

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 exercises 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 exercises 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>

- Choose the following
 - Snowflake edition – Enterprise
 - Snowflake cloud provider – AWS
 - Region – Northern Virginia – US EAST

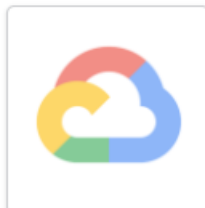
Choose your Snowflake edition*

- ☐ **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*



Amazon Web
Services



Google Cloud
Platform



Microsoft Azure

US East (Northern Virginia) ▼

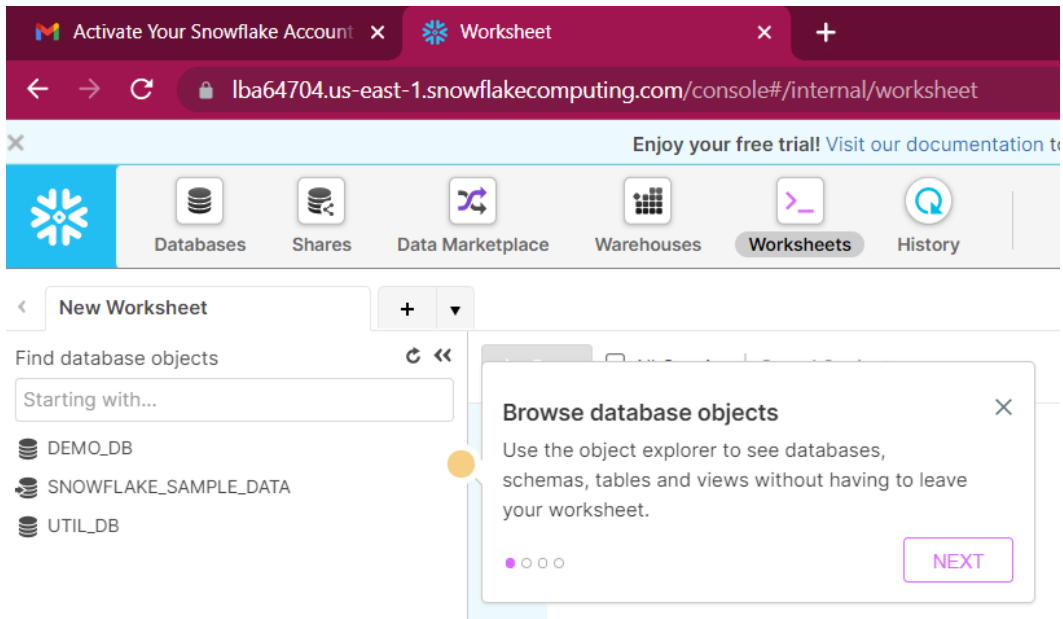
- ☒ Check here to indicate that you have read and agree to the terms of the [Snowflake Self Service On Demand Terms](#).

GET STARTED

- 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

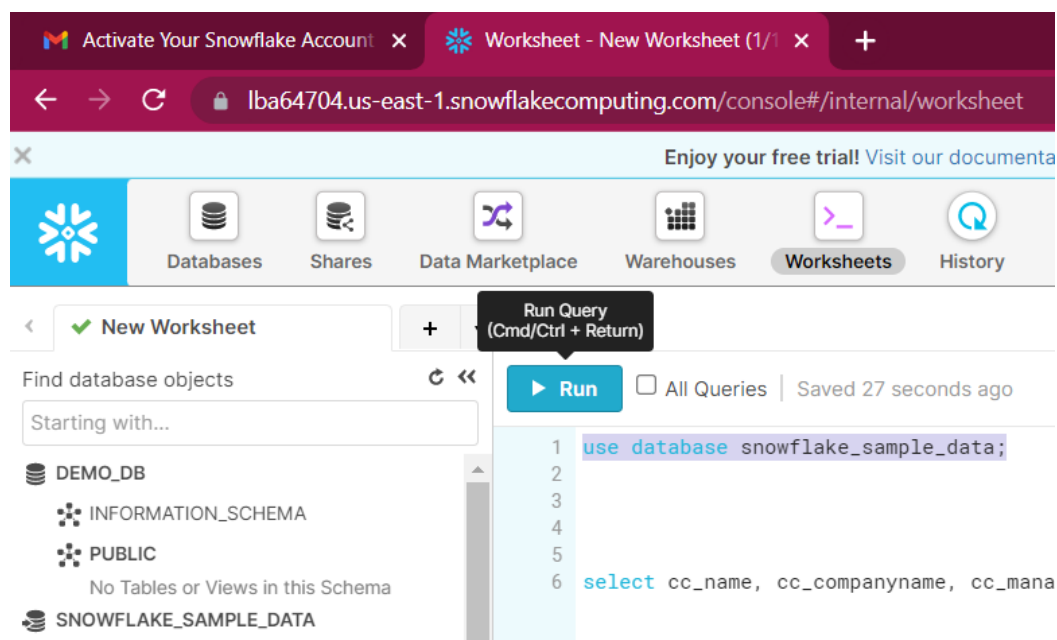
URL you can always login through **<https://app.snowflake.com/>**

