Zero to Snowflake in 90 minutes

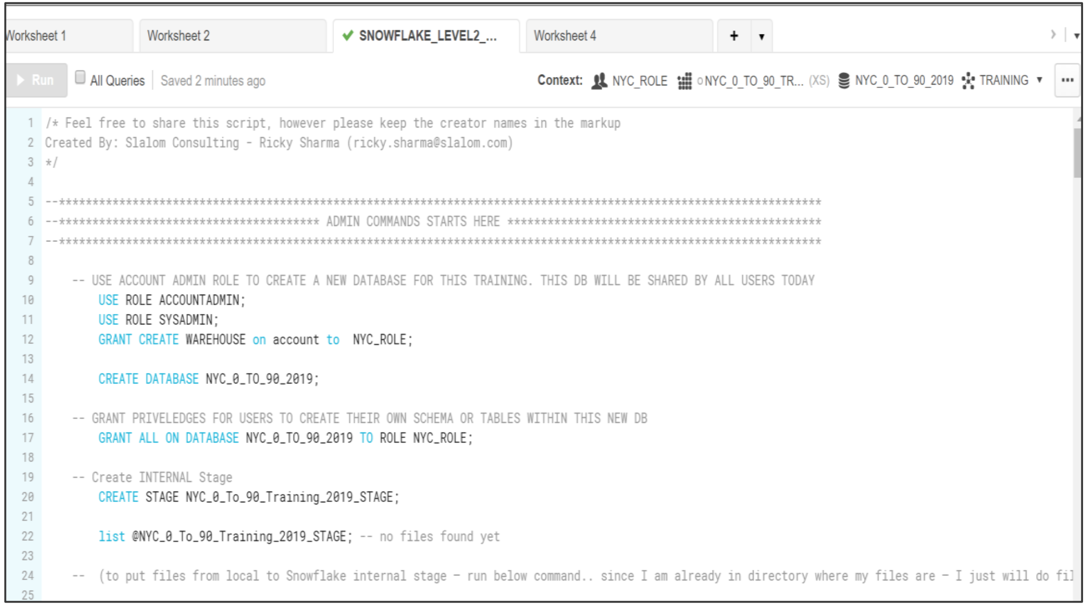
Exercise1: Let’s get the scripts

Get the scrips for the labs

- Click below link –

https://github.com/rsharma84/snowflakedemo

Copy or Load the Script from the file provided and Load/paste it into your Snowflake Web-UI Worksheet

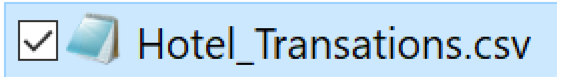


```
/* Feel free to share this script, however please keep the creator names in the markup
Created By: Slalom Consulting - Ricky Sharma (ricky.sharma@slalom.com)
*/

--******************************************************************************************************************
--*************************************** ADMIN COMMANDS STARTS HERE ***********************************************
--******************************************************************************************************************

    -- USE ACCOUNT ADMIN ROLE TO CREATE A NEW DATABASE FOR THIS TRAINING. THIS DB WILL BE SHARED BY ALL USERS TODAY
        USE ROLE ACCOUNTADMIN;
        USE ROLE SYSADMIN;
        GRANT CREATE WAREHOUSE on account to  NYC_ROLE;

        CREATE DATABASE NYC_0_TO_90_2019;

    -- GRANT PRIVELEDGES FOR USERS TO CREATE THEIR OWN SCHEMA OR TABLES WITHIN THIS NEW DB
        GRANT ALL ON DATABASE NYC_0_TO_90_2019 TO ROLE NYC_ROLE;

    -- Create INTERNAL Stage
        CREATE STAGE NYC_0_To_90_Training_2019_STAGE;

        list @NYC_0_To_90_Training_2019_STAGE; -- no files found yet

    --  (to put files from local to Snowflake internal stage - run below command.. since I am already in directory where my files are - I just will do fil

```

NYC_0-to-90-2019-Scripts.txt

Exercise1: Load Hotels Transactions Batch file to Snowflake

File Name:

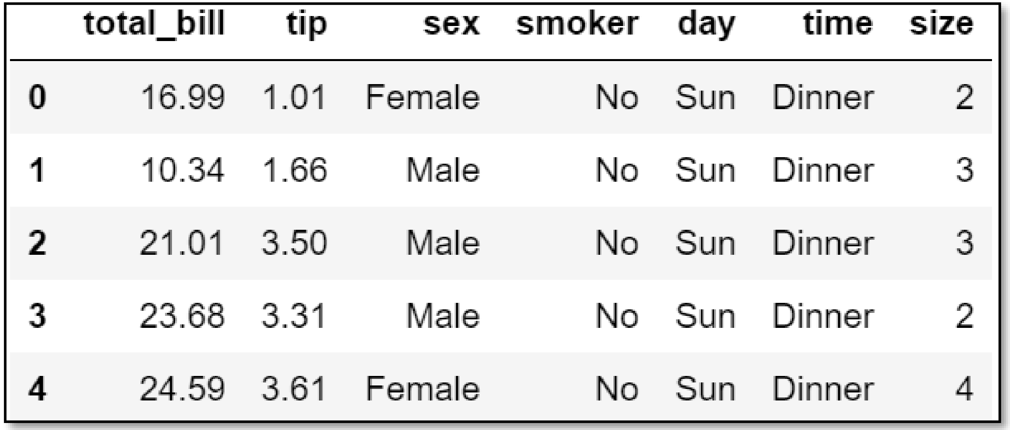


☑ Hotel_Transations.csv

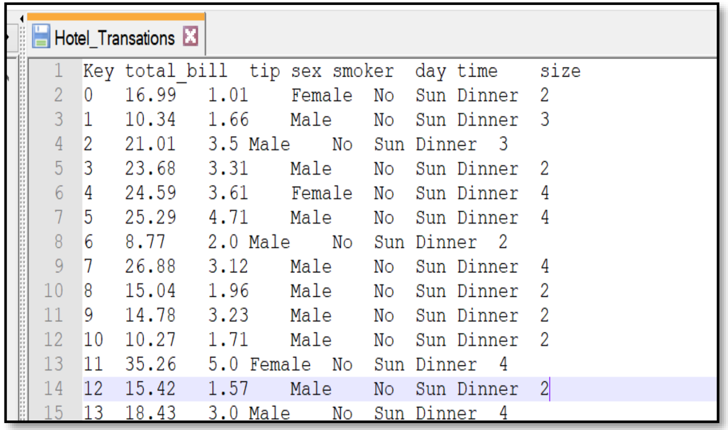
File Location:

https://github.com/rsharma84/snowflakedemo - Hotel_Transations.csv

Preview:



	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2
1	10.34	1.66	Male	No	Sun	Dinner	3
2	21.01	3.50	Male	No	Sun	Dinner	3
3	23.68	3.31	Male	No	Sun	Dinner	2
4	24.59	3.61	Female	No	Sun	Dinner	4



Hotel_Transations

```
Key total_bill  tip sex smoker  day time    size
0   16.99   1.01    Female  No  Sun Dinner  2
1   10.34   1.66    Male    No  Sun Dinner  3
2   21.01   3.5 Male    No  Sun Dinner  3
3   23.68   3.31    Male    No  Sun Dinner  2
4   24.59   3.61    Female  No  Sun Dinner  4
5   25.29   4.71    Male    No  Sun Dinner  4
6   8.77    2.0 Male    No  Sun Dinner  2
7   26.88   3.12    Male    No  Sun Dinner  4
8   15.04   1.96    Male    No  Sun Dinner  2
9   14.78   3.23    Male    No  Sun Dinner  2
10  10.27   1.71    Male    No  Sun Dinner  2
11  35.26   5.0 Female  No  Sun Dinner  4
12  15.42   1.57    Male    No  Sun Dinner  2
13  18.43   3.0 Male    No  Sun Dinner  4
```

Exercise1: Goal

Perform Data Analysis on this file to understand customer behavior patterns and trends on Sex code, Meal type, Day of the week and Smoker with respect to Bill amount and tips.

