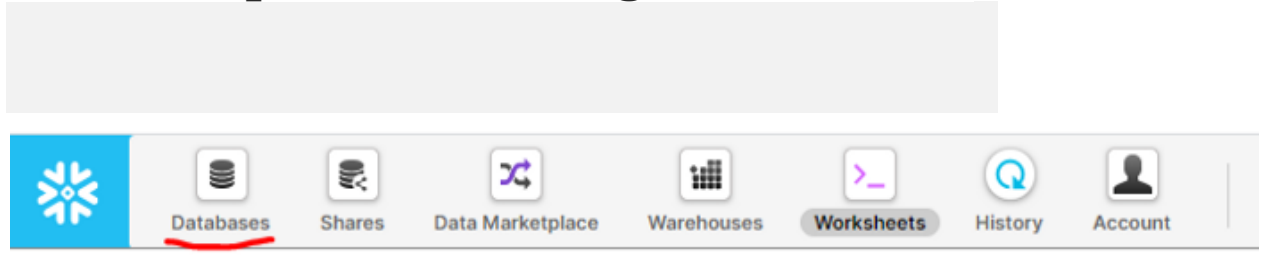


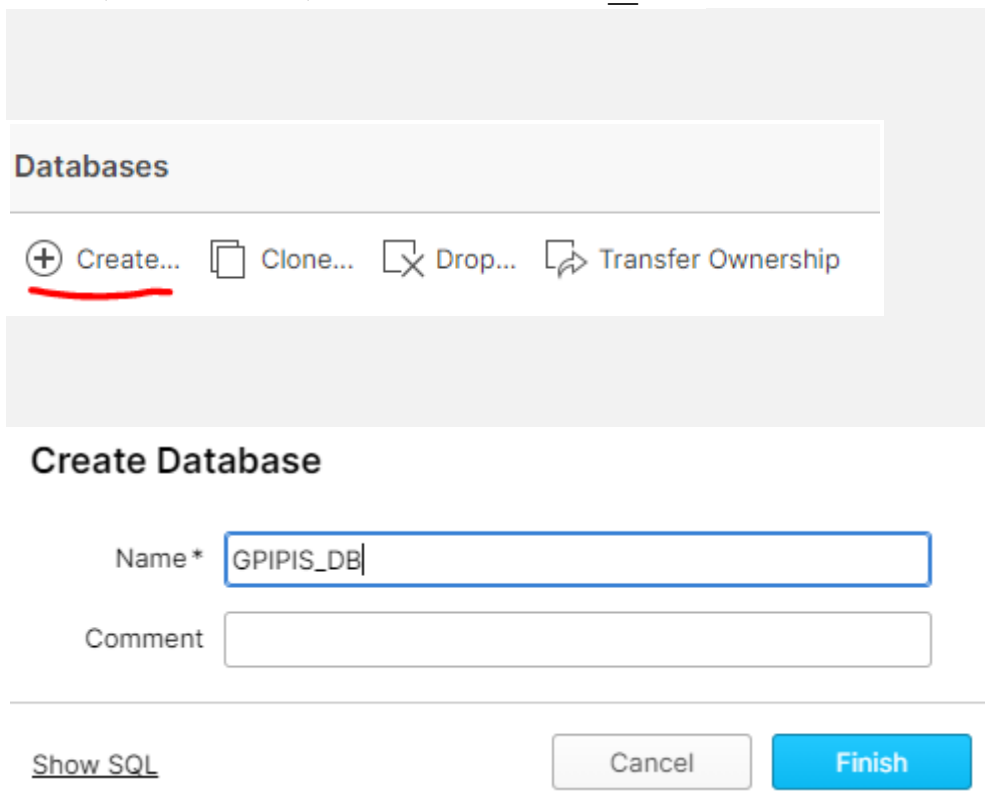
Snowflake Data Loading from S3

Create a Database

The first thing that we need to do is to create a database. This is easy. We login into the Snowflake platform and we go to Databases.