- 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.
- 1.1.4 We'll select Worksheets from the navigation to do a little initial exploration

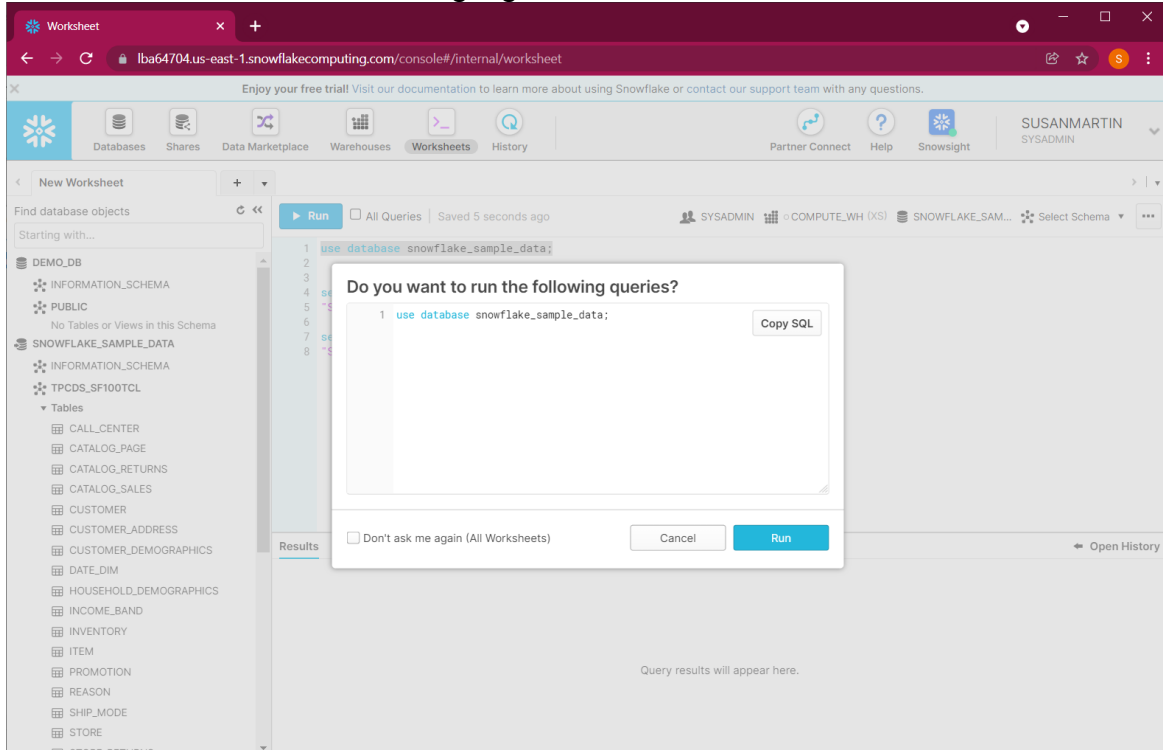


- 1.1.5 To use a database, we'll enter the SQL command

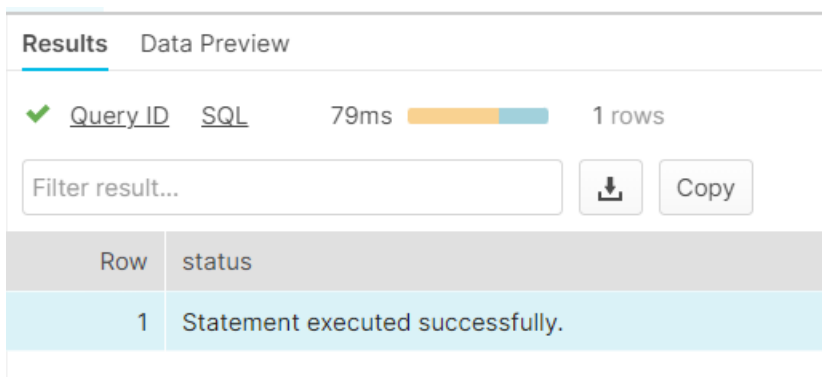
- `use database snowflake_sample_data;`



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



1.1.7 Selecting **Run** again in the dialog presented allows us to execute the SQL command. We could also have any further command automatically executed from the worksheet on Run without the additional dialog by check the **Don't ask me again** box.



