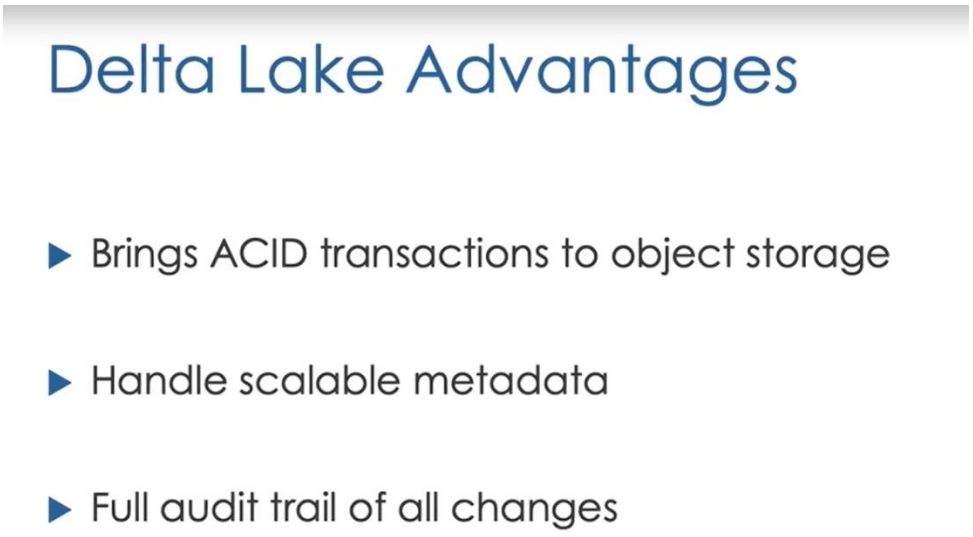
- Delta lake is open-source storage framework that brings reliability to data lakes.
- Data lakes have inconsistency and performance issues.

- Delta lake is not data warehouse and not a database service.
- Delta lake is component which is deployed on cluster as part of Databricks runtime.
- If you create delta lake table, it will be stored in one or more data files in parquet format.
- Delta log (Transaction log) is ordered records of every transaction performed on table.
- Delta log serves as single source of truth.
- JSON file contains commit information – Operations performed + predicates used; data files affected (added / removed).
- Delta lake guarantee that you will get the most recent version of the data. Read operation will not have a deadlock or conflicts with any ongoing operation on the table.



- Delta lake is default format for any table in Databricks, no need to mention it specifically.

- **DESCRIBE DETAIL <tablename>** statement will provide metadata information about table. We can get details like location of the table, number of data files (parquet format) in current table version.

- For single insert, we will have 4 parquet files created in Databricks. This is because Spark work in parallel, Check the cluster configuration and view the number of nodes in the cluster.

- DESCRIBE HISTORY **<tablename>** will provide history of the table with different versions of table.

- Delta log files will be in JSON format and check some files of Delta log files.

Advanced Delta Lake features

- Time travel – Audit data changes, Describe history command, query old version of data, version number. Roll back versions, restore table command

  SELECT * FROM <table name> **VERSION AS OF** <version number>

  SELECT * FROM <table name**>@v<version number>**

  RESTORE TABLE <table name> **TO VERSION AS OF** <version number>

- OPTIMIZE command for compacting small parquet files since spark work in parallel. Having small files negatively affects the performance of Delta table.

- Z-order indexing in Delta Lake is about co-locating and reorganizing column information in the same set (used with OPTIMIZE command). It speeds up data retrieval when filtering on provided fields by grouping data.

  OPTIMIZE <table name>
  ZORDER BY <column name(s)>

- Vacuum a delta table - Cleaning up unused or old data files, uncommitted files, files that are no longer in latest table state.

  VACUUM <table name> [retention period], default retention period is 7 days.
  VACCUM <table name> RETAIN 0 HOURS

  Retention period means vacuum operation will prevent you from deleting files less than 7 days old. This is to ensure that no long running operations are still referencing to any of the files to be deleted. You must change default **retentionDurationCheck** spark parameter if needed to turn off the retention duration check value of 7 days and this parameter shouldn't be changed in production environment.

- Note that once you run a vacuum command, you will lose the ability to time and travel back the table to an older version than the specified retention period.