To achieve this goal:

1. We will load this dataset to Snowflake
2. Perform simple Data exploration and analysis using SQL, Python and Tableau

Note: #2 can also be replaced with any other BI tool

Lets do it & **Learn Snowflake along the way!**

SQL Scripting and Web UI

Script your objects or create them in the friendly Snowflake web interface

Script your SQL

Use Web UI features

For purposes of education, you will see both

Exercise1: Let’s create a database

Create the database using the Snowflake Interface

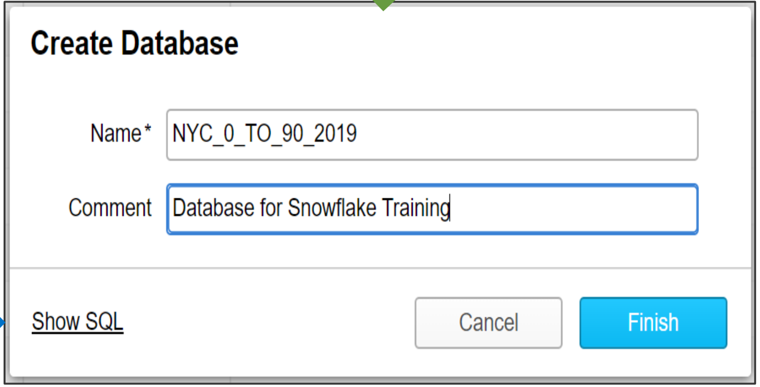
ONLY INSTRUCTOR WILL RUN THIS COMMAND

Option1: SQL

CREATE DATABASE NYC_0_TO_90_2019 Comment “Database for Snowflake Training”;

For this class, we have variables set for the ease of scripting our solution.

Option 2: Web UI ✓



Create Database

Name* NYC_0_TO_90_2019

Comment Database for Snowflake Training

Click here

Exercise1: Let’s create a database schema

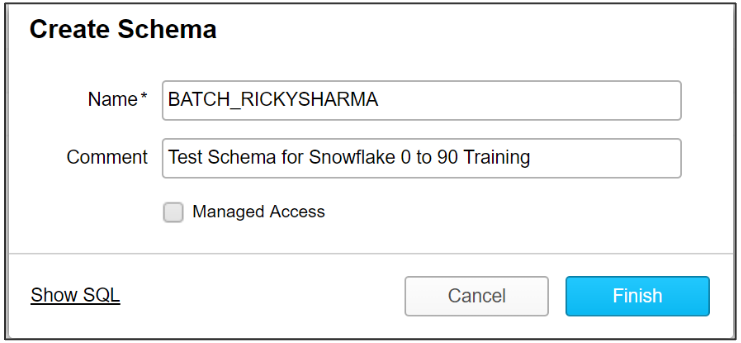
BATCH_<your user name>

Create the schema using the Snowflake Interface

Option1: SQL

```
 -- WE ALL WILL SHARE SAME DATABASE AND EACH USER WILL CREATE THEIR
OWN SCHEMA, WAREHOUSE, TABLE, FILE FORMAT AND PEFORM DATA LOADING
      USE ROLE NYC_TEMP;

   -- BELOW VARIABLES WILL HELP YOU EASILY ASSOCIATE YOUR OBJECTS BY
USING YOUR SNOWFLAKE USERID
      SET Current_User = CURRENT_USER();
      SET Current_DB = 'NYC_0_TO_90_2019';
      SET Current_schema = 'BATCH_' || $Current_User;
  --    SET Current_stage = 'NYC_TRN_STG_' || $Current_User;
      SET Current_FF = 'FF_CSV_' || $Current_User;
      SET Current_FF_JSON = 'FF_JSON_' || $Current_User;
      SET Current_WH = 'NYC_TRN_' || $Current_User;


      USE Database IDENTIFIER($Current_DB);
   -- Create a new schema.. eg. BATCH_RICKYSHARMA
      Create Schema IDENTIFIER($Current_schema)   COMMENT = 'Test Schema for
Snowflake 0 to 90 Training';
      USE Schema IDENTIFIER($Current_schema);

   -- Optional: Execute grants if user groups operate in different roles and need access to
your data
      GRANT ALL ON Schema IDENTIFIER($Current_schema) TO ROLE NYC_TEMP;
```

Option 2: Web UI ✓



Create Schema

Name* BATCH_RICKYSHARMA

Comment Test Schema for Snowflake 0 to 90 Training

☐ Managed Access

Exercise1: Create a Warehouse

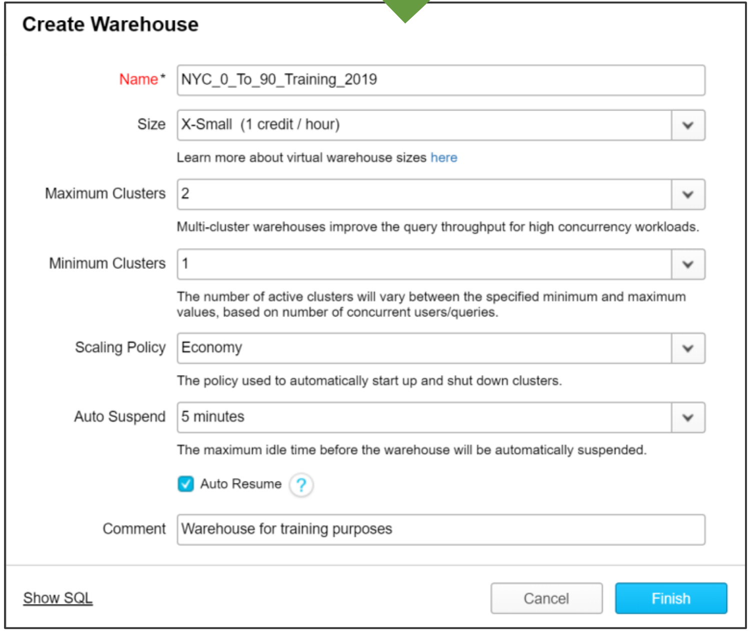
Create the Warehouse using the Snowflake Interface

NYC_TRN_<your user name>

Option1: SQL

```
 -- CREATE A NEW WAREHOUSE. Eg;
NYC_TRN_RICKYSHARMA
       CREATE WAREHOUSE
IDENTIFIER($Current_WH) WITH WAREHOUSE_SIZE
= 'XSMALL' WAREHOUSE_TYPE = 'STANDARD'
          AUTO_SUSPEND = 300 AUTO_RESUME =
TRUE MIN_CLUSTER_COUNT = 1
MAX_CLUSTER_COUNT = 2 SCALING_POLICY =
'ECONOMY'
          COMMENT = 'Warehouse for 0 to 90 training at
NYC';
```

Option 2: Web UI ✓



Create Warehouse

Name*	NYC_0_To_90_Training_2019
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	Economy
	The policy used to automatically start up and shut down clusters.
Auto Suspend	5 minutes
	The maximum idle time before the warehouse will be automatically suspended.
	☑ Auto Resume ?
Comment	Warehouse for training purposes

