# **CSYE 7245 - Big Data Sys and Int Analytics**

## **Lab - Snowflake**

### **Team Information:**

|  |  |
| --- | --- |
| **Name** | **NUID** |
| Tanvi Gurav | 001306848 |
| Keerti Ojha | 001050173 |
| Priyanka Malpekar | 001302741 |

**Lab Completion Date**: 17th February’ 21

# **About**

This lab demonstrates working with Snowflake data cloud, querying, data loading, caching, cloning, user roles and permissions and time travel concepts.

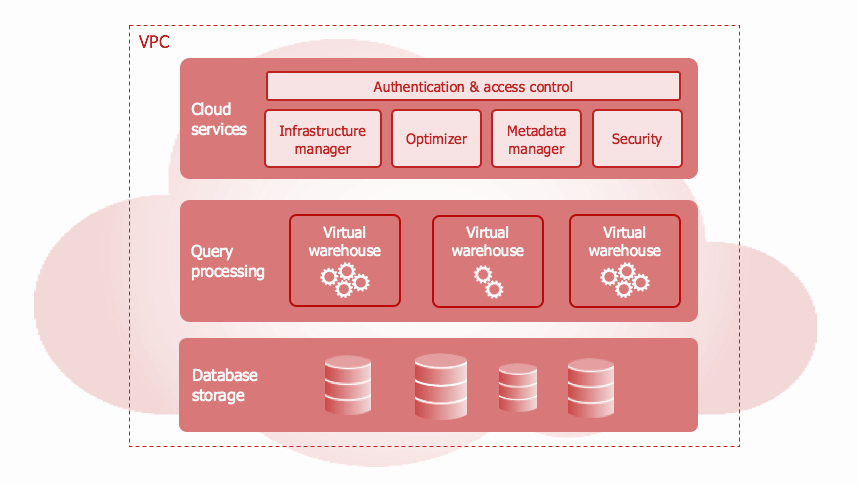
# 

Snowflake’s Data Cloud is powered by an advanced data platform provided as Software-as-a-Service (SaaS). Snowflake enables data storage, processing, and analytic solutions that are faster, easier to use, and far more flexible than traditional offerings.Snowflake combines a completely new SQL query engine with an innovative architecture natively designed for the cloud. To the user, Snowflake provides all of the functionality of an enterprise analytic database, along with many additional special features and unique capabilities.

Snowflake is a true SaaS offering. More specifically:

* There is no hardware (virtual or physical) to select, install, configure, or manage.
* There is virtually no software to install, configure, or manage.
* Ongoing maintenance, management, upgrades, and tuning are handled by Snowflake.

Snowflake runs completely on cloud infrastructure. All components of Snowflake’s service (other than optional command line clients, drivers, and connectors), run in public cloud infrastructures.



Snowflake’s architecture is a hybrid of traditional shared-disk and shared-nothing database architectures. Similar to shared-disk architectures, Snowflake uses a central data repository for persisted data that is accessible from all compute nodes in the platform. But similar to shared-nothing architectures, Snowflake processes queries using MPP (massively parallel processing) compute clusters where each node in the cluster stores a portion of the entire data set locally. This approach offers the data management simplicity of a shared-disk architecture, but with the performance and scale-out benefits of a shared-nothing architecture.

