DP-203 Q&As

Data Engineering on Microsoft Azure

Pass Microsoft DP-203 Exam with 100% Guarantee

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

https://www.pass4itsure.com/dp-203.html

100% Passing Guarantee
100% Money Back Assurance

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

QUESTION 1

You are creating an Apache Spark job in Azure Databricks that will ingest JSON-formatted data.

You need to convert a nested JSON string into a DataFrame that will contain multiple rows.

Which Spark SQL function should you use?

A. explode

B. filter

C. coalesce

D. extract

Correct Answer: A

Convert nested JSON to a flattened DataFrame

You can to flatten nested JSON, using only $"column.*" and explode methods.

Note: Extract and flatten

Use $"column.*" and explode methods to flatten the struct and array types before displaying the flattened DataFrame.

Scala

display(DF.select($"id" as "main_id",$"name",$"batters",$"ppu",explode($"topping")) // Exploding the topping column using explode as it is an array type withColumn("topping_id",$"col.id") // Extracting topping_id from col using DOT form

withColumn("topping_type",$"col.type") // Extracting topping_tytpe from col using DOT form drop($"col")

select($"*",$"batters.*") // Flattened the struct type batters tto array type which is batter drop($"batters") select($"*",explode($"batter"))

drop($"batter")

withColumn("batter_id",$"col.id") // Extracting batter_id from col using DOT form withColumn ("battter_type",$"col.type") // Extracting battter_type from col using DOT form drop($"col"))

Reference: https://learn.microsoft.com/en-us/azure/databricks/kb/scala/flatten-nested- columns- dynamically

QUESTION 2

DRAG DROP

You have an Azure Synapse Analytics workspace named WS1.

You have an Azure Data Lake Storage Gen2 container that contains JSON-formatted files in the following format.

```
{
    "id": "66532691-ab20-11ea-8b1d-936b3ec64e54",
    "context": {
        "data": {
            "eventTime": "2020-06-10T13:43:34.553Z",
            "samplingRate": "100.0",
            "isSynthetic": "false"
        },
        "session": {
            "isFirst": "false",
            "id": "38619c14-7a23-4687-8268-95862c5326b1"
        },
        "custom": {
            "dimensions": [
                {
                    "customerInfo": {
                        "ProfileType": "ExpertUser",
                        "RoomName": "",
                        "CustomerName": "diamond",
                        "UserName": "XXXX@yahoo.com"
                    }
                },
                {
                    "customerInfo" {
                        "ProfileType": "Novice",
                        "RoomName": "",
                        "CustomerName": "topaz",
                        "UserName": "XXXX@outlook.com"
                    }
                }
            ]
        }
    }
}
```