Exercise1: Create a File Format

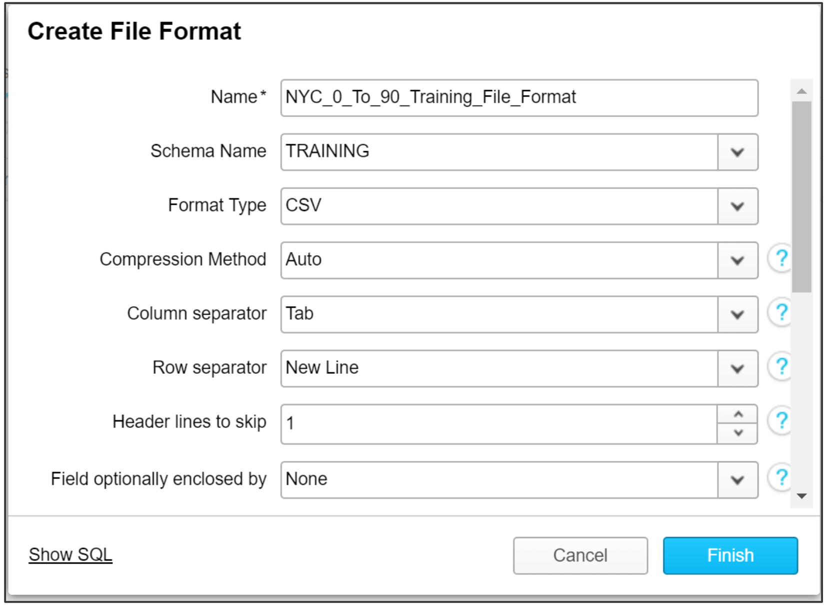
Create the File Format using the Snowflake Interface

FF_CSV_<your user name>

Option1: SQL

```
-- CREATE A NEW FILE FORMAT FOR LOADING CSV FILES
THAT ARE TAB DELIMITED AND HAVE COLUMN
HEADERS.
    -- Eg:  FF_CSV_RICKYSHARMA
        CREATE FILE FORMAT IDENTIFIER($Current_FF)
TYPE = 'CSV' COMPRESSION = 'AUTO' FIELD_DELIMITER =
'\t'
            RECORD_DELIMITER = '\n' SKIP_HEADER = 1
FIELD_OPTIONALLY_ENCLOSED_BY = 'NONE'
TRIM_SPACE = TRUE
            ERROR_ON_COLUMN_COUNT_MISMATCH = TRUE
ESCAPE = 'NONE' ESCAPE_UNENCLOSED_FIELD = '\134'
DATE_FORMAT = 'AUTO'
            TIMESTAMP_FORMAT = 'AUTO' NULL_IF = ('\\N');
```

Option 2: Web UI



Create File Format

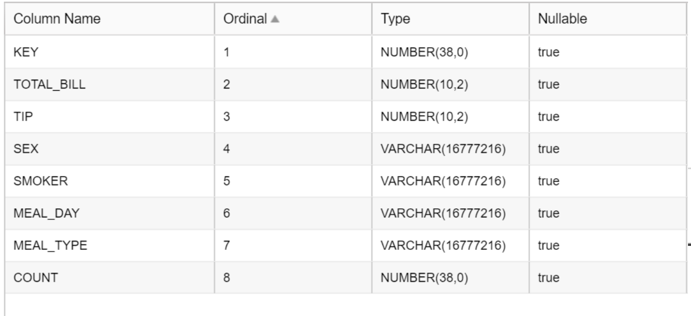
Name*	NYC_0_To_90_Training_File_Format
Schema Name	TRAINING
Format Type	CSV
Compression Method	Auto
Column separator	Tab
Row separator	New Line
Header lines to skip	1
Field optionally enclosed by	None

Exercise1: Let’s create a database table

Create the table(s) using the Snowflake Interface or DDL

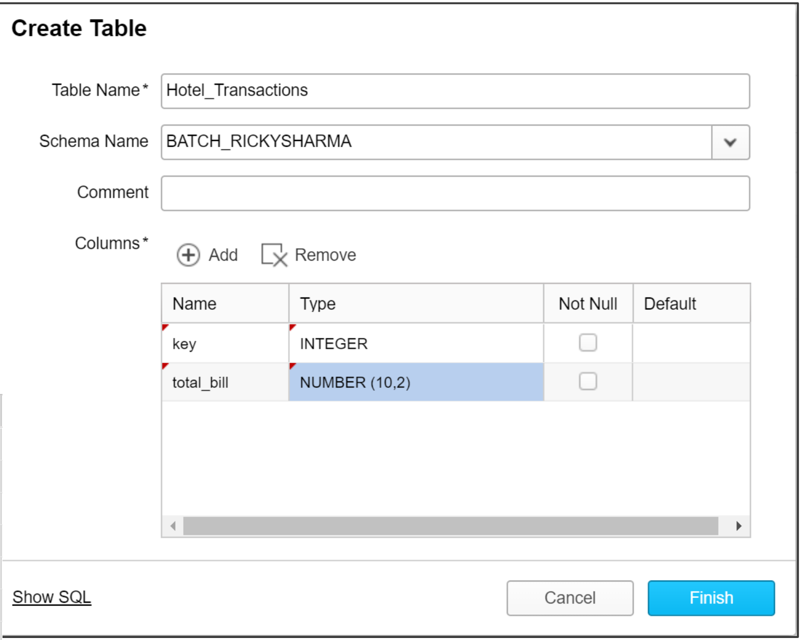
Option1: SQL

```
 -- CREATE A NEW TABLE FOR HOTEL
TRANSACTIONS DATASET
      Create or Replace Table Hotel_Transactions  --
provide your table name here
          (
          key integer,
          total_bill decimal(10,2),
          tip decimal(10,2),
          sex varchar,
          smoker varchar,
          meal_day varchar,
          meal_type varchar,
          count integer
          );
```



Column Name	Ordinal ▲	Type	Nullable
KEY	1	NUMBER(38,0)	true
TOTAL_BILL	2	NUMBER(10,2)	true
TIP	3	NUMBER(10,2)	true
SEX	4	VARCHAR(16777216)	true
SMOKER	5	VARCHAR(16777216)	true
MEAL_DAY	6	VARCHAR(16777216)	true
MEAL_TYPE	7	VARCHAR(16777216)	true
COUNT	8	NUMBER(38,0)	true

Option 2: Web UI



Create Table

Table Name*: Hotel_Transactions

Schema Name: BATCH_RICKYSHARMA

Comment:

Columns*

Name	Type	Not Null	Default
key	INTEGER	□	
total_bill	NUMBER (10,2)	□	

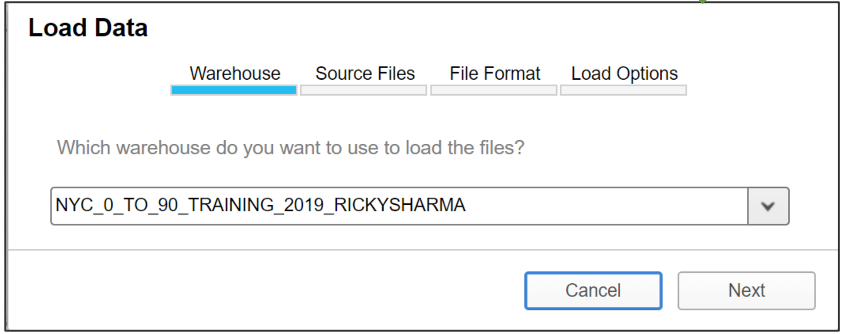
Exercise1: Load the File

Load the dataset into a Snowflake Table

Option 1: From Stage using SQL

```
 -- PICK A WAREHOUSE TO BE USED FOR DATA LOADING AND ANALYSIS

        Use warehouse IDENTIFIER($Current_WH) ;

-------------------------- DATA LOADING -----------------------------------

-- OPTION 2: LOAD THE FILE FROM SNOWFLAKE STAGE INTO THE TABLE. THE FILE WAS COPIED TO STAGE BY ADMIN SCRIPTS SHARED ABOVE

        Copy into Hotel_Transactions from
@NYC_0_TO_90_2019.TRAINING.nyc_0_to_90_training_2019_stage/Hotel_Transations.csv.gz

            file_format= $Current_FF;
```