# **Dataset:**

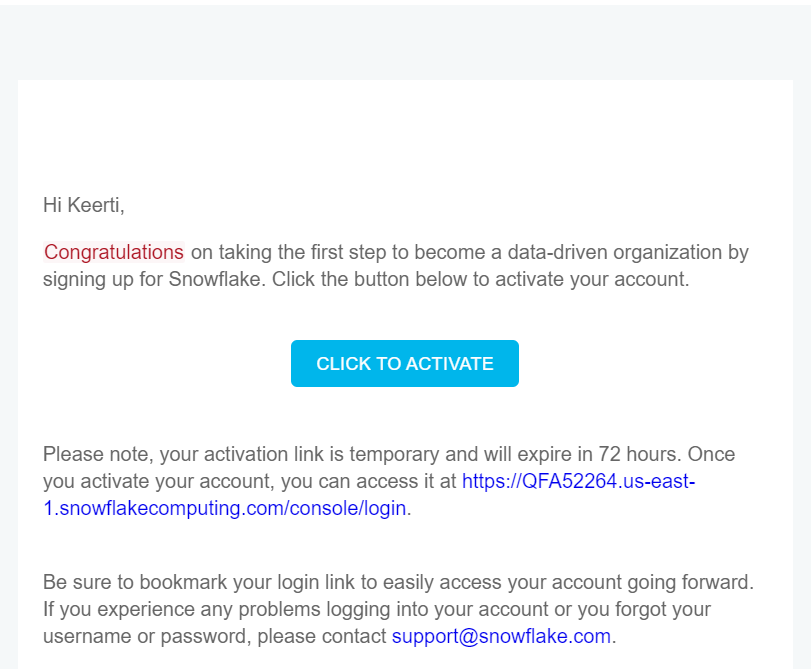
We used 2 data sets in our lab one was trips and the other was weather and it was staged on Snowflake from S3 bucket.

s3://snowflake-workshop-lab/citibike-trips

s3://snowflake-workshop-lab/weather-nyc

# **Experiment setup:**

To Prepare for our Lab Environment we registered for a [Snowflake free 30-day trial](https://trial.snowflake.com/) and used Snowflake Enterprise Edition, AWS cloud provider, and selected US East region. After registering, we received an email with an activation link of our Snowflake account URL



