



# The Snowflake Data Platform





# Prerequisites

- Comfortable writing SQL queries to analyze data
- Some familiarity with working on the cloud





# Set up for demos

- Please sign up for a free Snowflake account  
<https://signup.snowflake.com/>
- GitHub link for resources  
<https://github.com/janani-ravi-loony/analyzing-processing-data-snowflake>



---

# General Poll

How comfortable are you with SQL?

- Never written queries in SQL before
- Somewhat comfortable writing SQL queries
- Very comfortable writing SQL queries



---

# General Poll

How much experience do you have working on cloud platforms?

- No experience at all
- Somewhat comfortable working on the cloud
- Very comfortable working on the cloud



# The Snowflake Data Platform





# Snowflake

Snowflake is a cloud-native, software-as-a-service advanced data platform for data storage, processing and analytics.



# Snowflake

Snowflake is a **cloud-native, software-as-a-service** advanced data platform for data storage, processing and analytics.



# Data Platform as a Cloud Service

Not hardware to  
setup, install,  
configure, or  
manage

No software to  
install, configure,  
or manage

All administration,  
maintenance,  
upgrades  
managed by  
Snowflake



# Data Platform as a Cloud Service



# Snowflake

- Does not use any existing database technology
  - e.g. MySQL, PostgreSQL, SQL Server
- Is not built on top of any existing big data framework
  - e.g. Hadoop, Spark
- Proprietary big data SQL engine
- Innovative architecture designed for the cloud





# Where Does Snowflake Run?





# Supported Cloud Platforms

- On each platform Snowflake provides one or more regions where the account is provisioned
- Can choose to host all Snowflake accounts on the platform where you run other cloud services





# Snowflake Architecture

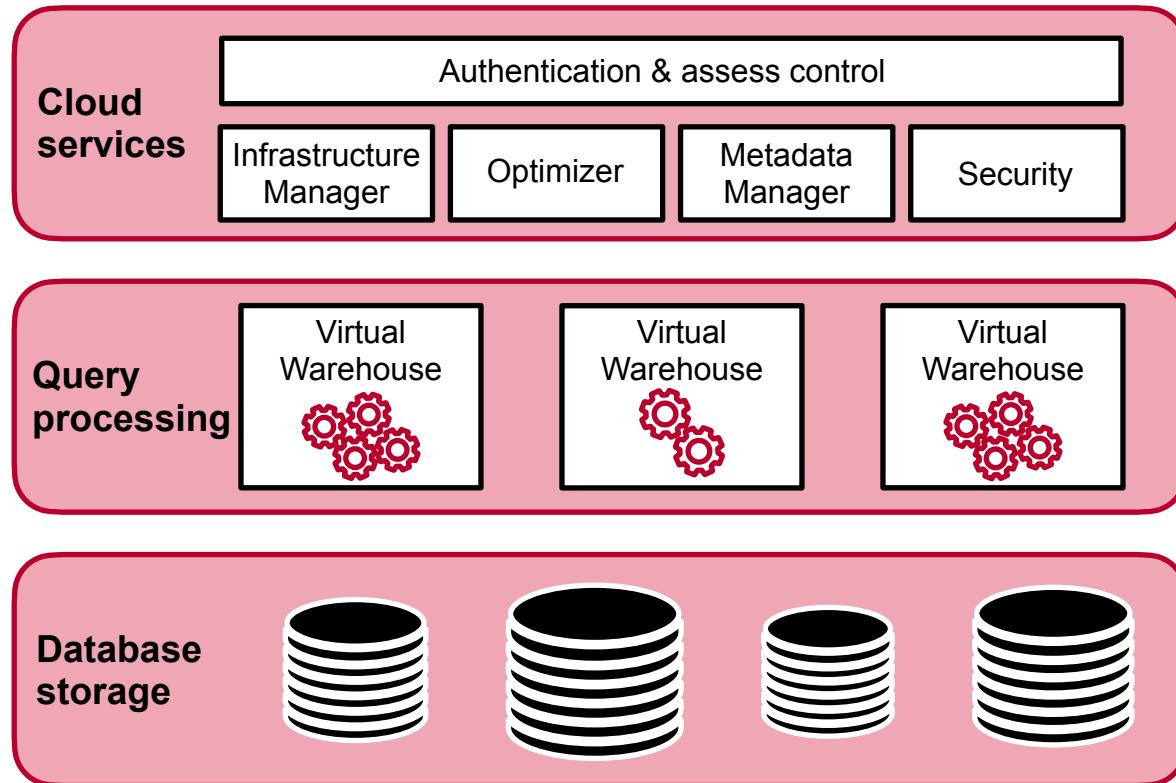
Database Storage

Query Processing

Cloud Services

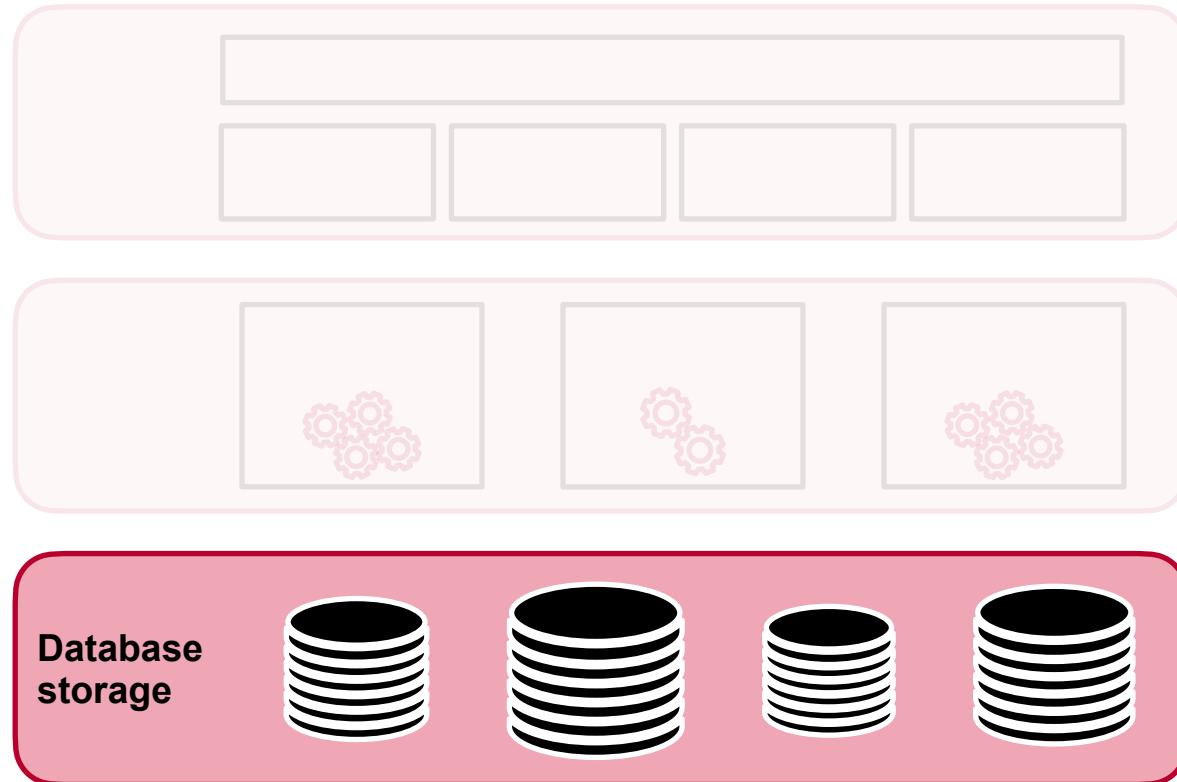


# Snowflake Architecture





# Snowflake Architecture





# Database Storage

- Persistent storage of data loaded into Snowflake
- Proprietary internal format:
  - Columnar, compressed, optimized
- Manages all aspects of data - organization, metadata, statistics, back up
- Snowflake users cannot access or view this data directly
- Data only accessible via SQL queries





# Row Storage

product	quantity	warehouse
dryer	30	warehouse #2
microwave	20	warehouse #1
top load washer	10	NULL
dishwasher	30	warehouse #3
...	...	...

Each row stored together - data of different types laid out one after other in contiguous memory locations



# Row Storage

product	quantity	warehouse
dryer	30	warehouse #2
microwave	20	warehouse #1
top load washer	10	NULL
dishwasher	30	warehouse #3
...	...	...

Efficient for looking up single records, but bad for analytical process since every field has to be read to access a record



# Columnar Storage

product
dryer
microwave
top load washer
dishwasher
...

quantity
30
20
10
30
...

warehouse
warehouse #2
warehouse #1
NULL
warehouse #3
...

Each column stored together - in contiguous memory locations. Makes it easy to compress and query data



# Columnar Storage

product
dryer
microwave
top load washer
dishwasher
...

