SQOOP

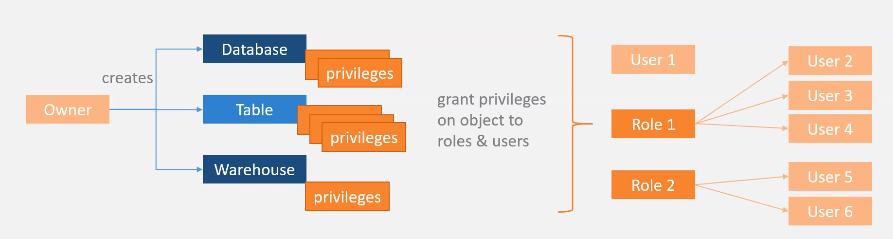
1. What is sqoop?
2. What are different sqoop commands?
3. What is the difference between target directory and warehouse directory arguments in sqoop import?
4. How to redirect output and log messages from console to files?
5. What are the different file formats and compression techniques in sqoop import?
6. Explain the arguments --columns, --where, --table in sqoop import?
7. What are free form query imports?
8. Is sqoop import a map-reduce job? Pls explain number of mappers argument in sqoop import.
   1. number of mappers
   2. split by
   3. controlling parallelism
   4. bounding vals query
9. How to format output in sqoop import?
10. How to append to existing data in sqoop import?
11. How to delete existing directory during sqoop import?
12. How to deal with nulls during sqoop import?
13. Explain incremental import in sqoop import.
14. What are the different ways of handling password in sqoop.
15. Explain sqoop import execution flow.
16. What is the concept of staging table in sqoop export and why it is needed?
17. Explain sqoop job.
18. How you schedule sqoop job in your project?

HIVE

1. What is hive? How it places itself in Hadoop ecosystem?
2. Why we use hive in place of MapReduce?
3. Can you explain hive internal architecture in detail?
   1. https://cwiki.apache.org/confluence/display/Hive/Design
4. What is thrift server / hive server?
5. Difference between hive and rdbms.
6. Difference between hql and sql.
7. What is the concept of a table in hive?
8. Why metadata is stored in a rdbms?
9. What are the different types of tables in hive? And what is the difference between them?
   1. managed
   2. external
   3. temporary
10. What are the 3 different ways to insert data into a hive table?
    1. insert command using values
    2. load command (from local (copy paste), from hdfs (cut paste))
    3. insert command using queries (one table to another table)
11. What are the different ways to connect to hive?
    1. hive shell
    2. hue ui
    3. beeline
12. How you run hive queries in production?
    1. beeline with hive.hql file containing hive queries
13. What are different data types in hive?
    1. https://cwiki.apache.org/confluence/display/Hive/LanguageManual+Types
14. What are hive subqueries?
    1. https://cwiki.apache.org/confluence/display/Hive/LanguageManual+SubQueries
15. What are hive views? Why we use views?
16. What are different set operations in hive?
    1. https://cwiki.apache.org/confluence/display/Hive/LanguageManual+Union
17. What are the different types of functions in hive?
    1. https://cwiki.apache.org/confluence/display/Hive/LanguageManual%20UDF
18. What is lateral view and how we use it?
    1. <https://spark.apache.org/docs/latest/sql-ref-syntax-qry-select-lateral-view.html>
    2. <https://cwiki.apache.org/confluence/display/Hive/LanguageManual+LateralView>
19. How to create temporary as well as permanent custom UDFs in hive?
    1. <https://cwiki.apache.org/confluence/display/Hive/HivePlugins>
20. What is the difference between order by, sort by, distribute by, cluster by?
21. What is the difference between normalization and de-normalization and what is preferred in hive?
22. What is msck repair command and why do we use it?
23. How will you skip header rows in hive?
24. How and why we make a table immutable?
25. Difference between drop, truncate, purge, delete.
26. What is the use of the configuration serilization.null.format?
    1. <https://stackoverflow.com/questions/43263038/empty-string-is-not-treated-as-null-in-hive>
27. What are hive variables? Why do we use it?
28. How can we print headers along with data in hive?
29. What if while inserting data, data types in table definition and of actual data didn’t match?
30. --------------------------------------------------------------------------------------------------------
31. How and why do you integrate hive with Spark?
32. Why we need ACID properties in hive if hive is used for analytical processing?
33. What are some of the important points that you need to remember while dealing with ACID properties in hive? What properties need to be set to enable ACID in hive?
    1. <https://medium.com/@randy-huang/abc-b7cbaf37e4f7>
    2. <https://sparkbyexamples.com/apache-hive/hive-enable-and-use-acid-transactions/>
    3. <https://blog.clairvoyantsoft.com/hive-acid-transactions-part-i-f08da70b591b>
    4. https://blog.clairvoyantsoft.com/hive-acid-transactions-part-ii-6f5e478bbff
34. How ACID property is implemented in hive?
35. What is compaction? Why it is needed?
36. Describe some of the properties of insert-only transactional tables.
37. Is it possible to convert a non-acid table to acid table & vice-versa? How will you do that?

