalProjectBudget (Rename SUM(Budget) as TotalProjectBudget).
144
145  SELECT e.ID, e.first_name, e.last_name, SUM(p.Budget) AS TotalProjectBudget
146  FROM "EMPLOYEES"."EMPLOYEES_SCHEMA"."EMPLOYEES" AS e
147  inner join "EMPLOYEES"."EMPLOYEES_SCHEMA"."PROJECTS" as p
148  ON e.ID = p.EMPLOYEEID
149  group by  e.ID, e.first_name, e.last_name
150  HAVING TotalProjectBudget > 150000;
151
152
```

Results | Chart

	ID	FIRST_NAME	LAST_NAME	TOTALPROJECTBUDGET
1	1	John	Doe	175000

Query Details

Query duration 91ms

Rows 1