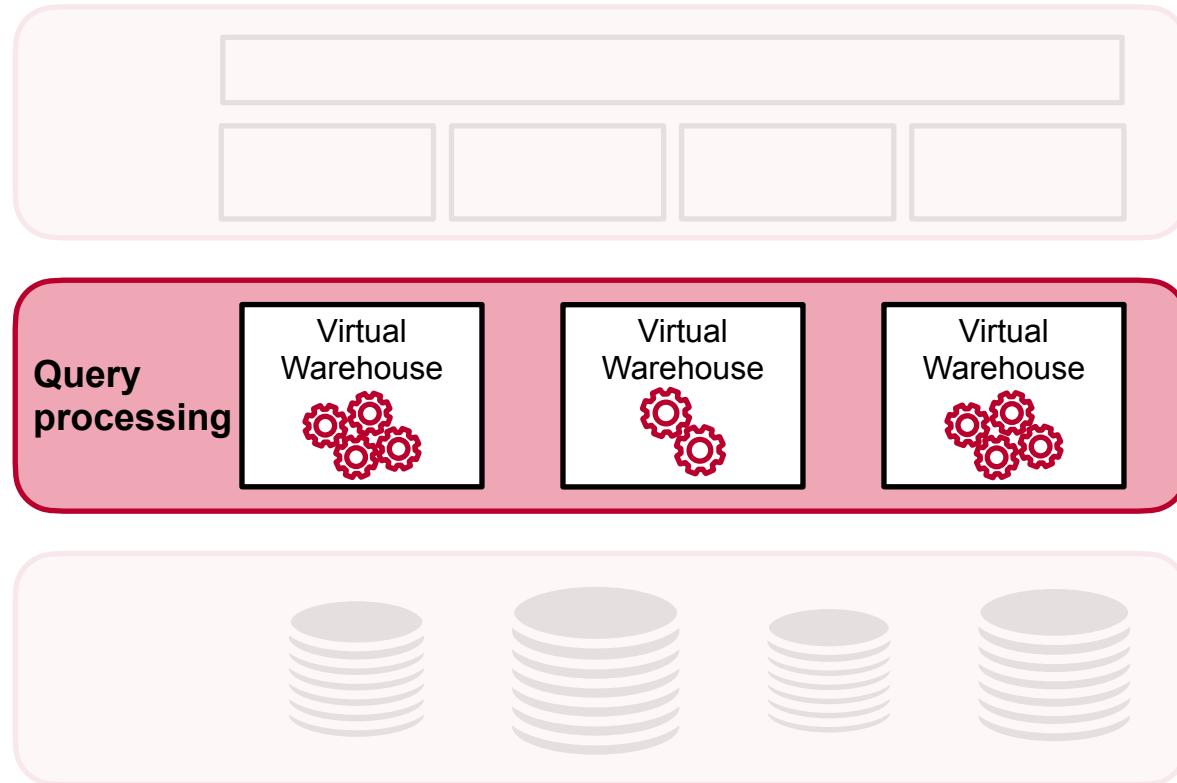
quantity
30
20
10
30
...

warehouse
warehouse #2
warehouse #1
NULL
warehouse #3
...

Very efficient at scanning individual columns over an entire dataset - great for large aggregations on a few fields



# Snowflake Architecture





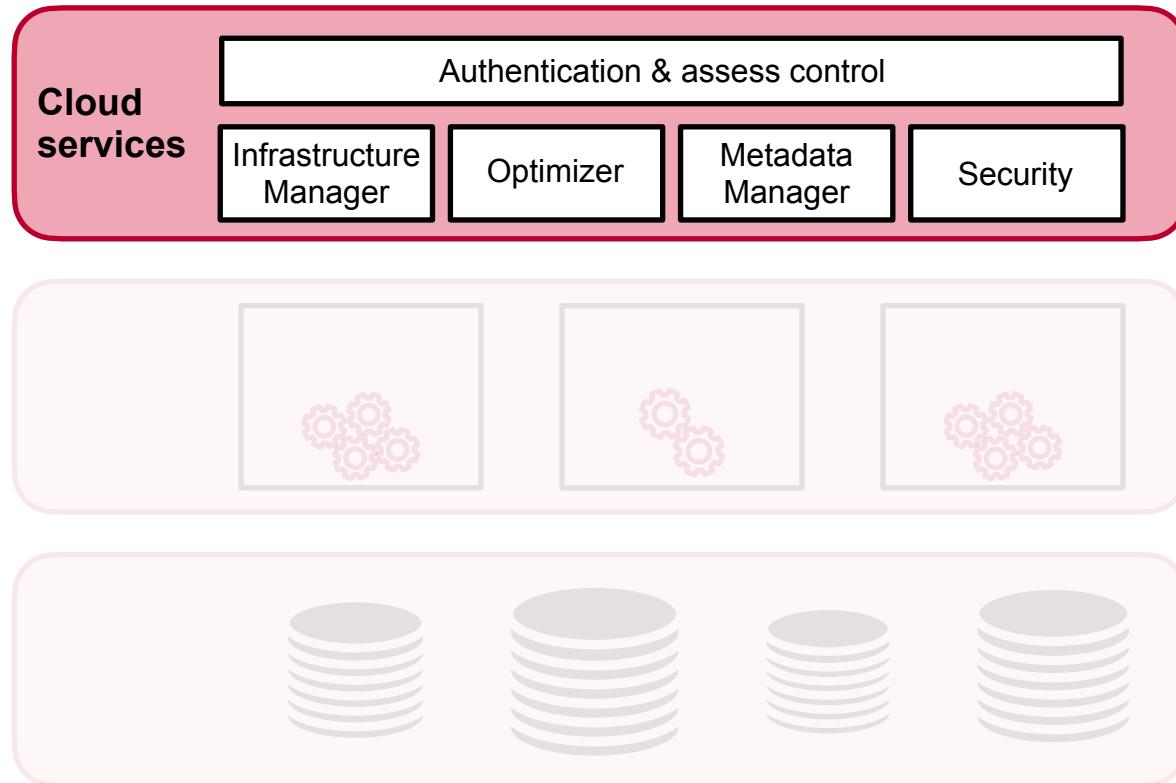
# Query Processing

- Processing layer uses data warehouses
- Each warehouse a massively, parallel cluster
- Warehouses can have one or more compute nodes
- Each warehouse an **independent** compute cluster
- If queries run on different warehouses they don't compete for the same resources





# Snowflake Architecture





# Cloud Services

- Collection of services that coordinate activities across Snowflake
- Services run on compute instances provisioned by Snowflake on the cloud platform





# Working with Snowflake

Web-based user interface

Command-line client  
(SnowSQL)

ODBC and JDBC drivers

Native connectors for  
Python, Spark etc.

Third-party connectors  
for BI, ETL tools



# Snowflake Editions

Standard

Enterprise

Business Critical

Virtual Private  
Snowflake



# Standard Edition

- Introductory offering, standard features
- Data encryption
- Object-level access control
- Time-travel up to 1 day
- Disaster recovery with fail-safe up to 7 days
- **No multi-cluster warehouses, may not meet regulatory needs (government), no private connectivity**





# Enterprise Edition

- Includes everything in the Standard edition
- Extended time travel up to 90 days
- Additional security configurations
- **May not meet regulatory needs  
(government), no private connectivity**





# Business Critical Edition

- Includes everything in the Enterprise edition
- Higher levels of data protection
- Database failover/failback for business continuity
- No dedicated metadata store and pool of compute resources





# Virtual Private Snowflake

- Includes everything in the Business Critical edition
- Highest level of security for organizations with strictest requirements
- Financial institutions, government
- Snowflake environment **completely isolated** from other Snowflake accounts - **no shared resources**





# Snowflake Billing Model

Storage

Compute

Cloud Services

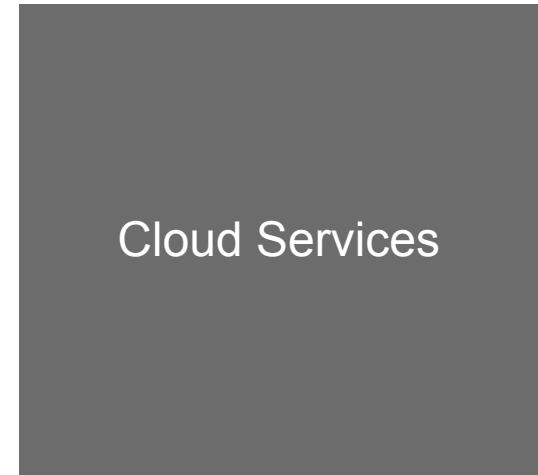
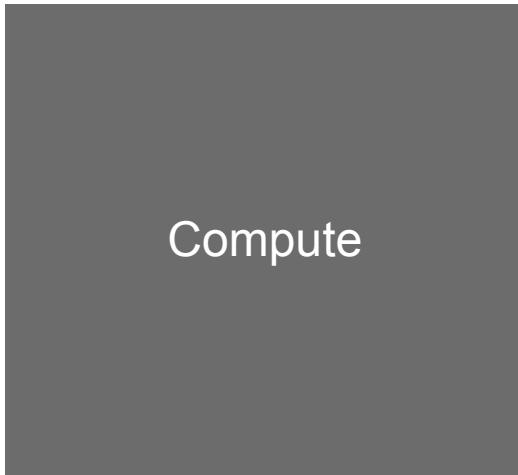
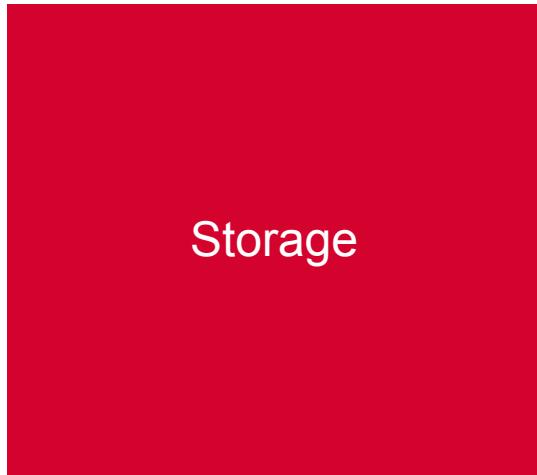


# Snowflake Credit

A unit of measure to pay for the consumption of resources on Snowflake. Credit is used only when the customer is using resources.