SNOWFLAKE

1. What is snowflake?
2. What are some of the properties of snowflake?
3. What is the difference between a traditional data warehouse and snowflake?
4. Can you explain in detail about snowflake architecture?
5. What are the different snowflake editions? Explain the features of each one of them.
6. What are snowflake credits?
7. On what factors the cost is dependent on in snowflake?
8. What the different types of cost in snowflake?
   1. storage cost
   2. compute cost
   3. cloud services cost
9. What is storage cost?
10. How will you choose which storage plan suits you the best?
11. What is compute cost?
12. What is cloud services cost?
13. What are the important concepts in snowflake?
    1. virtual warehouse
    2. storage (micro-partitioning, clustering)
    3. storage integration object
    4. file format object
    5. external stage
    6. internal stage
    7. copy
    8. snowpipe
    9. unloading
    10. semi-structured data
    11. caching
    12. time travel and fail-safe
    13. zero-copy cloning
    14. table types
    15. external tables
    16. access control
    17. views and materialized views
    18. column level security
    19. data sharing
    20. tasks
    21. streams
    22. data sampling
    23. UDFs
    24. stored procedures
    25. alerts
    26. notifications
    27. deployment
    28. resource monitor
14. ---
15. What is a virtual warehouse?
16. How will you choose a virtual warehouse?
    1. requirement specific
    2. environment specific
17. What are the different sizes in which virtual warehouses come?
18. What will happen if a virtual warehouse is not enough to handle the queries / data being processed?
19. What are the different ways to increase the performance of a virtual warehouse?
    1. scale up (increasing the size)
    2. scale out (increasing the number of clusters)
20. What is the auto suspend feature?
21. What is the auto resume feature?
22. ---
23. How is data stored in snowflake?
    1. columnar format
    2. efficient compression
    3. micro-partitioning
    4. clustering
24. What is the concept of micro-partitioning?
25. What are the benefits of micro-partitioning?
26. What is clustering?
27. What are the benefits of clustering?
28. How will you choose a column based on which clustering will be done?
29. ---
30. What is storage integration object?
31. What are the steps to integrate AWS with snowflake?
32. What are the steps to integrate Azure with snowflake?
33. ---
34. What is file format object?
35. ---
36. What is a stage?
37. What are the different types of stages?
    1. external
    2. internal
38. What is external stage?
39. What is internal stage?
40. What are the different types of internal stages? Explain each one of them in detail.
    1. user
    2. table
    3. named internal
41. ---
42. What are the different ways of bringing data into snowflake?
    1. copy command
    2. snowpipe
    3. other etl tools like adf etc.
43. Explain copy command. When is it used?
44. What are the different storages from which we can bring data into snowflake using copy command?
    1. cloud storages using external stage
    2. local machines using internal stage
45. Is it possible to do data transformations while using copy command?
46. What are the different options available for copy command?
47. How will you see the history of copy command?
    1. information\_schema
48. ---
49. Explain snowpipe. When is it used?
50. Can you explain the architecture of snowpipe?
51. Explain the steps needed to setup snowpipe.
52. How will you troubleshoot your snowpipe in case of any issues?
    1. check the pipe status if its working fine
    2. viewing the copy history for any errors
    3. validating the data files
53. What are the steps you will take to modify a snowpipe?
54. ---
55. What do you mean by data unloading in snowflake?
56. What are the steps to perform data unloading in snowflake?
57. What are the different options available in copy command while performing data unloading?
58. ---
59. How can you process semi-structured data like json, xml in snowflake?
    1. first we need to bring the data into snowflake and store it in a stage table with variant datatype
    2. then we need to process the variant column and get the data into desired format
    3. then we need to load that desired data into final table
60. ---
61. What is caching?
62. Explain the cache architecture in snowflake.
63. What are the different types of caches in snowflake?
    1. results cache
    2. local disk cache
64. Explain results cache.
65. Explain local disk cache.
66. ---
67. What is the continuous data protection lifecycle?
68. What is time travel and retention period?
69. What are the different ways of querying historical data?
    1. at timestamp
    2. at offset
    3. before statement
70. How can you restore deleted objects?
71. What is fail-safe?
72. ---
73. What happens when we do a zero-copy clone of a table?
74. ---
75. What are the different types of tables?
    1. permanent
    2. transient
    3. temporary
76. Explain permanent tables.
77. Explain transient tables.
78. Explain temporary tables.
79. Can we convert one type of table to another?
80. Lets say I have a table named A. I dropped it and created another table with exactly the same name A. Now I am trying to un-drop that table A and not able to. How can I un-drop my previous table A?
    1. alter and change the name of the newly created table A to something else, say B and then try to un-drop the table A
81. Can we create a transient database / schema?
82. Can we create a temporary table with exactly the same name as a permanent / transient table?
83. ---
84. What are external tables in snowflake?
85. What are the limitations of external tables in snowflake?
86. What are the advantages of using external tables in snowflake?
87. What are the metadata associated with external tables in snowflake?
88. What are the steps for creating an external table in snowflake?
89. Will an external table get refreshed automatically in case of any changes in external location?
90. ---
91. What is the object hierarchy in snowflake?
92. What is access control in snowflake?
93. What are the different models of access control supported by snowflake?
    1. discretionary
    2. role based
94. Can u explain the terms - securable object, privilege, role, user - in the context of above 2 access control models and how they are related?



1. What is a role?
2. What are different types of roles in snowflake?
   1. system defined
   2. custom
3. What are the different system defined roles?