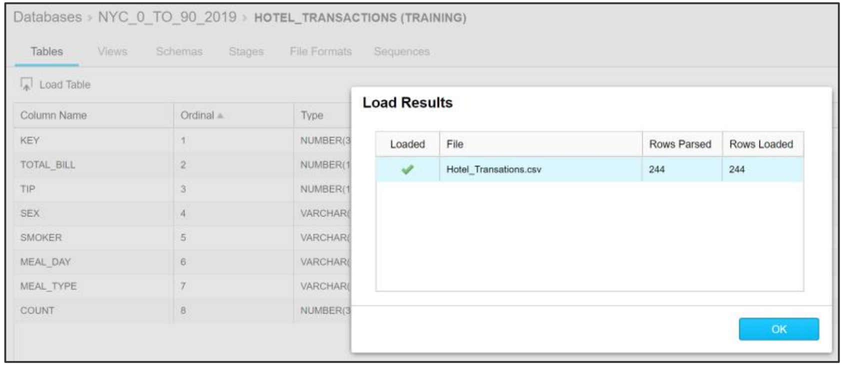
Option 2: From Local using UI



Load Data

Which warehouse do you want to use to load the files?

NYC_0_TO_90_TRAINING_2019_RICKYSHARMA



Databases › NYC_0_TO_90_2019 › HOTEL_TRANSACTIONS (TRAINING)

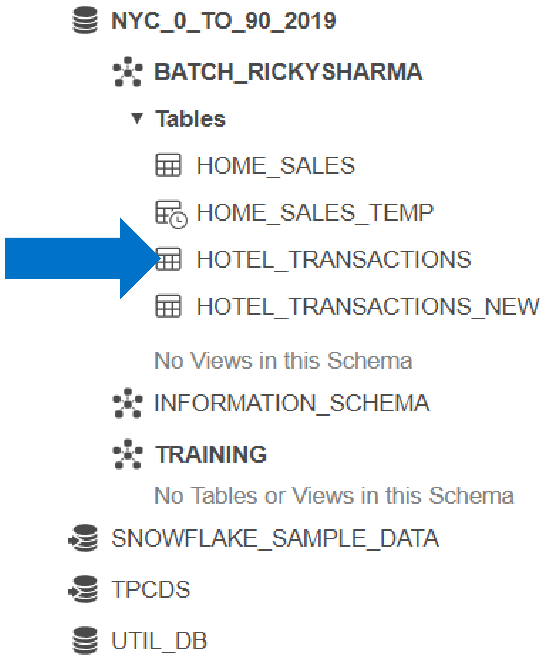
Column Name	Ordinal ▲	Type
KEY	1	NUMBER(3
TOTAL_BILL	2	NUMBER(1
TIP	3	NUMBER(1
SEX	4	VARCHAR(
SMOKER	5	VARCHAR(
MEAL_DAY	6	VARCHAR(
MEAL_TYPE	7	VARCHAR(
COUNT	8	NUMBER(3

Load Results

Loaded	File	Rows Parsed	Rows Loaded
✓	Hotel_Transations.csv	244	244

Exercise1: Review Actions done so far

After the Scripts run click database browser on the left side of your screen. Your content should be similar to the images below.



- NYC_0_TO_90_2019
  - BATCH_RICKYSHARMA
    - ▼ Tables
      - HOME_SALES
      - HOME_SALES_TEMP
      - HOTEL_TRANSACTIONS
      - HOTEL_TRANSACTIONS_NEW
    - No Views in this Schema
  - INFORMATION_SCHEMA
  - TRAINING
    - No Tables or Views in this Schema
- SNOWFLAKE_SAMPLE_DATA
- TPCDS
- UTIL_DB

You have created a Schema, Warehouse, File Format and Table and loaded data into that table

Exercise1: Perform Data Analysis using SQL

Execute data analysis scripts

```
------------------------- DATA ANALYSIS -----------------------------------

  -- Query the Data
      Select * from Hotel_Transactions limit 100;

  -- Identify Average Bill Vs Avg. Tips
    Select
      Avg(TOTAL_BILL) As Avg_Bill ,
      Avg(Tip) as Avg_Tip,
      Avg_Tip / Avg_Bill * 100 As Percentage_Tip
      FROM  Hotel_Transactions;

  -- Analyze Maximum and Minimum tips Vs Bills for each Gender and Meal Type
    Select
      Meal_type, sex,
      max(TOTAL_BILL) As max_Total_bill,
      min(TOTAL_BILL) As min_Total_bill,
      max(TIP) As max_TIP,
      min(TIP) As min_TIP
      FROM  Hotel_Transactions Group by Meal_type, sex;


  -- Identify maximum Bill per meal day
    Select
        max(total_bill) as max_total_bill, meal_day
        FROM  Hotel_Transactions Group by meal_day order by max_total_bill desc;
```

Exercise1: Perform Data Analysis using Python

We are using Anaconda Navigator to create a Jupyter Notebook which connects to Snowflake and loads data into a Data Frame for further plotting and statistical analysis

```
# 0 to 90 Workshop
# Connect to Snowflake and fetch the table into a Data Frame
#!/usr/bin/env python
import snowflake.connector
import pandas as pd

from sqlalchemy import create_engine
from snowflake.sqlalchemy import URL

url = URL(
    account = ‘xxx',
    user = 'xxx',
    password = 'xxx',
    database = 'NYC_0_TO_90_2019',
    schema = 'TRAINING',
    warehouse = 'DEMO_WH',
    role='NYC_TEMP’,
)
#    authenticator='https://xxxxx.okta.com',
engine = create_engine(url)


connection = engine.connect()

query = '''
select * from Hotel_Transactions
;
'''

tips = pd.read_sql(query, connection)
#print (df)
tips
```

```
# before running the Python commands, login to CMD and
type below commands to install Python libraries for
Snowflake
# pip install snowflake-connector-python
# pip install snowflake-sqlalchemy
```

Exercise1: Perform Data Analysis using Python

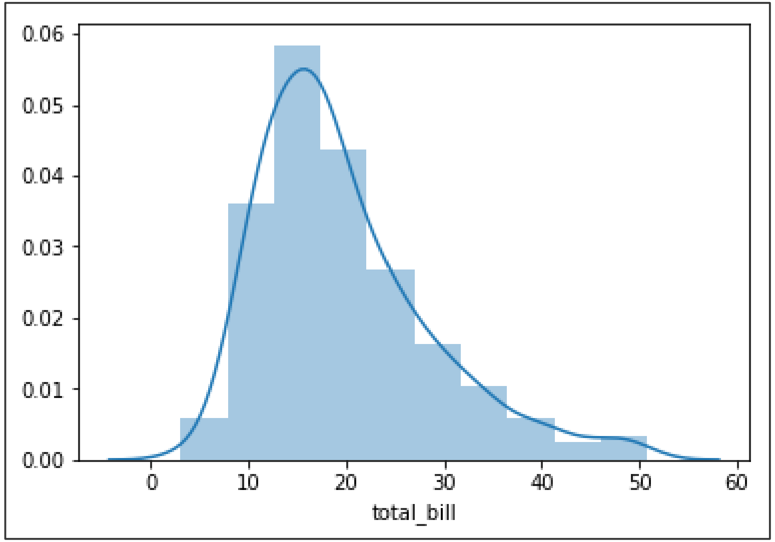
We are using Anaconda Navigator to create a Jupyter Notebook which connects to Snowflake and loads data into a Data Frame for further plotting and statistical analysis