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

```
use schema TPCDS_SF100TCL;
```

1.1.9 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;
```

1.1.10 Highlighting that SQL and running it shows us the results

▶ Run

☐ All Queries | Saved 19 seconds ago

SYSADMIN COMPUTE_WH (XS)

1 use database snowflake_sample_data;

2 use schema TPCDS_SF100TCL;

3

4 select cc_name,cc_manager from call_center;

5

Results

Data Preview

✔ Query_ID SQL 889ms 60 rows

Filter result...

Copy

Row	CC_NAME	CC_MANAGER
1	NY Metro	Bob Belcher
2	Mid Atlantic	Felipe Perkins
3	Mid Atlantic	Mark Hightower

1.1.11 Next, we'll expand the Tables under the SNOWFLAKE_SAMPLE_DATA database and TPCDS_SF100TCL schema. Select the ellipses ... next to the CALL_CENTER table name and choose **Preview Data**

Databases

Shares

Data Marketplace

Warehouses

Worksheets

< ✔ New Worksheet

+ ▼

Find database objects

Starting with...

DEMO_DB

- INFORMATION_SCHEMA
- PUBLIC
 - No Tables or Views in this Schema

SNOWFLAKE_SAMPLE_DATA

- INFORMATION_SCHEMA
- TPCDS_SF100TCL
 - ▼ Tables
 - CALL_CENTER ...

Table: CALL_CENTER

Created on: 8/19/2019, 11:57:04 PM

Rows: 60

Size: 11.5KB

Cluster by: LINEAR(CC_CALL_CENTER_SK)
 - CATALOG ...
 - CATALOG ...
 - CATALOG ...
 - CUSTOM ...

▶ Run

☐ All Queries | Saved 0 se

1

2

3

Results

Data Preview

Table: SNOWFLAKE_SAMPLE_DATA.TPCE

2 P_PROMO_ID VARCHAR

3 P START DA... NUMBER

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

Results

Data Preview

Table: SNOWFLAKE_SAMPLE_DATA.TPCDS_SF100TCL.CALL_CENTER

Data

Details

Filter result...

Row	CC_CALL_CENT	CC_CALL_CENT	CC_REC_START_	CC_REC_END_D	CC_CLOSED_DA	CC_OPEN_DATE
1	1	AAAAAAAAB...	1998-01-01			2450952
2	2	AAAAAAA...	1998-01-01	2000-12-31		2450806
3	3	AAAAAAA...	2001-01-01			2450806

1.1.13 Next we'll select the Databases view icon in the top level navigation. This displays the top-level details for the three databases in our Trial Account.

Databases						
Create... Clone... Drop... Transfer Ownership						
Search Databases			3 databases			
Database	Origin	↓ Creation Time	Owner	Comment		
SNOWFLAKE_SAMPLE_DATA	SFC_SAMPLES.SA...	11/7/2021, 7:23 AM	ACCOUNTADMIN	TPC-H, OpenWe		
DEMO_DB		11/7/2021, 7:23 AM	SYSADMIN	demo database		
UTIL_DB		11/7/2021, 7:23 AM	SYSADMIN	utility database		

1.1.14 Select the SNOWFLAKE_SAMPLE_DATA under the Database column. Then select the Table column header Sort by Ascending

Databases					
Databases > SNOWFLAKE_SAMPLE_DATA					
Tables	Views	Schemas	Stages	File Formats	Sequences