A screenshot of a document

Description automatically generated

1. What are custom roles in snowflake?
2. Explain the role hierarchy in snowflake.

A diagram of a company

Description automatically generated

1. ---
2. What are views in snowflake?
3. When are the different types of views in snowflake?
   1. normal
   2. secure
   3. materialized
4. When should you consider a normal view?
5. What is a secure view?
6. When should you consider a secure view?
7. What is a materialized view?
8. Does materialized view refresh itself when there is a change in the base table data?
9. Does materialized view increases cost?
10. When should you consider a materialized view?
11. What are the advantages of a materialized view?
12. What are the limitations of a materialized view?
13. ---
14. What is column level security in snowflake?
15. What are the different ways to implement column level security?
16. What is dynamic data masking?
17. What is a masking policy?
18. How to create a masking policy?
19. How to set a masking policy?
20. How to unset a masking policy?
21. How to alter a masking policy?
22. How to drop a masking policy?
23. ---
24. What is data sharing in snowflake?
25. What are the two types of customers we have?
    1. snowflake users
    2. non-snowflake users
26. What is a share in snowflake?
27. What are different objects that can be shared?
28. Can you explain the architecture of sharing data with snowflake users?

A diagram of a diagram

Description automatically generated

1. Can you explain the process of sharing data with snowflake users?
   1. create a share
   2. allow this share to be accessed by other snowflake users
2. Can you explain the architecture of sharing data with non-snowflake users?

A diagram of a server

Description automatically generated

1. What is a reader account and why it’s needed?
2. Can you explain the process of sharing data with non-snowflake users?
   1. create a share
   2. create a reader account
   3. allow reader account to access this share
   4. share the reader account with non-snowflake user
3. What are the two ways to work with data sharing features?
   1. sql queries
   2. UI
4. ---
5. What is a task?
6. How to create a task?
7. How to alter a task?
8. What is a DAG of tasks?
9. How to create a DAG of tasks?
10. How to check tasks history?
11. How will you troubleshoot a task?
12. ---
13. What is a stream in snowflake?
14. Are streams similar to cdc feature in databricks? How streams help in implementing a cdc like use case in snowflake?
    1. metadata columns (to capture changes in source table)

A diagram of a stream

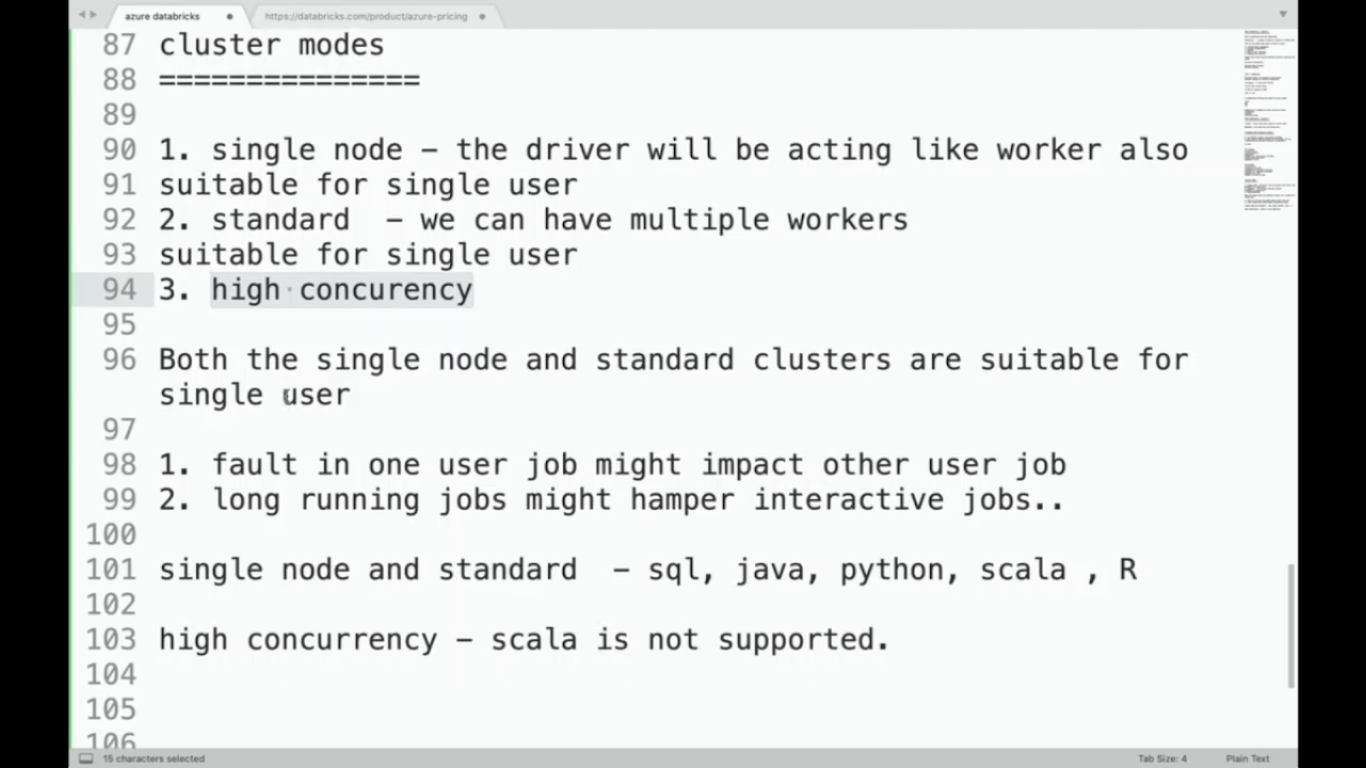
Description automatically generated

* 1. merge command (to merge changes captured via streams to destination table)