# We will use Seaborn - statistical plotting library

Plot Total Bill Distribution

Python Code:

```
sns.distplot(tips['total_bill'],bins=10)
```



Average bill is centered around $18

Exercise1: Perform Data Analysis using Python

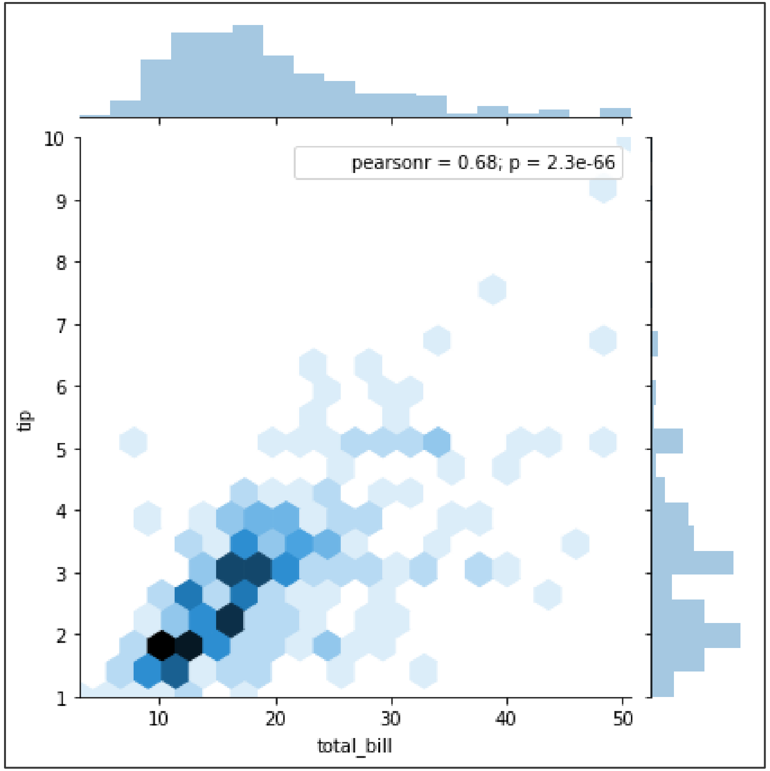
We are using Anaconda Navigator to create a Jupyter Notebook which connects to Snowflake and loads data into a Data Frame for further plotting and statistical analysis

# We will use Seaborn - statistical plotting library

Plot - bill Vs tips

Python Code:

```
sns.jointplot(x='total_bill',y='tip',data=tips,
kind='hex')
```



Tips are usually 15% of Total Bill Amount

Exercise1: Perform Data Analysis using Python

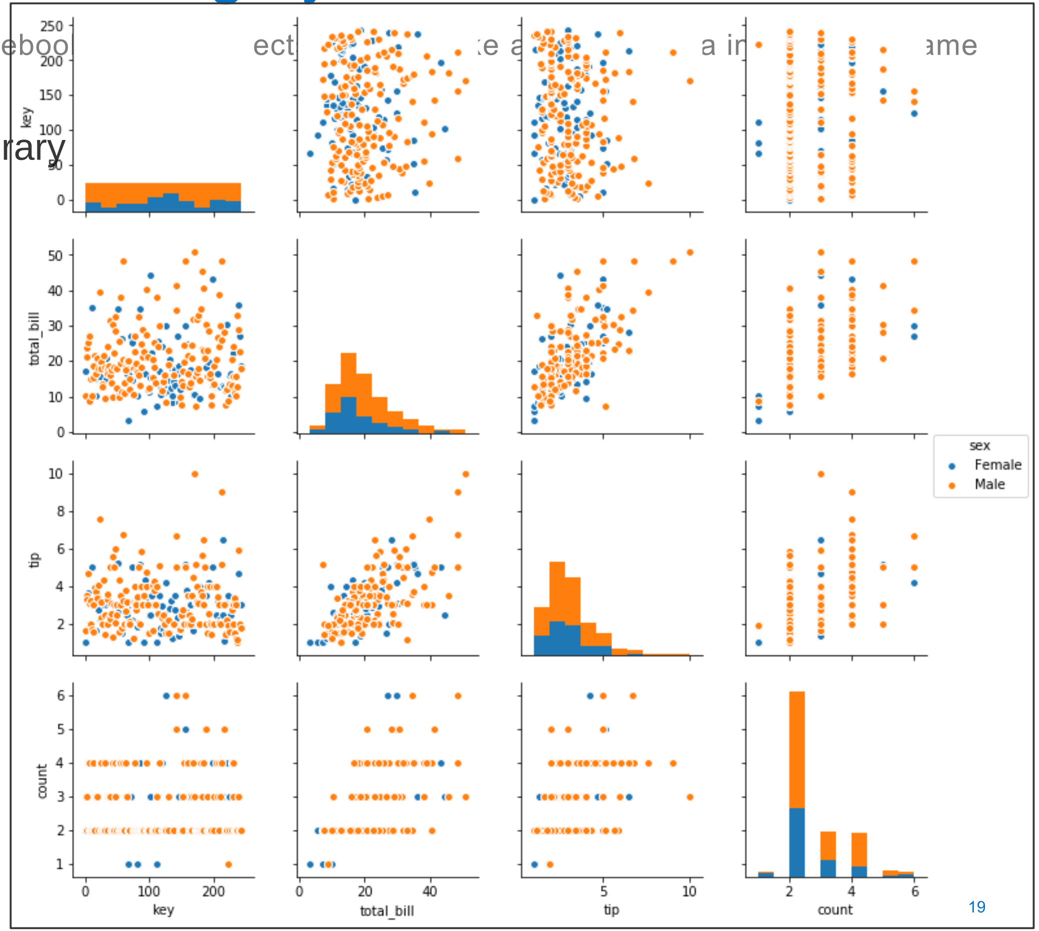
We are using Anaconda Navigator to create a Jupyter Noteboo ect te a a ir ame for further plotting and statistical analysis

# We will use Seaborn - statistical plotting library

pair plot - Auto-generates various statistics based on numerical columns in Dataframe