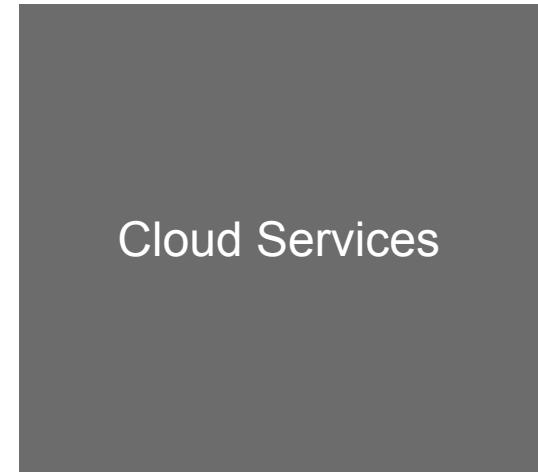
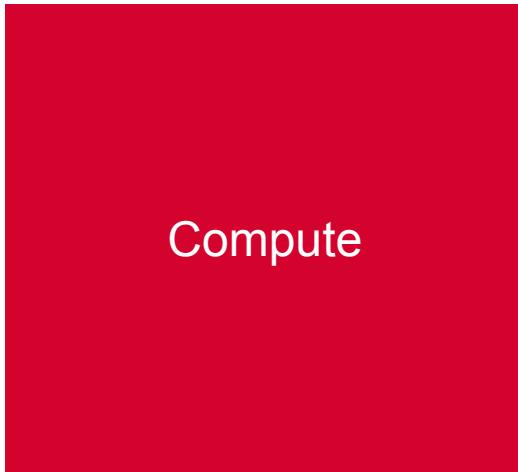
# Rate Per Terabyte (Data + Backups)



Computed using average storage used  
per month after compression



# Virtual Warehouses



Storage

Compute

Cloud Services

Compute needed to load data and  
perform queries



# Management and Coordination

Storage

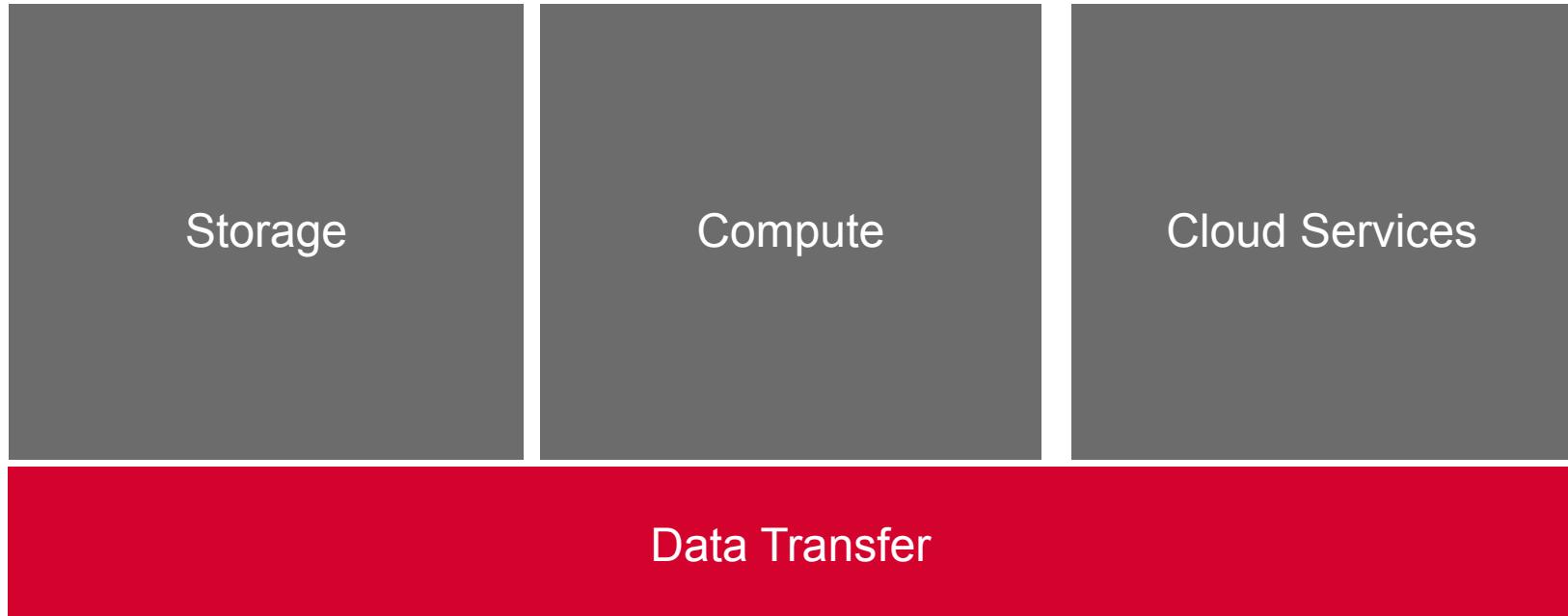
Compute

Cloud Services

Uses additional compute



# Cloud Provider Charge for Moving Data in and Out



---

# Poll 1

Which of the following Snowflake architecture components is responsible for query processing operations?

- Processing engine
- Virtual Warehouse
- Cloud Services
- Data Storage



---

# Poll 1

Which of the following Snowflake architecture components is responsible for query processing operations?

- Processing engine
- Virtual Warehouse
- Cloud Services
- Data Storage



---

# Poll 2

Which Snowflake edition offers a dedicated pool of compute resources so your Snowflake account can be entirely isolated from other accounts?

- Standard
- Enterprise
- Business Critical
- Virtual Private Snowflake



---

## Poll 2

Which Snowflake edition offers a dedicated pool of compute resources so your Snowflake account can be entirely isolated from other accounts?

