

Snowflake Foundation Experiments 1-10 Overview

This series of experiments introduces you to the user interface and capabilities of Snowflake, and is designed specifically for use with the Snowflake, free 30-day trial at <https://trial.snowflake.com>. When done with the experiment you will be ready to load your own data into Snowflake and use its more advanced capabilities.

Target Audience

Database and Data Warehouse Administrators and Architects

What you'll learn

The tasks in this series of experiments will walk you through the steps to:

- Create stages, databases, tables, views, and warehouses
- Load structured and semi-structured data
- Query data including joins between tables
- Clone objects
- Undo user errors
- Create roles and users, and grant them privileges
- Securely and easily share data with other accounts



Remember that these experiments are for your benefit. Like TSA, if you see something, say something. We want to make sure that you have the best experience possible in this session and with these experiments. Thank you for active participation. 😊

Experiment 1: Snowflake Foundation

1.1 Steps to Prepare Your Experiment Environment

1.1.1 If not yet done, register for a Snowflake free 30-day trial at <https://trial.snowflake.com>. This is outlined in the Experiment 00-Getting-Started, as well. Remember that if you are in a China location you will have to be in VPN to another region, Hong Kong office or other to be able to successfully provision your Snowflake account.

- We chose the following

- Snowflake edition – Enterprise
- Snowflake cloud provider – AWS
- Region – Ohio – US EAST-2

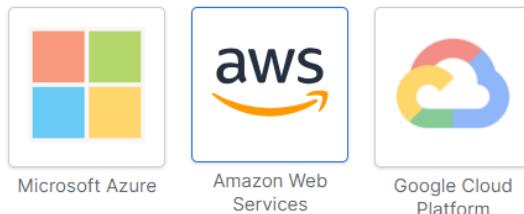
Choose your Snowflake edition*

Standard
A strong balance between features, level of support, and cost.

Enterprise
Standard plus 90-day time travel, multi-cluster warehouses, and materialized views.

Business Critical
Enterprise plus enhanced security, data protection, and database failover/fallback.

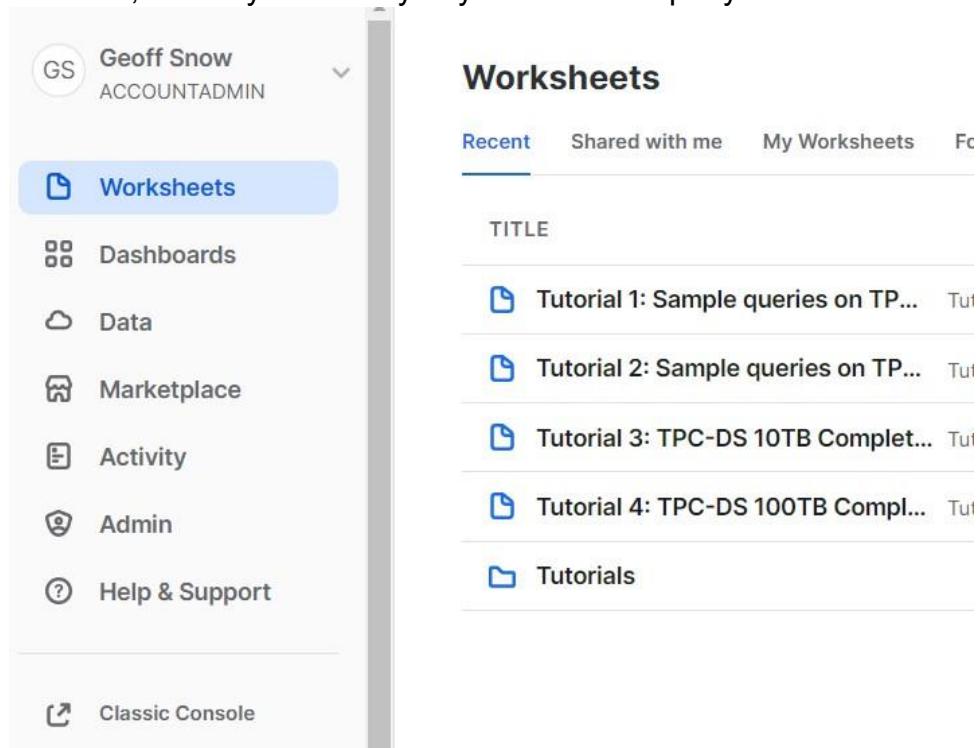
Choose your cloud provider*



- After registering, you will receive an email with an activation link and your Snowflake account URL. Bookmark this URL for easy, future access. After activation, you will create a user name and password. Write down these credentials. If you forget your URL you can always login through <https://app.snowflake.com/>