Create... Create Like... Clone... Load Data... Drop...					
Table Name	Schema	Creation Time			
CALL_CENTER		11/11/2021, 3:17:32 ...			
CALL_CENTER		11/11/2021, 3:17:32 ...			
CATALOG_PAGE		11/11/2021, 3:17:32 ...			

1.1.15 We notice that we have two CALL_CENTER tables, 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.

Databases > SNOWFLAKE_SAMPLE_DATA

Tables Views Schemas Stages File Formats Sequences

+ Create... + Create Like... Clone... Load Data... Drop... Transfer Ownership

Table Name	Schema	Creation Time	On
CALL_CENTER	TPCDS_SF10TCL	11/11/2021, 3:17:32 ...	
CALL_CENTER	TPCDS_SF100TCL	11/11/2021, 3:17:32 ...	

1.1.16 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.

Databases > SNOWFLAKE_SAMPLE_DATA

Tables Views Schemas Stages File Formats Sequences Pipes

+ Create... + Create Like... Clone... Load Data... Drop... Transfer Ownership

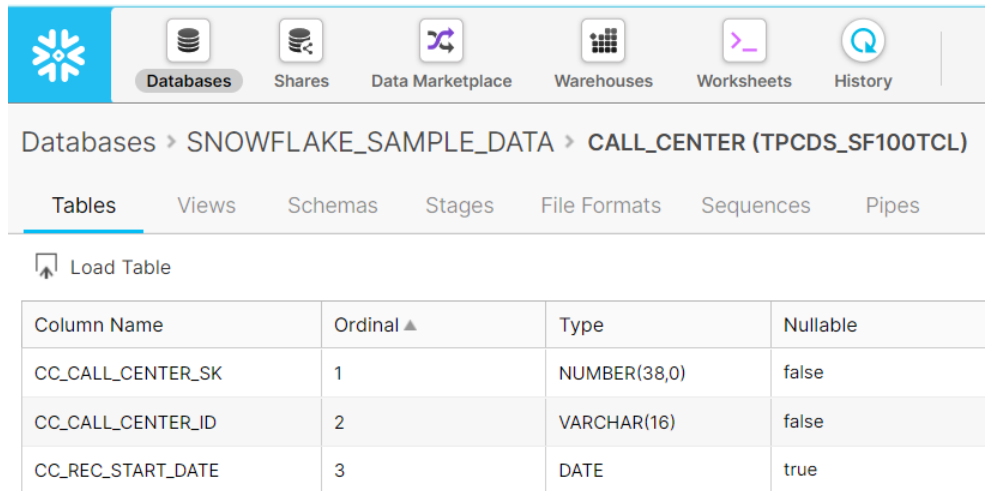
Table Name	Schema	Creation Time	Rows
CALL_CENTER	TPCDS_SF10TCL	11/11/2021, 3:17:32 ...	54
CALL_CENTER	TPCDS_SF100TCL	11/11/2021, 3:17:32 ...	60
CATALOG_PAGE	TPCDS_SF10TCL	11/11/2021, 3:17:32 ...	40K
CATALOG_PAGE	TPCDS_SF100TCL	11/11/2021, 3:17:32 ...	50K
CATALOG_RETURNS	TPCDS_SF10TCL	11/11/2021, 3:17:32 ...	1.4G
CATALOG_RETURNS	TPCDS_SF100TCL	11/11/2021, 3:17:32 ...	14.4G
CATALOG_SALES	TPCDS_SF10TCL	11/11/2021, 3:17:32 ...	14.4G



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.

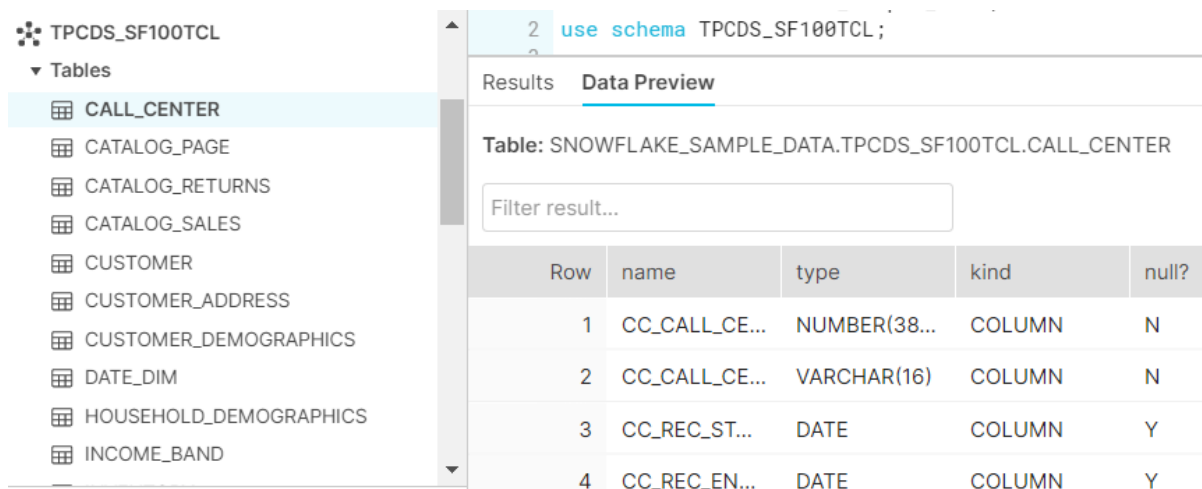
1.1.17 Select the CALL_CENTER Table Name for the TPCDS_SF100TCL schema. This will show us the details of that table as shown.



The screenshot shows the Snowflake web interface. The top navigation bar includes icons for Databases, Shares, Data Marketplace, Warehouses, Worksheets, and History. Below the navigation bar, the breadcrumb path is 'Databases > SNOWFLAKE_SAMPLE_DATA > CALL_CENTER (TPCD_SF100TCL)'. Underneath, there are tabs for Tables, Views, Schemas, Stages, File Formats, Sequences, and Pipes. The 'Tables' tab is selected. Below the tabs, there is a 'Load Table' button. The main content area displays a table with the following columns: Column Name, Ordinal, Type, and Nullable.

Column Name	Ordinal ▲	Type	Nullable
CC_CALL_CENTER_SK	1	NUMBER(38,0)	false
CC_CALL_CENTER_ID	2	VARCHAR(16)	false
CC_REC_START_DATE	3	DATE	true

1.1.18 Navigate back to the Worksheets view. From there, select the ellipses for CALL_CENTER and choose **View Details**. The following view is shown noting the column details for the table.