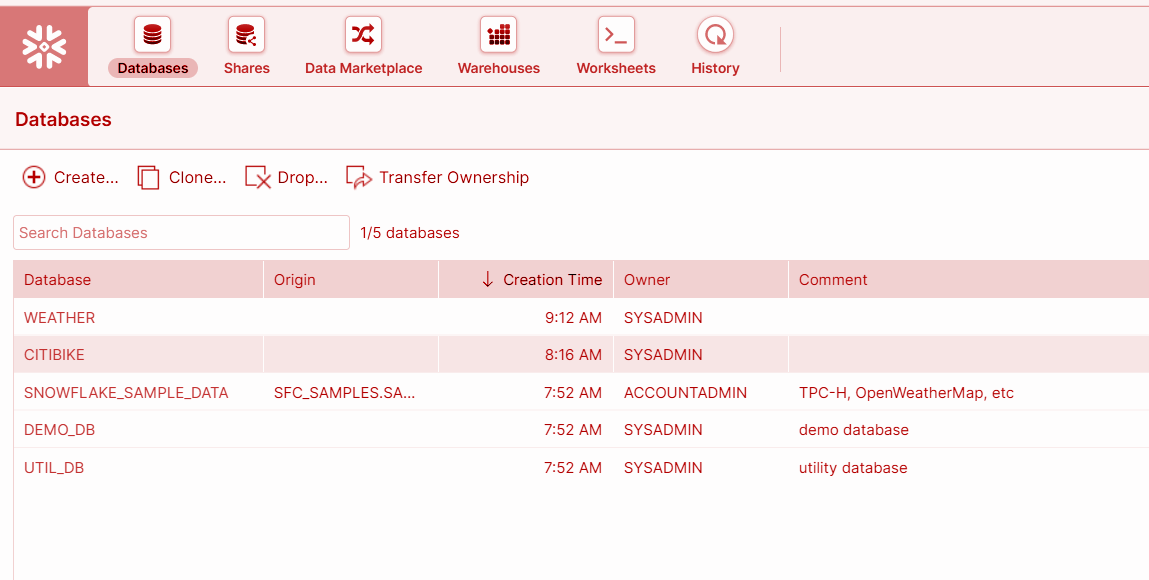
# **Test Cases**

## **Preparing to Load Data**

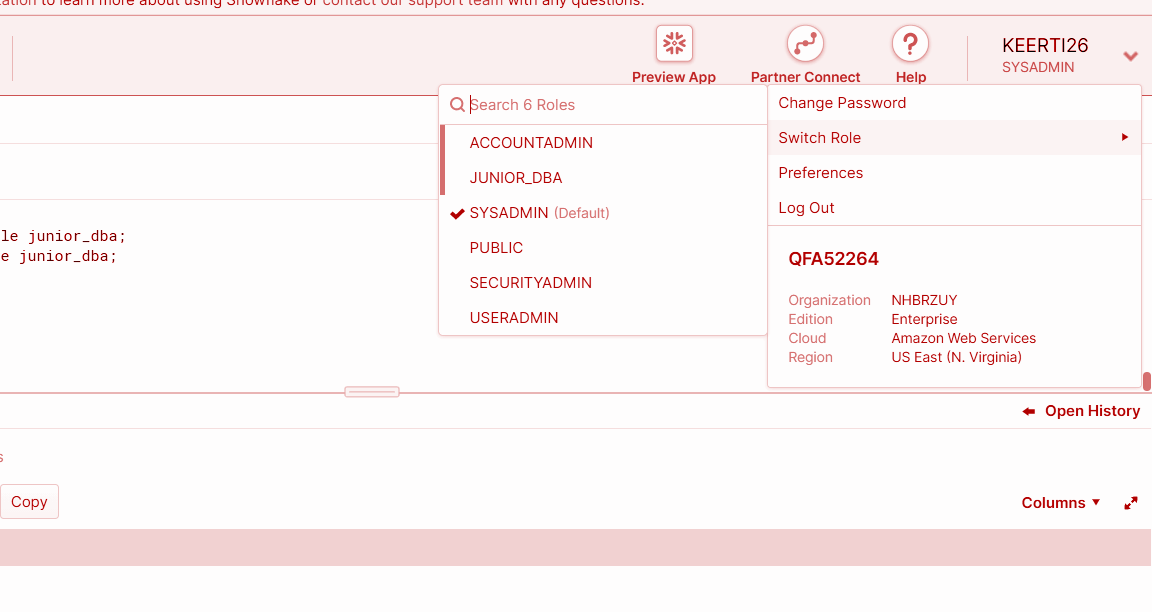
We performed below steps in this section:

* created a database and table
* created an external stage
* created a file format for the data

