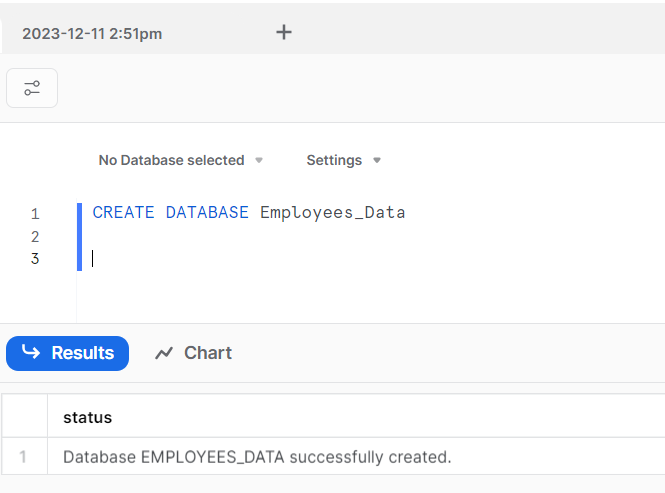
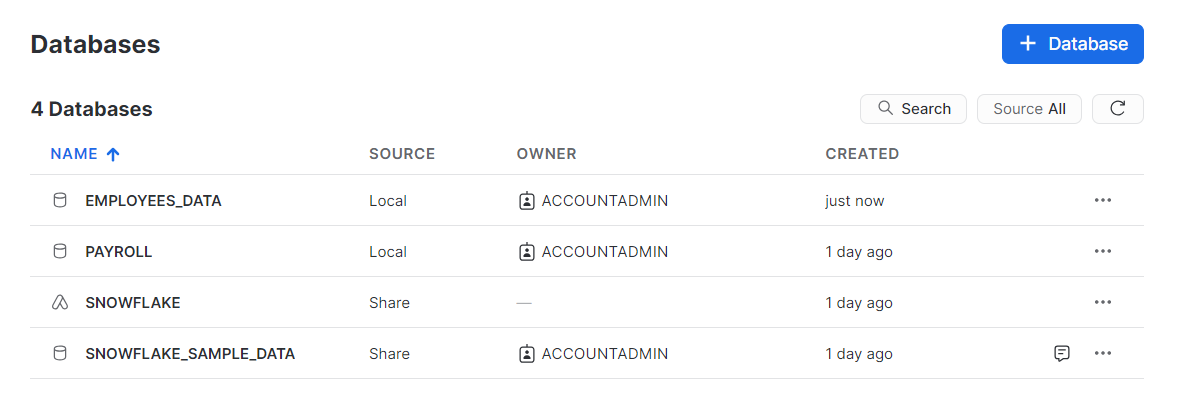
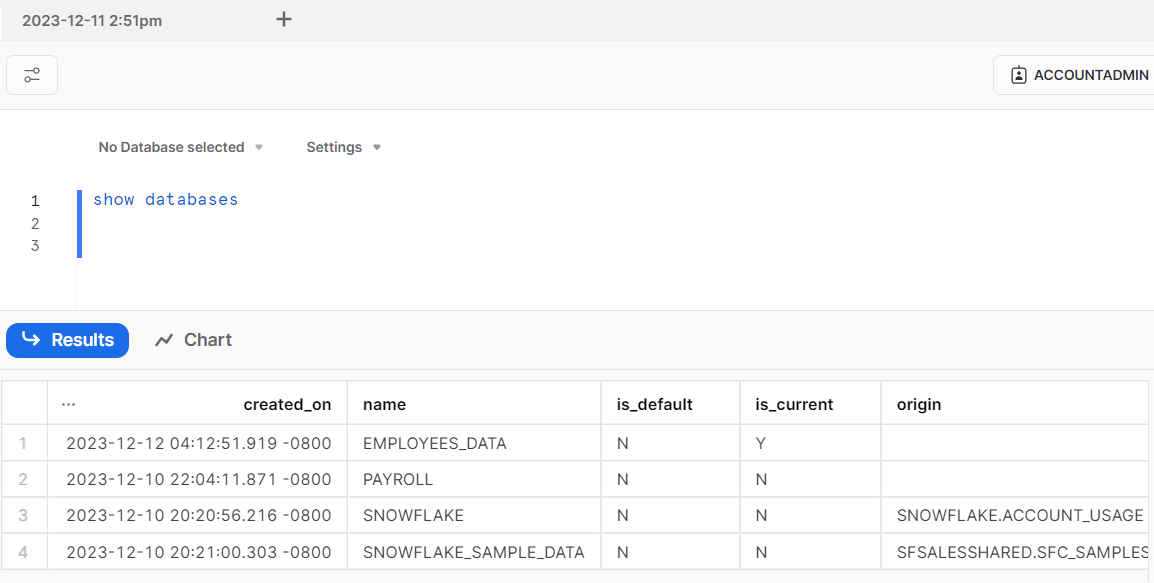
**SNOWFLAKE TASK**