The screenshot shows the Snowflake web interface with the 'View Details' pane open for the CALL_CENTER table. The left sidebar shows the 'TPCDS_SF100TCL' schema with a list of tables. The 'CALL_CENTER' table is selected. The right pane shows the 'Data Preview' tab. At the top, there is a text input field with the SQL query 'use schema TPCDS_SF100TCL;'. Below this, the 'Results' section is titled 'Data Preview'. The table name is 'Table: SNOWFLAKE_SAMPLE_DATA.TPCDS_SF100TCL.CALL_CENTER'. There is a 'Filter result...' input field. The table data is displayed with columns: Row, name, type, kind, and null?.

Row	name	type	kind	null?
1	CC_CALL_CE...	NUMBER(38...	COLUMN	N
2	CC_CALL_CE...	VARCHAR(16)	COLUMN	N
3	CC_REC_ST...	DATE	COLUMN	Y
4	CC_REC_EN...	DATE	COLUMN	Y

1.1.19 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.

Results Data Preview					
Table: SNOWFLAKE_SAMPLE_DATA.TPCDS_SF100TCL.CALL_CENTER					
Filter result...					
Row	CC_CALL_CENT	CC_CALL_CENT	CC_REC_START	CC_REC_END_D	CC_CLOSED_DA
1	1	AAAAAAAAB...	1998-01-01		
2	2	AAAAAAA...	1998-01-01	2000-12-31	
3	3	AAAAAAA...	2001-01-01		
4	4	AAAAAAA...	1998-01-01	2000-01-01	
5	5	AAAAAAA...	2000-01-01	2001-12-31	

1.1.20 Now enter the following fully qualified SQL statement

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

The screenshot shows the Snowflake Data Studio interface. At the top, there's a navigation bar with icons for Databases, Shares, Data Marketplace, Warehouses, Worksheets (selected), and History. Below this, a 'New Worksheet' button is visible. On the left, a sidebar shows the database structure: SNOWFLAKE_SAMPLE_DATA, INFORMATION_SCHEMA, and TPCDS_SF100TCL. Under TPCDS_SF100TCL, the 'CALL_CENTER' table is selected. The main area displays the SQL query: `select cc_name,cc_manager from "SNOWFLAKE_SAMPLE_DATA"."TPCDS_SF100TCL"."CALL_CENTER";`. A 'Run' button is highlighted next to the query. Below the query, there's a 'Results' tab and a 'Data Preview' tab.

1.1.21 Select/highlight the SQL and choose Run.

Run All Queries Saved 6 minutes ago SYS

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

Do you want to run the following queries?

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

Copy SQL

☐ Don't ask me again (All Worksheets)

Cancel Run

Results Data Preview

Table: SNOWFLAKE_SAMPLE_DATA

Filter result...

Row	name	ty
1	P_PROMO_SK	N
2	P_PROMO_ID	V
3	P_START_DA...	N
4	P_END_DATE...	N
5	P_ITEM_SK	N



Don't ask me again

Since this is an experimental environment we normally suggest you don't check this box, but that's up to you.

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