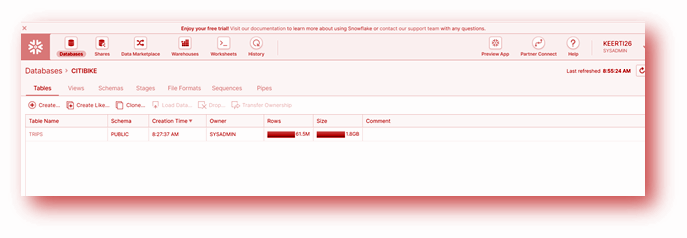
Databases



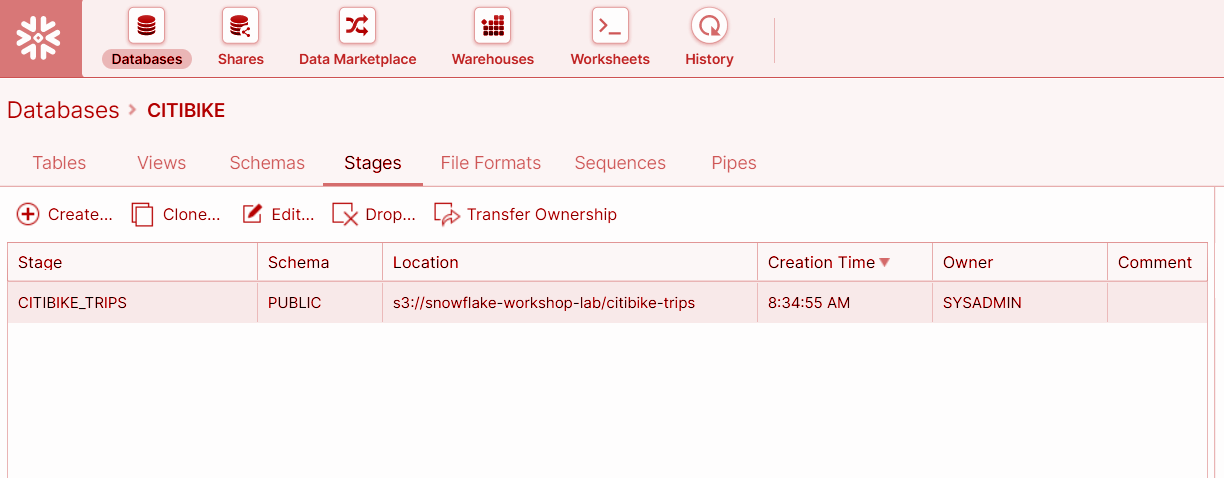
Roles



Table

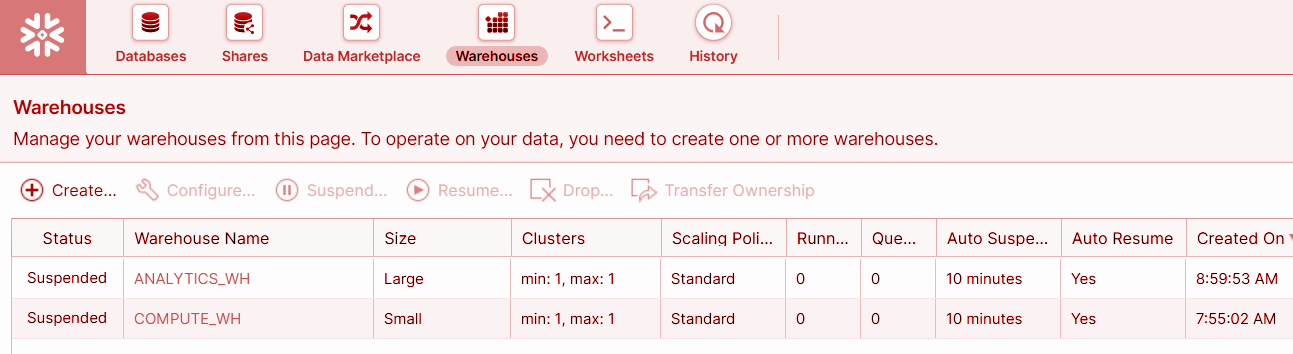


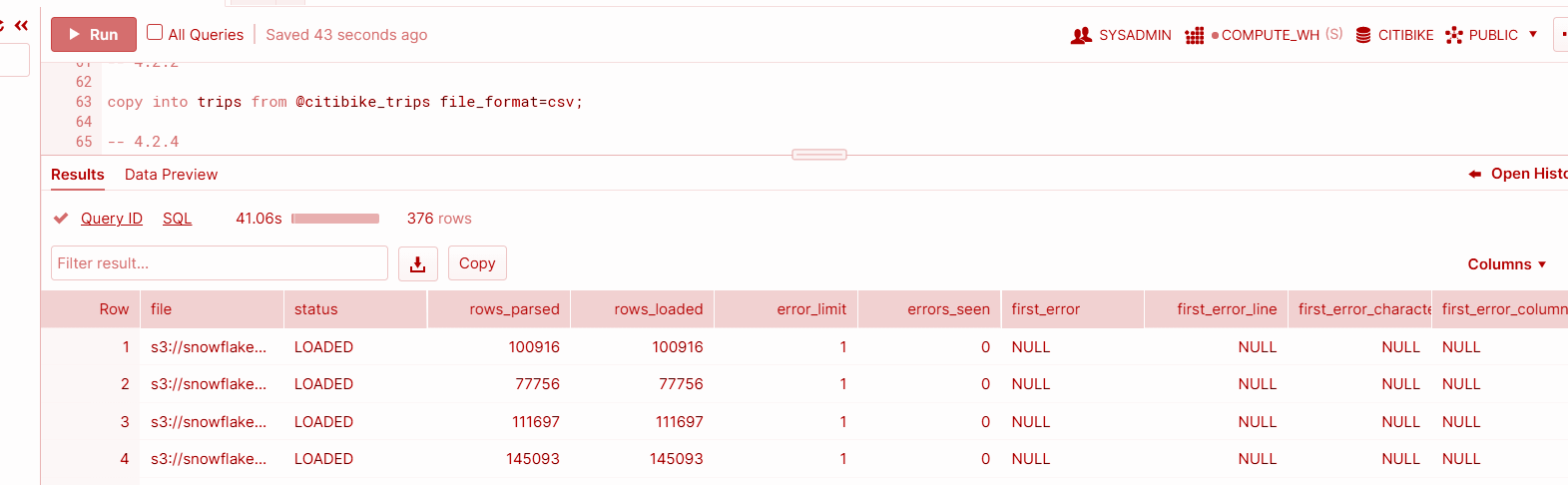
Stages



## Loading Data

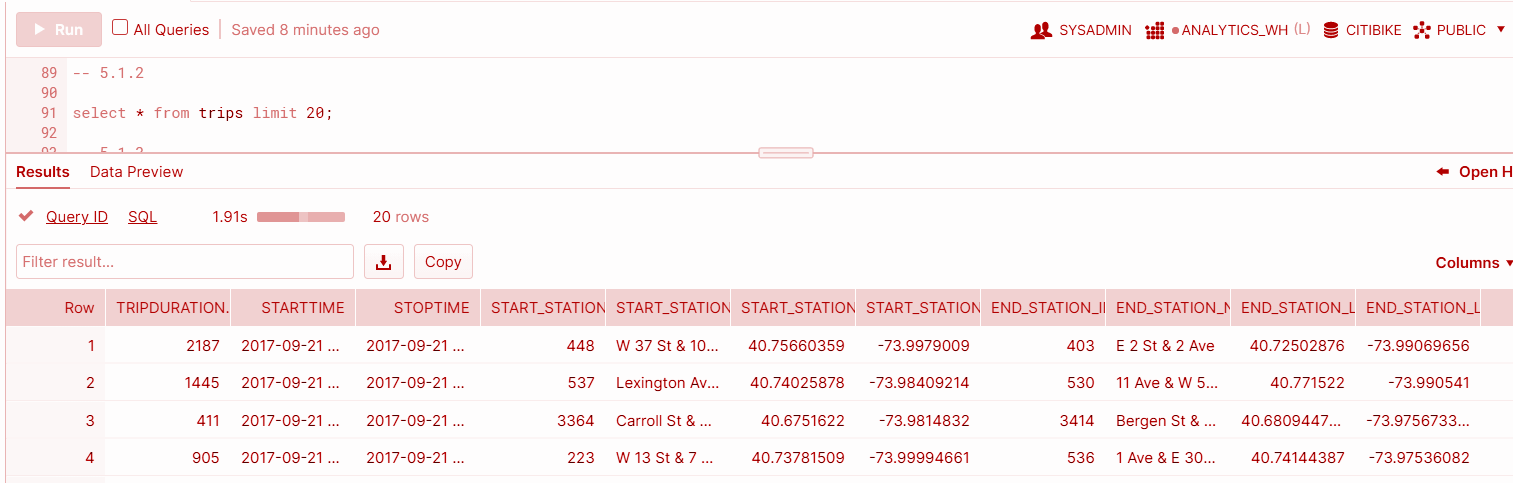