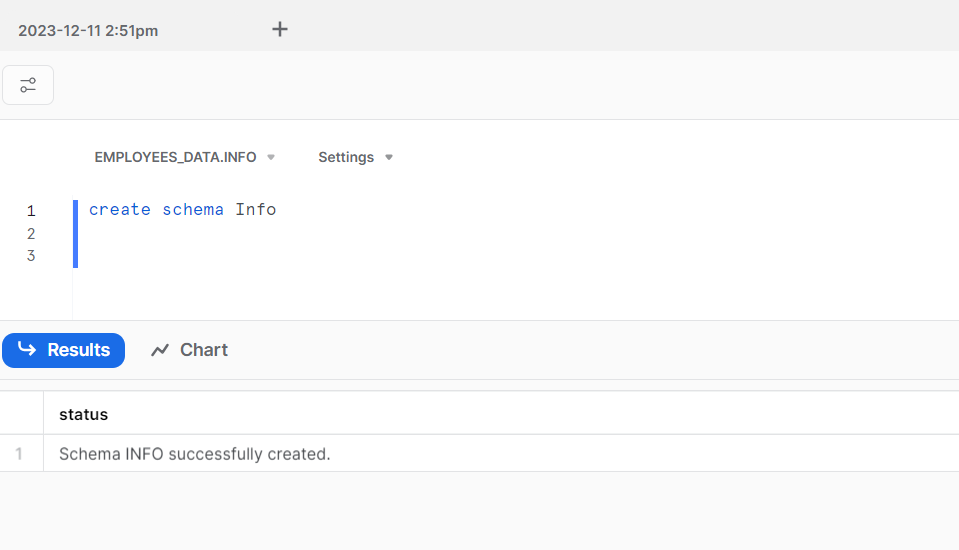
**Name: ADVITEEYA SHRAV**

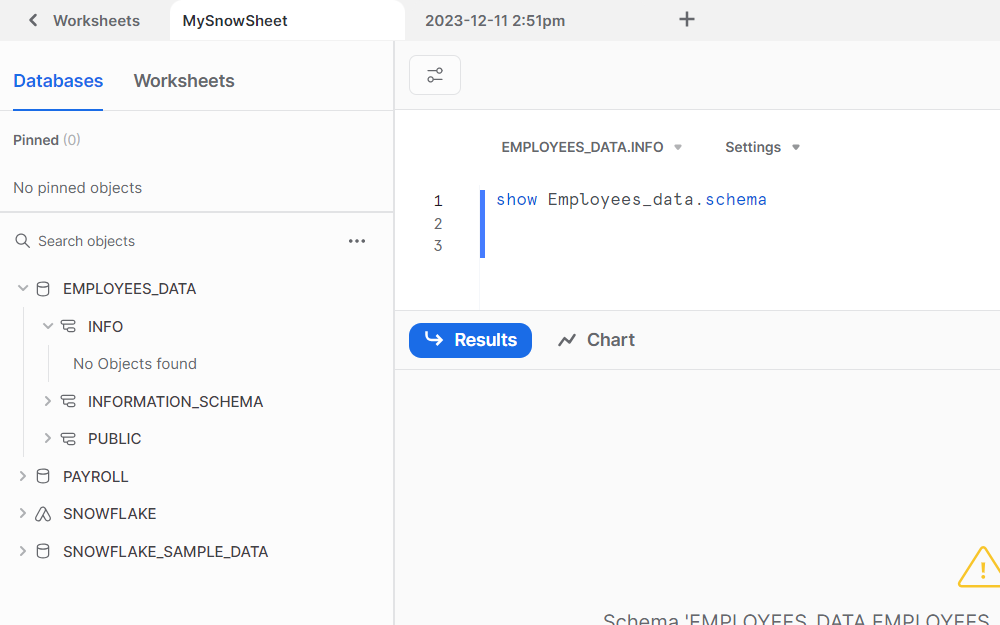
1. Create database called "Employees\_data", and schema by name "Info" using command line. [Do not use wizard]