- Standard
- Enterprise
- Business Critical
- Virtual Private Snowflake



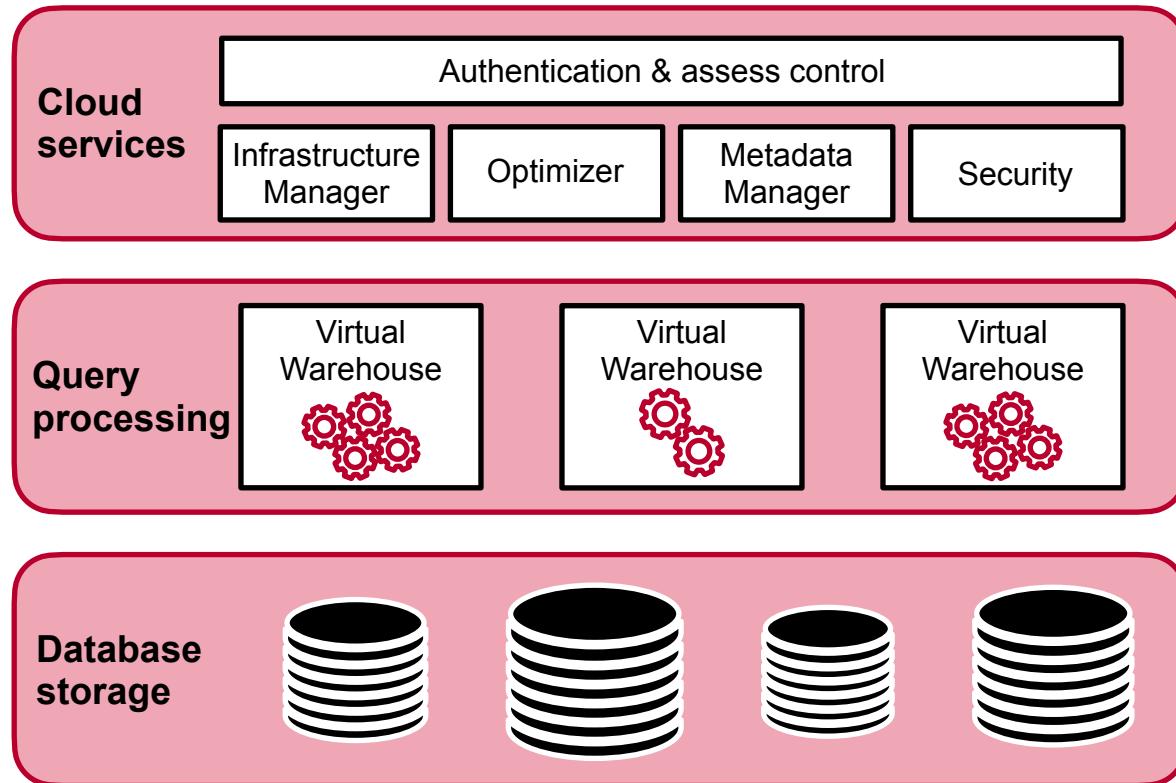
O'REILLY®

# Virtual Warehouses



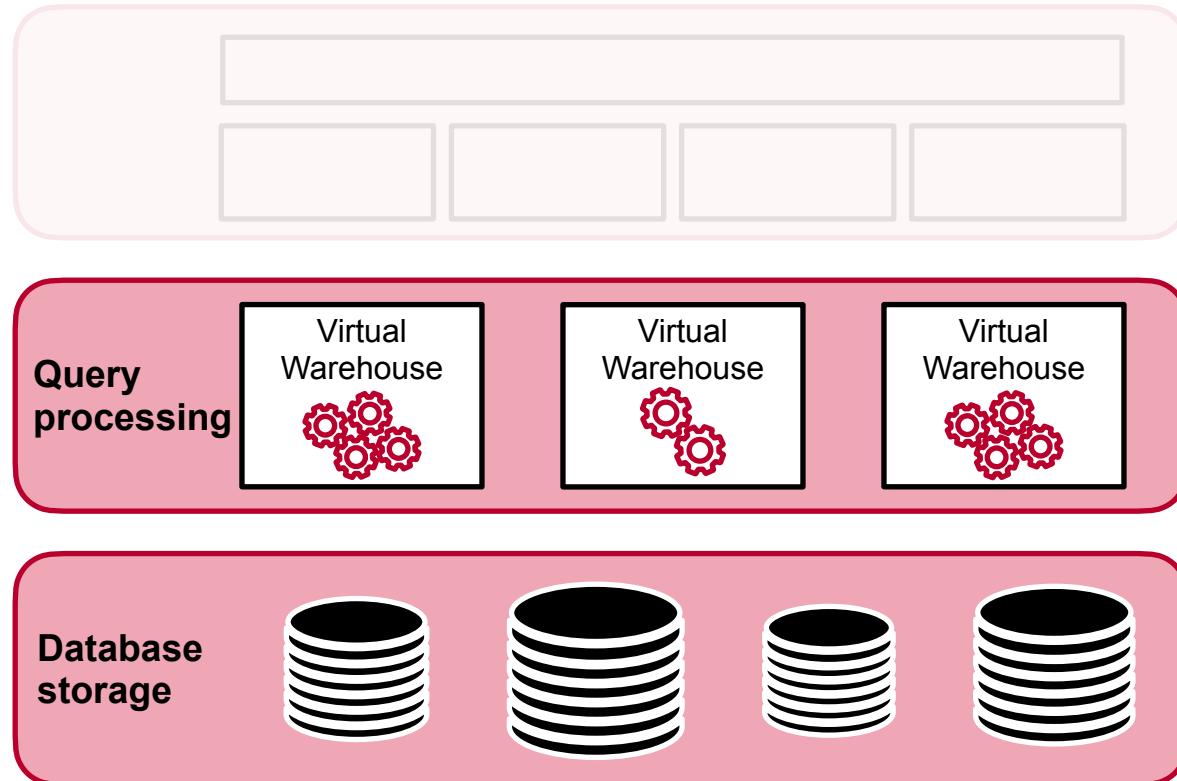


# Snowflake Architecture



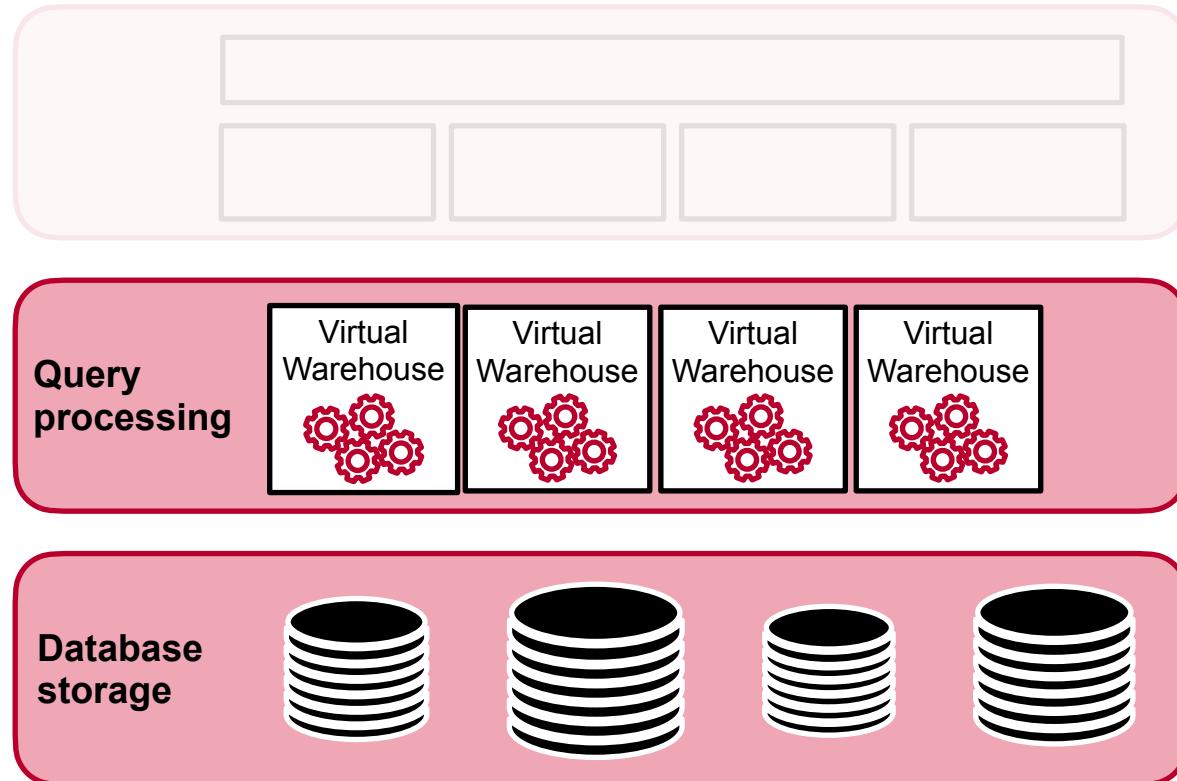


# Independent Storage and Compute



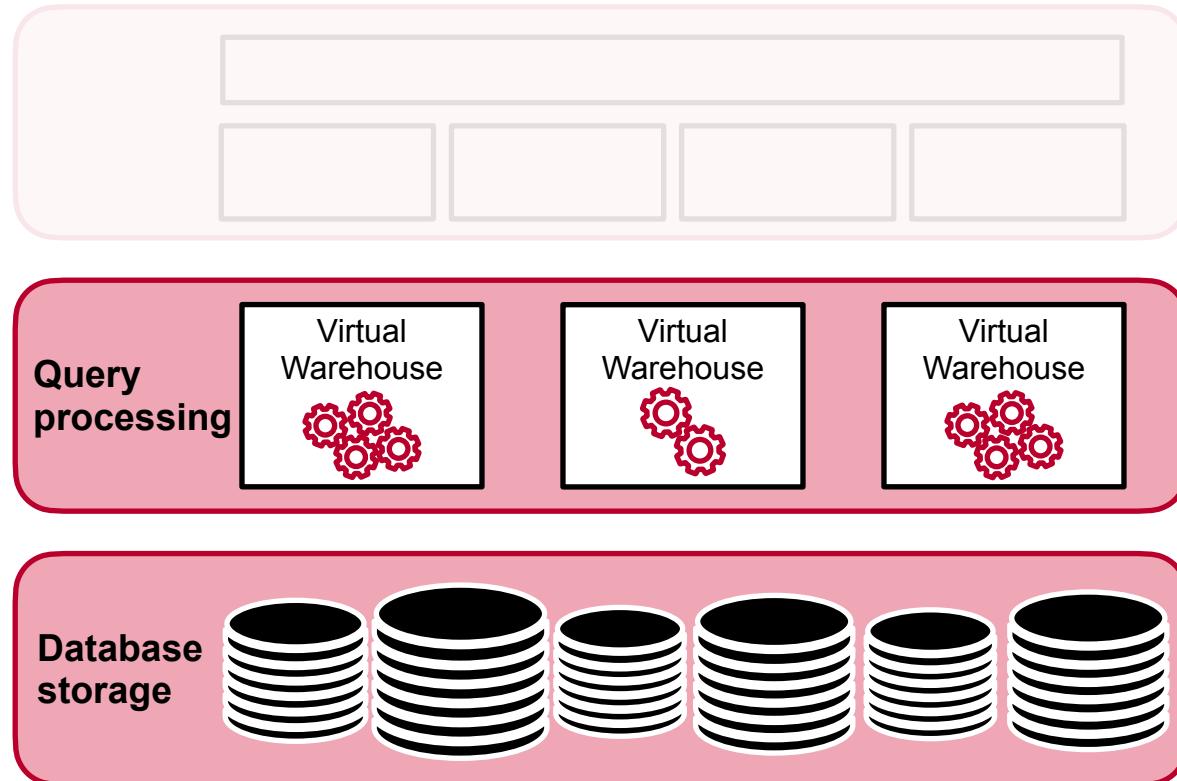


# Scale Compute





# Scale Storage





# Virtual Warehouse

Cluster of compute resources, such as CPU, memory, and temporary storage to perform operations in Snowflake.



# Virtual Warehouses



Standard



Snowpark-optimized

Snowpark-optimized warehouse recommended for workloads with large memory requirements



# Snowpark

Allows developers to write code in Java, Python, or Scala with custom support to build pipelines, ML workflows, and other data applications.



# Virtual Warehouse Operations

Retrieving rows from  
tables and views

Updating rows in  
tables

Loading data into  
tables

Unloading data from  
tables



# Virtual Warehouse Sizes

Warehouse Size	Credits / Hour	Credits / Second	Notes
X-Small	1	0.0003	Default size for warehouses created in Snowsight and using CREATE WAREHOUSE
Small	2	0.0006	
Medium	4	0.0011	
Large	8	0.0022	
X-Large	16	0.0044	Default size for warehouses created using the Classic Console
2X-Large	32	0.0089	
3X-Large	64	0.0178	
4X-Large	128	0.0356	
5X-Large	256	0.0711	Generally available in Amazon Web Services (AWS) and Microsoft Azure regions, and in preview in US Government regions
6X-Large	512	0.1422	Generally available in Amazon Web Services (AWS) and Microsoft Azure regions, and in preview in US Government regions



# Gen1 and Gen2 Warehouses

- **Gen2** is the next generation of standard warehouses (not Snowpark optimized)
- Built on faster hardware and has better optimizations
- Currently only available for some regions and some cloud providers





# Virtual Warehouse

- Size and usage of warehouse impacts **billing**
- **Data loading** performance may not depend on warehouse size
  - Also impacted by **size and number of files**
- Query processing time directly impacted by warehouse size
  - **Larger warehouses = more compute resources = faster queries** (especially complex queries)





# Virtual Warehouse

- Automatically suspended or resumed
- Warehouses consume credits only when they are running (not in suspended state)
- Query **concurrency** determined by size and complexity of queries
- Query is **queued** to be processed if the warehouse does not have sufficient resources





# Multi-cluster Warehouses

Snowflake can statically or dynamically allocate additional clusters\* to make more compute resources available.