* Created new Warehouse
* Loaded data into table





# **Results**

## Analytical Queries, Results Cache, Cloning

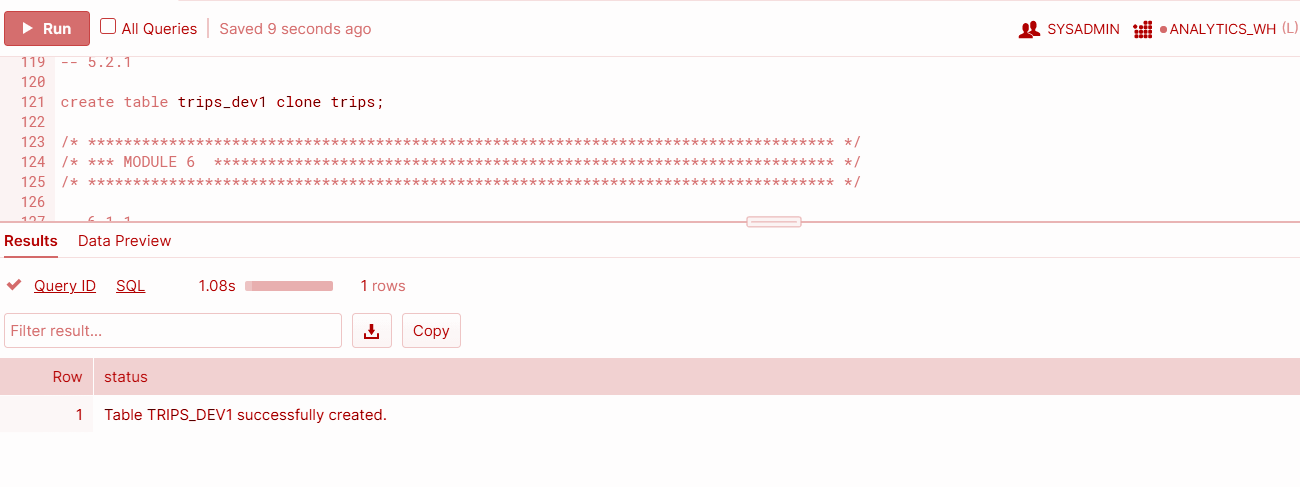
* Executing SELECT Statements and Result Cache 