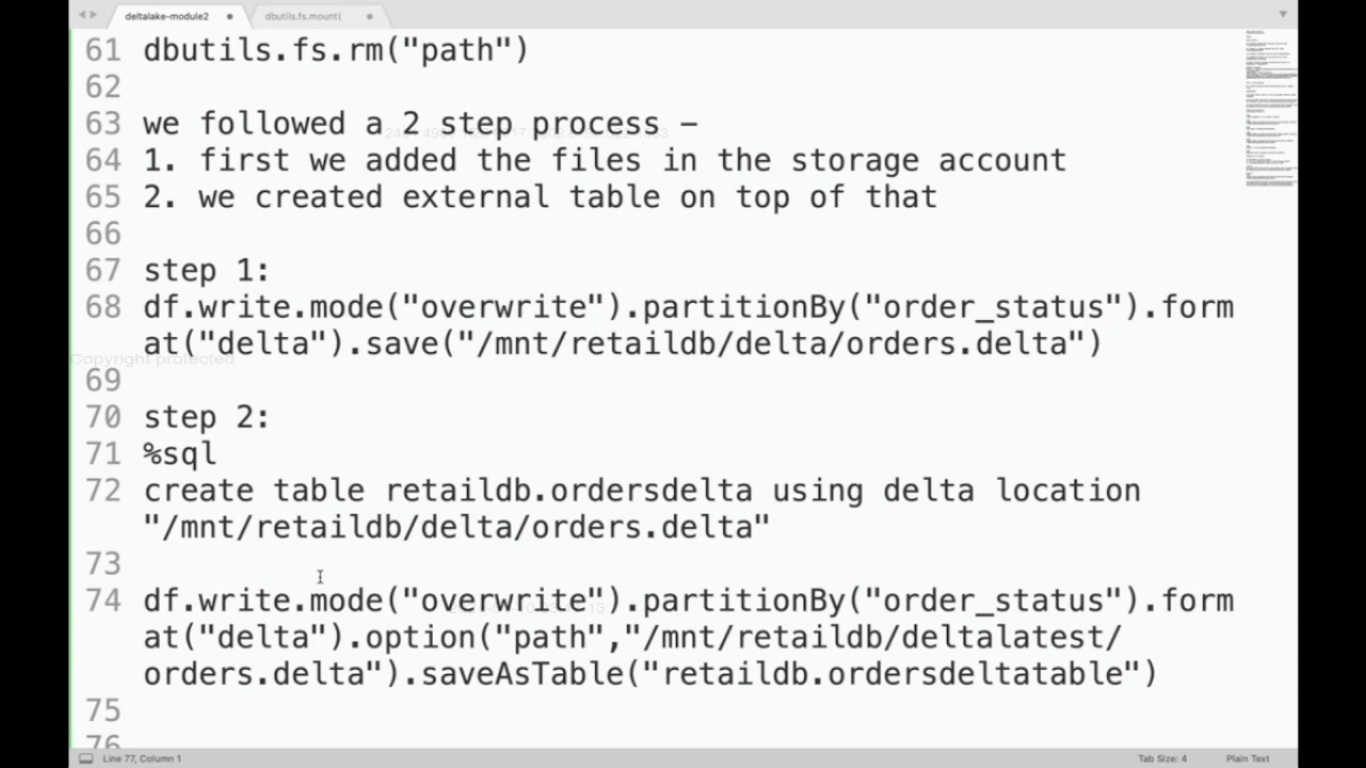
1. What are the different types of streams?
2. How will you implement a continuous data load pipeline?
   1. using snowpipe (for ingestion) + streams (data capture & merge) + tasks (scheduling the merge)
3. ---
4. What is data sampling?
5. Why we need data sampling in snowflake?
6. What are the different sampling techniques supported by snowflake?
7. Why to do data sampling if zero-copy cloning is available?
8. Why to do data sampling if we can filter the data based on some conditions to get a subset of data?
9. ---
10. What are UDFs in snowflake?
11. What are the different types of UDFs?
    1. based on return type - scalar, tabular
12. Is function overloading supported for UDFs in snowflake?
13. Which languages are supported by snowflake for writing UDFs?
14. ---
15. What are stored procedures?
16. What are some of the uses of stored procedures in snowflake?
17. Difference between UDFs and stored procedures.
18. ---
19. What are alerts?
20. When do we use alerts?
21. What are the privileges needed to create and use alerts?
22. ---
23. What are notifications?
24. What are email notifications?
25. How to create an email notification?
    1. create email type notification integration object
    2. use SYSTEM$SEND\_MAIL stored procedure and provide the notification integration object created before with recipient’s emails, subject etc.
26. Can email notifications be integrated with alerts?
    1. yes, using if then
27. ---
28. Discuss ci/cd implementation for snowflake code.
    1. one way is to use github and github actions. put code to run in one folder using github and schedule it to run in snowflake using workflows under github actions.
29. ---
30. What is resource monitor and why we use it?
31. What properties does resource monitor has?
    1. credit quota
    2. monitor type
    3. schedule
    4. actions
32. How to restart suspended warehouses?
33. What are different ways of creating/modifying a resource monitor?
    1. ui
    2. sql
34. ---

PYSPARK, DATABRICKS

1. What is databricks?
2. What are the challenges with open-source version of spark? Why we use databricks?
3. What is the pricing structure in databricks?
4. Is databricks available only on azure? Why databricks with azure?
5. What is a cluster?
6. What is a notebook?
7. What are the different types of cluster options?
8. What are the different types of cluster modes?



