

DATA 226- DATAWAREHOUSE

Homework 1

Name : Vimalanandhan Sivanandham

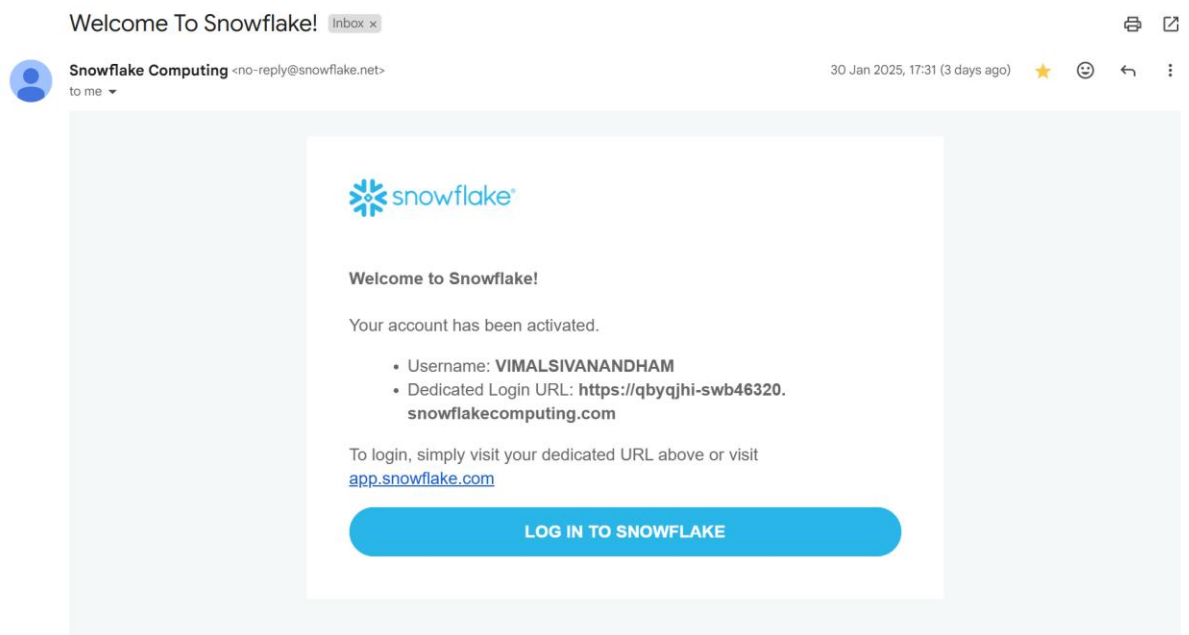
SJSU ID: 017596436

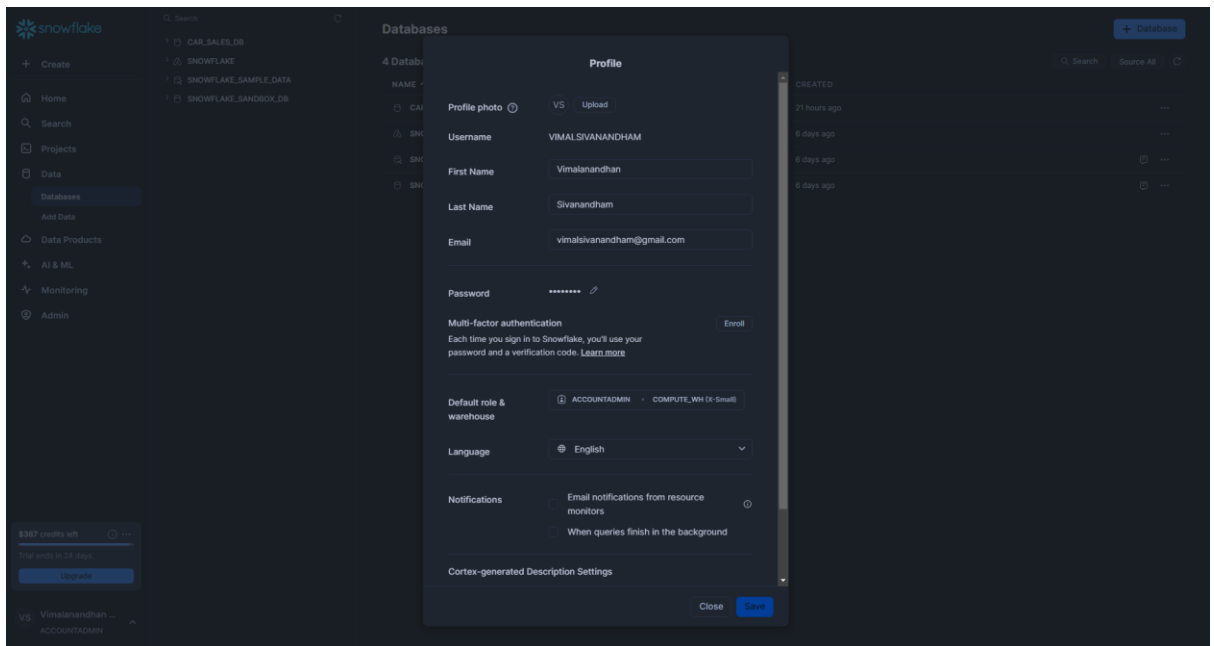
Assignment - Snowflake Environment Setup