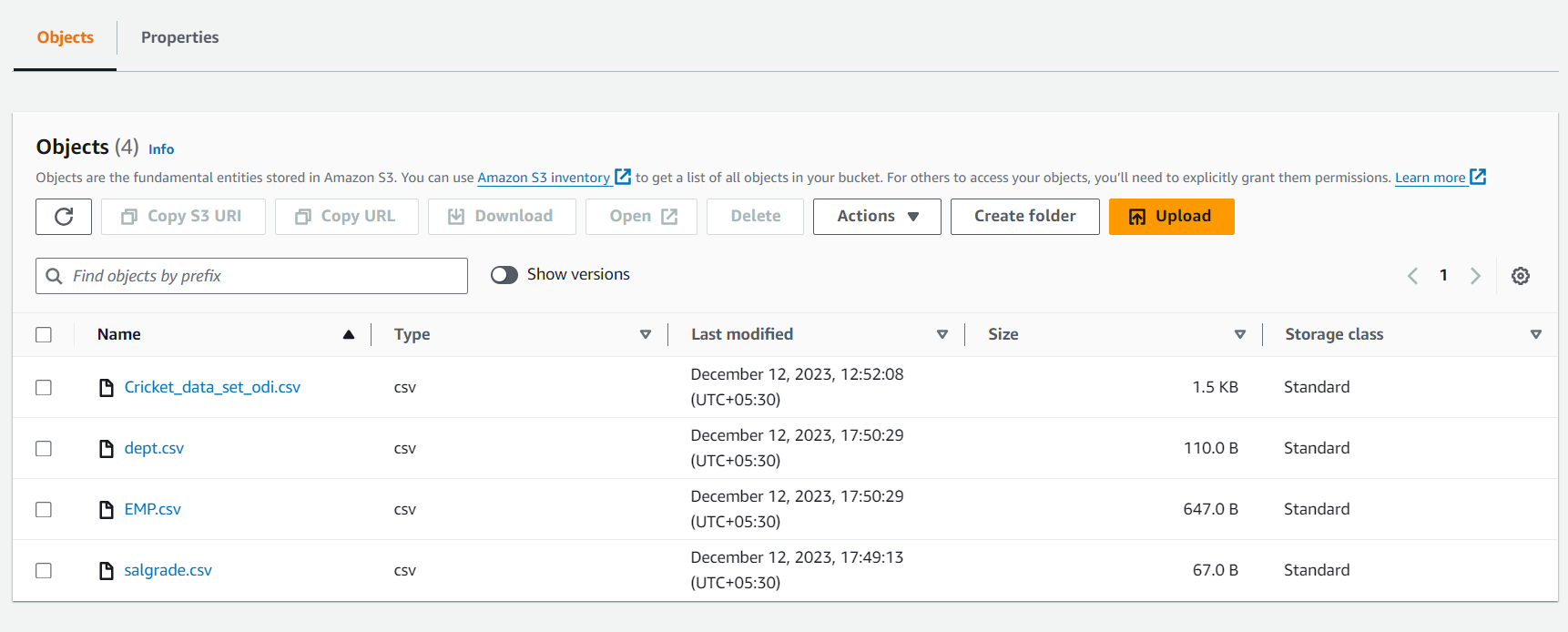




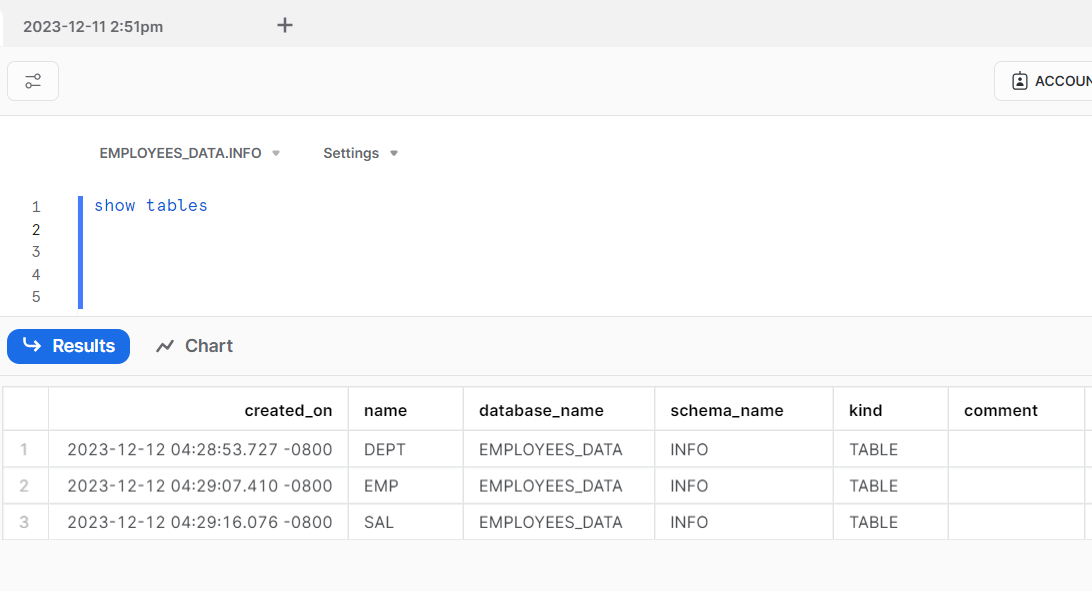




1. Upload emp.csv, dept.csv, and salgrade.csv files into S3 bucket under /data



1. Create appropriate tables and load the data accordingly.



1. Write 6 SQL queries of your own against the tables using normal SQL capabilities. Make sure queries contains grouping, sorting, and joins.

