DATABRICKS-CERTIFIED-ASSOCIAT

Q&As

Databricks Certified Associate Developer for Apache Spark 3.0

Pass Databricks DATABRICKS-CERTIFIED-ASSOCIATE-DEVELOPER-FOR-APACHE-SPARK Exam with 100% Guarantee

Free Download Real Questions & Answers **PDF** and **VCE** file from:

https://www.leads4pass.com/databricks-certified-associate-developer-for-apache-spark.html

100% Passing Guarantee
100% Money Back Assurance

Following Questions and Answers are all new published by Databricks Official Exam Center

QUESTION 1

The code block displayed below contains an error. The code block should merge the rows of DataFrames transactionsDfMonday and transactionsDfTuesday into a new DataFrame, matching column names and inserting null values where column names do not appear in both DataFrames. Find the error.

Sample of DataFrame transactionsDfMonday:

1.+-------------+---------+-----+-------+---------+----+

2.|transactionId|predError|value|storeId|productId| f|

3.+-------------+---------+-----+-------+---------+----+

4.| 5| null| null| null| 2|null|

5.| 6| 3| 2| 25| 2|null|

6.+-------------+---------+-----+-------+---------+----+

Sample of DataFrame transactionsDfTuesday:

1.+-------+-------------+---------+-----+

2.|storeId|transactionId|productId|value|

3.+-------+-------------+---------+-----+

4.| 25| 1| 1| 4|

5.| 2| 2| 2| 7|

6.| 3| 4| 2| null|

7.| null| 5| 2| null|

8.+-------+-------------+---------+-----+

Code block:

sc.union([transactionsDfMonday, transactionsDfTuesday])

A. The DataFrames\\' RDDs need to be passed into the sc.union method instead of the DataFrame variable names.

B. Instead of union, the concat method should be used, making sure to not use its default arguments.

C. Instead of the Spark context, transactionDfMonday should be called with the join method instead of the union method, making sure to use its default arguments.

D. Instead of the Spark context, transactionDfMonday should be called with the union method.

E. Instead of the Spark context, transactionDfMonday should be called with the unionByName method instead of the union method, making sure to not use its default arguments.

Correct Answer: E

Correct code block:

transactionsDfMonday.unionByName(transactionsDfTuesday, True) Output of correct code block:

+-------------+---------+-----+-------+---------+----+ |transactionId|predError|value|storeId|productId| f|

+-------------+---------+-----+-------+---------+----+ | 5| null| null| null| 2|null|

| 6| 3| 2| 25| 2|null|

| 1| null| 4| 25| 1|null|

| 2| null| 7| 2| 2|null|

| 4| null| null| 3| 2|null|

| 5| null| null| null| 2|null|

+-------------+---------+-----+-------+---------+----+ For solving this question, you should be aware of the

difference between the DataFrame.union() and DataFrame.unionByName() methods. The first one

matches columns independent of their

names, just by their order. The second one matches columns by their name (which is asked for in the

---

QUESTION 2

Which of the following describes a way for resizing a DataFrame from 16 to 8 partitions in the most efficient way?

A. Use operation DataFrame.repartition(8) to shuffle the DataFrame and reduce the number of partitions.

B. Use operation DataFrame.coalesce(8) to fully shuffle the DataFrame and reduce the number of partitions.

C. Use a narrow transformation to reduce the number of partitions.

D. Use a wide transformation to reduce the number of partitions.

E. Use operation DataFrame.coalesce(0.5) to halve the number of partitions in the DataFrame.

Correct Answer: C

QUESTION 3

Which of the following describes the role of tasks in the Spark execution hierarchy?

A. Tasks are the smallest element in the execution hierarchy.

B. Within one task, the slots are the unit of work done for each partition of the data.

C. Tasks are the second-smallest element in the execution hierarchy.

D. Stages with narrow dependencies can be grouped into one task.

E. Tasks with wide dependencies can be grouped into one stage.

Correct Answer: A

Stages with narrow dependencies can be grouped into one task. Wrong, tasks with narrow dependencies can be grouped into one stage. Tasks with wide dependencies can be grouped into one stage. Wrong, since a wide transformation causes a shuffle which always marks the boundary of a stage. So, you cannot bundle multiple tasks that have wide dependencies into a stage. Tasks are the second-smallest element in the execution hierarchy. No, they are the smallest element in the execution hierarchy. Within one task, the slots are the unit of work done for each partition of the data. No, tasks are the unit of work done per partition. Slots help Spark parallelize work. An executor can have multiple slots which enable it to process multiple tasks in parallel.