Perform the following in Snowflake:

Car Sales csv(synthetic) dataset is under the Files/Homeworks/Car_Sales_Data.csv

1. (+1) Create Snowflake Account:





Account Created.

2. (+2) Create Database 'car_sales_db', Schema 'car_sales_schema', Table 'car_sales_tbl' (understand the dataset structure). Use the warehouse of size XSMALL.

Created **Database** named as '**car_sales_db**'

Created **Schema** named as '**car_sales_schema**'

Created **Table** named as '**car_sales_tbl**'

While creating the warehouse of Size **XSMALL**

CODE:

-- Create the database

```
CREATE DATABASE car_sales_db;
```

```
CREATE WAREHOUSE compute_wh
```

```
WITH WAREHOUSE_SIZE = 'XSMALL';
```

-- Create the schema inside the database

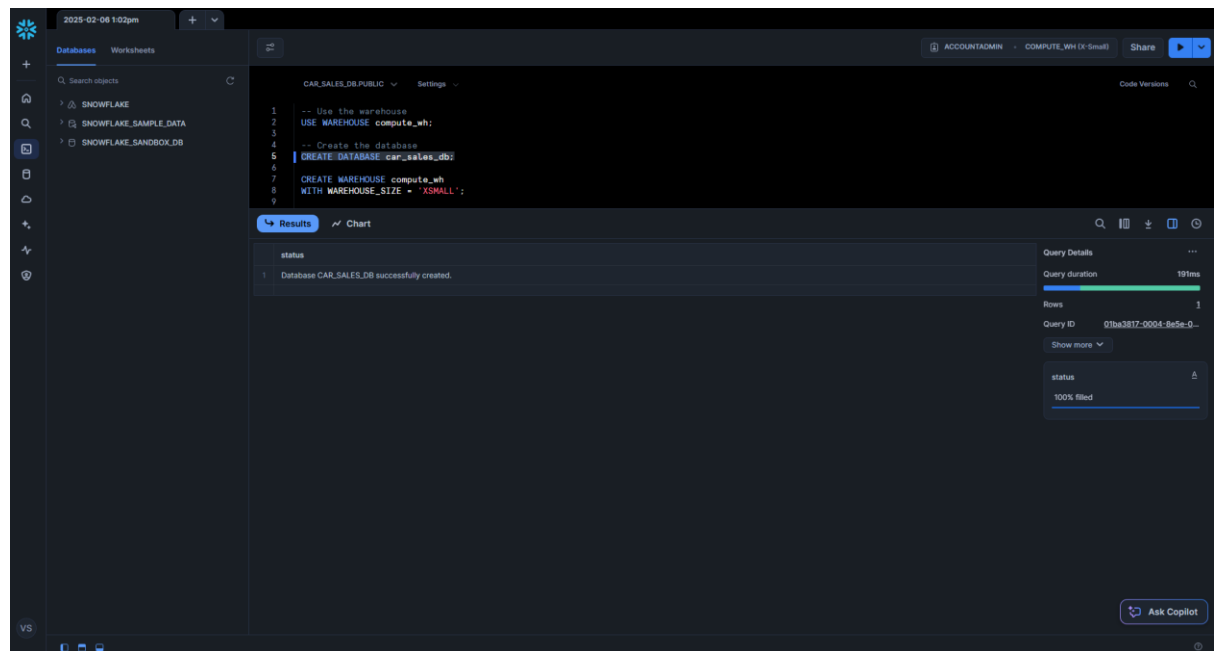
```
CREATE SCHEMA car_sales_schema;
```

-- Use the schema

USE SCHEMA car_sales_schema;

