Bronze & Silver Guiding Documentation

Greetings adventurer!

If you seek to build a bronze layer leveraging Apache Spark with the PySpark language, you must first gather the necessary tools and knowledge. The first step is to obtain a working Synapse Analytics workspace and familiarize yourself with the PySpark language.

Once you have the necessary tools, you will need to use several things to build your bronze layer. These include:

* [**sql**](https://spark.apache.org/sql/): to initialize database using SQL and drop tables if exist during initial load – please note it is forbidden to use SQL for anything else. Useful links: [DROP TABLE - Azure Databricks - Databricks SQL | Microsoft Learn](https://learn.microsoft.com/en-us/azure/databricks/sql/language-manual/sql-ref-syntax-ddl-drop-table) and [CREATE DATABASE - Azure Databricks - Databricks SQL | Microsoft Learn](https://learn.microsoft.com/en-us/azure/databricks/sql/language-manual/sql-ref-syntax-ddl-create-database).
* [**dataframe**](https://spark.apache.org/docs/3.1.1/api/python/reference/api/pyspark.sql.DataFrame.html): allows you to manipulate the data using pySpark.
* [**load**](https://spark.apache.org/docs/latest/sql-data-sources-load-save-functions.html): This function will allow you to load data into your PySpark DataFrame. You can use it to load data from a variety of sources, including JSON files, Parquet files, and more.
* [**withColumn**](https://spark.apache.org/docs/3.1.3/api/python/reference/api/pyspark.sql.DataFrame.withColumn.html): This function will allow you to add auditing columns to DataFrame.
* [**current\_timestamp**](https://spark.apache.org/docs/3.1.3/api/python/reference/api/pyspark.sql.functions.current_timestamp.html): This function will allow you to get the current date and time. You can use it in conjunction with **lit** to add the current date and time to your DataFrame.
* [**sha2**](https://spark.apache.org/docs/3.1.2/api/python/reference/api/pyspark.sql.functions.sha2.html): This function allows you to calculate row hash to check for row changes.
* [**lit**](https://spark.apache.org/docs/3.1.3/api/python/reference/api/pyspark.sql.functions.lit.html): This function will allow you to add a literal value to your DataFrame. You can use it to add constants, such as the current date and time, to your DataFrame.
* [**write to format delta**](https://docs.delta.io/latest/delta-batch.html#write-to-a-table): to write your DataFrame to a Delta format. Delta is a powerful data storage format that provides several advantages over other formats, including improved query performance and better data management.
* [**Delta merge**](https://docs.delta.io/latest/delta-update.html#upsert-into-a-table-using-merge): This function will allow you to merge two DataFrames based on a common key.

As you build your bronze layer, keep in mind that the process is iterative. You will need to experiment with different functions and data sources to find the best approach for your particular use case. Be sure to test your code thoroughly to ensure that it is working correctly.

May your quest for a powerful silver layer be successful!