Caching:





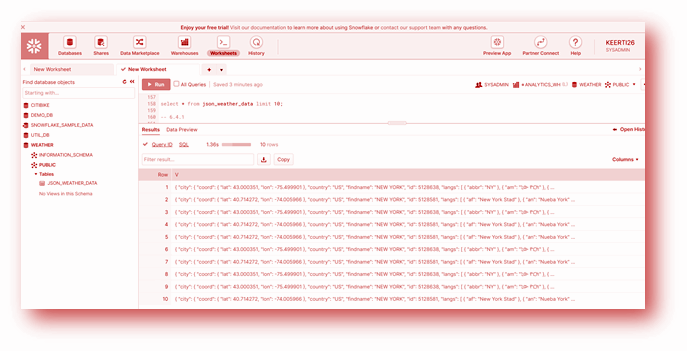
Cloning:



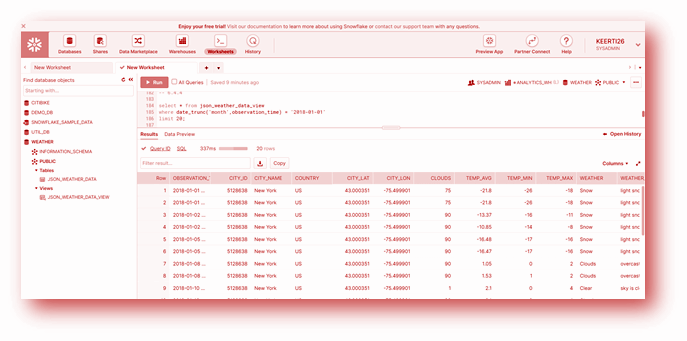
## Working With Semi-Structured Data, Views, JOIN

* Loaded weather data in JSON format held in a public S3 bucket
* Created a View and query the semi-structured data using SQL dot notation