--- Creation Table

```
CREATE OR REPLACE TABLE car_sales_tbl (  
  car_name STRING,  
  sale_date DATE,  
  units_sold INT  
);
```



The screenshot displays the Snowflake SQL Editor interface. The top navigation bar includes the date and time '2025-02-06 1:02pm', a '+' icon, and a dropdown menu. The left sidebar contains a search bar and a tree view of the database structure, showing 'SNOWFLAKE', 'SNOWFLAKE_SAMPLE_DATA', and 'SNOWFLAKE_SANDBOX_DB'. The main editor area shows a SQL script with the following content:

```

1
2
3 -- Create the database
4 CREATE DATABASE car_sales_db;
5
6
7 CREATE WAREHOUSE compute_wh
8 WITH WAREHOUSE_SIZE = 'XSMALL';
9
10 -- Create the schema inside the database
11
12 CREATE SCHEMA car_sales_schema;
13
14 -- Use the schema

```

Below the SQL editor, the 'Results' tab is active, displaying a single row of status information:

status
1 Schema CAR_SALES_SCHEMA successfully created.

On the right side, the 'Query Details' panel provides additional information:

- Query duration:** 118ms
- Rows:** 1
- Query ID:** 01ba3818-0004-8a38-0...
- Show more:** (dropdown menu)
- status:** 100% filled

At the bottom right, there is an 'Ask Copilot' button.