Creating policy stage for database EMPLOYEES\_DATA and schema Info

Creating policy : policy1234

{

"Version": "2012-10-17",

"Statement": [

{

"Effect": "Allow",

"Action": [

"s3:PutObject",

"s3:GetObject",

"s3:GetObjectVersion",

"s3:DeleteObject",

"s3:DeleteObjectVersion"

],

"Resource": "arn:aws:s3::: bootcamp123/data/\*"

},

{

"Effect": "Allow",

"Action": [

"s3:ListBucket",

"s3:GetBucketLocation"

],

"Resource": "arn:aws:s3::: bootcamp123",

"Condition": {

"StringLike": {

"s3:prefix": [

"data/\*"

]

}

}

}

]

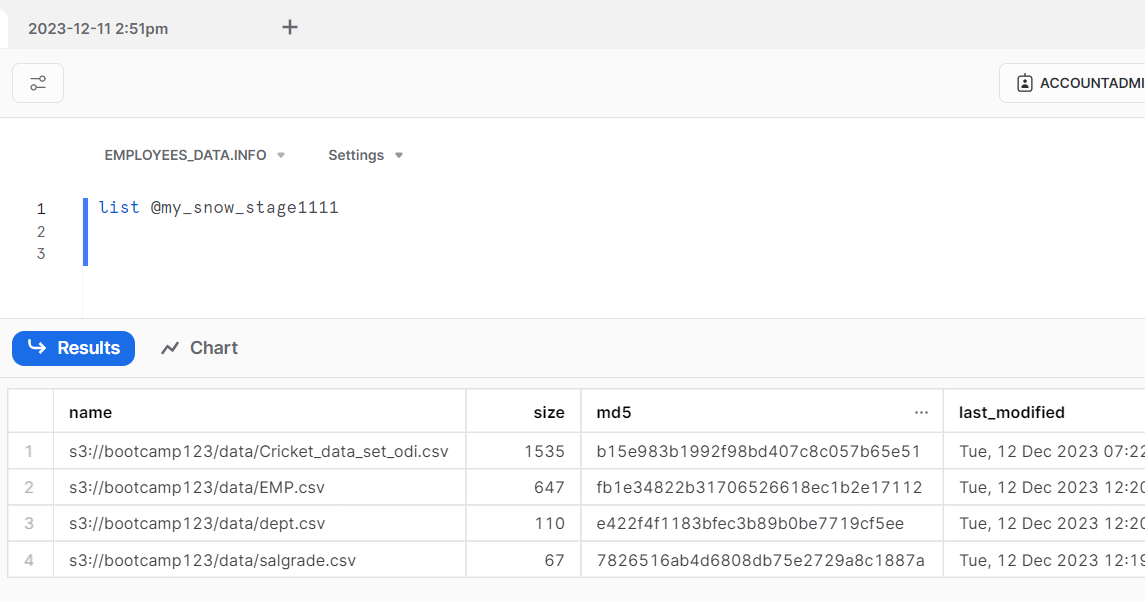
}

CREATE OR REPLACE STAGE my\_snow\_stage1111

URL='s3://bootcamp123/data/'

CREDENTIALS=(AWS\_KEY\_ID=' xxxxxxx'

AWS\_SECRET\_KEY=' xxx')