```
6
7 select cc_name,cc_manager from
8 "SNOWFLAKE_SAMPLE_DATA"."TPCDS_SF100TCL"."CALL_CENTER"
9
```

Results Data Preview

✓ Query ID SQL 1.39s 60 rows

Filter result...

Download Copy

Row	CC_NAME	CC_MANAGER
1	NY Metro	Bob Belcher
2	Mid Atlantic	Felipe Perkins

1.1.23 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.

1.1.24 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_STATU	CD_EDUCATION_STA	CD_PURCHASE_ESTI	CD_PURCHASE_ESTI
1	1425409	M	U	Unknown	1500	1500
2	1425410	F	U	Unknown	1500	1500
3	1425411	M	M	Primary	2000	2000
4	1425412	F	M	Primary	2000	2000
5	1425413	M	S	Primary	2000	2000
6	1425414	F	S	Primary	2000	2000

1.1.25 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";

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

[Run](#)
☐ All Queries
 | Saved 5 seconds ago

 SYSADMIN
 o COMPUTE_WH

```

1 use database snowflake_sample_data;
2 use schema TPCDS_SF100TCL;
3 DESC TABLE "SNOWFLAKE_SAMPLE_DATA"."TPCDS_SF100TCL"."CALL_CENTER";

```

Results
[Data Preview](#)

✓ [Query ID](#) [SQL](#)
38ms
31 rows

[Download](#)
[Copy](#)

Row	name	type	kind	null?	default	primary
1	CC_CALL_CE...	NUMBER(38...	COLUMN	N	NULL	Y
2	CC_CALL_CE...	VARCHAR(16)	COLUMN	N	NULL	N

1.1.27 Next we'll select the History tab of from the top level navigation. Notice that the queries we've been running will show here, including both successful and unsuccessful queries.

[Databases](#)
[Shares](#)
[Data Marketplace](#)
[Warehouses](#)
[Worksheets](#)
[History](#)

History

[Hide Filters](#)
[View SQL](#)
[Abort...](#)

Display queries that meet all of the following criteria:

User is

 - +

☐ Include client-generated statements
☒ Include queries executed by user tasks

Status	Query ID	SQL Text	User	Warehouse	Clust...	Size	Session ID
✓	01a021ce-...	select * from "SNOWFLAKE_SAMPLE_DATA"."TP...	SUSANMAR...	COMPUTE...	1	X-Small	477999865...
✓	01a02176-...	DESC TABLE "SNOWFLAKE_SAMPLE_DATA"."TPC...	SUSANMAR...	COMPUTE...			477999865...
✓	01a02176-...	select cc_name, cc_manager from "SNOWFLAKE...	SUSANMAR...	COMPUTE...	1	X-Small	477999865...
✗	01a02175-...	select cc_name, cc_companyname, cc_manager ...	SUSANMAR...	COMPUTE...			477999865...
✗	01a02174-...	select cc_name, cc_companyname, cc_manager ...	SUSANMAR...	COMPUTE...			477999865...
✓	01a02171-...	use database snowflake_sample_data;	SUSANMAR...	COMPUTE...			477999865...
✗	01a02170-...	use database snowflake_sampledata;	SUSANMAR...	COMPUTE...			477999865...

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

History

Hide Filters View SQL Abort...

Display queries that meet all of the following criteria:

User is SUSANMARTIN

☐ Include client-generated statements

☒ Include queries executed by user tasks

Status	Query ID	SQL Text
✓	01a021ce-...	select * from "SNOWFLAKE_SAMPLE_DATA"."TPCDS_SF100TCL"."CUSTOMER_DEMOGRAPHIC
✓	01a02176-...	DESC TABLE "SNOWFLAKE_SAMPLE_DATA"."TPCDS_SF100TCL"."CALL_CENTER";
✓	01a02176-...	select cc_name, cc_manager from "SNOWFLAKE_SAMPLE_DATA"."TPCDS_SF100TCL"."CALL_CENTER";
✗	01a02175-...	select cc_name, cc_companyname, cc_manager from "SNOWFLAKE_SAMPLE_DATA"."TPCDS_SF100TCL"."CALL_CENTER";

SQL Text