Then we click on Create. For this tutorial the database will be the GPIPIIS_DB

A screenshot of the 'Create Database' form in the Snowflake interface. The form is titled 'Databases' and includes a row of action buttons: '+ Create...' (underlined with a red line), 'Clone...', 'Drop...', and 'Transfer Ownership'. Below this, the 'Create Database' section contains a 'Name*' field with 'GPIPIIS_DB' entered, and an empty 'Comment' field. At the bottom, there are three buttons: 'Show SQL' (with a link underline), 'Cancel', and 'Finish'.

In the database, we can create a schema, where the default is the “public” and we can create a table or a schema by clicking on the Create button.

Let’s say that I want to create the MYIRISTABLE. In this tutorial, we will create the tables and the stages programmatically.

Create Table

Table Name *

MYIRISTABLE

Schema Name

PUBLIC

▼

Comment

Columns *


⊕ Add


✕ Remove

Name	Type ▲	Not Null	Default
sepal_length	NUMBER (precision, scale)	<input type="checkbox"/>	


Create a Warehouse

Similarly, we can create the warehouse by clicking on Warehouses and then on the Create button.







Databases




Shares




Data Marketplace




Warehouses



Worksheets









History



Account

Warehouses

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

 Create... Configure... Suspend... Resume... Drop... Transfer Ownership

We will create the smallest possible Warehouse called TEST_WH

Create Warehouse

Name * TEST_WH

Size X-Small (1 credit / hour)

[Learn more about virtual warehouse sizes here](#)

Maximum Clusters 2

Multi-cluster warehouses improve the query throughput for high concurrency workloads.

Minimum Clusters 1

The number of active clusters will vary between the specified minimum and maximum values, based on number of concurrent users/queries.

Scaling Policy Standard

The policy used to automatically start up and shut down clusters.

Auto Suspend 10 minutes

The maximum idle time before the warehouse will be automatically suspended.

☒ Auto Resume ?

Comment

[Show SQL](#)

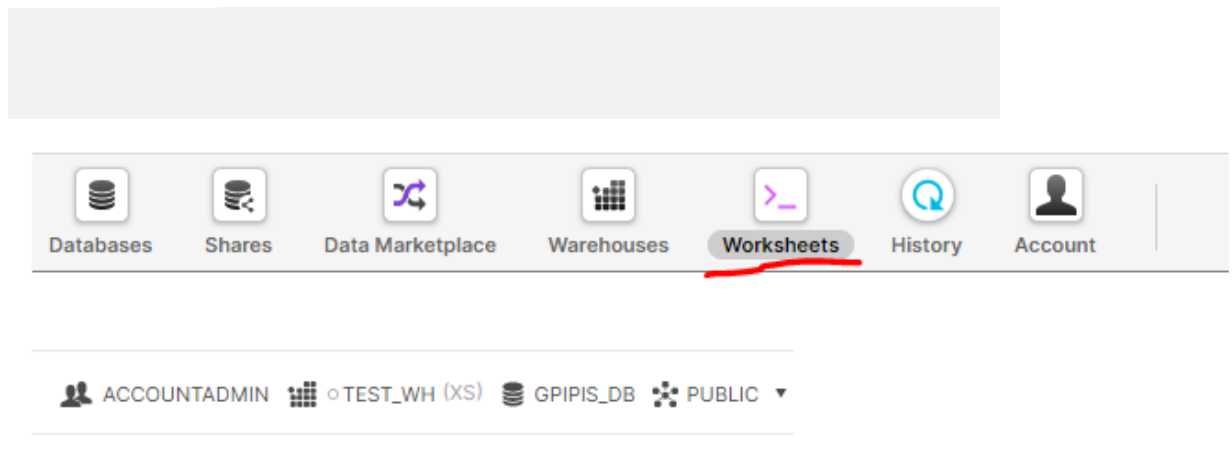
Cancel

Finish

By clicking Finish, the Warehouse is ready.

Load Data from S3

In order to load data from S3, we can go to Worksheets. Do not forget to choose your database.