\*cluster refers to the compute resources made available to the warehouse for executing queries



# Multi-cluster Warehouses

- Snowflake automatically starts and stops additional clusters as needed to handle **fluctuating workloads**
- Enables a large number of users to connect to the same warehouse





# Multi-cluster Warehouses

Improve  
concurrency for  
users and queries

Do NOT improve  
the performance of  
slow running  
queries





# Needs a Larger Warehouse

Improve  
concurrency for  
users and queries

Do NOT improve  
the performance of  
slow running  
queries



# Warehouse Considerations

- Charges for warehouses depend on:
  - Warehouse size
  - Number of clusters (for multi-cluster warehouses)
  - Time the warehouse is running (and how many clusters in the warehouse are running)





# Warehouse Considerations

- Compute resources for a query depends on query size and complexity
- Queries generally scale linearly w.r.t warehouse size
- Size of the tables being queried has more impact than number of rows
- Query filtering, joins, also impact processing





**Best practice: Use separate warehouses for  
data load operations vs. compute operations**



# Data Loading Considerations

- Load multiple files with your data - each file between 100-250MB compressed
- Loading very large files 100GB or larger is not recommended
- Loading large datasets can impact query performance
- Separate warehouses for data loading and compute
- Data loading requires **minimal resources** so warehouses can be small



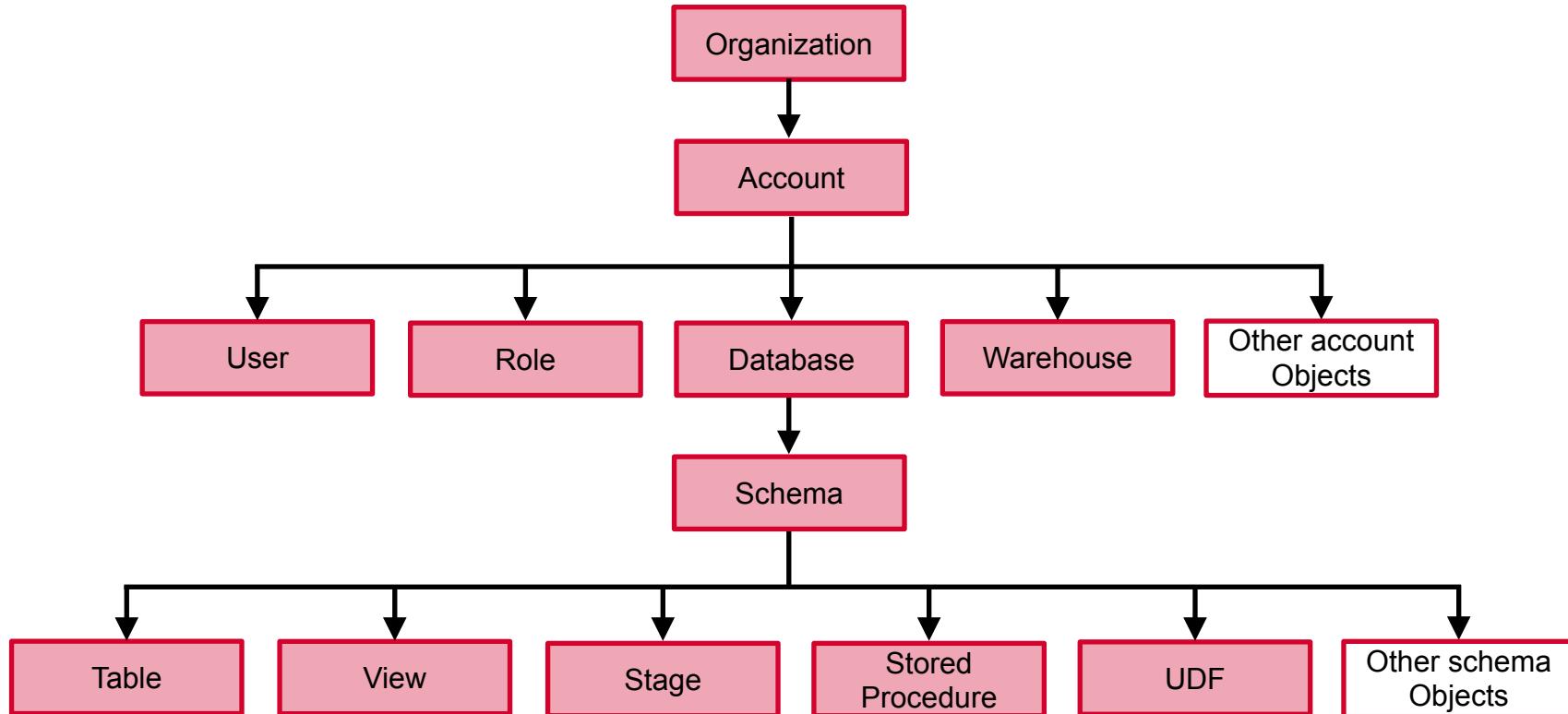
O'REILLY®

# Snowflake Objects



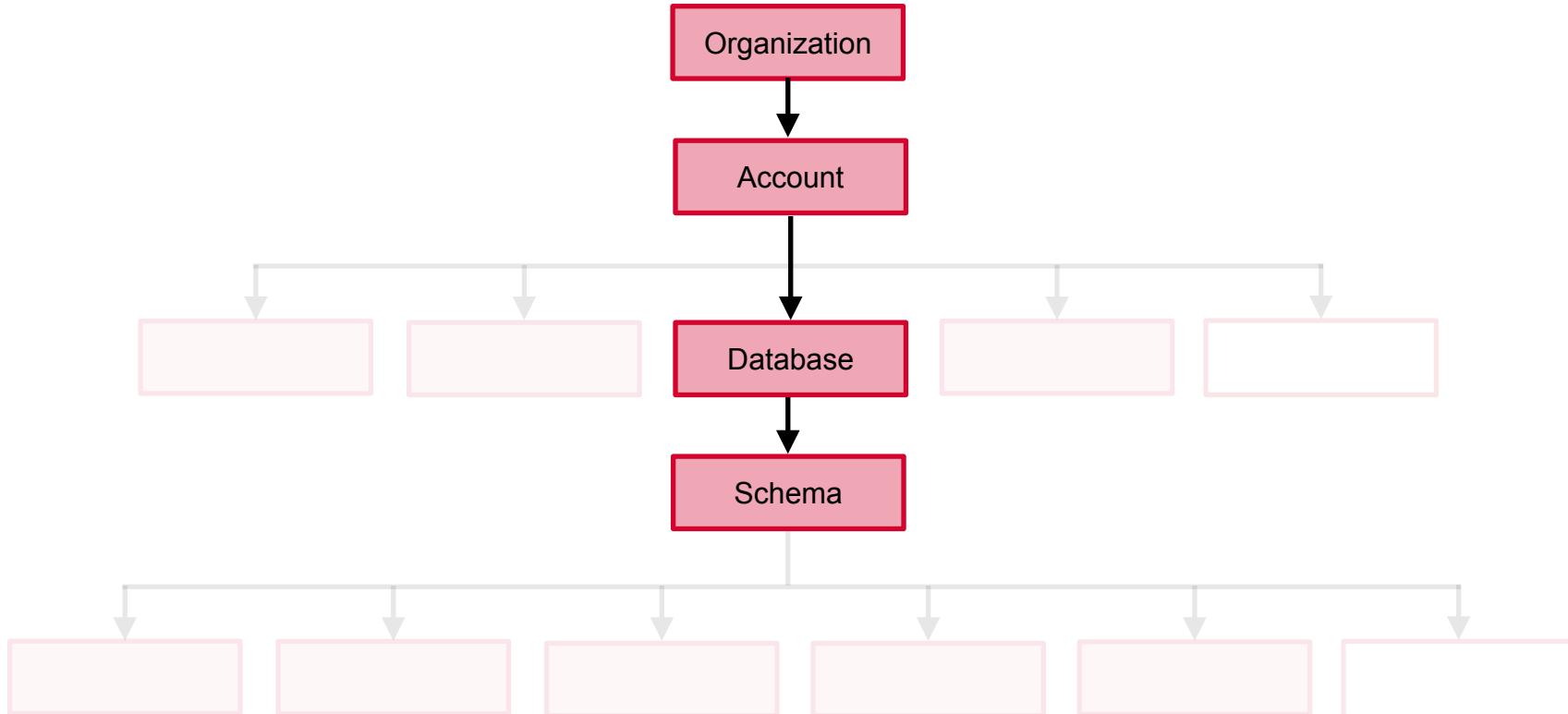


# Snowflake Hierarchy





# Snowflake Hierarchy





# Database

Logical collection of schemas, each schema belongs to a single database.

