Final Quiz

Quiz, 8 questions

7/8 points (87.50%)

✓ **Congratulations! You passed!**

Next Item

✗ 0 / 1 point

1.

Narrow dependency implies that:

□ The data should be partitioned in a particular way

Un-selected is correct

□ Partitions either depend on one parent or a unique subset of the parent partitions that is known at design time

This should be selected

□ It does not depend on the values of the records in the parent partitions

Correct

Yes, a narrow transformation can be applied to arbitrary rows

□ It can be determined at the design time

Correct

Yes, Spark doesn’t need any extra information to compute the narrow dependency

✓ 1 / 1 point

2.

May there be a situation when join transformation does not shuffle data?

□ Yes, if RDDs are co-partitioned

Un-selected is correct

□ No way, join is shufflin' every day

Un-selected is correct

□ Yes, if RDDs are co-located

Correct
Spot on! If RDDs are co-located their partitions are already stored in memory of the same executor.

✓ 1 / 1 point

3.
What problems does PySpark introduce?

□ It introduces serialization overhead

Correct
Yes. PySpark has to serialize Python objects and then Spark has to serialize Scala objects

□ Kryo can't boost the performance of your app

Correct
That's right. JVM serializer gets byte array that is already serilized with Pickle. There is nothing much a serializer can do with the byte array

□ It generates DAGs which are hard to understand

Correct
Correct. PySpark tries to pipeline some transformations inside the interpreter, so DAGs become hard to understand

---

✓ 1 / 1 point

4.

What are examples of the Catalyst rules?

□ Join elimination

Un-selected is correct

□ Filter pushdown

Correct

Yes! One of the most useful rules

□ Constant folding

Correct

Correct. Another small but useful optimization

□ Column pruning

Correct

Right. Great rule to reduce the volume of the data being processed

---

✓ 1 / 1 point

5.

How can you force Catalyst to use broadcast join?

□ Use broadcast hint

Correct

Yes. Just import it from pyspark.sql.functions

□ Increase the spark.sql.autoBroadcastJoinThreshold configuration option

Correct

Correct. The value of this option is used to check if broadcast join can be applied

□ Set spark.sql.preferBroadcastJoin to True

Un-selected is correct

✓ 1 / 1 point

6.
How does checkpointing differ from persisting?

□ You can't checkpoint a DataFrame

Correct
Yes. There is no method to checkpoint a DataFrame. You can create a checkpoint of the underlying RDD.

□ Persisting truncates the lineage graph

Un-selected is correct

□ A checkpoint is always stored in a stable storage

Correct
Correct. When you persist a DataFrame it can be stored in memory and/or on disk. A checkpoint is stored in HDFS

□ Creating a checkpoint is faster

Un-selected is correct

✓ 1 / 1 point

7.
What are the premises of the Unified Memory Management?

□ Minimum unevictable amount of cached

Correct
Yes. Spark gives you that configuration option for workloads which rely heavily on caching

□ Evict storage, not execution

Correct
That's right. If you evict storage data, you will probably not read it back. But if you evict execution data, you will definitely use it again

□ Evict storage using FIFO (First In First Out) strategy

Un-selected is correct

□ Unlimited memory growth

Un-selected is correct

✓ 1 / 1
point

8.
Give examples of workloads which benefit from dynamic allocation

□ Machine learning algorithms

Un-selected is correct

□ ETL jobs with non-uniform input

Correct
Yes. If the input is not distributed uniformly, there may be payload spikes which require additinal resources

□ Interactive applications

Correct
Correct. Jupyter notebook (for example) is often used to analyse data. This analysis is often done locally on the driver and executors become idle

□ Applications with large shuffles

Correct

True. Shuffle may produce much more partitions than executors available