```
1 select * from "SNOWFLAKE_SAMPLE_DATA"."TPCDS_SF100TCL"."CUSTOMER_DEMOGRAPHIC
```

Select SQL

1.1.29 Selecting the Query ID value for one of the entries shows us the details of the statement execution as noted below.

Databases Shares Data Marketplace Warehouses Worksheets History Partner Connect

History > 11:34:10 AM for 48ms

Details Profile

Status	Success
User	SUSANMARTIN
Warehouse	COMPUTE_WH
Start Time	11:34:10 AM
End Time	11:34:10 AM
Total Duration	48ms

SQL Text

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

Query Result

Results

row#	name	type	kind	null?
1	CC_CALL_CE...	NUMBER(38...	COLUMN	N

1.1.30 Note that if we have a significant number of queries we may want to add a filter condition, other than the default that is automatically added by User as shown. Add a 2nd query filter condition by clicking the + sign, choosing **SQL Text, Contains** and entering the value **select**. Notice that now only select statements are visible in our history.

History

Hide Filters

View SQL

Abort...

Display queries that meet all of the following criteria:

User

is

SUSANMARTIN

⊖

SQL Text

Contains

select

⊖

⊕

☐ Include client-generated statements
 ☒ Include queries executed by user tasks

Status	Query ID	SQL Text	User	Warehouse	Clust...	Size
✓	01a021ce-...	select * from "SNOWFLAKE_SAMPLE_DATA"."TP...	SUSANMAR...	COMPUTE_...	1	X-Small
✓	01a02176-...	select cc_name, cc_manager from "SNOWFLAKE...	SUSANMAR...	COMPUTE_...	1	X-Small