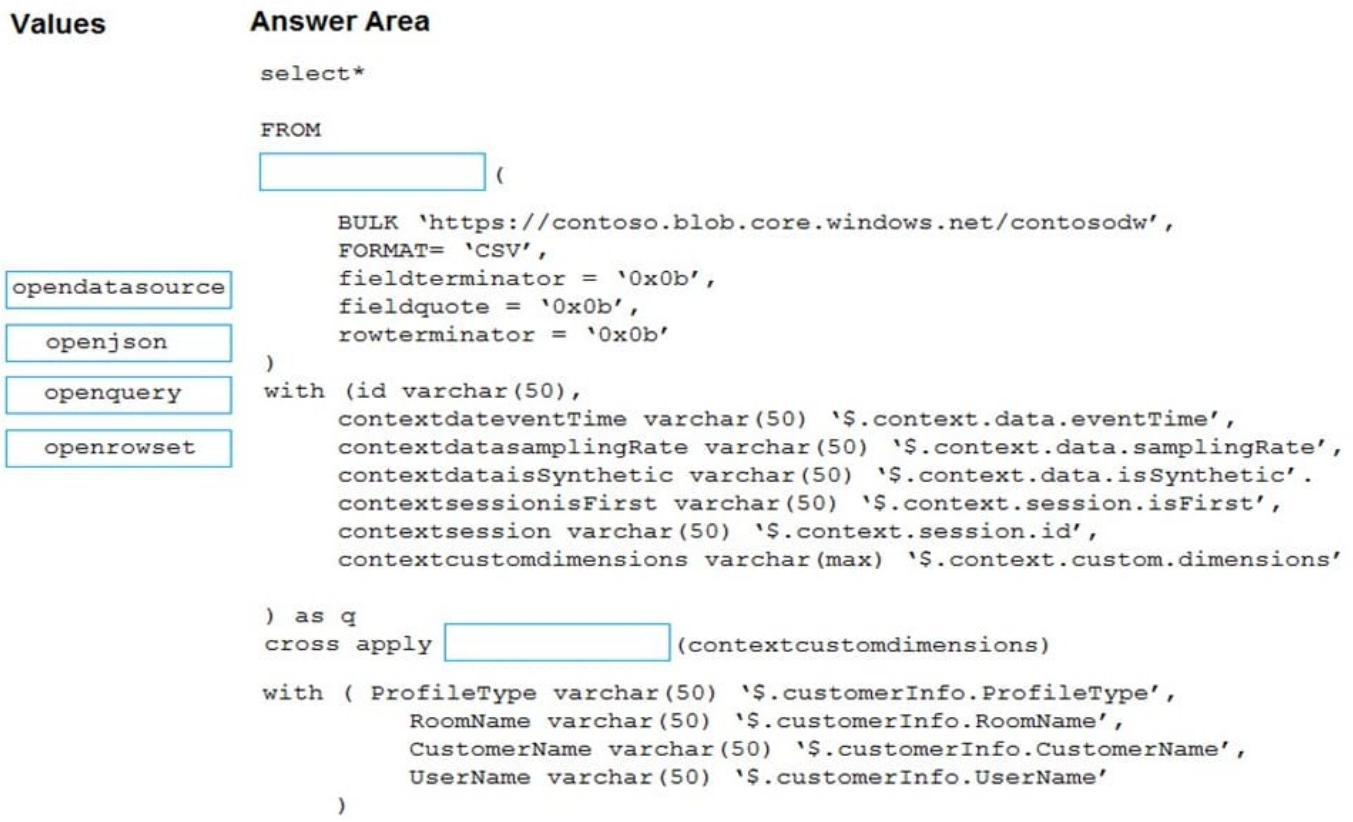

You need to use the serverless SQL pool in WS1 to read the files.

How should you complete the Transact-SQL statement? To answer, drag the appropriate values to the correct targets. Each value may be used once, more than once, or not at all. You may need to drag the split bar between panes or scroll to