- 1.1.2 Resize your browser windows so you can view this experiment guide PDF and your web browser side-by-side to more easily follow the experiment instructions. If possible, use a secondary display dedicated to the experiment guide.
- 1.1.3 Open your snowflake environment, remember this is SaaS so you'll be logging into the Snowflake Management console application. Optimistically you bookmarked that account link but you can navigate to the login from snowflakecomputing.com, as well. The The

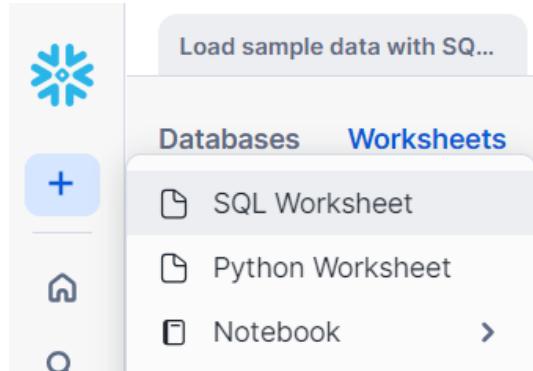
Classic Console in the lower left hand dialog pane view will not be visible in the Trial accounts, but may be for any of your older company accounts.



The screenshot shows the Snowflake Classic Console. On the left, there is a sidebar with a user profile (Geoff Snow, ACCOUNTADMIN) and a navigation menu. The 'Worksheets' option is highlighted. Other menu items include Dashboards, Data, Marketplace, Activity, Admin, Help & Support, and a 'Classic Console' link. The main content area is titled 'Worksheets' and shows a list of recent worksheets. The list includes:

| TITLE | TYPE |
|-------------------------------------|------|
| Tutorial 1: Sample queries on TP... | Tut |
| Tutorial 2: Sample queries on TP... | Tut |
| Tutorial 3: TPC-DS 10TB Complet... | Tut |
| Tutorial 4: TPC-DS 100TB Compl... | Tut |
| Tutorials | |

- 1.1.5 We'll select + SQL Worksheet from the newer console navigation to do a little initial exploration



The screenshot shows the Snowflake newer console navigation. On the left, there is a sidebar with a snowflake icon, a '+' button, a home icon, and a search icon. The 'Worksheets' option is highlighted. A dropdown menu is open, showing options: 'Load sample data with SQ...', 'Databases', 'Worksheets', 'SQL Worksheet' (which is selected and highlighted in grey), 'Python Worksheet', and 'Notebook'.

- 1.1.6 To use a database, we'll enter the SQL command

```
use database snowflake_sample_data;
```

2023-07-17 10:07am + PREVIEW

ACCOUNTADMIN • COMPUTE_WH Share ▶ ▾

SNOWFLAKE_SAMPLE_DATA.TPCDS_SF100TCL ▾ Settings ▾ Latest Version ▾

```
1 use database SNOWFLAKE_SAMPLE_DATA;
2 use schema tpcds_sf100tcl;
3 select cc_name, cc_manager from call_center;
4
5
6
```

↳ Results ▾ Chart

status

| | |
|---|----------------------------------|
| 1 | Statement executed successfully. |
|---|----------------------------------|

Query Details

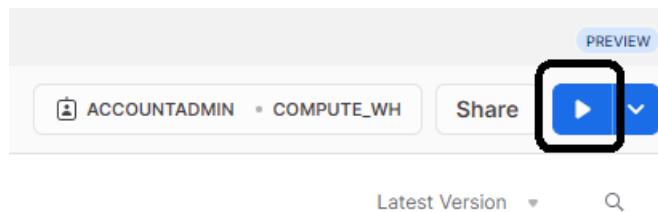
Query duration 86ms

Rows 1

Query ID 01adbe59-0001-3413-0...

status A

1.1.7 To execute the command we highlight the line of SQL and select the **Run** Arrow button



1.1.8 This allows us to see the query results and the dynamic query performance details.

PREVIEW

ACCOUNTADMIN • COMPUTE_WH Share ▶ ▾

Latest Version ▾

Query Details

Query duration 86ms

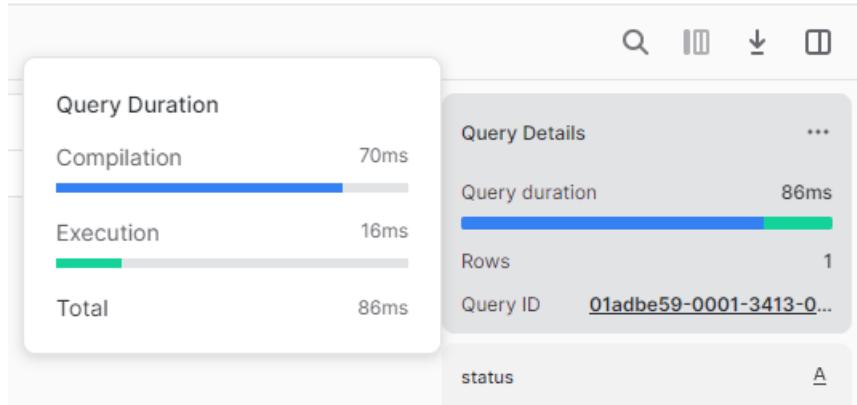
Rows 1

Query ID 01adbe59-0001-3413-0...