The screenshot shows the Snowflake SQL Editor interface. The top bar displays the date and time '2025-02-06 1:02pm' and a dropdown menu. The left sidebar contains a 'Databases' tab and a 'Worksheets' tab. The main editor area displays a SQL script for creating a database, schema, and table. The script includes comments and SQL commands like 'CREATE DATABASE', 'CREATE WAREHOUSE', 'CREATE SCHEMA', and 'CREATE OR REPLACE TABLE'. The bottom status bar shows 'Results' and 'Chart' tabs, with a message indicating 'Table CAR_SALES_TBL successfully created.'

```

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
-- Create the database
CREATE DATABASE car_sales_db;

-- Create the warehouse
CREATE WAREHOUSE compute_wh
WITH WAREHOUSE_SIZE = 'XSMALL';

-- Create the schema inside the database
CREATE SCHEMA car_sales_schema;

-- Use the schema
USE SCHEMA car_sales_schema;

CREATE OR REPLACE TABLE car_sales_schema.car_sales_tbl (
  car_name STRING,
  car_price INT,
  purchase_date DATE,
  brand_name STRING
);

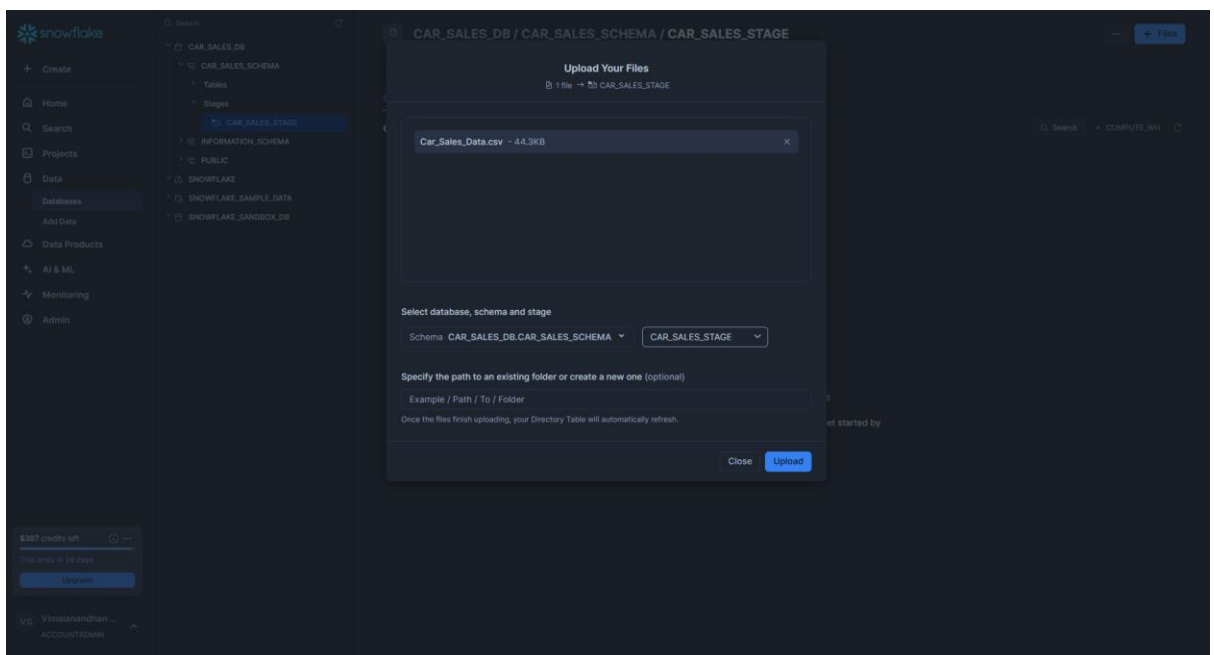
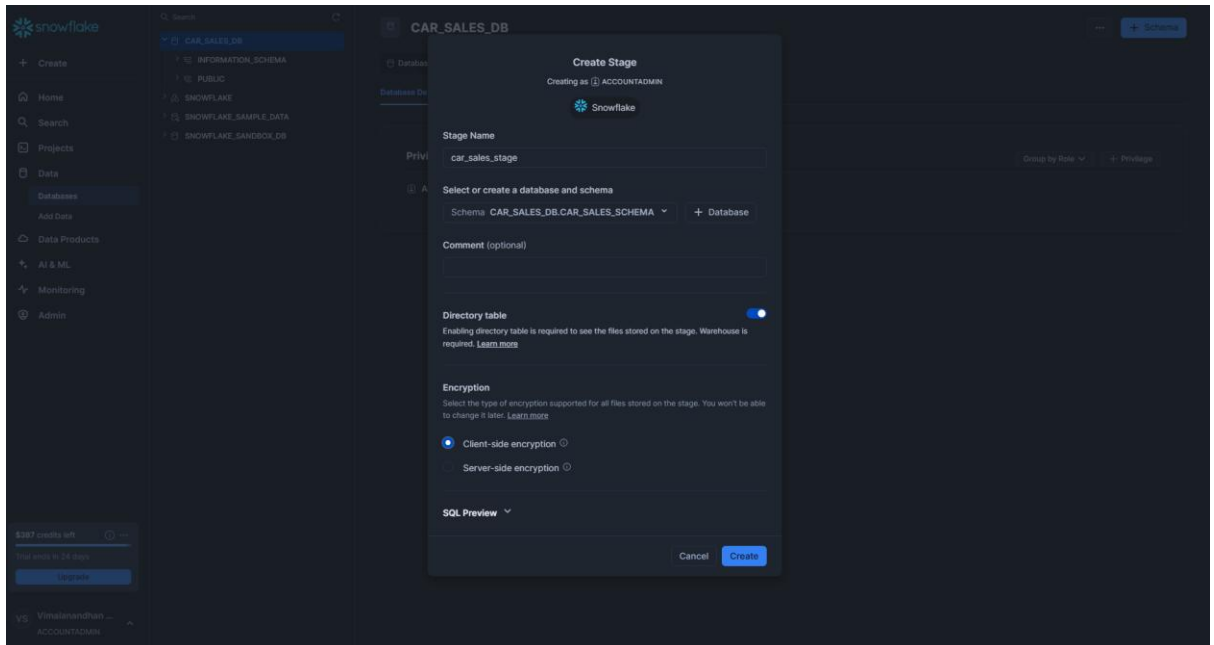
```