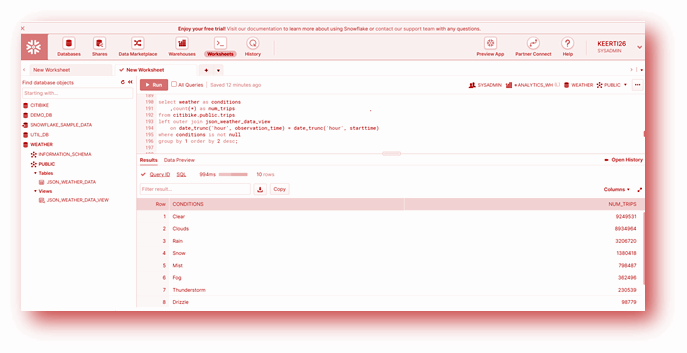
Json data



Creating view



Joining trips data and weather view

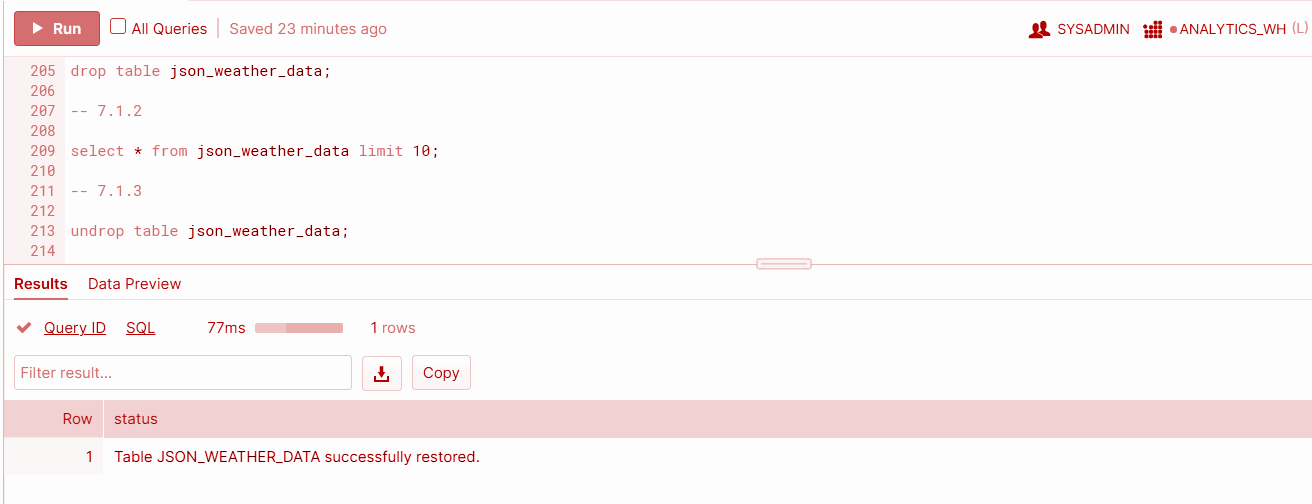


## 

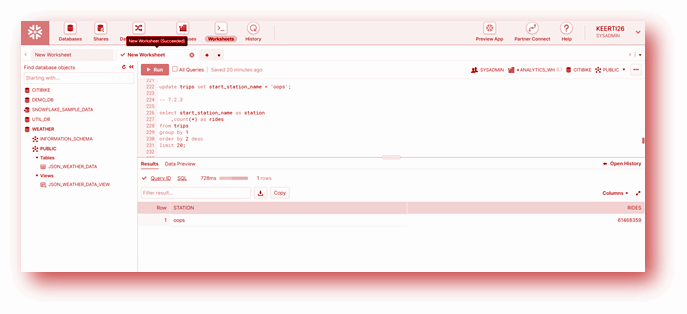
## 

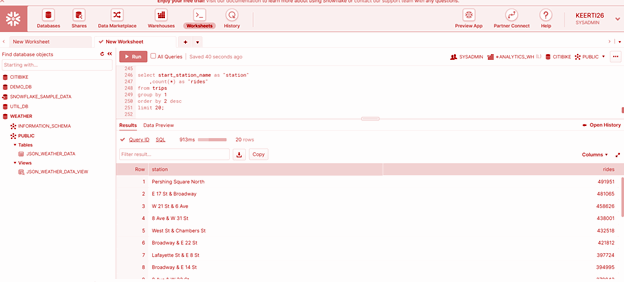
## Time Travelling

Restoring weather table after dropping it accidentally



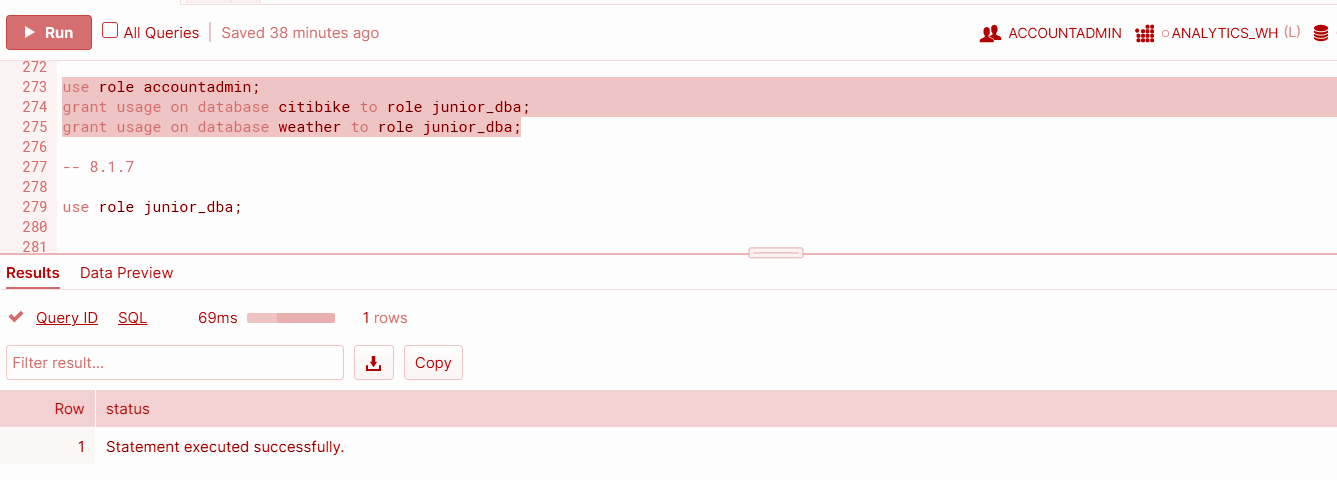
Updating the records with wrong data and then rolling back the action