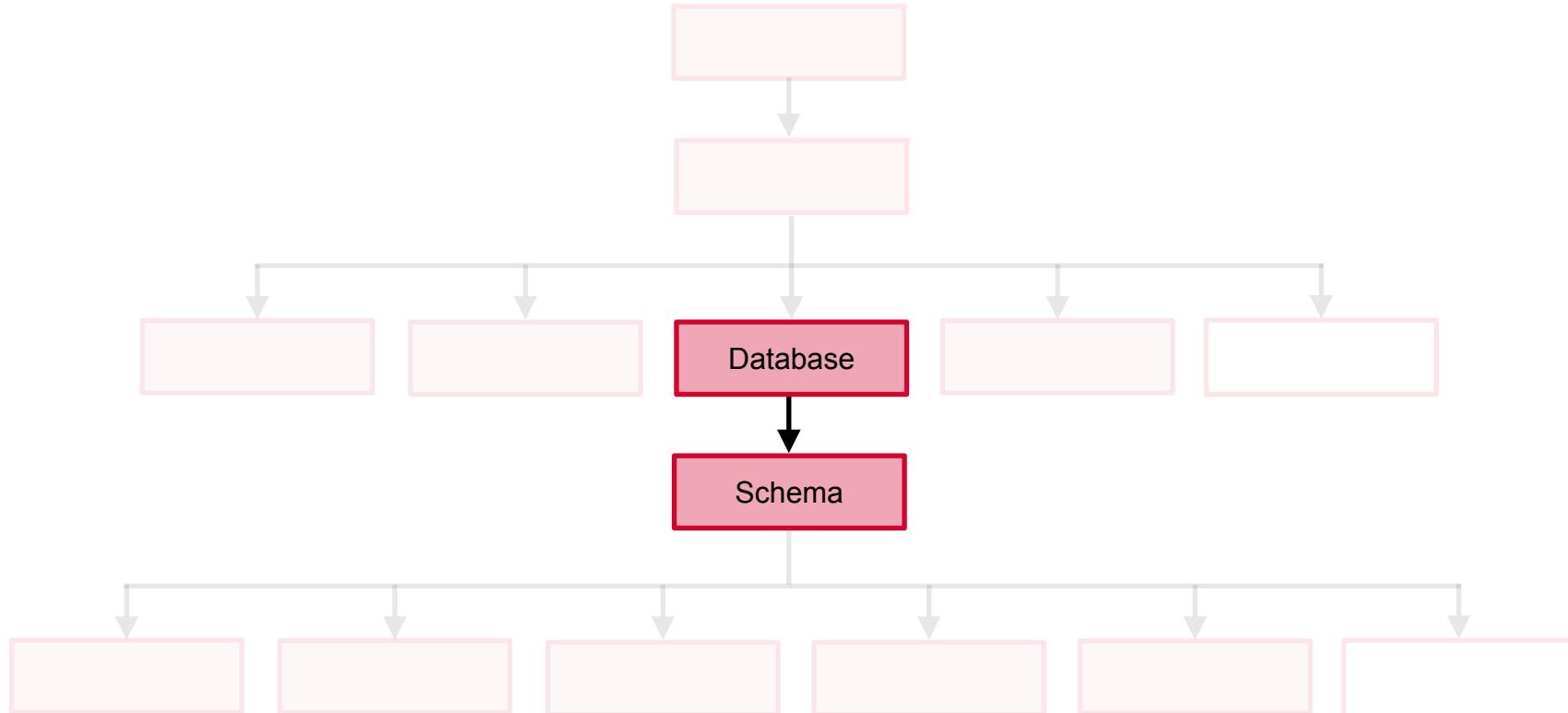


# Schema

Logical grouping of database objects (tables, views, and other constructs).

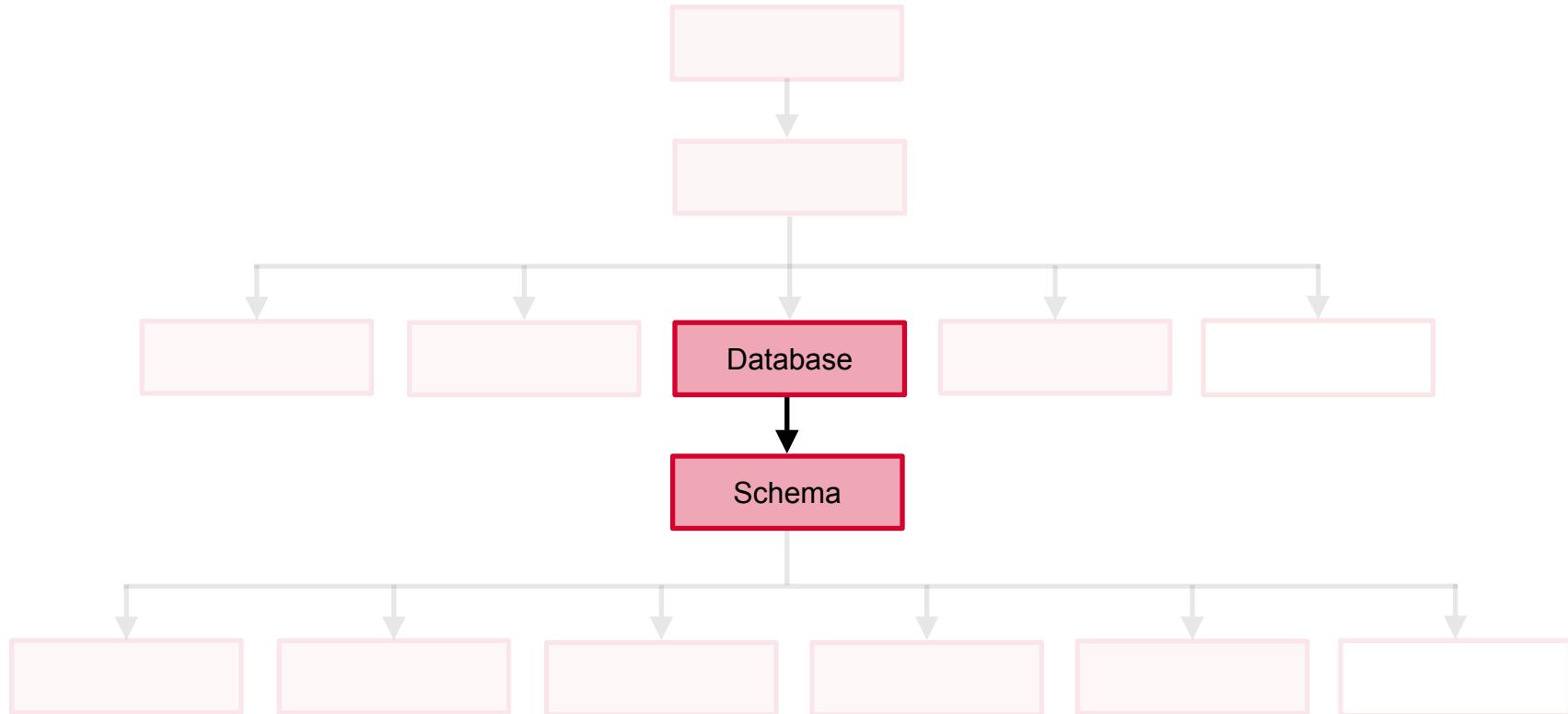


# Namespace = Database + Schema





# Namespace Required for DB Ops





**Hands-on demos:  
Getting Started with  
Snowflake, Databases  
and Warehouses,  
Running Queries**



---

# Poll 3

What characteristic of Snowflake does its three layer architecture enable?

- The ability to specify database tables using a hierarchy
- The ability to extend the SQL query language for complex ops
- The ability to store data in a columnar, compressed format
- The ability to independently scale compute and storage



---

# Poll 3

What characteristic of Snowflake does its three layer architecture enable?

- The ability to specify database tables using a hierarchy
- The ability to extend the SQL query language for complex ops
- The ability to store data in a columnar, compressed format
- The ability to independently scale compute and storage



---

# Poll 4

What does the schema represent in Snowflake?

- Names and data types of columns in your data
- The format of the data loaded into Snowflake
- A logical grouping of database objects
- The path that needs to be specified when accessing tables



---

# Poll 4

What does the schema represent in Snowflake?

- Names and data types of columns in your data
- The format of the data loaded into Snowflake
- A logical grouping of database objects
- The path that needs to be specified when accessing tables



# Table Structures in Snowflake





# Querying Big Data

Several issues when queries are run on tables that are terabytes or petabytes in size:

- Queries run slowly
- No parallelism in query processing
- Entire data needs to be scanned for each query





**Better performance and scaling is obtained using partitioning of tables**



# Partitioning

A unit of management of table data. Large datasets are split into partitions which help parallelize queries and reduce the amount of data scanned.



# Table Structures in Snowflake

Micro-partitioning

Clustering



# Micro-partitioning

- All tables divided into micro-partitions
- **Contiguous units of storage** which contain between 50MB to 500MB of uncompressed data
- Allows granular pruning of very large tables
- Snowflake stores metadata for each micro-partition
  - Range of values for each column
  - Number of distinct values
  - Additional properties for optimization and efficient query processing





# Micro-partitioning Benefits

- Automatic, no manual intervention from the user
- Fine-grained pruning == faster queries
- Uniform size, will not result in skew
- Queries run in parallel on each micro-partition
- Columnar storage (only selected columns in the result will be scanned)





# Clustering

- Data in tables **sorted along natural dimensions** based on columns i.e. clustered
- Clustering metadata recorded for each micro-partitioning when data is loaded into tables
  - Total number of micro-partitions
  - Number of micro-partitions with overlapping values
  - Depth of the overlapping micro-partitions (smaller depth = better clustering)





# Clustering

- Optimize range and equality predicates of queries
- Scan only the micro-partitions which are likely to contain the required data





# Not Clustered

Order_Date	Country	Status
2022-08-02	US	Shipped
2022-08-04	JP	Shipped
2022-08-05	UK	Canceled
2022-08-06	KE	Shipped
2022-08-02	KE	Canceled
2022-08-05	US	Processing
2022-08-04	JP	Processing
2022-08-04	KE	Shipped
2022-08-06	UK	Canceled
2022-08-02	UK	Processing
2022-08-05	JP	Canceled
2022-08-06	UK	Processing
2022-08-05	US	Shipped
2022-08-06	JP	Processing
2022-08-02	KE	Shipped
2022-08-04	US	Shipped



# Clustered Table

Order_Date	Country	Status
2022-08-04	JP	Shipped
2022-08-04		Processing
2022-08-05		Canceled
2022-08-06		Processing
2022-08-06	KE	Shipped
2022-08-02		Canceled
2022-08-04		Shipped
2022-08-02		Shipped
2022-08-05	UK	Processing
2022-08-06		Canceled
2022-08-02		Canceled
2022-08-06		Processing
2022-08-02	US	Shipped
2022-08-05		Processing
2022-08-05		Shipped
2022-08-04		Shipped



# Search Optimization

- Needs to be explicitly enabled for a table
- Applies to all columns in a table
- Creates an additional data structure containing access paths for column values
- Improves performance of **selective point lookup queries**





# Search Optimization

- Relies on a persistent data structure that serves as an optimized search access path
- Maintenance service runs in the background for creating and maintaining the search access path
- When data is updated, the service automatically updates the paths





# Data Storage in Tables

Active Data

Time Travel

Fail-safe



# Data Storage in Tables

Active Data

Time Travel

Fail-safe



# Active Data

Data that can be actively queried. SQL queries, DDL, DML operations are all permitted on this data.