Our goal is to load the iris.csv dataset that we have in an S3 bucket called gpipis-iris-dataset. The first thing that we need to do, is to specify the file format, wherein my case is CSV.

```
create or replace file format mycsvformat
```

```
type = 'CSV'
```

```
field_delimiter = ','
```

```
skip_header = 1;
```

Then, we need to create the table, called myiristable.

```
create or replace table myiristable (
```

```
  sepal_length numeric,
```

```
  sepal_width numeric,
```

```
  petal_length numeric,
```

```
  petal_width numeric,
```

```
  variety string);
```

The next step is to create an S3 stage. You have to put your credentials.

```
create or replace stage my_s3_stage  
url='s3://gpipeis-iris-dataset/'
```

```
  credentials=(aws_key_id='xxx'  
aws_secret_key='xxx')
```

```
file_format = mycsvformat;
```

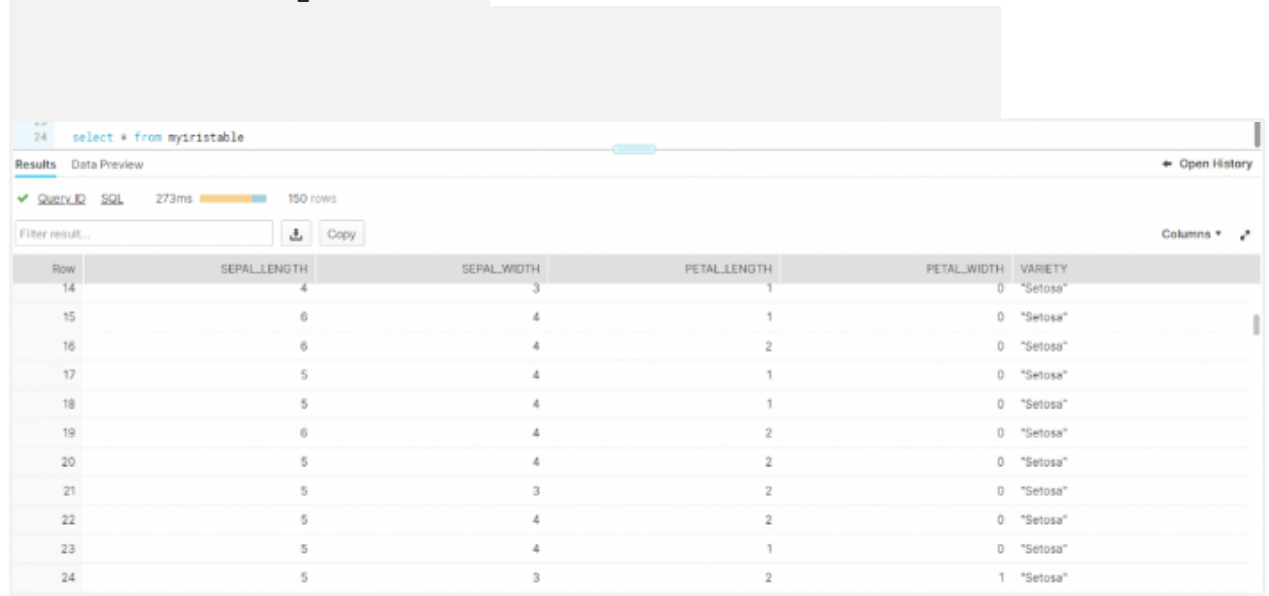
Finally, we can copy the data to our table.

```
copy into myiristable
```

```
from @my_s3_stage;
```

and we can confirm that the data are in Snowflake!

```
select * from myiristable
```



The screenshot displays the Snowflake query results interface. At the top, the query `select * from myiristable` is shown. Below the query, the results are presented in a table format. The table has 6 columns: Row, SEPAL_LENGTH, SEPAL_WIDTH, PETAL_LENGTH, PETAL_WIDTH, and VARIETY. The first 10 rows are visible, showing data for the 'Setosa' variety. The interface also includes a 'Filter result...' input, a 'Copy' button, and a 'Columns' dropdown menu.

Row	SEPAL_LENGTH	SEPAL_WIDTH	PETAL_LENGTH	PETAL_WIDTH	VARIETY
14	4	3	1	0	"Setosa"
15	6	4	1	0	"Setosa"
16	6	4	2	0	"Setosa"
17	5	4	1	0	"Setosa"
18	5	4	1	0	"Setosa"
19	6	4	2	0	"Setosa"
20	5	4	2	0	"Setosa"
21	5	3	2	0	"Setosa"
22	5	4	2	0	"Setosa"
23	5	4	1	0	"Setosa"
24	5	3	2	1	"Setosa"