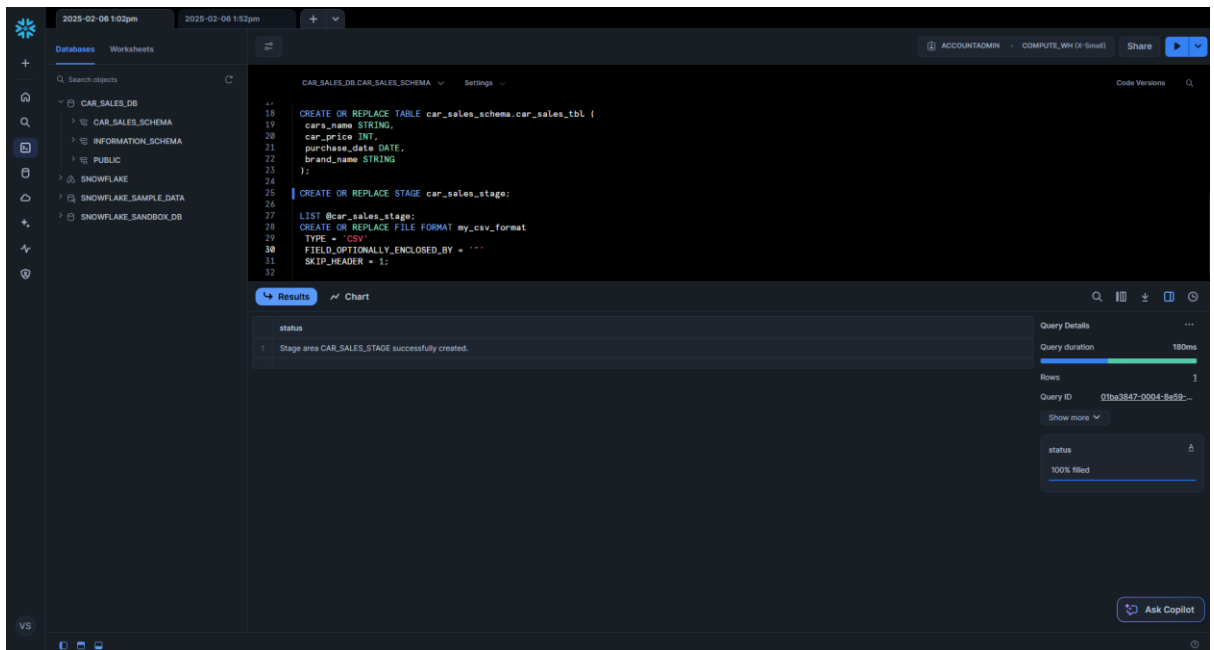
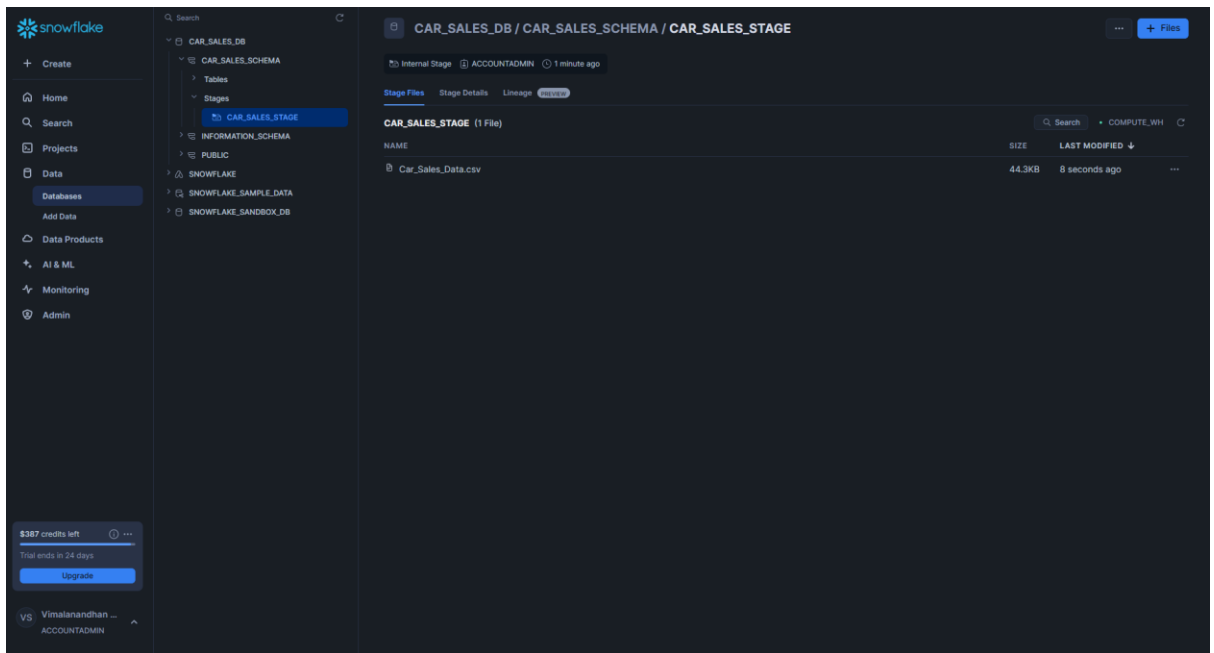
The bottom status bar shows 'Results' and 'Chart' tabs, with a message indicating 'Table CAR_SALES_TBL successfully created.'