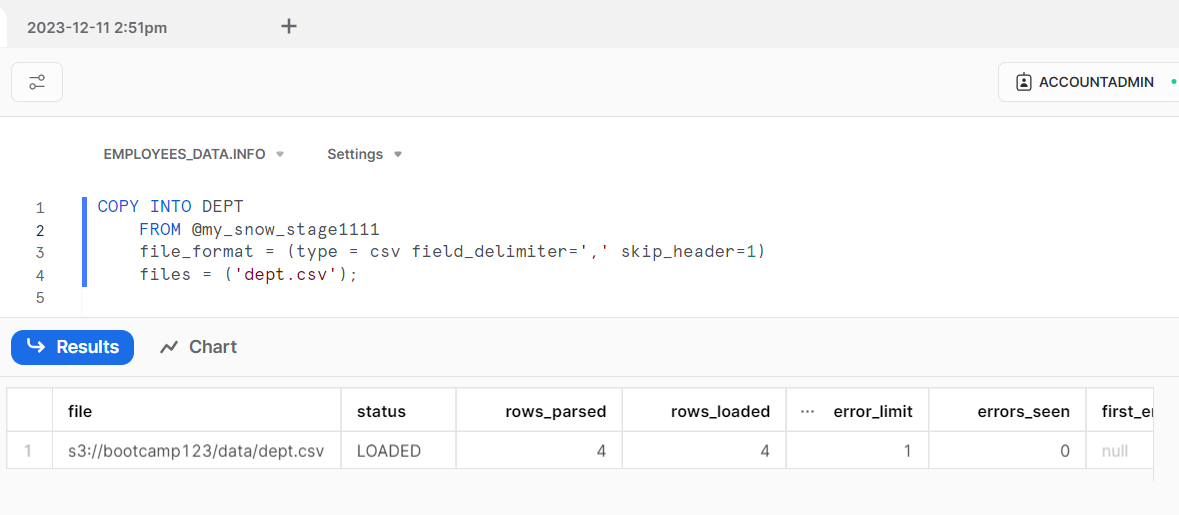


Loading data into the tables

COPY INTO DEPT

FROM @my\_snow\_stage1111

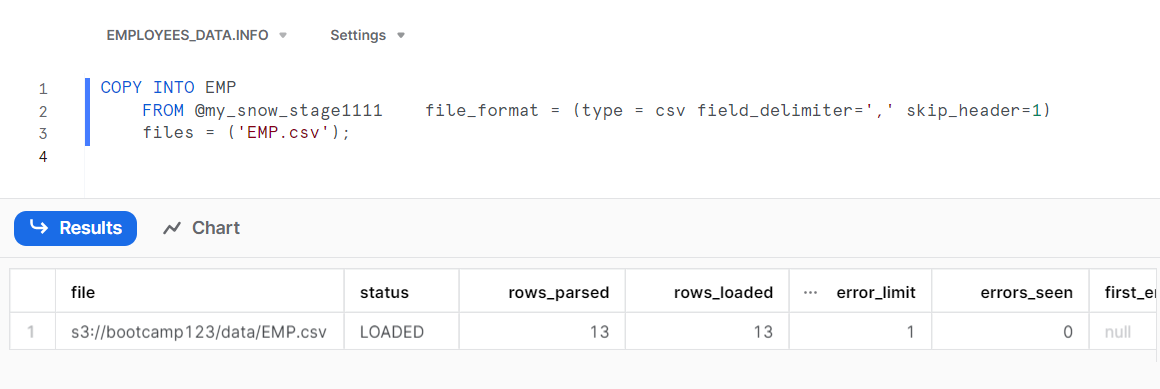
file\_format = (type = csv field\_delimiter=',' skip\_header=1)

files = ('dept.csv');

COPY INTO EMP

FROM @my\_snow\_stage1111 file\_format = (type = csv field\_delimiter=',' skip\_header=1)

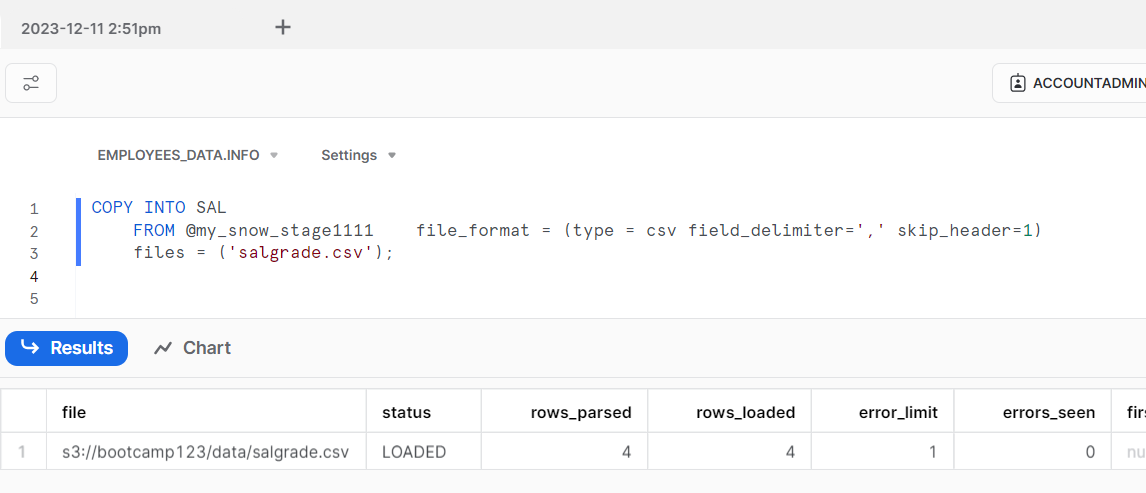
files = ('EMP.csv');



COPY INTO SAL

FROM @my\_snow\_stage1111 file\_format = (type = csv field\_delimiter=',' skip\_header=1)

files = ('salgrade.csv');