# Data Storage in Tables

Active Data

Time Travel

Fail-safe



# Time Travel

- **Query data from the past** that has since been updated or deleted
- **Create clones** of tables, schemas, and databases at or before specific points in the past
- **Restore** tables, schemas, and databases that have been dropped





# Data Storage in Tables

Active Data

Time Travel

Fail-safe



# Fail-safe

- Non-configurable 7-day period during which historical data may be recoverable by Snowflake
- Period starts immediately after the time travel retention period ends





# Fail-safe

- No user operations allowed
- Snowflake makes best effort to recover data
- Data recovery may take hours or days to complete





# Types of Tables

Permanent

Temporary

Transient



# Types of Tables

Type	Persistence	Time Travel Retention Period	Fail-safe Period
Permanent (other editions)	Until explicitly dropped	0 to 90 (default configurable)	7
Permanent (Standard Edition)	Until explicitly dropped	0 or 1 (default 1)	7
Temporary	Remainder of session	0 or 1 (default 1)	0
Transient	Until explicitly dropped	0 or 1 (default 1)	0



# External Tables

- Reference data files located on cloud storage
  - Amazon S3
  - Azure Data Lake
  - Google Cloud Storage
- Read-only tables, no updates allowed
- Query and join operations allowed



O'REILLY®

Hands-on  
demos: Time  
travel  
SnowSQL



# Snowflake vs. Databricks





Cloud-based data platform powered by Apache Spark for processing and analyzing big data.



# Databricks

- **Databricks SQL**
  - Platform to run SQL queries to analyze data
- **Databricks Data Science and Engineering**
  - Interactive workspace for collaboration and to generate insights using Spark
- **Databricks Machine Learning**
  - Integrated end-to-end machine learning environment with managed services





# Databricks Key Features

Delta Lake for the entire  
data lifecycle

Optimized Spark Engine  
for data processing

Machine Learning  
Environments

Collaborative Notebooks  
for Scala R, SQL, Python



# Snowflake vs. Databricks

## Snowflake

- Managed data platform
- Integrated with the 3 major cloud platforms
- Decoupled storage and processing layers
- Highly scalable

## Databricks

- Managed data platform
- Integrated with the 3 major cloud platforms
- Decoupled storage and processing layers
- Highly scalable



# Snowflake vs. Databricks

## Snowflake

- SQL-based so **easier and intuitive** to use
- Data processing with Snowpark within Snowflake
- Batch-centric but can work with continuous data

## Databricks

- SQL + R, Python, Scala more options to process data
- Spark at the core of the Databricks ecosystem
- Strong **batch and streaming** support (with Spark)

---

# Poll 5

Which of the following is NOT a data storage component in Snowflake?

- Backup
- Fail-safe
- Time travel
- Active or current data



---

# Poll 5

Which of the following is NOT a data storage component in Snowflake?

- Backup
- Fail-safe
- Time travel
- Active or current data



---

# Poll 6

Which of the following types of tables is only available within a single session?

- Permanent
- Temporary
- Transient



---

# Poll 6

Which of the following types of tables is only available within a single session?

- Permanent
- Temporary
- Transient



# Regular, Secure, Materialized Views





# View

A query which produces a result, this result can then be accessed as a table.

A view helps combine, segregate, or protect data.



# Views in Snowflake

Non-materialized  
views

Materialized  
views



# Non-materialized Views

Named definition of a query.

When a view is referenced in another query the view's query is executed. The view's results are not stored for future use.



# Materialized Views

Named definition of a query, however results of the view's query are stored as though they were a table.

When a view is referenced in another query that uses the stored query results of the materialized view.

Queries run faster but the stored results take up additional space.



# Secure Views

Secure

Non-materialized  
views

Materialized  
views



# Secure Views

Views with improved data privacy and sharing.

Does not allow query optimizations that might inadvertently expose data from the underlying table to the user.

---

# Poll 8

Which of the following statements about views is true?

- Queries on a secure view run faster than queries on regular views
- Queries on a materialized view run faster than queries on regular views
- Queries on regular views run faster than queries on materialized views



---

# Poll 8

Which of the following statements about views is true?

- Queries on a secure view run faster than queries on regular views
- Queries on a materialized view run faster than queries on regular views
- Queries on regular views run faster than queries on materialized views



# Hands-on demos: Views



O'REILLY®

# Dynamic Tables





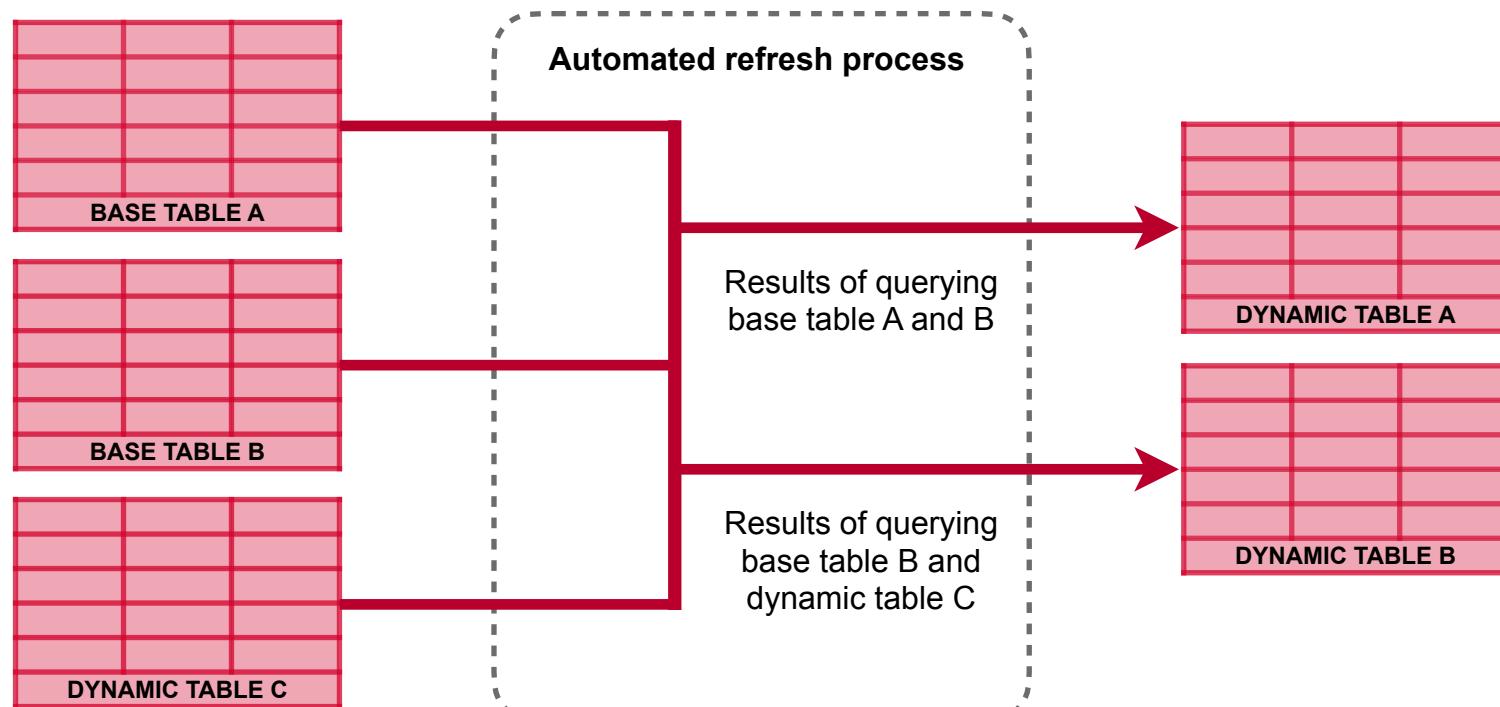
# Dynamic Tables

Dynamic tables are tables that automatically refresh based on a defined query and target freshness.

Makes data transformation and pipeline management simple - does not require any manual intervention.



# Dynamic Tables



Transformation query executed on a regular basis to update the dynamic table



## Ideal For

- Automated management of data dependencies and refresh schedules
- Multiple tables that are part of a data transformation pipeline
- You only need a target freshness for the data in a table - no fine-grained control





# Initialization of Dynamic Tables



On Create



On Schedule

On Create, the table is initialized immediately. If you set On Schedule, initialization happens within the specified target lag timeframe.



# Target Lag

Target lag specifies how outdated the data in a dynamic table can be. Table refresh schedule is determined by the target lag.



# Dynamic Table Refresh Mode

Auto

Incremental

Full



# Dynamic Table Refresh Mode

Auto

Incremental

Full

Incremental refresh by default. When this is not supported or might not perform well, full refresh used

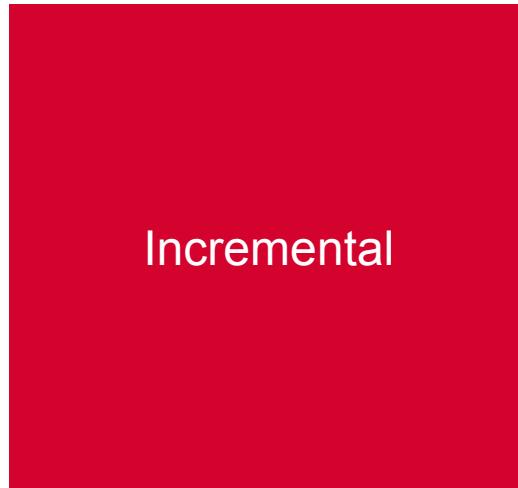
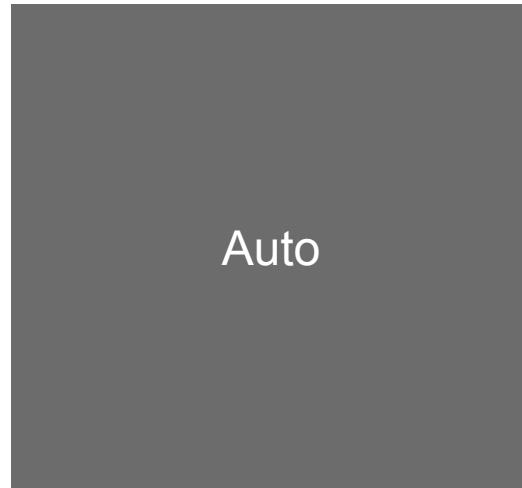


# Dynamic Table Refresh Mode

Auto

Incremental

Full



Updates the dynamic table with only the changes since the last refresh, ideal for large datasets with frequent small updates.