3. (+1) Create Stage 'car_sales_stage' in Snowflake and load the csv.

CODE:

```
CREATE OR REPLACE STAGE car_sales_stage; LIST @car_sales_stage; CREATE OR  
REPLACE FILE FORMAT my_csv_format TYPE = 'CSV'  
FIELD_OPTIONALLY_ENCLOSED_BY = '"' SKIP_HEADER = 1;
```





4. (+2) Create file format and Load Data into the 'car_sales_data' table (COPY INTO).

CODE:

-----Creation of File

CREATE OR REPLACE FILE FORMAT car_sales_file_format

TYPE = 'CSV'

FIELD_OPTIONALLY_ENCLOSED_BY = ''

SKIP_HEADER = 1;

-----Load Data into Table

COPY INTO car_sales_tbl

FROM @car_sales_stage/Car_Sales_Data.csv

FILE_FORMAT = (FORMAT_NAME = car_sales_file_format);

DESC TABLE car_sales_tbl;

The screenshot displays the Snowflake SQL IDE interface. On the left, the 'Databases' sidebar shows a tree view with 'CAR_SALES_DB' expanded, containing 'Tables' (CAR_SALES_TBL), 'Stages' (CAR_SALES_STAGE), and 'File Formats' (MY_CSV_FORMAT). The main editor shows a SQL script with the following content:

```
CREATE OR REPLACE TABLE @car_sales_stage/car_sales_tbl (  
  car_name STRING,  
  car_price INT,  
  purchase_date DATE,  
  brand_name STRING  
);  
  
CREATE OR REPLACE STAGE car_sales_stage;  
  
LIST @car_sales_stage;  
  
CREATE OR REPLACE FILE FORMAT my_csv_format  
  TYPE = CSV  
  FIELD_OPTIONALLY_ENCLOSED_BY = ''  
  SKIP_HEADER = 1;
```

The 'Results' tab at the bottom shows a single row with the status: 'File format MY_CSV_FORMAT successfully created.' The 'Query Details' panel on the right indicates a query duration of 103ms, 1 row returned, and a status of '100% filled'.

```

1  -----Creation of Database
2  CREATE DATABASE car_sales_db;
3  USE DATABASE car_sales_db;
4  -----Creation of SCHEMA
5  CREATE SCHEMA car_sales_schema;
6  USE SCHEMA car_sales_schema;
7  --- Creation Table
8  CREATE OR REPLACE TABLE car_sales_tbl (
9    car_name STRING,
10   sale_date DATE,
11   units_sold INT
12 );
13 -----Creation of STAGE
14 CREATE OR REPLACE STAGE car_sales_stage;
15 -----Creation of File
16 CREATE OR REPLACE FILE FORMAT car_sales_file_format
17 TYPE = 'CSV'
18 FIELD_OPTIONALLY_ENCLOSED_BY = '|'
19 SKIP_HEADER = 1;
20 -----Load Data into Table
21 COPY INTO car_sales_tbl
22 FROM @car_sales_stage/Car_Sales_Data.csv
23 FILE_FORMAT = (FORMAT_NAME = car_sales_file_format);
24 DESC TABLE car_sales_tbl;
25
26 -- 5)Write an SQL queries for each that shows the following insights
27 -- 1)The average number of cars sold per month in 2024.
28
29 SELECT
30   MONTH(purchase_date) AS MONTH,
31   COUNT(*) AS AVG_CARS_SOLD
32 FROM car_sales_tbl
33 WHERE YEAR(purchase_date) = 2024
34 GROUP BY MONTH
35 ORDER BY MONTH;

```

```

14 -- Use the schema
15 USE SCHEMA car_sales_schema;
16
17 CREATE OR REPLACE TABLE car_sales_schema.car_sales_tbl (
18   car_name STRING,
19   car_price INT,
20   purchase_date DATE,
21   brand_name STRING
22 );
23
24 CREATE OR REPLACE STAGE car_sales_stage;
25
26 LIST @car_sales_stage;
27
28 CREATE OR REPLACE FILE FORMAT my_csv_format
29 TYPE = 'CSV'
30 FIELD_OPTIONALLY_ENCLOSED_BY = '|'
31 SKIP_HEADER = 1;
32
33 SELECT * FROM "CAR_SALES_DB"."CAR_SALES_SCHEMA"."CAR_SALES_TBL" LIMIT 10;