status A

1.1.9 Selecting **Run** again in the dialog presented allows us to execute the same SQL command. This would allow us to identify the effect of the statement execution on

caching and other Snowflake optimizations. Note that we can hover the Query duration to see the breakdown of the query Compilation and Execution timing.



1.1.10 We can then select the schema for our worksheet session by entering the SQL command

```
use schema TPCDS_SF100TCL;
```

1.1.10 Highlight that SQL, and select Run to execute our choice of schema at that point we can run queries that don't have the database and schema qualifiers like

```
select cc_name,cc_manager from call_center;
```

1.1.11 Highlighting that SQL and running it shows us the results

SNOWFLAKE_SAMPLE_DATA.TPCDS_SF100TCL ▾ Settings ▾

```

1 use database SNOWFLAKE_SAMPLE_DATA;
2
3 use schema tpcds_sf100tcl;
4
5 select cc_name, cc_manager from call_center;
6

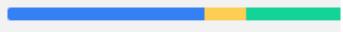
```

↳ Results ▾ Chart

| | CC_NAME | CC_MANAGER |
|---|-------------------|----------------|
| 1 | NY Metro | Bob Belcher |
| 2 | Mid Atlantic | Felipe Perkins |
| 3 | Mid Atlantic | Mark Hightower |
| 4 | North Midwest | Larry Mccray |
| 5 | North Midwest | Larry Mccray |
| 6 | North Midwest | Larry Mccray |
| 7 | Pacific Northwest | Alden Snyder |
| 8 | California | Wayne Ray |
| 9 | California | Wayne Ray |

Query Details

Query duration



Rows

Query ID [01adbe83-0001-3413](#)

CC_NAME

- North Midwest
- Hawaii/Alaska
- North Midwest_1

+ 27 more

1.1.11 Next, we'll expand the Tables under the SNOWFLAKE_SAMPLE_DATA database and TPCDS_SF100TCL schema. Select the CALL_CENTER table.

Databases Worksheets

Pinned (0)

No pinned objects

All Objects

...

> ⏺ SNOWFLAKE

▼ ⏺ SNOWFLAKE_SAMPLE_DATA

 > ⏺ INFORMATION_SCHEMA

 ▼ ⏺ TPCDS_SF100TCL

 ▼ Tables

 CALL_CENTER

 CATALOG_PAGE

1.1.12 The definition will come up in the pane below the Table selection. Select the ellipses ... next to the CALL_CENTER table name and choose **Preview Data**

| CALL_CENTER | | 60 Rows | ... |
|-------------|--------------|--------------|-----------------------|
| # | CC_CALL_CEN | | Place Name in Editor |
| A | CC_CALL_CEN | | Add Columns in Editor |
| (L) | CC_REC_STA | | View Definition |
| (L) | CC_REC_END | | Preview Data |
| # | CC_CLOSED_ | | Load Data |
| # | CC_OPEN_DA | | Pin |
| A | CC_NAME | | |
| A | CC_CLASS | | |
| # | CC_EMPLOYEES | NUMBER(38,0) | |
| # | CC_SQ_FT | NUMBER(38,0) | |
| | | | □ □ □ |

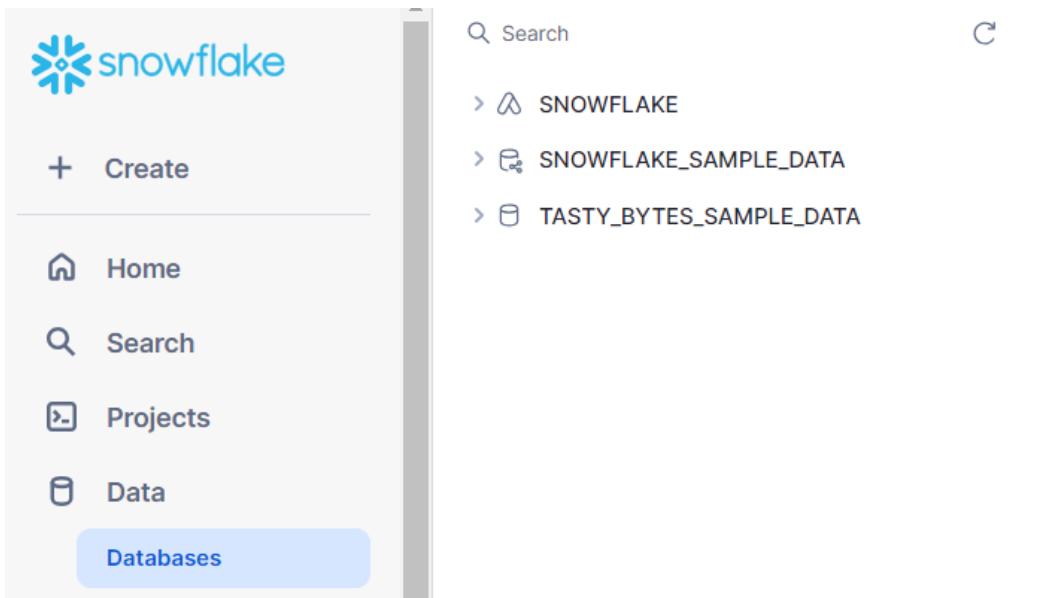
1.1.13 That displays the preview of the data in the CALL_CENTER table