1. What are the different types of nodes in databricks?
2. What is the difference between azure databricks and azure synapse?
3. What is the use of magic command?
   1. To write various types of codes (sql, python, fs etc.) in same notebook.
4. What is DBFS?
5. Explain the architecture of databricks.
   1. <https://medium.com/@melissa.milligan/getting-started-with-data-engineering-on-databricks-88b4ebaf7a94>
6. What is dbutils?
7. Explain the functionality of fs utility in dbutils.
8. Explain the functionality of mount in fs utility in dbutils.
   1. mount(), unmount(), mounts(), updateMount()
9. Explain the functionality of data utility in dbutils.
   1. summarize
10. Explain the functionality of notebook utility in dbutils.
    1. exit
    2. run
11. Explain the functionality of widgets utility in dbutils.
    1. combobox, dropdown, multiselect, text
    2. get, remove, removeAll
12. How to pass parameters from one notebook to another?
    1. Using an argument in run command and widgets
13. What is databricks cli? How to install it?
14. How to configure databricks cli to connect to workspace?
    1. databricks configure --token
15. What is a datalake?
16. What are some of the advantages of datalake?
17. What are the challenges of using a datalake?
18. What are some of the scenarios which shows ACID properties violation and other problems with datalake?
19. What is deltalake?
20. How deltalake solves the problems which are there in datalake?
    1. crud with ACID compliance
    2. constraints check
    3. schema management
    4. time travel
21. How crud happens in deltalake?
    1. create, insert (append, overwrite)
    2. read
    3. update
    4. delete
22. How to create tables on top of delta data?



1. How to insert new data into a delta table / delta location?
   1. insert command
   2. copy command
   3. dataframe write with different modes