Python Code:

```
sns.pairplot(tips, hue='sex') # color
based on sex
```



Males have more transactions and tends to Bill more and Tip more than females

Exercise1: Perform Data Analysis using Python

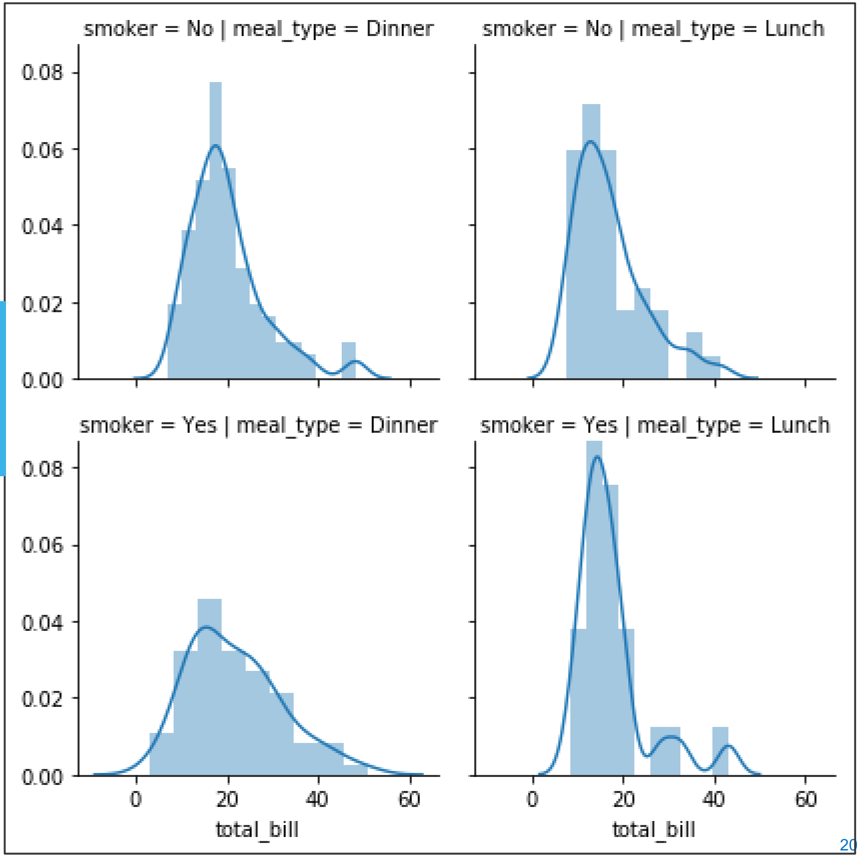
We are using Anaconda Navigator to create a Jupyter Notebook which connects to Snowflake and loads data into a Data Frame for further plotting and statistical analysis

# We will use Seaborn - statistical plotting library

Total Bills By Smoker and Meal Type

Python Code:

```
import numpy as np
import seaborn as sns
g = sns.FacetGrid(data=tips,col='meal_type',row='smoker')
g.map(sns.distplot,'total_bill')
```



Non Smokers tend to bill slightly more than smokers during Dinner time however its reverse at Lunch

Exercise1: Perform Data Analysis using Python

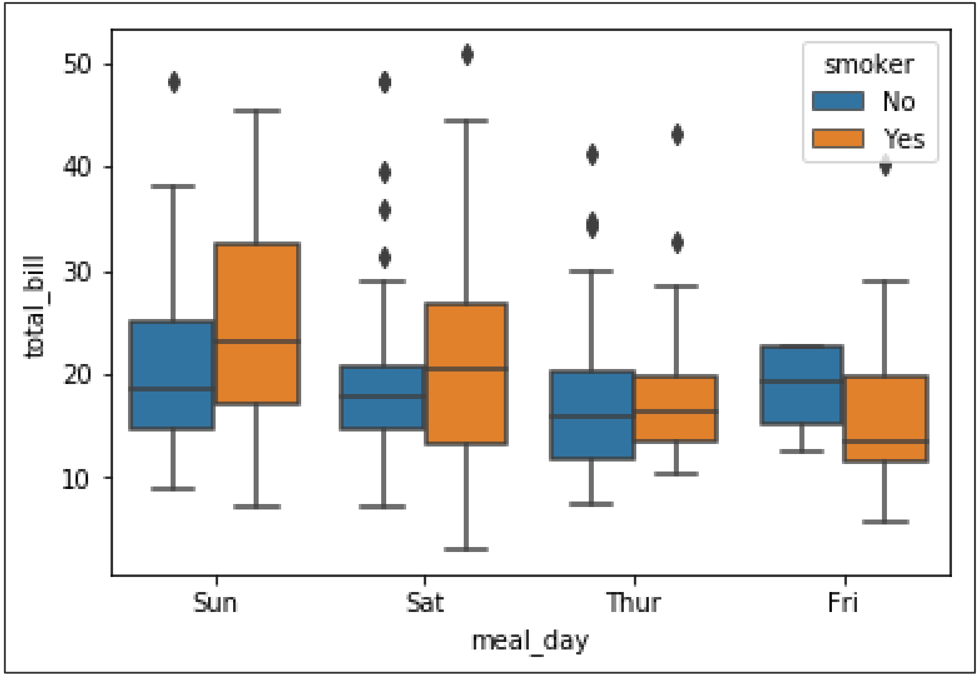
We are using Anaconda Navigator to create a Jupyter Notebook which connects to Snowflake and loads data into a Data Frame for further plotting and statistical analysis

# We will use Seaborn - statistical plotting library

Distribution of Bill by Day of the week and Smoker Vs non-smoker

Python Code:

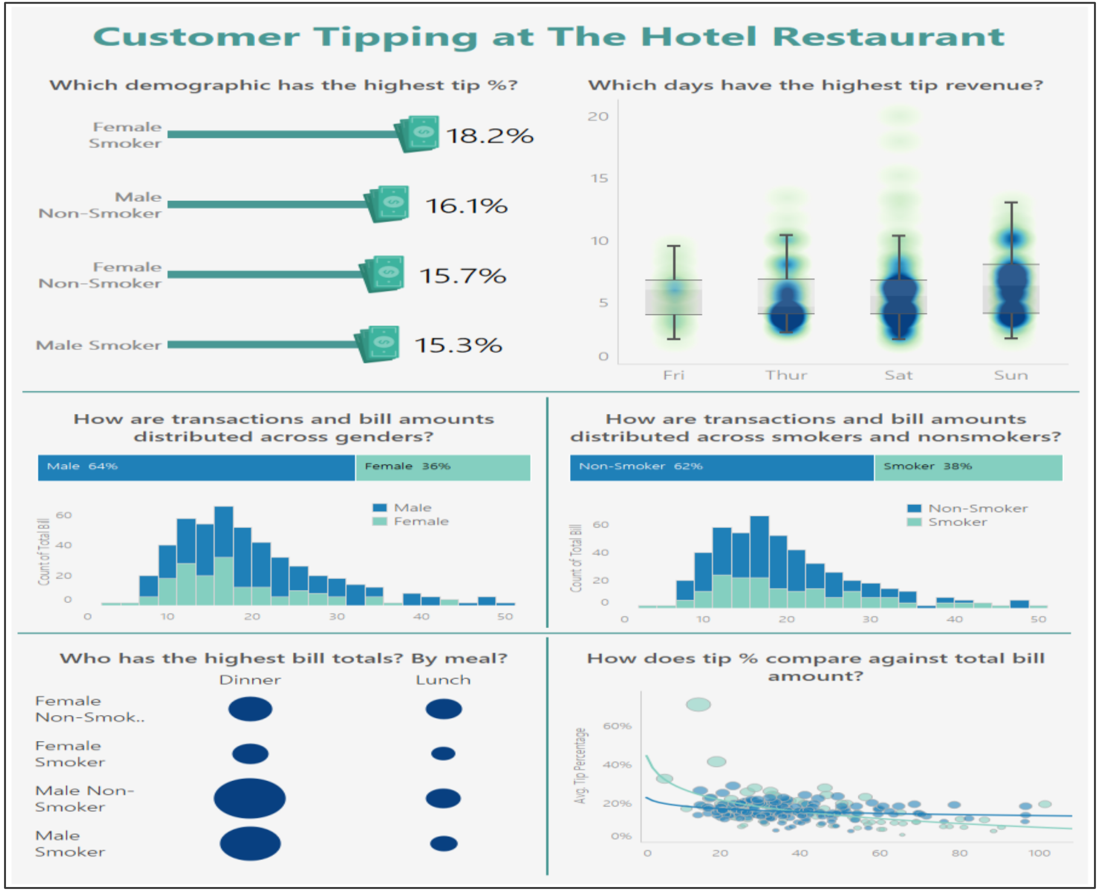
```
# distribution of data
sns.boxplot(x='meal_day',y='total_bill',data=tips, hue='smoker')
# total bill - y axis and tips are bars
#bars - distributed by smoker
```



Sundays brings the top earnings!