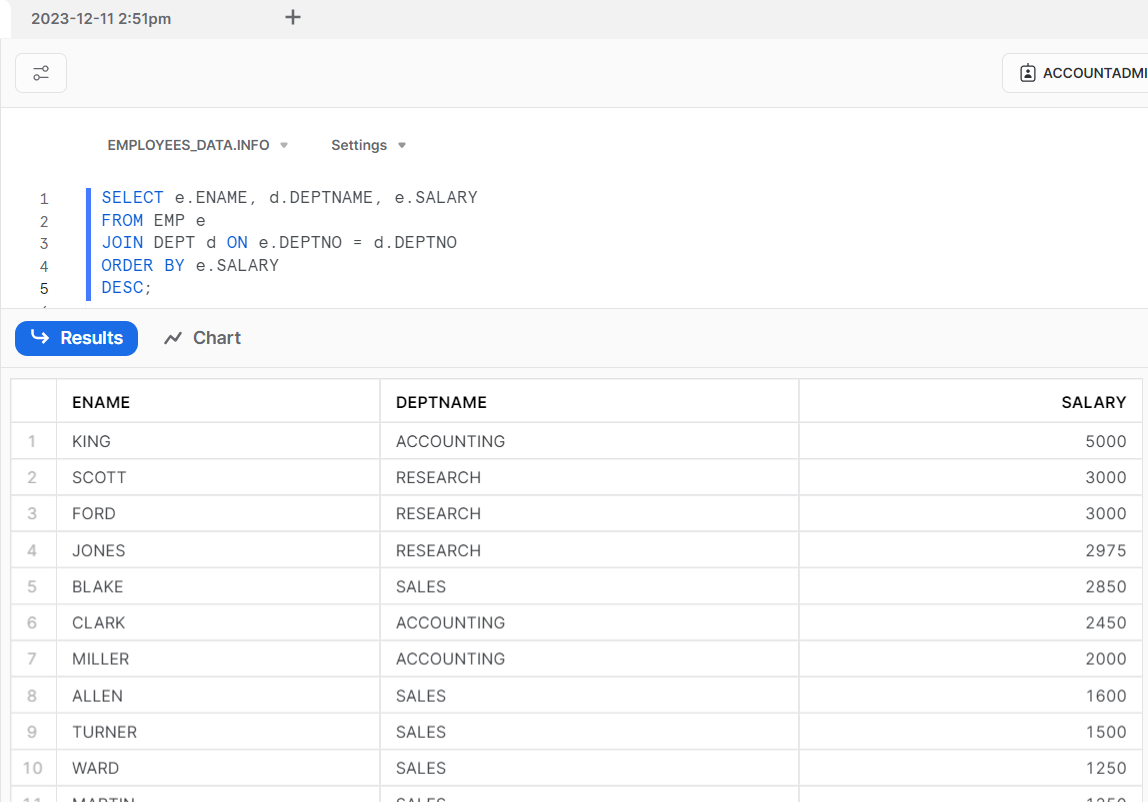
1.Retrieve the employee names, department names, and salaries in descending order of salary.

SELECT e.ENAME, d.DNAME, e.SAL

FROM EMP e

JOIN DEPT d ON e.DEPTNO = d.DEPTNO

ORDER BY e.SAL DESC;



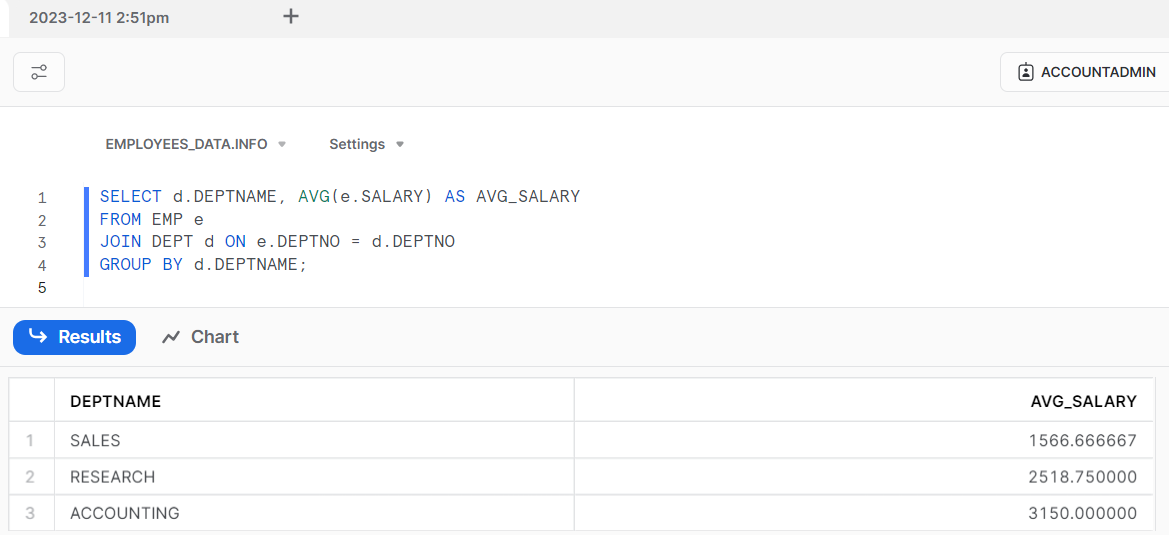
2. Find the average salary for each department.