1. How to evolve schema in delta?
   1. .option(“mergeSchema”, “true”)
2. How to update some existing data in delta table?
   1. update command
3. How to delete some existing data from delta table?
   1. delete command
4. How to apply constraints on delta table?
   1. alter command with check
5. How will you time travel across different versions in delta table?
   1. describe history tablename
   2. version as of, timestamp as of
   3. restore table tablename to version as of x
6. What is change data feed feature in databricks? Why we need it?
7. What are the different ways to enable change data feed feature?
8. How the change data feed feature works in databricks?
   1. https://docs.databricks.com/en/delta/delta-change-data-feed.html
9. What is data governance? How unity catalog comes into play here?
10. What are some key features of unity catalog?
11. What are the limitations of implementing data governance without unity catalog?
12. How to create a metastore in unity catalog?
13. Explain unity catalog object hierarchy model.
    1. metastore root -> catalog -> schema -> table
14. What are the different ways of adding users in databricks?
    1. manual
    2. automatic using scim provisioning
15. What are the different roles in databricks?
    1. account admin
    2. metastore admin
    3. workspace admin
    4. workspace user
16. What all an account admin can do?
17. What all a metastore admin can do?
18. What all a workspace admin can do?
19. What all a workspace user can do?
20. What are the different ways in which a workspace admin can provide permissions to developers for creating/using clusters?
    1. unrestricted cluster creation
    2. restricted cluster creation
    3. use shared cluster (created by workspace admin)
21. -----------------------------------------------------------------------------------------------------
22. What is data engineering?
23. What is Apache spark?
24. What is the architecture of Apache spark framework?
25. Why Apache spark?
26. What are some of the things missing from plane vanilla version of Apache spark?
27. What are some of the important platforms that integrate spark to provide the above missing features?
    1. cloudera hadoop platform
    2. amazon emr
    3. azure hdinsight
    4. google data proc
    5. databricks
28. What features databricks provide on top of vanilla spark?
29. On which platforms databricks is available?
30. Explain databricks architecture.
31. What are magic commands? List some magic commands.
    1. %lsmagic
32. What is dbutils?
33. Explain the functionality of fs utility in dbutils.
34. Explain the functionality of notebook utility in dbutils.
35. Explain the functionality of widgets utility in dbutils.
36. What is DBFS?
37. What is dbfs root? What info does it hold? Why its not recommended to use it?
    1. logs, job results, notebook revisions
38. Which file system is accessed using file:/?
    1. local file system of driver machine
39. Why its recommended to setup and mount another separate storage container to dbfs for data engineering use cases?
40. What are the different ways of setting up the mount point to dbfs?
    1. using service principal (app registration)
41. Why its not recommended to mount using dbfs mount utility?
    1. because once mounted, the /mnt/<path> will be accessible to all who uses that workspace
42. What is unity catalog? What all features it provides?
43. Where unity catalog fits inside medallion architecture?
    1. check diagram by Prashant
44. Can you draw a diagram to show how unity catalog looks like and its components?
45. What are the steps to create a metastore in unity catalog?
46. How to perform user provisioning in unity catalog?
47. Which catalog is local to the workspace?
    1. hive\_metastore
48. How to create a new catalog?
49. How to create a new database inside a catalog?
50. How to create a new table / volume?
51. Where does the data of a managed table / volume gets stored and how it can be accessed?
52. How to grant required permissions on catalog/database/object to users/groups?
    1. grant <permission\_name> on catalog/database <catalog/database name> to <user/group name>
53. What is the recommended way of having a storage container other than dbfs root?
    1. unity catalog metastore external location feature
54. How will you setup an external location inside unity catalog metastore?
    1. UI
    2. create table statement
55. What is delta lake? What are the key features it provides?
56. How to create delta tables?
    1. create table statement
    2. df.write….saveAsTable()
    3. DeltaTable api
57. How to read delta tables?
    1. spark.sql
    2. df.read……table()
58. How to update records in delta table?
    1. spark.sql update
    2. DeltaTable api
59. How to delete records in delta table?
    1. spark.sql delete
    2. DeltaTable api
60. How to perform merge operation with delta tables?
    1. spark sql merge
    2. DeltaTable api
61. How to create, read, insert etc. into delta tables present in external location?
    1. same methods as above
62. How will you time travel across different versions in delta table?
    1. methods – spark sql, DeltaTable api, dataframe api
    2. describe history tablename
    3. version as of, timestamp as of
    4. restore table tablename to version as of x
63. How will you convert parquet to delta in place?
    1. using convert to delta command
64. What are the different ways a schema can change and how different insert methods work out?
65. What are the different ways we can allow schema evolution?
    1. manual - alter table command
    2. automatic
       1. spark.databricks.delta.schema.autoMerge.enabled=true
       2. .option(“mergeSchema”. “true”)
    3. different insert methods work out differently with the above schema evolution tactics. preferred is dataframe with option mergeSchema
66. How delta works internally for crud operation?
67. -----------------------------------------------------------------------------------------------------
68. What is the difference between batch and stream processing?
69. Explain spark as a streaming engine.
70. What are the two different modules in spark for stream processing?
71. (In/using spark streaming module) What is a dstream?
72. (In/using spark streaming module) What are stateless transformations?
73. (In/using spark streaming module) Explain working of stateful transformations in detail.
    1. processing of a micro-batch depends on all the previous micro-batches
    2. micro-batches present in a window are processed together
74. (In/using spark streaming module) Can you write a word count program using stateless transformations?
    1. reduceByKey()
75. (In/using spark streaming module) What is checkpointing and why it is needed?
76. (In/using spark streaming module) Can you write a program to calculate frequency of words since beginning?
    1. updateStateByKey(update\_func)
77. (In/using spark streaming module) Can you write a program to calculate frequency of words in a window of a particular size?
    1. reduceByKeyAndWindow(summary\_func, inverse\_func, window\_size, sliding\_interval)
78. (In/using spark streaming module) Can you explain the functioning of reduceByWindow(summary\_func, inverse\_func, window\_size, sliding\_interval) and countByWindow(window\_size, sliding\_interval) transformations?
79. (In/using spark streaming module) What are the limitations of spark streaming module and how spark structured streaming module overcomes those limitations?
80. (In/using spark structured streaming module) What are the challenges we face in a streaming use case and how spark structured streaming module overcomes all those?
81. (In/using spark structured streaming module) Can you write a simple program to calculate frequency of words since beginning?
82. (In/using spark structured streaming module) How actually a spark structured streaming code runs?
83. (In/using spark structured streaming module) Explain different output modes.
84. (In/using spark structured streaming module) Explain different triggers.
85. (In/using spark structured streaming module) Explain different input sources.
86. (In/using spark structured streaming module) Explain different output sinks.
87. (In/using spark structured streaming module) What is the use of checkpointing?
88. (In/using spark structured streaming module) How will you achieve exactly once semantics?
89. (In/using spark structured streaming module) What are the different types of transformations?
90. (In/using spark structured streaming module) How will you clean your state store?
    1. watermark
    2. proper output mode
91. (In/using spark structured streaming module) How do you handle late coming records?
92. (In/using spark structured streaming module) What are the different types and ways of aggregation?
93. (In/using spark structured streaming module) Can you write a simple program showing aggregation in tumbling window with/without watermark?
94. (In/using spark structured streaming module) What are the different types of joins?
95. (In/using spark structured streaming module) Explain streaming-static joins with all their validity conditions.
96. (In/using spark structured streaming module) Explain streaming-streaming joins with all their validity conditions.
97. (In/using spark structured streaming module) Can you write a simple program showing streaming-static join?
98. (In/using spark structured streaming module) Can you write a simple program showing streaming-streaming join?
99. -------------------------------------------------------------------------------------------------------
100. What are the 3 ways of ingestion?
     1. copy into
     2. spark streaming
     3. autoloader
101. Explain copy into.
102. Explain manual and automatic schema evolution methods for copy into.
103. Explain spark streaming mode of ingesting data.
104. Explain manual and automatic schema evolution methods for spark streaming ingestion.
105. What is autoloader?
106. Explain different schema inference methods in autoloader.
107. Explain different schema evolution modes in autoloader.
108. How will you work with autoloader with different cloud storages?
     1. gen2
     2. s3
109. Explain different file detection modes in autoloader.
110. -------------------------------------------------------------------------------------------------------

OPTIMIZATIONS

1. What are the different optimization techniques you have used?
2. What is partitioning?
3. How do you select proper columns for partitioning?
4. What is the difference between static and dynamic partitioning?
5. What is bucketing?
6. How do you select proper columns for bucketing?
7. How to calculate the number of buckets needed?
8. Can we have partitions inside buckets? Can we have buckets inside partitions?
9. Tell some differences between partitioning and bucketing.
10. Given following operations - filter, joins, group by - which of them get optimized with the help of partitioning, which of them get optimized with the help of bucketing?
11. What are the configurations that need to be set for partitioning and bucketing?
    1. hive> SET hive.exec.dynamic.partition=true;
    2. hive> SET hive.exec.dynamic.partition.mode=non-strict;
    3. hive> SET hive.enforce.bucketing =true;​
    4. <https://medium.com/@sujathamudadla1213/how-can-you-configure-hive-for-dynamic-partitioning-and-specify-the-number-of-reducers-for-a-job-1a6464da4143>