| | | |
|-----------------------|--------------------|---|
| 1: Sample queries ... | 2023-07-17 10:07am | + |
|-----------------------|--------------------|---|

Data Preview

| | CC_CALL_CENTER_SK | CC_CALL_CENTER_ID | CC_REC_START_DATE | CC_REC_END_DATE | CC_CLOSED_DATE_SK | C |
|---|-------------------|-------------------|-------------------|-----------------|-------------------|---|
| 1 | 1 | AAAAAAAABAAAAAAA | 1998-01-01 | null | null | |
| 2 | 2 | AAAAAAAACAAAAAAA | 1998-01-01 | 2000-12-31 | null | |
| 3 | 3 | AAAAAAAACAAAAAAA | 2001-01-01 | null | null | |
| 4 | 4 | AAAAAAAEEAAAAAAA | 1998-01-01 | 2000-01-01 | null | |
| 5 | 5 | AAAAAAAEEAAAAAAA | 2000-01-02 | 2001-12-31 | null | |
| 6 | 6 | AAAAAAAEEAAAAAAA | 2002-01-01 | null | null | |
| 7 | 7 | AAAAAAAHHAAAAAAA | 1998-01-01 | null | null | |
| 8 | 8 | AAAAAAIAAAAAAAA | 1998-01-01 | 2000-12-31 | null | |

1.1.14 Next we'll select Data -> Databases in the navigation. This displays the top-level details for the databases in our Trial Account.