Smokers tend to pay more bill amounts than non smokers.

Exercise1: Perform Data Analysis using Tableau



Customer Tipping at The Hotel Restaurant

Which demographic has the highest tip %?

Which days have the highest tip revenue?

How are transactions and bill amounts distributed across genders?

How are transactions and bill amounts distributed across smokers and nonsmokers?

Who has the highest bill totals? By meal?

How does tip % compare against total bill amount?

Lab 3: Explore Snowflake differentiating features

Demonstrate Time Travel and Cloning features

Explore unique Snowflake Features

Explore Snowflake deeper

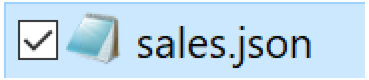
```
---------------------- EXPLORE UNIQUE SNOWFLAKE FEATURES ----------------------------

      -- Accidently DROP this table!!
        DROP table  Hotel_Transactions;

      -- Recover dropped table back
        UNDROP table Hotel_Transactions;
        Select * from Hotel_Transactions;


      -- Clone this table into a new table - without replicating data
        CREATE  TABLE Hotel_Transactions_NEW CLONE  Hotel_Transactions;
        Select * from Hotel_Transactions_NEW;

      -- Update Column Count by adding 5 to its current value
        Update  Hotel_Transactions Set Count = count + 5;
        Select * from Hotel_Transactions;

      -- Now, Use time travel feature to look at the previous state of this data.. 5 mins ago
        Select * from Hotel_Transactions at(offset => -60*5);
```

Lab 4: Configure and Load Semi-Structured Data to Snowflake

This will demonstrate how to configure your database and related objects to process and store data

Exercise2: Load Semi-structured data to Snowflake

File Name

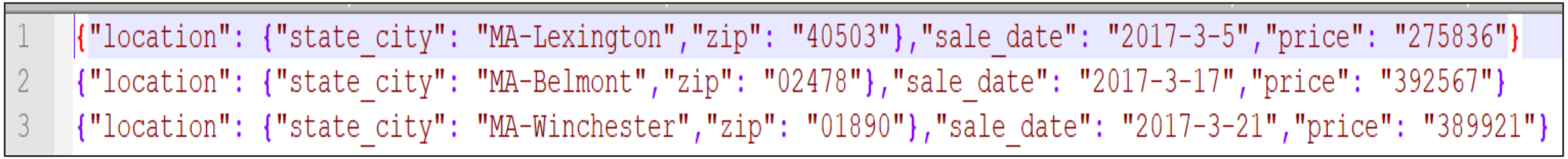


☑ sales.json

sales.json

File Location: https://github.com/rsharma84/snowflakedemo sales.json

Preview:



```
{"location": {"state_city": "MA-Lexington","zip": "40503"},"sale_date": "2017-3-5","price": "275836"}
{"location": {"state_city": "MA-Belmont","zip": "02478"},"sale_date": "2017-3-17","price": "392567"}
{"location": {"state_city": "MA-Winchester","zip": "01890"},"sale_date": "2017-3-21","price": "389921"}
```

Exercise2: Goal

Convert JSON dataset into a structured format and perform data analysis using Snowflake

To achieve this goal:

1. We will load this dataset to Snowflake into a Temporary table (stage area) retaining JSON structure
2. Use Snowflake SQL parsing features to parse data from #1 into a structured format and load into a Permanent table
3. Perform simple Data exploration and analysis

Let’s prep Snowflake for our batch data

Simple steps to create the space and service to operate on our data

1

Configure Compute Service and File format

Snowflake separates data and storage, we need compute to execute our loads and queries. These are called Virtual Warehouses in Snowflake.

File Format helps defining file structures for Data Loading

2

Run the load process – Copy to Stage (JSON format)

In the Snowflake Worksheet drop in the Copy statement and load the date from internal (Stage) , external storage (S3, Blob) or Local File system to Snowflake target table

3

Parse JSON to structured format

Use Snowflake SQL features to parse JSON into a Structured format and load into a Permanent Target Table

4

Review the load history and profile

Examine the load history as the data is being loaded in the Worksheet. Shift over to History tab and examine the Load History and the Query Profile for much more detail

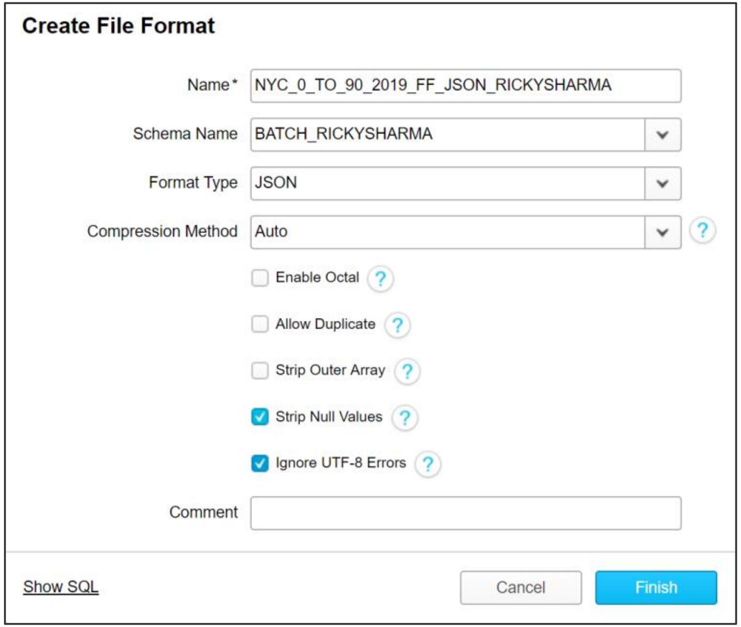
Exercise2: Create a File Format

Create the File Format using the Snowflake Interface

Option1: SQL ✓

```
 -- Create File format for JSON Dataset
        CREATE FILE FORMAT
IDENTIFIER($Current_FF_JSON) TYPE = 'JSON'
COMPRESSION = 'AUTO'
                ENABLE_OCTAL = FALSE
ALLOW_DUPLICATE = FALSE
STRIP_OUTER_ARRAY = FALSE
STRIP_NULL_VALUES = TRUE
IGNORE_UTF8_ERRORS = TRUE;
```

Option 2: Web UI



Create File Format

Name* NYC_0_TO_90_2019_FF_JSON_RICKYSHARMA

Schema Name BATCH_RICKYSHARMA

Format Type JSON

Compression Method Auto

□ Enable Octal

□ Allow Duplicate

□ Strip Outer Array

☑ Strip Null Values

☑ Ignore UTF-8 Errors

Comment

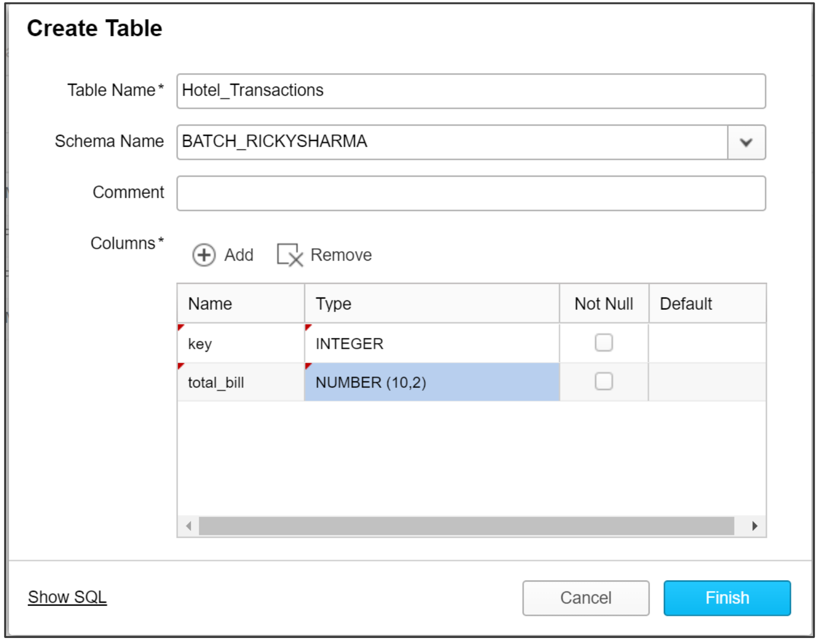
Exercise2: Let’s create a database table