I'd be glad to explain hive.exec.dynamic.partition.mode in Hive and provide an example:

**What is** hive.exec.dynamic.partition.mode**?**

In Apache Hive, hive.exec.dynamic.partition.mode is a configuration property that controls how Hive handles partitioning during data insertion into partitioned tables using dynamic partitioning. Dynamic partitioning allows you to insert data without explicitly specifying the partition values in your INSERT statement. Hive infers the partition values from the data itself based on the specified partition columns.

**Dynamic Partition Modes:**

This property has two main modes:

* **STRICT (default):** (Value: STRICT)
  + In this mode (the default), Hive strictly enforces that every row in the data being inserted must have values for all the partition columns. If any row is missing a partition value, the entire insertion operation fails. This mode is more secure as it prevents accidental creation of empty partitions.
* **NONSTRICT:** (Value: NONSTRICT)
  + With NONSTRICT mode, Hive inserts rows that have valid partition values into their corresponding partitions. However, rows missing partition values are simply skipped and not inserted into any partition. This mode offers more flexibility but requires caution to avoid unintended data loss.

**Choosing the Right Mode:**

* Use STRICT mode when data quality and completeness are crucial, and you want to ensure every row has all necessary partition values.
* Consider NONSTRICT mode if you anticipate some rows might lack partition values, but you still want to insert the rest of the valid data. However, be mindful of the potential for skipped rows and unintended consequences.

**Example:**

Suppose you have a partitioned table sales\_data with partition columns year and month. You're inserting data from a non-partitioned table temp\_sales.

**Scenario 1: STRICT Mode (Default)**

SQL

-- Assuming temp\_sales has all columns, including year and month

SET hive.exec.dynamic.partition=true;

INSERT OVERWRITE TABLE sales\_data PARTITION (year, month)

SELECT \* FROM temp\_sales;

Use code [with caution.](https://gemini.google.com/faq#coding)

content\_copy

In this case, with STRICT mode (the default), Hive will:

* Read each row from temp\_sales.
* Attempt to infer year and month values from the corresponding columns in the row.
* If all rows have valid year and month values, they'll be inserted into their respective partitions in sales\_data.
* If even one row is missing a year or month value, the entire insertion will fail.

**Scenario 2: NONSTRICT Mode**

SQL

SET hive.exec.dynamic.partition=true;

SET hive.exec.dynamic.partition.mode=NONSTRICT;

INSERT OVERWRITE TABLE sales\_data PARTITION (year, month)

SELECT \* FROM temp\_sales;

Use code [with caution.](https://gemini.google.com/faq#coding)

content\_copy

Here, with NONSTRICT mode:

* Hive will process each row from temp\_sales.
* Rows with valid year and month values will be inserted into their corresponding partitions.
* Rows missing year or month values will be **skipped** and not inserted into any partition.

**Key Points:**

* Dynamic partitioning can improve performance and data organization, but it's essential to choose the appropriate mode based on your data quality and desired behavior.
* Consider using STRICT mode for stricter data validation.
* Use NONSTRICT mode cautiously, understanding that rows might be skipped.
* Always test your queries thoroughly in a non-production environment before running them on critical data.