Search

- SNOWFLAKE
- SNOWFLAKE_SAMPLE_DATA
- TASTY_BYTES_SAMPLE_DATA

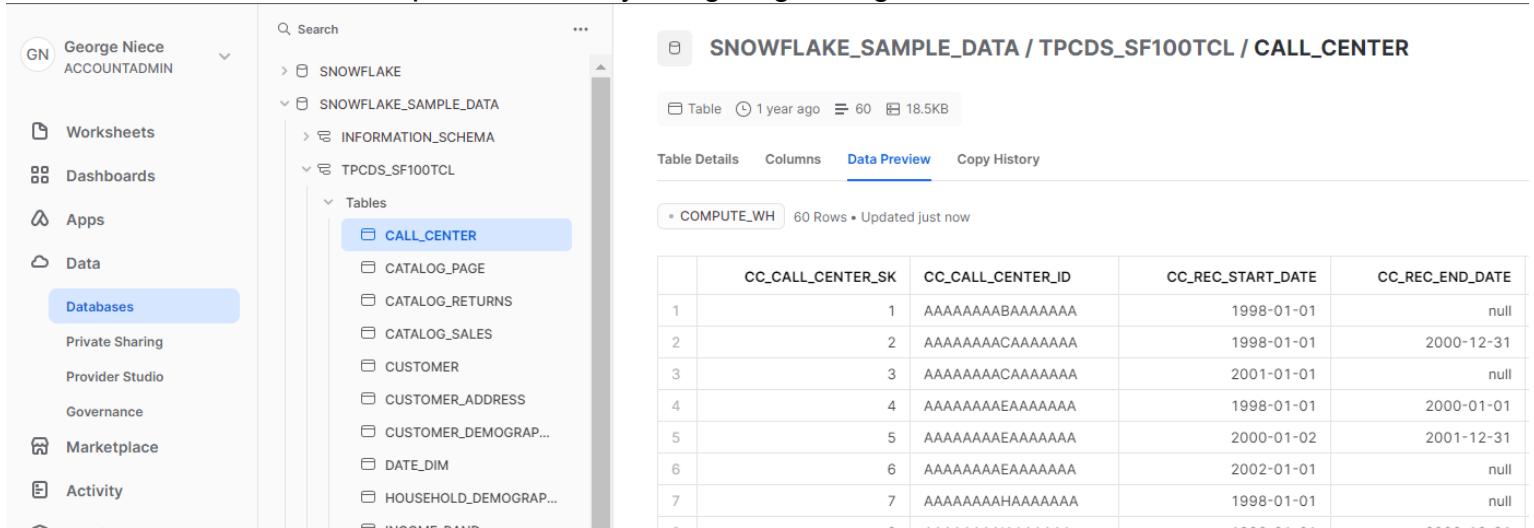
Databases

3 Databases

NAME ↑

| SNOWFLAKE |
|-------------------------|
| SNOWFLAKE_SAMPLE_DATA |
| TASTY_BYTES_SAMPLE_DATA |

1.1.15 You can also preview data by navigating through the search for data.



George Niece ACCOUNTADMIN

Worksheets

Dashboards

Apps

Data

Databases

SNOWFLAKE_SAMPLE_DATA / TPCDS_SF100TCL / CALL_CENTER

Table Details Columns Data Preview Copy History

COMPUTE_WH 60 Rows • Updated just now

| | CC_CALL_CENTER_SK | CC_CALL_CENTER_ID | CC_REC_START_DATE | CC_REC_END_DATE |
|---|-------------------|-------------------|-------------------|-----------------|
| 1 | | 1 | 1998-01-01 | null |
| 2 | | 2 | 1998-01-01 | 2000-12-31 |
| 3 | | 3 | 2001-01-01 | null |
| 4 | | 4 | 1998-01-01 | 2000-01-01 |
| 5 | | 5 | 2000-01-02 | 2001-12-31 |
| 6 | | 6 | 2002-01-01 | null |
| 7 | | 7 | 1998-01-01 | null |

1.1.16 Select the SNOWFLAKE_SAMPLE_DATA under the Database column. Then select a Table column header and choose Sort by Ascending



SNOWFLAKE_SAMPLE_DATA / TPCDS_SF100TCL / CALL_CENTER

Table 1 year ago 60 18.5KB

Table Details Columns Data Preview Copy History

• COMPUTE_WH 60 Rows • Updated 1 minute ago



| | CC_CALL_CENTER_SK | CC_CALL_CENTER_ID | CC_REC_START_DATE | CC_REC_END_DATE |
|---|-------------------|-------------------|-------------------|-----------------|
| 1 | | 1 | AAAAAAAAABAAAAAAA | 1998-01-01 |
| 2 | | 2 | AAAAAAAAACAAAAAAA | 1998-01-01 |

We may have noticed that we have two CALL_CENTER tables, in our the SNOWFLAKE_SAMPLE_DATA, and looking further that they're in different schemas (although the names are very similar). In our worksheet we've been using the TPCDS_SF100TCL schema

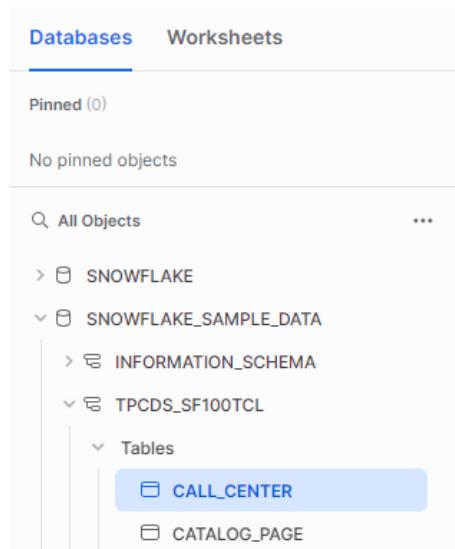
1.1.17 Notice that we have only 60 Rows in our CALL_CENTER table, but there are tables with significantly more like CATALOG_SALES with more than 14 billion rows. You'll also notice that we have those noted in G for Billions.



Query execution

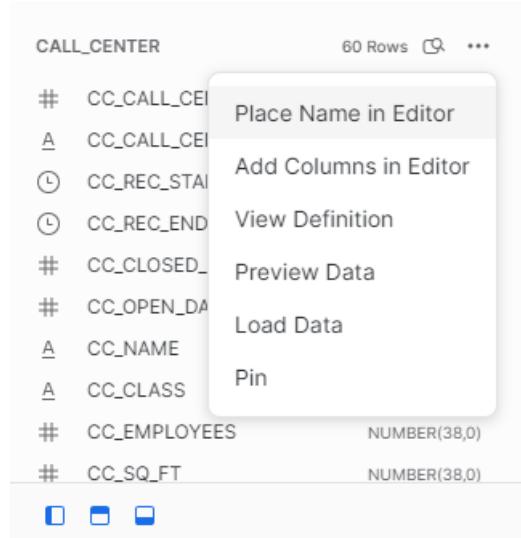
Queries will fail when there are syntax issues. That's a good thing, since when we break things we learn from that experience. When that occurs don't hesitate to correct and try again or ask for help.