# Dynamic Table Refresh Mode

Auto

Incremental

Full

Reprocesses the entire dataset. Used for complex queries or when significant data changes require a complete update.

# Hands-on demos: Dynamic Tables



O'REILLY®

# Sampling





# Sampling

Return a subset of rows from the specified table

Snowflake allows you to:

- Sample a fraction of the table with a specified probability “p” for including a certain row
- Sample a fixed, specified number of rows



# Sampling in Snowflake

Bernoulli or Row  
sampling

System or Block  
sampling



# Bernoulli Sampling

Samples each row with a certain probability =  $p/100$

Works well for smaller datasets as it reduces the bias of sampled data

Is slower for larger datasets because each row considered and included with a certain probability



# Block Sampling

Samples each block with a certain probability =  $p/100$

May result in biased samples for smaller datasets

Works well for larger dataset, is faster and more efficient

O'REILLY®

# Hands-on demos: Sampling Common Table Expressions



---

# Poll 9

If you have a very large table what kind of sampling technique would you choose for that table for fast and efficient sampling?

- Bernoulli sampling
- Random sampling
- Block sampling



---

# Poll 9

If you have a very large table what kind of sampling technique would you choose for that table for fast and efficient sampling?

- Bernoulli sampling
- Random sampling
- Block sampling



O'REILLY®

# Window Functions





# Window Functions

Operate on windows - which are groups of rows which are related to one another.



# Window Functions

Department	Name	Salary
Engineering	Carrie	7000
Sales	Dora	5000
Engineering	Jackson	6600
Sales	Stewart	6200
Sales	Ruby	4000
Engineering	Sonia	6850
Sales	Ivan	4200
Engineering	Steven	6550



# Highest Salary in Department

Department	Name	Salary
Engineering	Carrie	7000
Sales	Dora	5000
Engineering	Jackson	6600
Sales	Stewart	6200
Sales	Ruby	4000
Engineering	Sonia	6850
Sales	Ivan	4200
Engineering	Steven	6550



# Partition by Department

Department	Name	Salary
Engineering	Carrie	7000
Sales	Dora	5000
Engineering	Jackson	6600
Sales	Stewart	6200
Sales	Ruby	4000
Engineering	Sonia	6850
Sales	Ivan	4200
Engineering	Steven	6550



# Partition by Department

Department	Name	Salary
Engineering	Carrie	7000
	Jackson	6600
	Sonia	6850
	Steven	6550
Sales	Dora	5000
	Stewart	6200
	Ruby	4000
	Ivan	4200



# Order by Salary DESC

Department	Name	Salary
Engineering	Carrie	7000
Engineering	Jackson	6600
Engineering	Sonia	6850
Engineering	Steven	6550
Sales	Dora	5000
Sales	Stewart	6200
Sales	Ruby	4000
Sales	Ivan	4200



# Order by Salary DESC

Department	Name	Salary
Engineering	Carrie	7000
Engineering	Sonia	6850
Engineering	Jackson	6600
Engineering	Steven	6550
Sales	Stewart	6200
Sales	Dora	5000
Sales	Ivan	4200
Sales	Ruby	4000



# Highest Salary in Department

Department	Name	Salary
Engineering	Carrie	7000
Engineering	Sonia	6850
Engineering	Jackson	6600
Engineering	Steven	6550
Sales	Stewart	6200
Sales	Dora	5000
Sales	Ivan	4200
Sales	Ruby	4000



# Highest Salary in Department

Department	Name	Salary
Engineering	Carrie	7000
Engineering	Sonia	6850
Engineering	Jackson	6600
Engineering	Steven	6550
Sales	Stewart	6200
Sales	Dora	5000
Sales	Ivan	4200
Sales	Ruby	4000



# Partition By Department Order by Salary

Department	Name	Salary
Engineering	Carrie	7000
	Sonia	6850
	Jackson	6600
	Steven	6550
Sales	Stewart	6200
	Dora	5000
	Ivan	4200
	Ruby	4000



# Partition By Department Order by Salary



Department	Name	Salary
Engineering	Carrie	7000
Engineering	Sonia	6850
Engineering	Jackson	6600
Engineering	Steven	6550
Sales	Stewart	6200
Sales	Dora	5000
Sales	Ivan	4200
Sales	Ruby	4000

ROWS BETWEEN UNBOUNDED PRECEDING AND CURRENT ROW



# Partition By Department Order by Salary



Department	Name	Salary
Engineering	Carrie	7000
Engineering	Sonia	6850
Engineering	Jackson	6600
Engineering	Steven	6550
Sales	Stewart	6200
Sales	Dora	5000
Sales	Ivan	4200
Sales	Ruby	4000

ROWS BETWEEN UNBOUNDED PRECEDING AND CURRENT ROW



# Partition By Department Order by Salary

Department	Name	Salary
Engineering	Carrie	7000
Engineering	Sonia	6850
Engineering	Jackson	6600
Engineering	Steven	6550
Sales	Stewart	6200
Sales	Dora	5000
Sales	Ivan	4200
Sales	Ruby	4000

ROWS BETWEEN UNBOUNDED PRECEDING AND CURRENT ROW



# Partition By Department Order by Salary

Department	Name	Salary
Engineering	Carrie	7000
	Sonia	6850
	Jackson	6600
	Steven	6550
Sales	Stewart	6200
Sales	Dora	5000
Sales	Ivan	4200
Sales	Ruby	4000

ROWS BETWEEN UNBOUNDED PRECEDING AND CURRENT ROW



# Partition By Department Order by Salary



Department	Name	Salary
Engineering	Carrie	7000
Engineering	Sonia	6850
Engineering	Jackson	6600
Engineering	Steven	6550
Sales	Stewart	6200
Sales	Dora	5000
Sales	Ivan	4200
Sales	Ruby	4000

ROWS BETWEEN CURRENT ROW AND UNBOUNDED FOLLOWING



# Partition By Department Order by Salary

Department	Name	Salary
Engineering	Carrie	7000
Engineering	Sonia	6850
Engineering	Jackson	6600
Engineering	Steven	6550
Sales	Stewart	6200
Sales	Dora	5000
Sales	Ivan	4200
Sales	Ruby	4000

ROWS BETWEEN CURRENT ROW AND UNBOUNDED FOLLOWING



# Partition By Department Order by Salary

Department	Name	Salary
Engineering	Carrie	7000
Engineering	Sonia	6850
Engineering	Jackson	6600
Engineering	Steven	6550
Sales	Stewart	6200
Sales	Dora	5000
Sales	Ivan	4200
Sales	Ruby	4000

ROWS BETWEEN CURRENT ROW AND UNBOUNDED FOLLOWING



# Partition By Department Order by Salary

Department	Name	Salary
Engineering	Carrie	7000
Engineering	Sonia	6850
Engineering	Jackson	6600
Engineering	Steven	6550
Sales	Stewart	6200
Sales	Dora	5000
Sales	Ivan	4200
Sales	Ruby	4000

ROWS BETWEEN CURRENT ROW AND UNBOUNDED FOLLOWING



# Partition By Department Order by Salary

Department	Name	Salary
Engineering	Carrie	7000
Engineering	Sonia	6850
Engineering	Jackson	6600
Engineering	Steven	6550
Sales	Stewart	6200
Sales	Dora	5000
Sales	Ivan	4200
Sales	Ruby	4000

ROWS BETWEEN 1 PRECEDING AND 1 FOLLOWING



# Partition By Department Order by Salary



Department	Name	Salary
Engineering	Carrie	7000
Engineering	Sonia	6850
Engineering	Jackson	6600
Engineering	Steven	6550
Sales	Stewart	6200
Sales	Dora	5000
Sales	Ivan	4200
Sales	Ruby	4000

ROWS BETWEEN 1 PRECEDING AND 1 FOLLOWING



# Partition By Department Order by Salary

Department	Name	Salary
Engineering	Carrie	7000
Engineering	Sonia	6850
Engineering	Jackson	6600
Engineering	Steven	6550
Sales	Stewart	6200
Sales	Dora	5000
Sales	Ivan	4200
Sales	Ruby	4000

ROWS BETWEEN 1 PRECEDING AND 1 FOLLOWING



# Partition By Department Order by Salary

Department	Name	Salary
Engineering	Carrie	7000
Engineering	Sonia	6850
Engineering	Jackson	6600
Engineering	Steven	6550
Sales	Stewart	6200
Sales	Dora	5000
Sales	Ivan	4200
Sales	Ruby	4000



ROWS BETWEEN 1 PRECEDING AND 1 FOLLOWING

# Hands-on demos: Window functions



O'REILLY®

# Data Loading in Snowflake





# Stage

Refers to the location of data files in cloud storage as a stage.



# Types of Stages

External Stage

Internal Stage



# External Stage

Cloud storage services that belong to the cloud platform that you are using:

- Amazon S3
- Azure Storage
- Google Cloud Storage



# Internal Stages

Cloud storage that belongs to your Snowflake account



# Internal Stages

User Stage

Table Stage

Named Stage



# User Stage

- Stage associated with a user
- Data in stage can be loaded into multiple tables





# Table Stage

- Implicit stage associated with a table
- Stage can be managed by one or more users
- Data in this stage can only be loaded into the corresponding table





# Named Stage

- A database object created in a schema
- Used and managed by one or more users
- Data in stage can be loaded into one or more tables



**Hands-on demos:**  
**Bulk Loading**  
**External stage**  
**Semi-structured**  
**Data**



---

# Poll 10

If you wanted the data loaded in your stage to be managed by multiple users and loaded into any number of tables, what stage would you use?

- Table stage
- User stage
- Named stage



---

# Poll 10

If you wanted the data loaded in your stage to be managed by multiple users and loaded into any number of tables, what stage would you use?

- Table stage
- User stage
- Named stage