SELECT d.DEPTNAME, AVG(e.SALARY) AS AVG\_SALARY

FROM EMP e

JOIN DEPT d ON e.DEPTNO = d.DEPTNO

GROUP BY d.DEPTNAME;

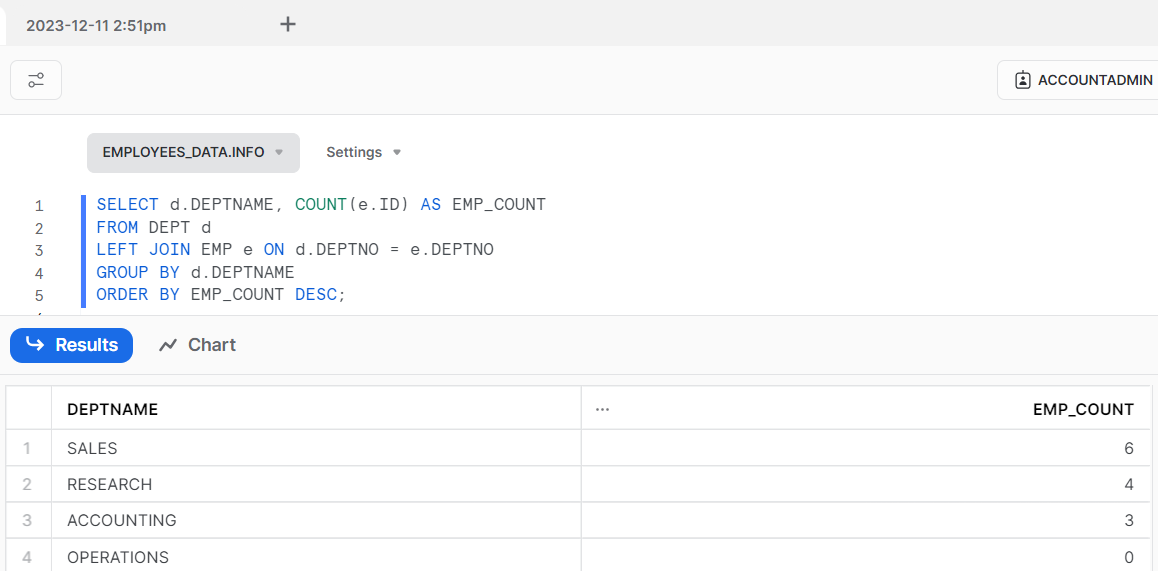


3. Retrieve the department name and the total number of employees in each department, sorted in descending order of employee count.

SELECT d.DEPTNAME, COUNT(e.ID) AS EMP\_COUNT

FROM DEPT d LEFT JOIN EMP e ON d.DEPTNO = e.DEPTNO

GROUP BY d.DEPTNAME ORDER BY EMP\_COUNT DESC;