1.1.18 Select the CALL_CENTER Table Name for the TPCDS_SF100TCL schema. This will show us the details of that table as shown.



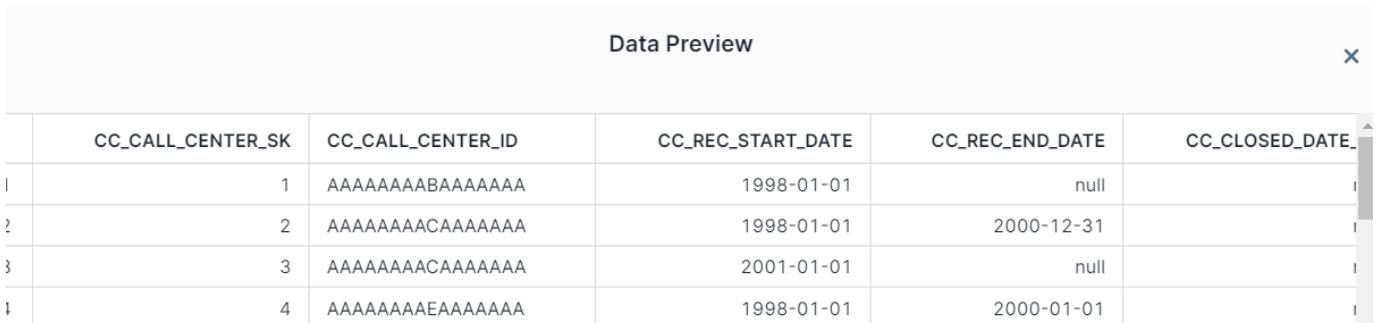
The screenshot shows the Snowflake interface for navigating databases and worksheets. The 'Databases' tab is selected, showing a tree structure of schemas and tables. The 'TPCDS_SF100TCL' schema is expanded, revealing the 'Tables' node, which contains the 'CALL_CENTER' table, which is highlighted with a blue box. Other tables like 'CATALOG_PAGE' are also listed under 'Tables'.

1.1.19 Navigate back to the Worksheets view. From there, select the ellipses for CALL_CENTER and choose **View Definition**. Note that we're presented with Nan error screen. This is because the SNOWFLAKE_SAMPLE_DATA is shared with our Trial Account.



| CALL_CENTER | | 60 Rows | ... |
|-------------|--------------|--------------|-----|
| # | CC_CALL_CEN | | |
| A | CC_CALL_CEN | | |
| (L) | CC_REC_STA | | |
| (L) | CC_REC_END | | |
| # | CC_CLOSED_ | | |
| # | CC_OPEN_DA | | |
| A | CC_NAME | | |
| A | CC_CLASS | | |
| # | CC_EMPLOYEES | NUMBER(38,0) | |
| # | CC_SQ_FT | NUMBER(38,0) | |

1.1.20 Next we'll select **Preview Data** from the ellipses menu for the CALL_CENTER table. Notice that there is a slider for Data Details in the Data Preview pane. Change the slide to **Data** from Details. Note the preview of the data in the CALL_CENTER table.

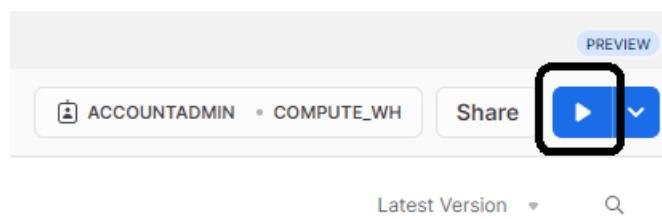


| | CC_CALL_CENTER_SK | CC_CALL_CENTER_ID | CC_REC_START_DATE | CC_REC_END_DATE | CC_CLOSED_DATE |
|---|-------------------|-------------------|-------------------|-----------------|----------------|
| 1 | 1 | AAAAAAAAABAAAAAAA | 1998-01-01 | null | |
| 2 | 2 | AAAAAAAAACAAAAAAA | 1998-01-01 | 2000-12-31 | |
| 3 | 3 | AAAAAAAAACAAAAAAA | 2001-01-01 | null | |
| 4 | 4 | AAAAAAAAEAAAAAAA | 1998-01-01 | 2000-01-01 | |

1.1.21 Now enter the following fully qualified SQL statement

```
select cc_name,cc_manager from  
"SNOWFLAKE_SAMPLE_DATA"."TPCDS_SF100TCL"."CALL_CENTER";
```

1.1.22 Select/highlight the SQL and choose the Blue Triangle in the upper right hand corner to run the query.





Query execution

Since this is an experimental environment we normally suggest you run each query individually.