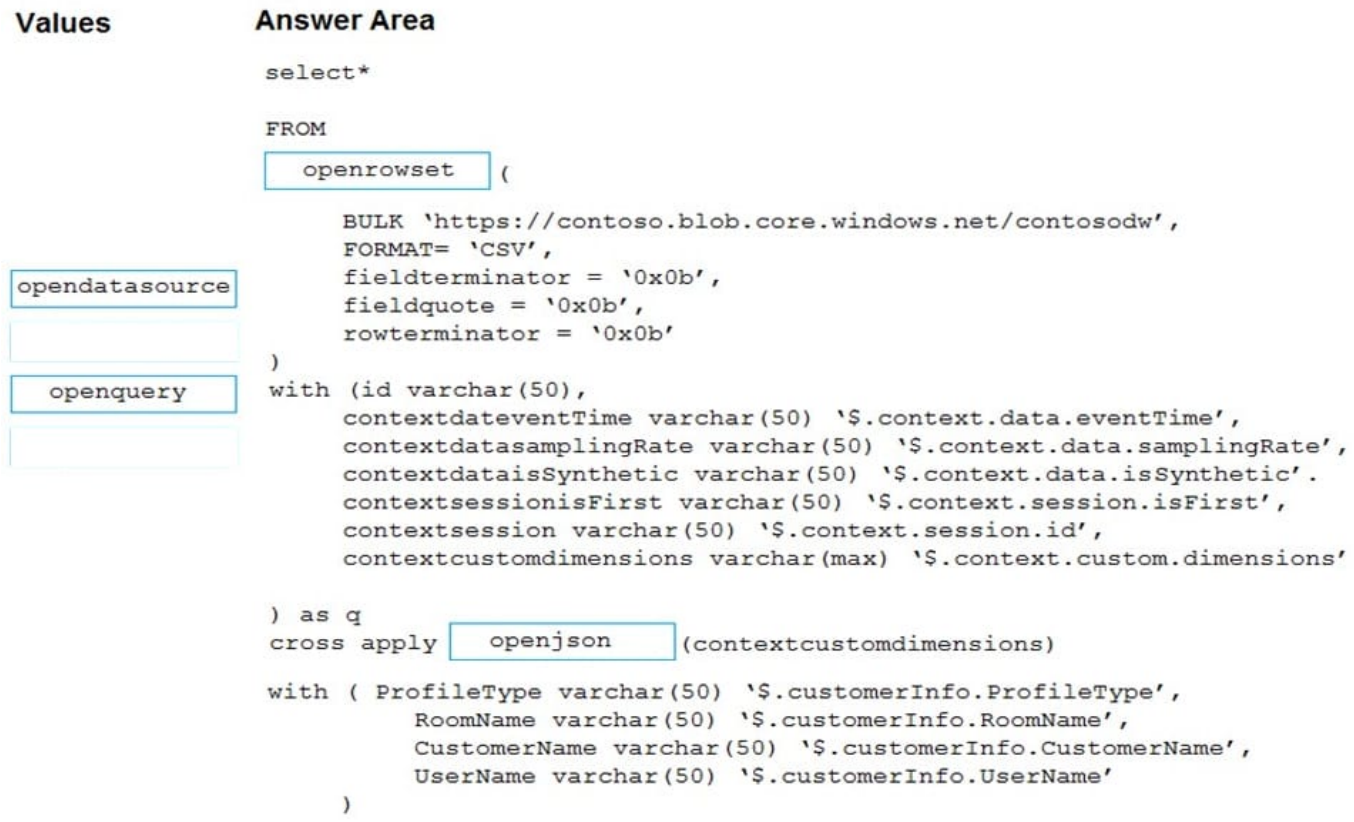
view content.

NOTE: Each correct selection is worth one point.

Select and Place:



Correct Answer:



Box 1: openrowset

The easiest way to see to the content of your CSV file is to provide file URL to OPENROWSET function, specify csv FORMAT.

Example:

SELECT *

FROM OPENROWSET(BULK \\'csv/population/population.csv\\', DATA_SOURCE = \\'SqlOnDemandDemo\\', FORMAT = \\'CSV\\', PARSER_VERSION = \\'2.0\\', FIELDTERMINATOR =\\',\\', ROWTERMINATOR = \\'\n\\'

Box 2: openjson

You can access your JSON files from the Azure File Storage share by using the mapped drive, as shown in the following example:

SELECT book.* FROM OPENROWSET(BULK N\\'t:\books\books.json\\', SINGLE_CLOB) AS json CROSS APPLY OPENJSON(BulkColumn) WITH(id nvarchar(100), name nvarchar(100), price float, pages_i int, author nvarchar(100)) AS book

Reference: https://docs.microsoft.com/en-us/azure/synapse-analytics/sql/query-single-csv-file https://docs.microsoft.com/en-us/sql/relational-databases/json/import-json-documents-into-sql-server

QUESTION 3

You are designing database for an Azure Synapse Analytics dedicated SQL pool to support workloads for detecting

ecommerce transaction fraud.

Data will be combined from multiple ecommerce sites and can include sensitive financial information such as credit card numbers.