```

CARS_NAME	CAR_PRICE	PURCHASE_DATE	BRAND_NAME
Chevrolet Silverado	35000	2024-11-15	Chevrolet
Audi A4	35000	2024-05-08	Tesla
Audi A4	24000	2024-01-31	BMW
Hyundai Elantra	21000	2024-05-24	Audi
Mercedes C-Class	45000	2024-02-28	Nissan
Mercedes C-Class	45000	2024-12-01	Hyundai
Audi A4	47000	2024-05-23	Chevrolet
Mercedes C-Class	45000	2024-10-25	BMW
Hyundai Elantra	25000	2024-10-13	Audi
Audi A4	47000	2024-05-17	Audi

5. (+6) Write an SQL queries for each that shows the following insights

1. The average number of cars sold per month in 2024.

With regard to step 5, the expected output format is mentioned below:

A. Expected Output

MONTH	AVG_CARS_SOLD
1	4
2	116

3	94
4	106
5	89
6	99

...Till Month 12

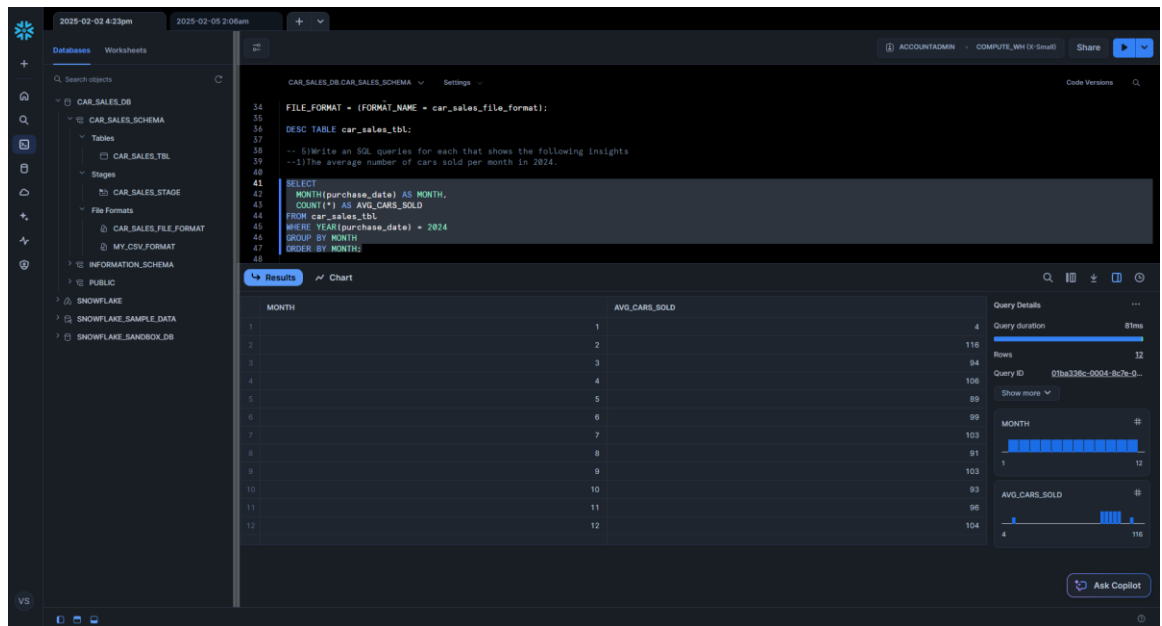
CODE:

5)Write an SQL queries for each that shows the following insights

1)The average number of cars sold per month in 2024.

```

SELECT
    MONTH(purchase_date) AS MONTH,
    COUNT(*) AS AVG_CARS_SOLD
FROM car_sales_tbl
WHERE YEAR(purchase_date) = 2024
GROUP BY MONTH
ORDER BY MONTH;
```



2. The top 5 best-selling cars along with the count in 2024.

B. Expected Output

CARS_NAME	TOTAL_SOLD
Mercedes C-Class	140
Hyundai Elantra	126

....till top 5

CODE:

2)The top 5 best-selling cars along with the count in 2024.