1.1.23 This shows us the output data from our query in the details pane.

```
SNOWFLAKE_SAMPLE_DATA.TPCDS_SF100TCL ▾ Settings ▾
```

```
1 use database SNOWFLAKE_SAMPLE_DATA;
2
3 use schema tpcds_sf100tcl;
4
5 select cc_name, cc_manager from call_center;
6
```

↳ Results Chart Q W Y

| | CC_NAME | CC_MANAGER |
|---|-------------------|----------------|
| 1 | NY Metro | Bob Belcher |
| 2 | Mid Atlantic | Felipe Perkins |
| 3 | Mid Atlantic | Mark Hightower |
| 4 | North Midwest | Larry Mccray |
| 5 | North Midwest | Larry Mccray |
| 6 | North Midwest | Larry Mccray |
| 7 | Pacific Northwest | Alden Snyder |
| 8 | California | Wayne Ray |
| 9 | California | Wayne Ray |

Query Details

Query duration

Rows

Query ID [01adbe83-0001-3413](#)

CC_NAME

North Midwest

Hawaii/Alaska

North Midwest_1

+ 27 more

1.1.24 Now we'll run another query from a different table to make use of the limit, which returns only the number of rows that we specify.

```
Select * from
snowflake_sample_data.tpcds_sf100tcl.customer_demographics limit 10
```

1.1.25 Notice that the returned details only show 10 of the nearly 2M rows in the CUSTOMER_DEMOGRAPHICS table.

| Row | CD_DEMO_SK | CD_GENDER | CD_MARITAL_STATUS | CD_EDUCATION_STA | CD_PURCHASE_ESTIM |
|-----|------------|-----------|-------------------|------------------|-------------------|
| 1 | 1425409 | M | U | Unknown | 1500 |
| 2 | 1425410 | F | U | Unknown | 1500 |
| 3 | 1425411 | M | M | Primary | 2000 |
| 4 | 1425412 | F | M | Primary | 2000 |
| 5 | 1425413 | M | S | Primary | 2000 |
| 6 | 1425414 | F | S | Primary | 2000 |

1.1.26 Now we'll use another command to show us a view of the table description, now as the Describe command or Desc for short. Enter the following SQL statement, highlight it and choose Run

```
DESC TABLE
  "SNOWFLAKE_SAMPLE_DATA"."TPCDS_SF100TCL"."CALL_CENTER";
```

1.1.27 Notice this shows us yet another view of the CALL_CENTER table details.

| | name | type | kind | null? | default | primary key | unique key |
|---|-------------------|--------------|--------|-------|---------|-------------|------------|
| 1 | CC_CALL_CENTER_SK | NUMBER(38,0) | COLUMN | N | null | Y | N |
| 2 | CC_CALL_CENTER_ID | VARCHAR(16) | COLUMN | N | null | N | N |
| 3 | CC_REC_START_DATE | DATE | COLUMN | Y | null | N | N |
| 4 | CC_REC_END_DATE | DATE | COLUMN | Y | null | N | N |
| 5 | CC_CLOSED_DATE_SK | NUMBER(38,0) | COLUMN | Y | null | N | N |
| 6 | CC_OPEN_DATE_SK | NUMBER(38,0) | COLUMN | Y | null | N | N |

1.1.28 Next we'll select the Activity -> Query History navigation. Notice that the queries we've been running will show here, including both successful and unsuccessful queries.

Query History

Status All ▾ User GEONIECE ▾

| SQL TEXT |
|---|
| 1 DESC TABLE "SNOWFLAKE_SAMPLE_DATA"."TPCDS_SF100TCL"."CALL_CENTER"; |
| 2 select * from IDENTIFIER('SNOWFLAKE_SAMPLE_DATA').TPCDS_SF100TCL.CALL_CENTER; |
| 3 SELECT system\$GET_NPS_FEEDBACK_TIMESTAMP(); |
| 4 select cc_name, cc_manager from call_center; |
| 5 use schema tpcds_sf100tcl; |

1.1.29 Choosing the SQL Text value from one of the entries shows us the SQL statement or command that was run.

< **Query - 01adc168-0001-347c-0000-0004469ea061**

COMPUTE_WH SNOWFLAKECOURSEUPDATE

Query Details Query Profile

Details

| | |
|---------------------|---|
| Status | Duration |
| Success |  1.8s |
| Start Time | Query ID |
| 7/20/2023, 10:36 AM | 01adc168-0001-347c-0000-0004469ea061 |

SQL Text

```
with active_contracts as (
  select distinct contract_number
  from SNOWFLAKE.ORGANIZATION_USAGE.CONTRACT_ITEMS
  where (EXPIRATION_DATE is null or EXPIRATION_DATE >= current_date())
  and START_DATE <= current_date())
```

