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

Which of the following code blocks returns about 150 randomly selected rows from the 1000-row DataFrame transactionsDf, assuming that any row can appear more than once in the returned DataFrame?

A. transactionsDf.resample(0.15, False, 3142)

B. transactionsDf.sample(0.15, False, 3142)

C. transactionsDf.sample(0.15)

D. transactionsDf.sample(0.85, 8429)

E. transactionsDf.sample(True, 0.15, 8261)

Correct Answer: E

Answering this correctly depends on whether you understand the arguments to the DataFrame.sample() method (link to the documentation below). The arguments are as follows: DataFrame.sample(withReplacement=None, fraction=None, seed=None). The first argument withReplacement specified whether a row can be drawn from the DataFrame multiple times. By default, this option is disabled in Spark. But we have to enable it here, since the question asks for a row being able to appear more than once. So, we need to pass True for this argument.

About replacement: "Replacement" is easiest explained with the example of removing random items from a box. When you remove those "with replacement" it means that after you have taken an item out of the box, you put it back inside. So, essentially, if you would randomly take 10 items out of a box with 100 items, there is a chance you take the same item twice or more times. "Without replacement" means that you would not put the item back into the box after removing it. So, every time you remove an item from the box, there is one less item in the box and you can never take the same item twice. The second argument to the withReplacement method is fraction. This referes to the fraction of items that should be returned. In the we are asked for 150 out of 1000 items ?a fraction of 0.15. The last argument is a random seed. A random seed makes a randomized processed repeatable. This means that if you would re-run the same sample() operation with the same random seed, you would get the same rows returned from the sample() command. There is no behavior around the random seed specified in the question. The varying random seeds are only there to confuse you!

More info: pyspark.sql.DataFrame.sample -- PySpark 3.1.1 documentation Static notebook | Dynamic notebook: See test 1, 49 (Databricks import instructions)

QUESTION 2

Which of the following statements about stages is correct?

A. Different stages in a job may be executed in parallel.

B. Stages consist of one or more jobs.

C. Stages ephemerally store transactions, before they are committed through actions.

D. Tasks in a stage may be executed by multiple machines at the same time.

E. Stages may contain multiple actions, narrow, and wide transformations.

Correct Answer: D

QUESTION 3

The code block shown below should return a copy of DataFrame transactionsDf with an added column cos. This column should have the values in column value converted to degrees and having the cosine of those converted values taken, rounded to two decimals. Choose the answer that correctly fills the blanks in the code block to accomplish this.

Code block:

transactionsDf.__1__(__2__, round(__3__(__4__(__5__)),2))

A. 1. withColumn

2.

col("cos")

3.

cos

4.

degrees

5.

transactionsDf.value

B. 1. withColumnRenamed

2.

"cos"

3.

cos

4.

degrees

5.

"transactionsDf.value"

C. 1. withColumn

2.

"cos"

3.

cos

4.

degrees

5.

transactionsDf.value

D. 1. withColumn

2.

col("cos")

3.

cos

4.

degrees

5.

col("value")

E. 1. withColumn

2.

"cos"

3.

degrees

4.

cos

5.

col("value")

Correct Answer: C

QUESTION 4

The code block shown below should add a column itemNameBetweenSeparators to DataFrame itemsDf.

The column should contain arrays of maximum 4 strings. The arrays should be composed of the values in

column itemsDf which are separated at - or whitespace characters. Choose the answer that correctly fills

the blanks in the code block to accomplish this.

Sample of DataFrame itemsDf:

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

2.|itemId|itemName |supplier |

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

4.|1 |Thick Coat for Walking in the Snow|Sports Company Inc.|

5.|2 |Elegant Outdoors Summer Dress |YetiX |

6.|3 |Outdoors Backpack |Sports Company Inc.|

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

Code block:

itemsDf.__1__(__2__, __3__(__4__, "[\s\-]", __5__))

A. 1. withColumn

2.

"itemNameBetweenSeparators"

3.

split

4.

"itemName"

5.

4

(Correct)

B. 1. withColumnRenamed

2.

"itemNameBetweenSeparators"

3.

split

4.

"itemName"

5.

4

C. 1. withColumnRenamed

2.

"itemName"

3.

split

4.

"itemNameBetweenSeparators"

5.

4

D. 1. withColumn

2.

"itemNameBetweenSeparators"

3.

split

4.

"itemName"

5.

5

E. 1. withColumn

2.

itemNameBetweenSeparators

3.

str_split

4.

"itemName"

5.

5

Correct Answer: A

QUESTION 5

Which of the following code blocks returns a DataFrame with an added column to DataFrame transactionsDf that shows the unix epoch timestamps in column transactionDate as strings in the format month/day/year in column transactionDateFormatted?

Excerpt of DataFrame transactionsDf:

A. transactionsDf.withColumn("transactionDateFormatted", from_unixtime("transactionDate", format="dd/ MM/yyyy"))

B. transactionsDf.withColumnRenamed("transactionDate", "transactionDateFormatted", from_unixtime ("transactionDateFormatted", format="MM/dd/yyyy"))

C. transactionsDf.apply(from_unixtime(format="MM/dd/yyyy")).asColumn("transactionDateFor matted")

D. transactionsDf.withColumn("transactionDateFormatted", from_unixtime("transactionDate", format="MM/ dd/yyyy"))

E. transactionsDf.withColumn("transactionDateFormatted", from_unixtime("transactionDate"))

Correct Answer: D