1. How do you usually perform partitioning?
   1. static - load command
   2. dynamic - transfer data from non-partitioned table to partitioned table
2. How do you usually perform bucketing?
   1. transfer data from non-bucketed table to bucketed table
3. Which among mappers and reducers are invoked during partitioning, bucketing?
   1. partitioning - only mappers
   2. bucketing - both mappers and reducers (number of buckets ~ number of reducers)
4. What are the different types of join optimizations that you have used?
5. How a normal join works?
6. When can we perform a map side join?
7. Explain working of map side join in detail.
8. Among inner, left, right, full outer joins, which one of them can be performed using a map side join if the left table is small and right table is big?
9. When can we perform a bucket map join?
10. Explain working of bucket map join in detail.
11. When can we perform a sort merge bucket join?
12. Explain working of sort merge bucket join in detail.
13. What are some of the configurations needed to be set for performing different joins?
14. What is broadcast join and when to use it?
15. What are window functions and how they help in optimizing our queries?
16. What is the difference between row and columnar file formats? Why columnar file formats are preferred?
17. Why we need to think about file formats and what are the different factors based on which we decide a suitable file format?
18. Explain text file format in detail.
19. Explain avro file format in detail.
20. Explain orc file format in detail.
21. Explain parquet file format in detail.
22. What are the different tradeoffs when it comes to compression? What are the different factors based on which we decide a suitable compression technique?
23. Explain different compression techniques in detail.
    1. https://stackoverflow.com/questions/32382352/is-snappy-splittable-or-not-splittable
24. What is vectorization? What is the configuration to enable it?
25. What are the different supported engines for hive? Which configuration is used to change the hive engine?
26. How to deal with UDFs in a proper way?
27. What is CBO in hive?
    1. <https://teepika-r-m.medium.com/cost-based-optimization-in-hive-ea0c6296894c#:~:text=CBO%20is%20one%20of%20the%20optimization%20techniques%20used%20to%20boost,they%20are%20expensive%20to%20compute>.
28. -----------------------------------------------------------------------------------------------------------
29. What is performance tuning?
30. What are some of the performance tuning challenges?
31. What is the end goal of performance tuning?
32. Explain benchmarking in detail.
33. How do you perform instrumentation and pinpointing?
34. How do you optimize reading schemas in spark?
35. What is disk caching and how it helps optimize queries?
36. How can you optimize read using cache()?
37. How can you optimize the response time of the query that performs caching? Will column elimination help?
38. How can we push filter condition down to storage layer?
39. What are crippled predicates and how can we avoid them?
40. What is directory scan overhead and how can we resolve it?
41. What is small file problem and how can we resolve it?
42. What is haystack query? How z-ordering helps us with haystack queries?
43. How can we estimate the initial number of partitions in a dataframe?
44. Can we tune spark.sql.files.maxPartitionBytes value? What is the best value for it?
45. Can we tune spark.sql.files.openCostInBytes value? What is the best value for it?
46. Explain Spark memory management in Databricks.
47. How to check the amount of memory mapped per core?
48. What is data spill? Why does it happen?
49. What is data explosion? What are some scenarios that may cause data explosion?
50. Why data spill is bad?
51. How will you detect data spill?
52. What options may not solve data spill?
53. What are different ways you can resolve data spill issues (before and after shuffle spills)?
54. What are the implications of tuning spark.memory.fraction value?
55. Can you resolve data spill by using larger VMs?
56. How will you tune shuffle partitions to resolve data spill?
57. You are doing an explode() and its causing data explosion before shuffle. How will you resolve it?
58. You are doing an union() with orderBy and its causing data explosion after shuffle. How will you resolve it?
59. What is skew and how it is created?
60. Why skew is bad?
61. How to detect skew?
62. What options may not work to resolve skew?
63. What are some of the different skew tuning approaches?
64. How salting works?
65. Can you write code to implement salting in a join?
    1. <https://medium.com/curious-data-catalog/sparks-salting-a-step-towards-mitigating-skew-problem-5b2e66791620>
66. Can you write code to implement salting in groupBy?
67. What are some disadvantages of salting?
68. What are skew hints in databricks? Do they resolve data skew?
69. How AQE resolves data skew?
70. How shuffle works?
71. How shuffle works in different wide transformations like repartition, orderBy, groupBy etc?
72. How will you tune shuffling?
    1. Avoid shuffle.
    2. If cant avoid shuffle, tune shuffle bottlenecks.
73. Explain working of shuffle sort merge join.
74. Explain working of broadcast hash join.
75. Explain working of shuffle hash join.
76. How broadcast performs with small datasets?
77. How broadcast performs with large datasets?
78. How will you tune a large-to-large table join?
79. How bucketing works?
80. Explain different bucket joins.
    1. bucket map join
    2. sort merge bucket join
81. What are some of the limitations of using bucketing in joins?
82. How will you perform a large-to-large table join with the help of bucketing?
83. What are the different techniques of saving the results of intermediate joins to be used later?
84. ---------------------------------------------------------------------------------------------------------
85. What is data skipping and how delta helps us to skip data?
    1. https://towardsdatascience.com/delta-lake-enables-effective-caching-mechanism-and-query-optimization-in-addition-to-acid-96c216b95134
86. What is disk caching and how it helps optimize queries?
87. What is the problem with small files in delta? How do they get created?
88. How can we resolve small files problem?
89. What is z-ordering and how does it help in data skipping?
90. Why we use vacuum command?
    1. vacuum tablename retain x hours
91. What is auto optimize in databricks?
    1. The term auto optimize is sometimes used to describe functionality controlled by the settings delta.autoOptimize.autoCompact and delta.autoOptimize.optimizeWrite.
    2. https://docs.databricks.com/en/delta/tune-file-size.html#auto-compact
92. What is photon engine and why we use it?
93. --------------------------------------------------------------------------------------------------------
94. How will you drop columns in an optimized way?
    1. alter table drop with tblproperties delta.columnMapping.mode set to name -> to drop those columns from metadata and not data
    2. reorg table <table\_name> apply(purge) -> to delete data marked for deletion
95. ----------------------------------------------------------------------------------------------------

SYSTEM DESIGN

1. Difference between transactional vs analytical processing.
2. Difference between database, data warehouse, datalake.
3. Difference between normalization and de-normalization.
4. What are the different normal forms?
5. What is SCD? What are the different types of SCD?
6. Can you write code for SCD1 in PySpark, SQL?
7. Can you write code for SCD2 in PySpark, SQL?
8. What do you mean by ACID properties?
9. What is modern data architecture / 2-tier architecture? What are the challenges with 2-tier architecture?
10. What is lakehouse architecture? How does it resolve the challenges of 2-tier architecture?
11. How databricks has implemented its lakehouse architecture?
    1. storage layer (data lake) -> transactional layer (delta lake) -> delta engine
12. What is medallion architecture?
13. Can u implement a sample pipeline having medallion architecture using change data feed feature?

PROJECT

1. Explain your project in detail.
2. How much amount of data that you are processing per day?
3. What you mean by incremental data? And how you process incremental data in your project?
4. What are the challenges/bottlenecks you faced while developing your pipeline and how did you resolve them?