You need to recommend a solution that meets the following requirements:

Users must be able to identify potentially fraudulent transactions. Users must be able to use credit cards as a potential feature in models. Users must NOT be able to access the actual credit card numbers.

What should you include in the recommendation?

A. Transparent Data Encryption (TDE)

B. row-level security (RLS)

C. column-level encryption

D. Azure Active Directory (Azure AD) pass-through authentication

Correct Answer: C

Use Always Encrypted to secure the required columns. You can configure Always Encrypted for individual database columns containing your sensitive data. Always Encrypted is a feature designed to protect sensitive data, such as credit card numbers or national identification numbers (for example, U.S. social security numbers), stored in Azure SQL Database or SQL Server databases.

Reference: https://docs.microsoft.com/en-us/sql/relational-databases/security/encryption/always-encrypted-database-engine

QUESTION 4

DRAG DROP

You use PySpark in Azure Databricks to parse the following JSON input.

```
{
    "persons":[
        {
            "name":"Keith",
            "age":30,
            "dogs":["Fido", "Fluffy"]
        },
        {
            "name":"Donna",
            "age":46,
            "dogs":["Spot"]
        }
    ]
}
```

You need to output the data in the following tabular format.

owner	age	dog
Keith	30	Fido
Keith	30	Fluffy
Donna	46	Spot

How should you complete the PySpark code? To answer, drag the appropriate values to the correct targets. Each value may be used once, more than once, or not at all. You may need to drag the spit bar between panes or scroll to view

content.

NOTE: Each correct selection is worth one point.

Select and Place:

Values

alias

array_union

createDataFrame

explode

select

translate

Answer Area

```
dbutils.fs.put("/tmp/source.json", source_json, True)
source_df = spark.read.option("multiline", "true").json("/tmp/source.json")
persons = source_df. [Value] [Value] ("persons").alias("persons"))
persons_dogs = persons.select(col("persons.name").alias("owner"), col("persons.age").alias("age"),
explode [Value] ("dog"))
("persons-dogs").
display(persons_dogs)
```

Correct Answer:

Values

array_union

createDataFrame

translate

Answer Area

```
dbutils.fs.put("/tmp/source.json", source_json, True)
source_df = spark.read.option("multiline", "true").json("/tmp/source.json")
persons = source_df. select explode ("persons").alias("persons"))
persons_dogs = persons.select(col("persons.name").alias("owner"), col("persons.age").alias("age"),
explode alias ("dog"))
("persons-dogs").
display(persons_dogs)
```

Box 1: select

Box 2: explode

Bop 3: alias

pyspark.sql.Column.alias returns this column aliased with a new name or names (in the case of expressions that return more than one column, such as explode).

Reference:

https://spark.apache.org/docs/latest/api/python/reference/api/pyspark.sql.Column.alias.html

https://docs.microsoft.com/en-us/azure/databricks/sql/language-manual/functions/explode

QUESTION 5

You have an Azure Data Lake Storage Gen2 container that contains 100 TB of data.

You need to ensure that the data in the container is available for read workloads in a secondary region if an outage occurs in the primary region. The solution must minimize costs.

Which type of data redundancy should you use?

A. geo-redundant storage (GRS)

B. read-access geo-redundant storage (RA-GRS)

C. zone-redundant storage (ZRS)

D. locally-redundant storage (LRS)

Correct Answer: B

Geo-redundant storage (with GRS or GZRS) replicates your data to another physical location in the secondary region to protect against regional outages. However, that data is available to be read only if the customer or Microsoft initiates a failover from the primary to secondary region. When you enable read access to the secondary region, your data is available to be read at all times, including in a situation where the primary region becomes unavailable.

Incorrect Answers:

A: While Geo-redundant storage (GRS) is cheaper than Read-Access Geo-Redundant Storage (RA-GRS), GRS does NOT initiate automatic failover. C, D: Locally redundant storage (LRS) and Zone-redundant storage (ZRS) provides redundancy within a single region.

Reference: https://docs.microsoft.com/en-us/azure/storage/common/storage-redundancy