QUESTION 4

The code block displayed below contains an error. The code block should read the csv file located at path data/transactions.csv into DataFrame transactionsDf, using the first row as column header and casting the columns in the most appropriate type. Find the error. First 3 rows of transactions.csv: 1.transactionId;storeId;productId;name 2.1;23;12;green grass 3.2;35;31;yellow sun 4.3;23;12;green grass Code block: transactionsDf = spark.read.load("data/transactions.csv", sep=";", format="csv", header=True)

A. The DataFrameReader is not accessed correctly.

B. The transaction is evaluated lazily, so no file will be read.

C. Spark is unable to understand the file type.

D. The code block is unable to capture all columns.

E. The resulting DataFrame will not have the appropriate schema.

Correct Answer: E

Correct code block:

transactionsDf = spark.read.load("data/transactions.csv", sep=";", format="csv", header=True,

inferSchema=True)

By default, Spark does not infer the schema of the CSV (since this usually takes some time). So, you need

to add the inferSchema=True option to the code block.

More info: pyspark.sql.DataFrameReader.csv -- PySpark 3.1.2 documentation

QUESTION 5

Which of the following code blocks reorders the values inside the arrays in column attributes of DataFrame

itemsDf from last to first one in the alphabet?

1.+------+-----------------------------+-------------------+

2.|itemId|attributes |supplier |

3.+------+-----------------------------+-------------------+

4.|1 |[blue, winter, cozy] |Sports Company Inc.|

5.|2 |[red, summer, fresh, cooling]|YetiX |

6.|3 |[green, summer, travel] |Sports Company Inc.|

7.+------+-----------------------------+-------------------+

A. itemsDf.withColumn(\\'attributes\\', sort_array(col(\\'attributes\\').desc()))

B. itemsDf.withColumn(\\'attributes\\', sort_array(desc(\\'attributes\\')))

C. itemsDf.withColumn(\\'attributes\\', sort(col(\\'attributes\\'), asc=False))

D. itemsDf.withColumn("attributes", sort_array("attributes", asc=False))

E. itemsDf.select(sort_array("attributes"))

Correct Answer: D

QUESTION 6

Which of the following describes a valid concern about partitioning?

A. A shuffle operation returns 200 partitions if not explicitly set.

B. Decreasing the number of partitions reduces the overall runtime of narrow transformations if there are more executors available than partitions.

C. No data is exchanged between executors when coalesce() is run.

D. Short partition processing times are indicative of low skew.

E. The coalesce() method should be used to increase the number of partitions.

Correct Answer: A

QUESTION 7

Which of the following code blocks shows the structure of a DataFrame in a tree-like way, containing both column names and types?

A. 1.print(itemsDf.columns) 2.print(itemsDf.types)

B. itemsDf.printSchema()

C. spark.schema(itemsDf)

D. itemsDf.rdd.printSchema()

E. itemsDf.print.schema()

Correct Answer: B

QUESTION 8

The code block displayed below contains an error. The code block is intended to perform an outer join of

DataFrames transactionsDf and itemsDf on columns productId and itemId, respectively.

Find the error.

Code block:

transactionsDf.join(itemsDf, [itemsDf.itemId, transactionsDf.productId], "outer")

A. The "outer" argument should be eliminated, since "outer" is the default join type.

B. The join type needs to be appended to the join() operator, like join().outer() instead of listing it as the last argument inside the join() call.

C. The term [itemsDf.itemId, transactionsDf.productId] should be replaced by itemsDf.itemId == transactionsDf.productId.

D. The term [itemsDf.itemId, transactionsDf.productId] should be replaced by itemsDf.col("itemId") == transactionsDf.col("productId").

E. The "outer" argument should be eliminated from the call and join should be replaced by joinOuter.

Correct Answer: C

Correct code block:

transactionsDf.join(itemsDf, itemsDf.itemId == transactionsDf.productId, "outer") Static notebook | Dynamic

notebook: See test 1, 33 (Databricks import instructions) (https://flrs.github.io/spark_practice_tests_code/

#1/33.html , https://bit.ly/sparkpracticeexams_import_instructions)

QUESTION 9

Which of the following code blocks returns a new DataFrame in which column attributes of DataFrame itemsDf is renamed to feature0 and column supplier to feature1?

A. itemsDf.withColumnRenamed(attributes, feature0).withColumnRenamed(supplier, feature1)

B. 1.itemsDf.withColumnRenamed("attributes", "feature0") 2.itemsDf.withColumnRenamed("supplier", "feature1")

C. itemsDf.withColumnRenamed(col("attributes"), col("feature0"), col("supplier"), col("feature1"))

D. itemsDf.withColumnRenamed("attributes", "feature0").withColumnRenamed("supplier", "feature1")

E. itemsDf.withColumn("attributes", "feature0").withColumn("supplier", "feature1")

Correct Answer: D

QUESTION 10

In which order should the code blocks shown below be run in order to create a table of all values in column attributes next to the respective values in column supplier in DataFrame itemsDf?