✗	01a02175-...	select cc_name, cc_companyname, cc_manager ...	SUSANMAR...	COMPUTE_...		
✗	01a02174-...	select cc_name, cc_companyname, cc_manager ...	SUSANMAR...	COMPUTE_...		
✗	01a02170-...	select cc_name, cc_companyname, cc_manager ...	SUSANMAR...	COMPUTE_...		

1.1.31 Select the Warehouses tab from the top level navigation.

Warehouses

←

→

↻

lba64704.us-east-1.snowflakecomputing.com/console#/warehouses

Enjoy your free trial! Visit our documentation to learn more about u

Warehouses

Databases

Shares

Data Marketplace

Worksheets

History

Warehouses

Manage your warehouses from this page. To operate on your data, you need to create one or

⊕ Create...

🔧 Configure...

⏸ Suspend...

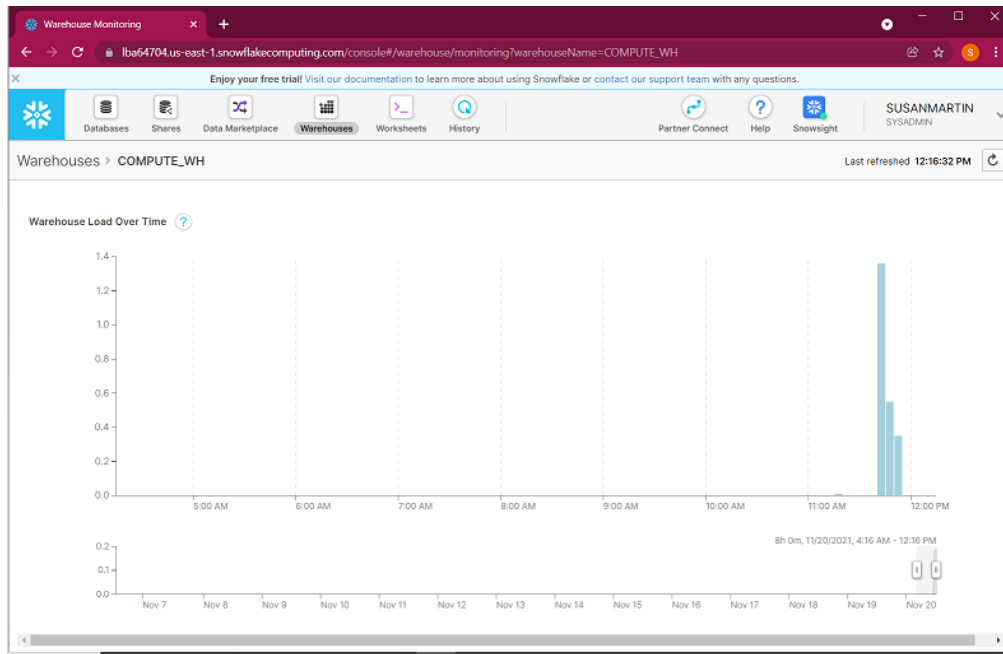
▶ Resume...

🗑 Drop...

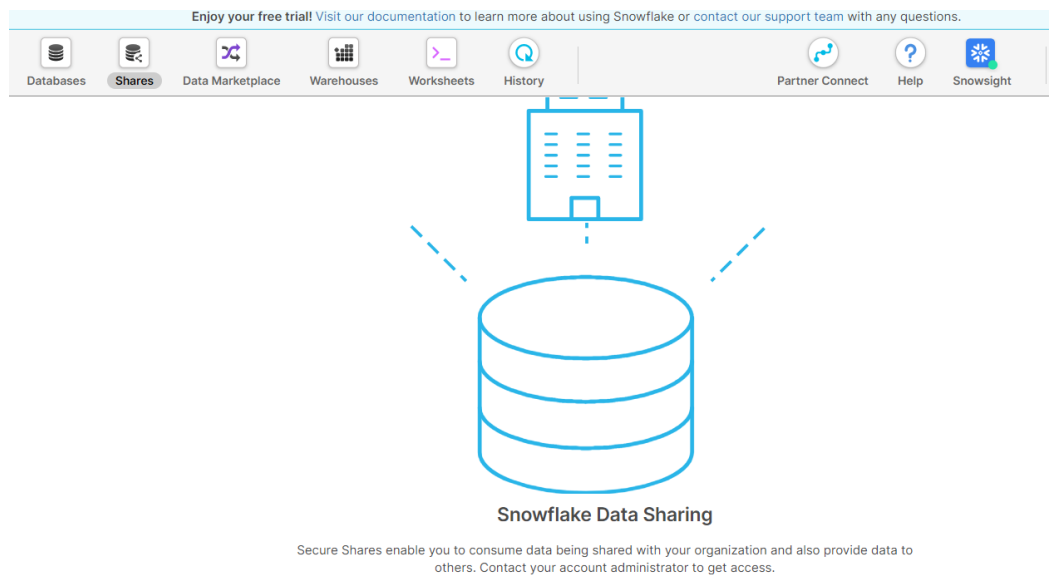
👤 Transfer Owner


Status	Warehouse Name	Size	Clusters	Scaling Poli...
Suspended	COMPUTE_WH	X-Small	min: 1, max: 1	Standard

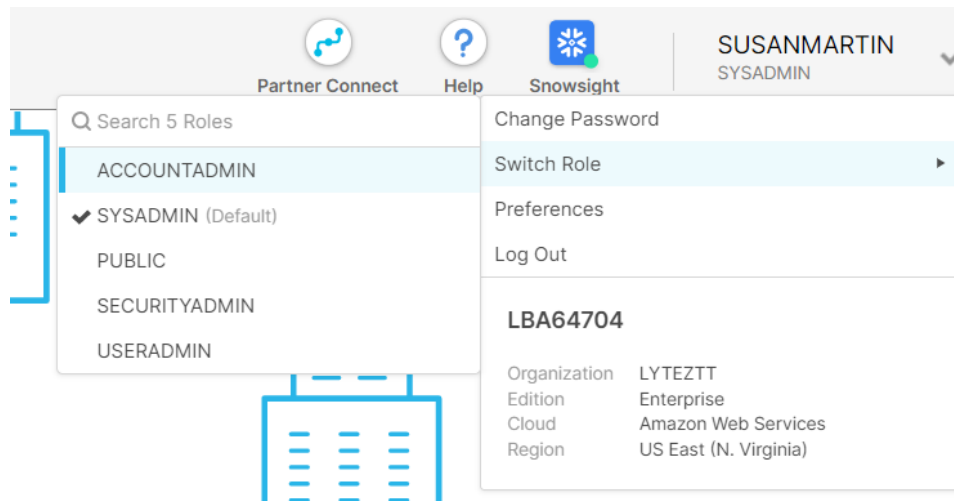
1.1.32 Choose the COMPUTE_WH under Warehouse Name. Note that this will show usage graph details for our warehouse.



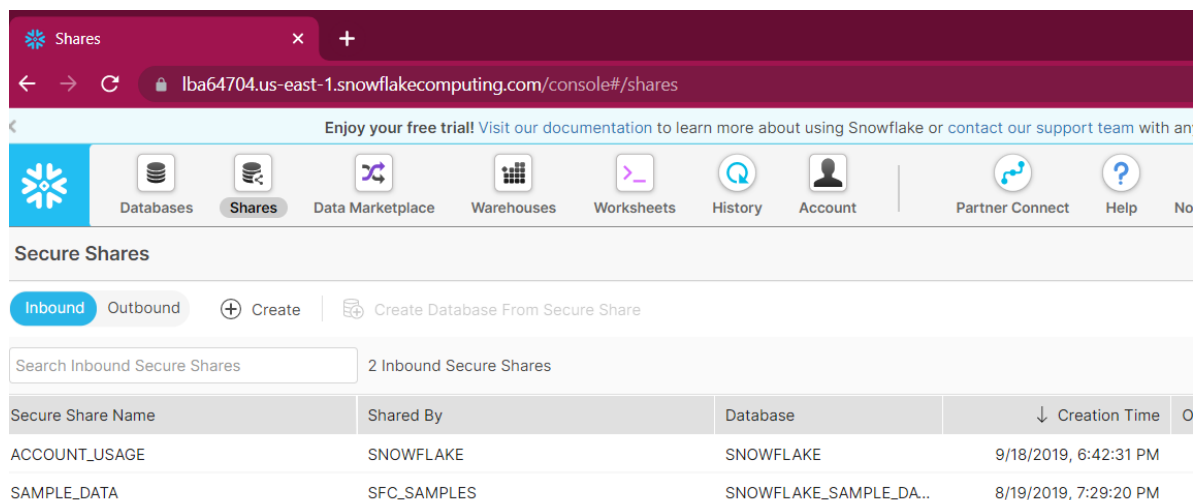
1.1.33 Next, select Shares from the top level navigation. Notice that we don't actually have rights to do that when we're the SYSADMIN role.



1.1.34 Select the  by the USERNAME, then choose **Switch Role** to **ACCOUNTADMIN**.



1.1.35 Notice that now we'll be able to view the Shares pane as shown.



1.1.36 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.