SELECT

cars_name,

COUNT(*) AS TOTAL_SOLD

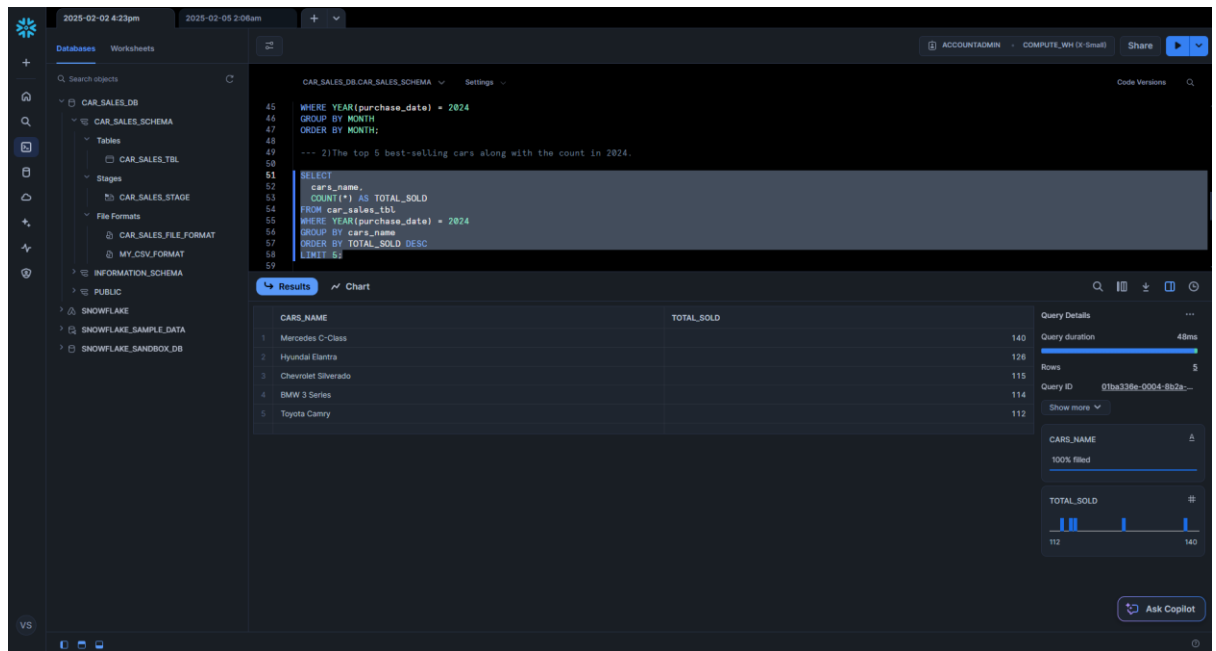
FROM car_sales_tbl

WHERE YEAR(purchase_date) = 2024

GROUP BY cars_name

ORDER BY TOTAL_SOLD DESC

LIMIT 5;



3. The top 3 least-selling cars along with the count in 2024.

C. Expected Output

CARS_NAME	TOTAL_SOLD
Audi A4	79

..till least 3

CODE:

3)The top 3 least-selling cars along with the count in 2024.

SELECT

cars_name,

COUNT(*) AS TOTAL_SOLD

FROM car_sales_tbl

WHERE YEAR(purchase_date) = 2024

GROUP BY cars_name

ORDER BY TOTAL_SOLD ASC

LIMIT 3;

2025-02-02 4:23pm2025-02-05 2:06am+

DatabasesWorksheets

Search objects

CAR_SALES_DB

CAR_SALES_SCHEMA

Tables

CAR_SALES_TBL

Stages

CAR_SALES_STAGE

File Formats

CAR_SALES_FILE_FORMAT

MY_CSV_FORMAT

INFORMATION_SCHEMA

PUBLIC

SNOWFLAKE

SNOWFLAKE_SAMPLE_DATA

SNOWFLAKE_SANDBOX_DB

CAR_SALES_DB.CAR_SALES_SCHEMA

Settings

Code Versions

58LIMIT 5;

59

60--- 3)The top 3 least-selling cars along with the count in 2024.

61

62SELECT

63cars_name,

64COUNT(*) AS TOTAL_SOLD

65FROM car_sales.tbl

66WHERE YEAR(purchase_date) = 2024

67GROUP BY cars_name

68ORDER BY TOTAL_SOLD ASC

69LIMIT 3;

70

71

ResultsChart

	CARS_NAME	TOTAL_SOLD	
1	Audi A4	79	
2	Nissan Altima	98	
3	Honda Civic	99	

Query Details

Query duration140ms

Rows3

Query ID01ba3372-0004-8b28-

Show more

CARS_NAME

100% filled

TOTAL_SOLD

#

79

99

Ask Copilot