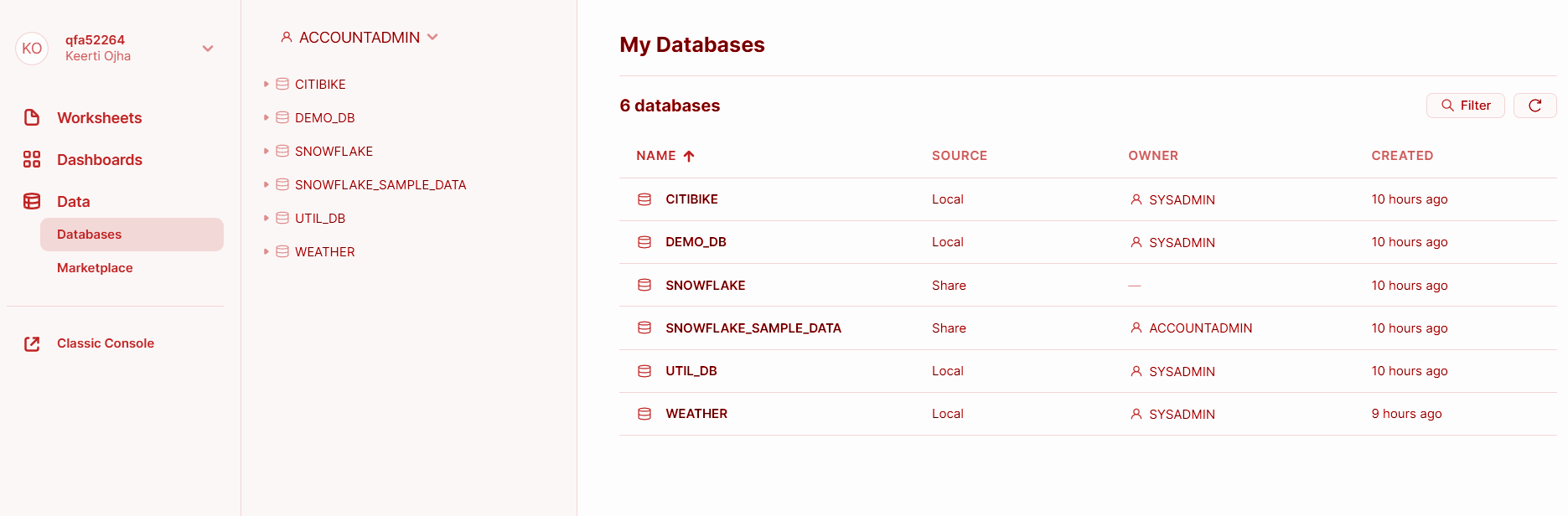


## Role based Access Controls





Preview



# Lessons learned

1. Learned to create stages, databases, tables, views, and warehouses on Snowflake
2. Learned to load structured and semi structured data, querying the tables and cloning the table
3. Also learned to grant privileges to roles and time travelling i.e rolling back the dropped table and updated records.

References

<https://guides.snowflake.com/guide/getting_started_with_snowflake/#11>