Create the table(s) using the Snowflake Interface or DDL

Option1: SQL ✓

```
 /* Create a target table for the JSON data. The table is
temporary, meaning it persists only for the duration
   of the user session and is not visible to other users. */

      Create or replace temporary table
home_sales_temp
      (col1 variant);

   /* Create a Structured Table to store parsed JSON
data into a columnar format */
      Create or replace table home_sales (
        city string,
        zip string,
        state string,
        type string default 'Residential',
        sale_date timestamp_ntz,
        price string
        );
```

Option 2: Web UI



Create Table

Table Name* Hotel_Transactions

Schema Name BATCH_RICKYSHARMA

Comment

Columns*

Name	Type	Not Null	Default
key	INTEGER	□	
total_bill	NUMBER (10,2)	□	

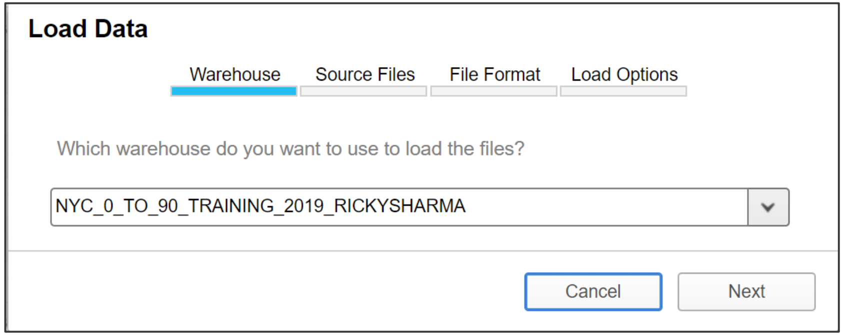
Exercise2: Load the File

Load the dataset into a Snowflake Table

Option 1: From Stage using SQL

```
-- PICK A WAREHOUSE TO BE USED FOR DATA LOADING AND ANALYSIS

        Use warehouse IDENTIFIER($Current_WH) ;


-- OPTION 2: LOAD THE FILE FORM STAGE INTO THE TABLE

      Copy into home_sales_temp from @NYC_0_TO_90_2019.TRAINING.nyc_0_to_90_training_2019_stage/sales.json

      file_format= $Current_FF_JSON;
```

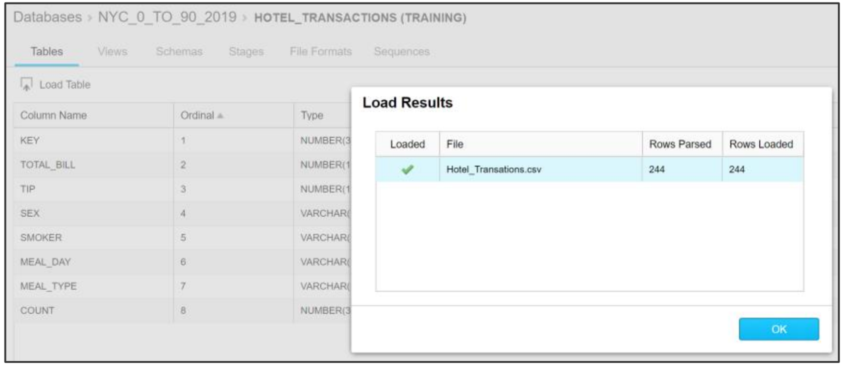
Option 2: From Local using UI



Load Data

Which warehouse do you want to use to load the files?

NYC_0_TO_90_TRAINING_2019_RICKYSHARMA



Databases › NYC_0_TO_90_2019 › HOTEL_TRANSACTIONS (TRAINING)

Column Name	Ordinal ▲	Type
KEY	1	NUMBER(3
TOTAL_BILL	2	NUMBER(1
TIP	3	NUMBER(1
SEX	4	VARCHAR(
SMOKER	5	VARCHAR(
MEAL_DAY	6	VARCHAR(
MEAL_TYPE	7	VARCHAR(
COUNT	8	NUMBER(3

Load Results

Loaded	File	Rows Parsed	Rows Loaded
✓	Hotel_Transations.csv	244	244

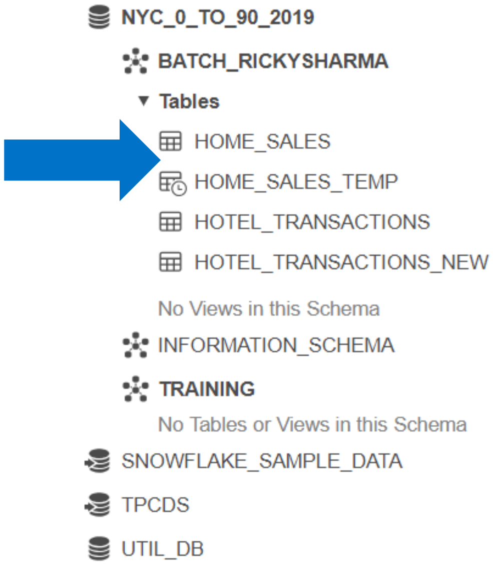
Exercise2: Convert JSON to Structured format

Use Snowflake SQL capabilities to parse JSON into a Structured Dataset

```
-- QUERY THE LOADED DATA (STILL JSON)
    Select * from home_sales_temp;


  -- CONVERT JSON INTO A STRUCTURED FORMAT AND LOAD INTO A STRUCTURED TABLE
     INSERT INTO home_sales (city, state, zip, sale_date, price)
     select substr(col1:location.state_city,4),
          substr(col1:location.state_city,1,2),
          col1:location.zip,
          to_timestamp_ntz(col1:sale_date),
          col1:price
      From home_sales_temp;


-- QUERY THE LOADED DATA (Structured format)
     Select * from home_sales;
```

Exercise2: Check out what has been accomplished so far..

After the Scripts run click database browser on the left side of your screen. Your content should be similar to the images below.



NYC_0_TO_90_2019
BATCH_RICKYSHARMA
▼ Tables
HOME_SALES
HOME_SALES_TEMP
HOTEL_TRANSACTIONS
HOTEL_TRANSACTIONS_NEW
No Views in this Schema
INFORMATION_SCHEMA
TRAINING
No Tables or Views in this Schema
SNOWFLAKE_SAMPLE_DATA
TPCDS
UTIL_DB

You have created a File Format, loaded file to a TEMPORARY Table in JSON structure and parsed it to a Structured format to store into a Permanent Structured Table

Exercise2: Goal - RECAP

Convert JSON dataset into a structured format and perform data analysis using Snowflake

To achieve this goal:

✓ 1. We will load this dataset to Snowflake into a Temporary table (stage area) retaining JSON structure

✓ 2. Use Snowflake SQL parsing features to parse data from #1 into a structured format and load into a Permanent table

✓ 3. Perform simple Data exploration and analysis