1.1.30 Hovering the Query Duration value for the entry shows us the details of the statement execution as noted below.

Query - 01adc168-0001-347c-0000-0004469ea061

COMPUTE_WH SNOWFLAKECOURSEUPDATE

Query Details Query Profile

Details

Status

Success

Start Time

7/20/2023, 10:36 AM

End Time

7/20/2023, 10:36 AM

Query Duration

Compilation 488ms

Queued provisioning 18ms

Execution 1.3s

Total 1.8s

Duration

1.8s

Query ID

01adc168-0001-347c-0000-0004469ea061

Query Tag

1.1.31 Note that if we have a significant number of queries we may want to add a filter condition. Note that only select statements matching our entered filter are visible in our history, when we enter a filter condition.

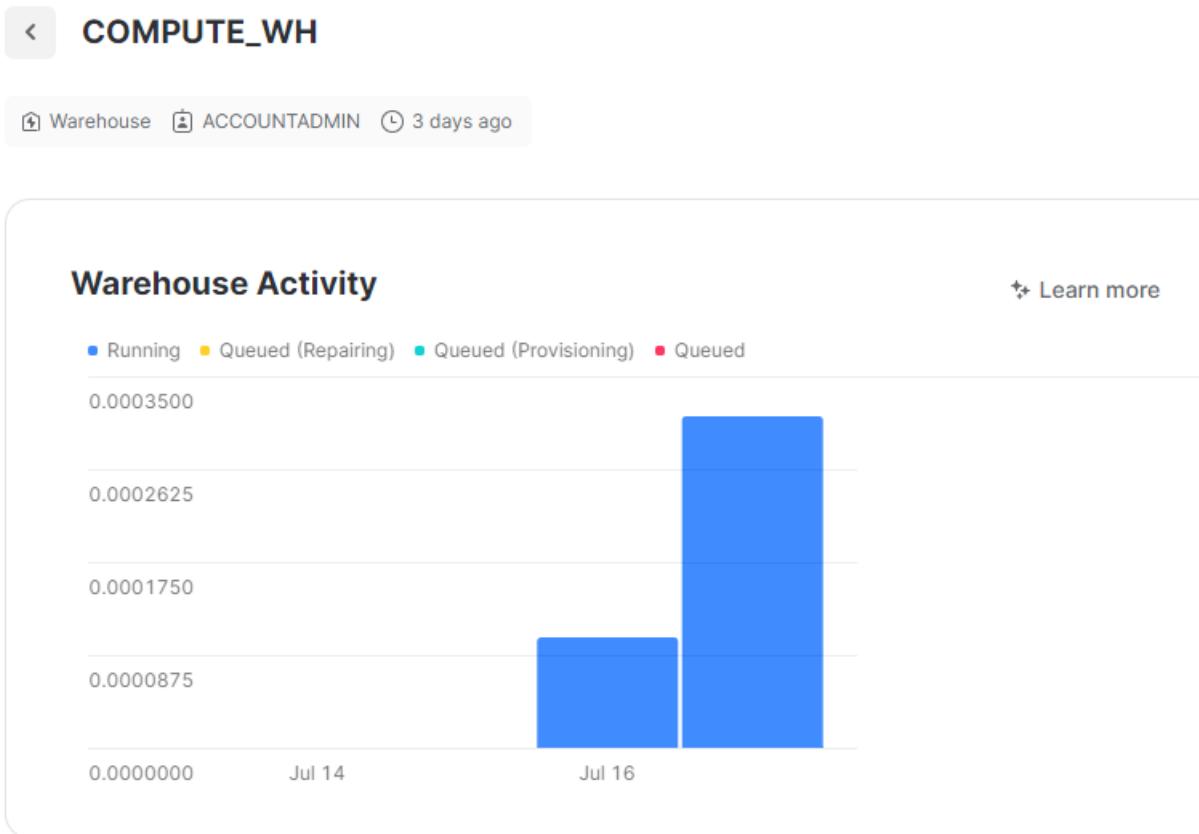
1.1.32 Select the Admin -> Warehouses view from the navigation.

Warehouses

| NAME | SIZE | STATUS |
|------------|------|---------|
| COMPUTE_WH | XS | Started |

Search Name Type All Size All Status All 1 Warehouse

1.1.33 Choose the COMPUTE_WH under Warehouse Name. Note that this will show usage graph details for our warehouse. We won't see usage initially, but by the last days of our course we'll see that populated with information from our time in the trial accounts we're using for the course.



1.1.37 Lastly, we'll download the **foundation_experiment_scripts.sql** from the course GitHub experiments folder to your local machine. This file contains pre-written SQL commands and we will use this file later in our Snowflake Foundation experiment group.