1.

itemsDf.createOrReplaceView("itemsDf")

2.

spark.sql("FROM itemsDf SELECT \\'supplier\\', explode(\\'Attributes\\')")

3.

spark.sql("FROM itemsDf SELECT supplier, explode(attributes)")

4.

itemsDf.createOrReplaceTempView("itemsDf")

A. 4, 3

B. 1, 3

C. 2

D. 4, 2

E. 1, 2

Correct Answer: A

Static notebook | Dynamic notebook: See test 1, 56 (Databricks import instructions)

QUESTION 11

Which of the following code blocks returns a new DataFrame with the same columns as DataFrame transactionsDf, except for columns predError and value which should be removed?

A. transactionsDf.drop(["predError", "value"])

B. transactionsDf.drop("predError", "value")

C. transactionsDf.drop(col("predError"), col("value"))

D. transactionsDf.drop(predError, value)

E. transactionsDf.drop("predError and value")

Correct Answer: B

QUESTION 12

Which of the following is a viable way to improve Spark\\'s performance when dealing with large amounts of data, given that there is only a single application running on the cluster?

A. Increase values for the properties spark.default.parallelism and spark.sql.shuffle.partitions

B. Decrease values for the properties spark.default.parallelism and spark.sql.partitions

C. Increase values for the properties spark.sql.parallelism and spark.sql.partitions

D. Increase values for the properties spark.sql.parallelism and spark.sql.shuffle.partitions

E. Increase values for the properties spark.dynamicAllocation.maxExecutors, spark.default.parallelism, and spark.sql.shuffle.partitions

Correct Answer: A

QUESTION 13

Which of the following code blocks reads in the JSON file stored at filePath, enforcing the schema expressed in JSON format in variable json_schema, shown in the code block below?

Code block: 1.json_schema = """ 2.{"type": "struct",

3.

"fields": [

4.

{

5.

"name": "itemId",

6.

"type": "integer",

7.

"nullable": true,

8.

"metadata": {}

9.

},

10.

{

11.

"name": "supplier",

12.

"type": "string",

13.

"nullable": true,

14.

"metadata": {}

15.

}

16.

] 17.} 18."""

A. spark.read.json(filePath, schema=json_schema)

B. spark.read.schema(json_schema).json(filePath) 1.schema = StructType.fromJson(json.loads(json_schema)) 2.spark.read.json(filePath, schema=schema)

C. spark.read.json(filePath, schema=schema_of_json(json_schema))

D. spark.read.json(filePath, schema=spark.read.json(json_schema))

Correct Answer: C

QUESTION 14

Which of the following code blocks reads in parquet file /FileStore/imports.parquet as a DataFrame?

A. spark.mode("parquet").read("/FileStore/imports.parquet")

B. spark.read.path("/FileStore/imports.parquet", source="parquet")

C. spark.read().parquet("/FileStore/imports.parquet")

D. spark.read.parquet("/FileStore/imports.parquet")

E. spark.read().format(\\'parquet\\').open("/FileStore/imports.parquet")

Correct Answer: D

QUESTION 15

Which of the following code blocks returns a DataFrame with a single column in which all items in column

attributes of DataFrame itemsDf are listed that contain the letter i?

Sample of DataFrame itemsDf:

1.+------+----------------------------------+----------------------------+-------------------+

2.|itemId|itemName |attributes |supplier |

3.+------+----------------------------------+----------------------------+-------------------+

4.|1 |Thick Coat for Walking in the Snow|[blue, winter, cozy] |Sports Company Inc.|

5.|2 |Elegant Outdoors Summer Dress |[red, summer, fresh, cooling]|YetiX |

6.|3 |Outdoors Backpack |[green, summer, travel] |Sports Company Inc.|

7.+------+----------------------------------+----------------------------+-------------------+

A. itemsDf.select(explode("attributes").alias("attributes_exploded")).filter(attributes_exploded.c ontains ("i"))

B. itemsDf.explode(attributes).alias("attributes_exploded").filter(col("attributes_exploded").con tains("i"))

C. itemsDf.select(explode("attributes")).filter("attributes_exploded".contains("i"))

D. itemsDf.select(explode("attributes").alias("attributes_exploded")).filter(col("attributes_explo ded").contains("i"))

E. itemsDf.select(col("attributes").explode().alias("attributes_exploded")).filter(col("attributes_e xploded").contains("i"))

Correct Answer: D

Latest DATABRICKS-CERTIFIED-ASSOCIATE-DEVELOPER-FOR-APACHE-SPARK Dumps

DATABRICKS-CERTIFIED-ASSOCIATE-DEVELOPER-FOR-APACHE-SPARK Practice Test

DATABRICKS-CERTIFIED-ASSOCIATE-DEVELOPER-FOR-APACHE-SPARK Exam Questions