Syed Babar Ali
School of Science and Engineering

CS5304 Big Data Services & MLOps Spring 2025 (Quiz # 4) March 19th, 2025

Name: ________________________________ Roll No: __________________

1. Briefly describe the concept of encoder in a Dataset API (2)

At the core of the Dataset API is a new concept called an encoder, which is responsible for converting between JVM objects and tabular representation. The tabular representation is stored using Spark's internal Tungsten binary format, allowing for operations on serialized data and improved memory utilization. Spark comes with support for automatically generating encoders for a wide variety of types, including primitive types (e.g. String, Integer, Long), Scala case classes, and Java Beans.

2. Explain the **concept** and **benefits** of lazy evaluation in Dataset in relation to transformations and actions. (5)

Lazy evaluation in Spark means that the computation will not happen until an action is triggered. Transformations are the ones that produce new Datasets, e.g., map, filter, select, and aggregate (groupBy). Actions are the ones that trigger computation and return results, e.g., count, show, or writing data.

Lazy evaluation in Spark delays the execution of transformations until triggered by an action, which helps Spark to analyze and optimize the computation, resulting in significant improvements in efficiency and scalability.

3. What is required to perform SQL queries on a spark DataSet? Provide the necessary **df** command. (2)

DataFrame/DataSet must be registered as a table/view to perform SQL Queries.

```
df.createOrReplaceTempView
```

4. Provide the spark command to read a CSV/JSON as a domain object. (2)

```
spark.read.json(path).as[<className>]
```

5. Change Data Feed (6)
   a. Brief explain change data feed (CDF) in a Deltalake.
   b. Provide one use case for CDF.
   c. How do you enable CDF on a table.

a. CDF is used to track row level changes between versions of a Delta table. 'Change events' are recorded by runtime for the data written to the table
   a. These events consist of data plus metadata whether data was inserted, updated or deleted

b. Using CDF, we can send changes downstream to any other system that is interested in the changed data. For example, if a flight time changes in an airline reservation system, using CDF, we can that flight change to a notification system to send email/sms notifications to the passengers on that flight.

c. *For a specific table:*
   ALTER TABLE myDeltaTable SET TBLPROPERTIES (delta.enableChangeDataFeed = true)
   For all new tables:
   set spark.databricks.delta.properties.defaults.enableChangeDataFeed = true;

6. Data Skipping (6)
   a. What is data skipping?
   b. Which column level statistics are required by data skipping?
   c. What is the role of Z-ORDER in optimizing data skipping?

a. As new data is inserted into a Delta table, file-level min/max statistics are collected for all columns of supported types. When there’s a lookup query against the table, Delta table first consults these statistics in order to determine which files can safely be skipped

b. Min/max

c. Z-Ordering is a technique to colocate related information in the same set of files. This co-locality is automatically used by data-skipping algorithms to dramatically reduce the amount of data that needs to be read

7. Delta Lake (6)
   a. How do you enable delta lake in Azure Databricks?
   b. What enhancements delta lake does in the underlying file structure to implement its functionality?
   c. Mention any 2 delta lake enhancements with brief explanation.

a. All tables on Databricks are Delta tables by default.

b. Delta Lake is an open source storage layer that brings reliability to data lakes. Delta Lake provides ACID transactions, scalable metadata handling, and unifies streaming and batch data processing. Delta Lake runs on top of your existing data lake and is fully compatible with Apache Spark APIs.

c. Delta Lake Enhancements:
   a. Schema enforcement: Automatically handles schema variations to prevent insertion of bad records during ingestion.
   b. Time travel: Data versioning enables rollbacks, full historical audit trails, and reproducible machine learning experiments.
   c. Upserts and deletes: Supports merge, update and delete operations to enable complex use cases like change-data-capture, slowly-changing-dimension (SCD) operations, streaming upserts, and so on.

8. Time Travelling (6)
    a. What is time travelling in Delta Lake?
    b. Which feature in delta lake allows you to time travel?
    c. Provide the command to retrieve time travelled data using either python or SQL?

a. Time Travel (a.k.a. data versioning): Delta Lake provides snapshots of data enabling developers to access and revert to earlier versions of data for audits, rollbacks or to reproduce experiments
    a. Each operation that modifies a Delta Lake table creates a new table version. You can use history information to audit operations, rollback a table, or query a table at a specific point in time using time travel.
b. Table Versioning
c. Various ways to read time travel data:
    a. DESCRIBE HISTORY bikeSharingDay
    b. **SELECT** count(1) **from** bikeSharingDay **VERSION AS OF 1**
    c. **SELECT** * **FROM** bikeSharingDay **TIMESTAMP AS OF** '2019-01-29 00:37:58'

9. What is the impact of VACUUM table utility command (2)

- The data files are never deleted automatically. You need to run VACUUM for this to happen.
- Vacuum commands remove files no longer referenced by a Delta table and are older than the retention threshold
    - The default retention threshold for the files is 7 days.

10. Labs: Provide a brief and concise answer to the question or explain the purpose of the commands below: (24 Total, 2 Points Each)

`df.withColumn('Tax', col('UnitPrice') * 0.08)`
Add new column to the dataframe based on UnitPrice existing column
`df.select(year("OrderDate").alias("Year")).groupBy("Year").count().orderBy("Year")`
Groups all rows with same year, aggregate year and sort in ascending order
What happens when you drop a managed table vs an external table?
For a managed table, databricks deletes the table and removes the directory associated with the table from the file system. In case of an external table, only the associated metadata information is removed from the metastore schema.
Given a notebook URL, how do you bring that in a databricks workspace?
Import URL
Given a DF with columns A...Z, how would you select a new DF with columns B,Y only?
new_df = df.select(col(“B”),col(“Y”))
How do you work with matplotlib with spark DF?
Covert spark dataframe to panda dataframe
What is the difference between matplotlib and seaborn?

Seaborn is based on matplotlib and provides simpler and easier to understand syntax.
How do you drop duplicate rows in DF?
Df.distinct -- checks all columns Df.dropDuplicates -- you can specific which column to check
What is the fundamental building block of databricks workflows?
task
What is the purpose of Delta Live Tables (DLT)? Provide the command to create a DLT pipeline?
---- Question Removed from Grading. This was from the extra credit assignment -----
How can you see last 10 changes in a delta table?
DESCRIBE HISTORY table_name LIMIT 10
How do you optimize table layout? Provide the necessary commands in the right order.
Optimize (with Z-ordering) Vacuum Or you can also simply enable liquid clustering which allows databricks to take care of everything byitself.