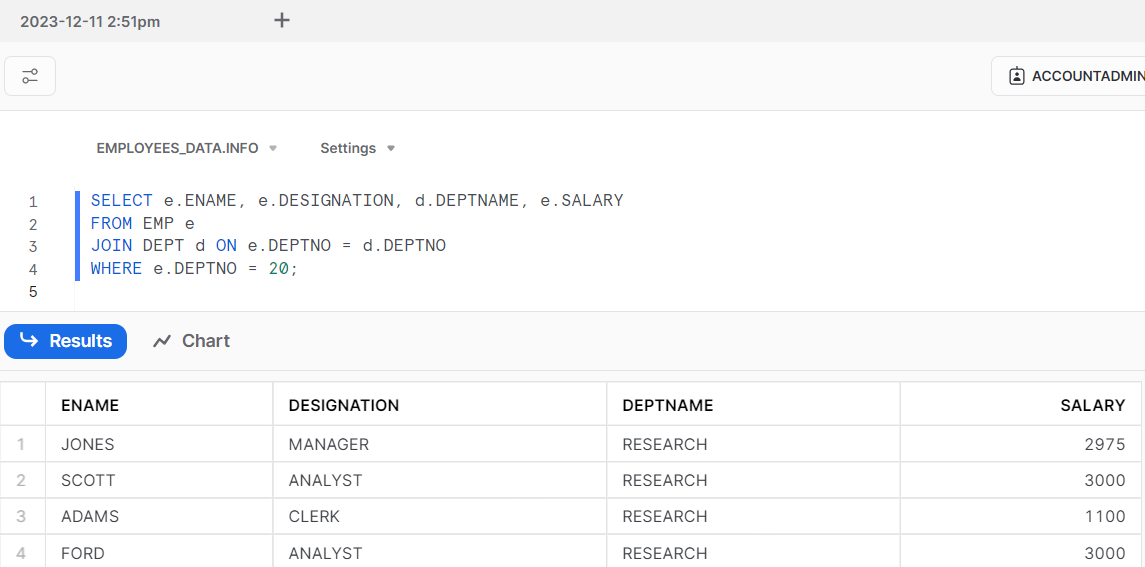


4.Retrieve the employee name, job title, department name, and salary for employees in department 20.

SELECT e.ENAME, e.DESIGNATION, d.DEPTNAME, e.SALARY

FROM EMP e JOIN DEPT d ON e.DEPTNO = d.DEPTNO

WHERE e.DEPTNO = 20;



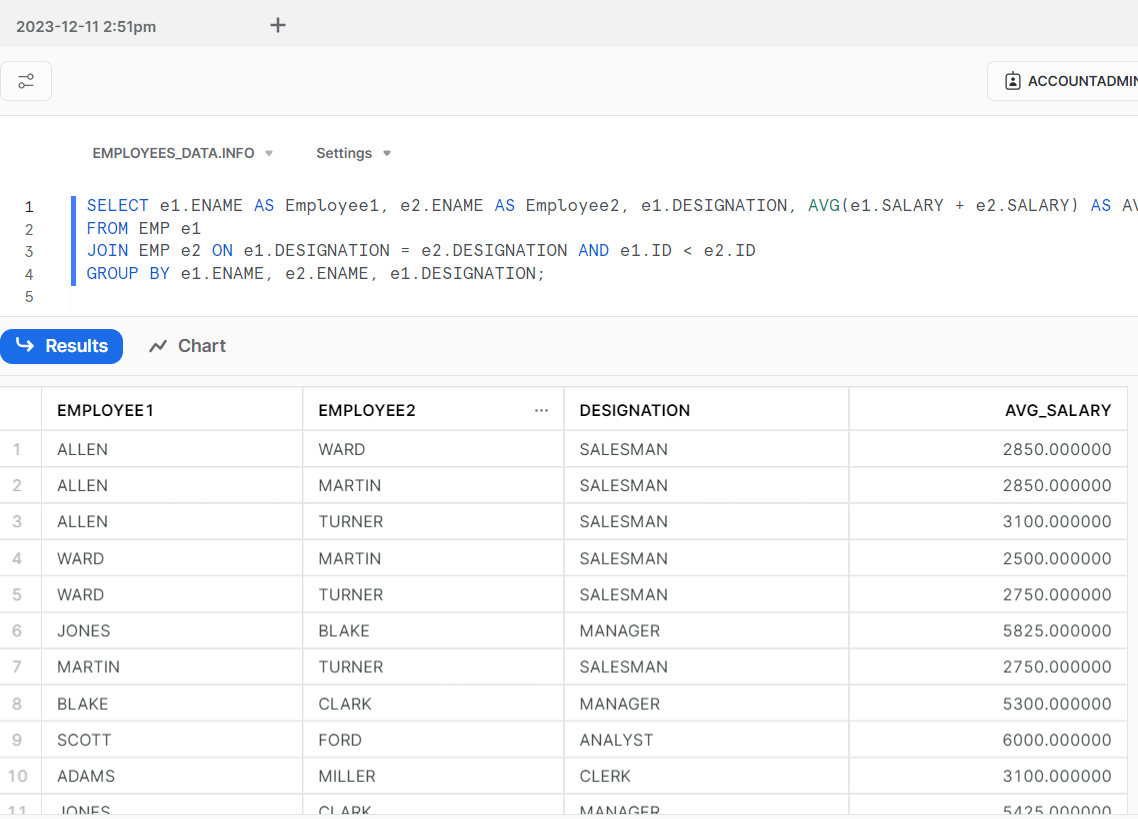
Find pairs of employees who have the same job title and calculate the average salary for each pair.

SELECT e1.ENAME AS Employee1, e2.ENAME AS Employee2, e1.JOB, AVG(e1.SALARY + e2.SALARY) AS AVG\_SALARY

FROM EMP e1

JOIN EMP e2 ON e1.DESIGNATION = e2.DESIGNATION AND e1.ID < e2.ID

GROUP BY e1.ENAME, e2.ENAME, e1.DESIGNATION;



6.USAGE OF WINDOWING FUNCTION.

