

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.leads4pass.com/dp-203.html>

100% Passing Guarantee
100% Money Back Assurance

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

-  **Instant Download** After Purchase
-  **100% Money Back** Guarantee
-  **365 Days** Free Update
-  **800,000+** Satisfied Customers



QUESTION 1

You have an Azure Synapse Analytics de contains a fact table named Table1.

You need to identify the extent of the data skew in Table1.

What should you do in Synapse Studio?

- A. Connect to the built-in pool and run DBCC PDW_SHOWSPACEUSED.
- B. Connect to the built-in pool and run DBCC CHECKALLOC.
- C. Connect to Pool1 and query sys.dm_pdw_node_status.
- D. Connect to Pool1 and query sys.dm_pdw_nodes_db_partition_stats.

Correct Answer: D

Microsoft recommends use of sys.dm_pdw_nodes_db_partition_stats to analyze any skewness in the data.

Reference: <https://docs.microsoft.com/en-us/azure/synapse-analytics/sql-data-warehouse/sql-data-warehouse-tables-distribute>

QUESTION 2

You need to implement a Type 3 slowly changing dimension (SCD) for product category data in an Azure Synapse Analytics dedicated SQL pool. You have a table that was created by using the following Transact-SQL statement.

```
CREATE TABLE [DBO].[DimProduct] (  
  [ProductKey] [int] IDENTITY(1,1) NOT NULL,  
  [ProductSourceID] [int] NOT NULL,  
  [ProductName] [nvarchar] (100) NULL,  
  [Color] [nvarchar] (15) NULL,  
  [SellStartDate] [date] NOT NULL,  
  [SellEndDate] [date] NULL,  
  [RowInsertedDateTime] [datetime] NOT NULL,  
  [RowUpdatedDateTime] [datetime] NOT NULL,  
  [ETLAuditID] [int] NOT NULL  
)
```

Which two columns should you add to the table? Each correct answer presents part of the solution. NOTE: Each correct selection is worth one point.

- A. [EffectiveStartDate] [datetime] NOT NULL,
- B. [CurrentProductCategory] [nvarchar] (100) NOT NULL,
- C. [EffectiveEndDate] [datetime] NULL,

D. [ProductCategory] [nvarchar] (100) NOT NULL,

E. [OriginalProductCategory] [nvarchar] (100) NOT NULL,

Correct Answer: BE

A Type 3 SCD supports storing two versions of a dimension member as separate columns. The table includes a column for the current value of a member plus either the original or previous value of the member. So Type 3 uses additional columns to track one key instance of history, rather than storing additional rows to track each change like in a Type 2 SCD.

This type of tracking may be used for one or two columns in a dimension table. It is not common to use it for many members of the same table. It is often used in combination with Type 1 or Type 2 members.

CustomerID	FirstName	LastName	CurrentEmail	OriginalEmail	CompanyName	InsertedDate	ModifiedDate
2	Keith	Harris	keith0@aw.com	keith0@aw.com	Progressive Sports	2021-03-20	2021-03-20
3	Donna	Carreras	donna0@aw.com	donna0@aw.com	A Bike Store	2021-03-20	2021-03-20

CustomerID	FirstName	LastName	CurrentEmail	OriginalEmail	CompanyName	InsertedDate	ModifiedDate
2	Keith	Harris	keith0@aw.com	keith0@aw.com	Progressive Sports	2021-03-20	2021-03-20
3	Donna	Carreras	dc3@aw.com	donna0@aw.com	A Bike Store	2021-03-20	2021-03-22

Reference: <https://k21academy.com/microsoft-azure/azure-data-engineer-dp203-q-a-day-2-live-session-review/>

QUESTION 3

HOTSPOT

You are processing streaming data from vehicles that pass through a toll booth.

You need to use Azure Stream Analytics to return the license plate, vehicle make, and hour the last vehicle passed during each 10-minute window.

How should you complete the query? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

Hot Area:

WITH LastInWindow AS

```

(
    SELECT
        

|        |   |
|--------|---|
|        | ▼ |
| COUNT  |   |
| MAX    |   |
| MIN    |   |
| TOPONE |   |

 (Time) AS LastEventTime
    FROM
        Input TIMESTAMP BY Time
    GROUP BY
        

|                |   |
|----------------|---|
|                | ▼ |
| HoppingWindow  |   |
| SessionWindow  |   |
| SlidingWindow  |   |
| TumblingWindow |   |

 (minute, 10)
)
SELECT
    Input.License_plate,
    Input.Make,
    Input.Time
FROM
    Input TIMESTAMP BY Time
    INNER JOIN LastInWindow
ON
    

|          |   |
|----------|---|
|          | ▼ |
| DATEADD  |   |
| DATEDIFF |   |
| DATENAME |   |
| DATEPART |   |

 (minute, Input, LastInWindow) BETWEEN 0 AND 10
    AND Input.Time = LastInWindow.LastEventTime

```

Correct Answer:

WITH LastInWindow AS

```
(
    SELECT
        (Time) AS LastEventTime
        COUNT
        MAX
        MIN
        TOPONE
    FROM
        Input TIMESTAMP BY Time
    GROUP BY
        (minute, 10)
        HoppingWindow
        SessionWindow
        SlidingWindow
        TumblingWindow
)
SELECT
    Input.License_plate,
    Input.Make,
    Input.Time
FROM
    Input TIMESTAMP BY Time
    INNER JOIN LastInWindow
ON
    (minute, Input, LastInWindow) BETWEEN 0 AND 10
    DATEADD
    DATEDIFF
    DATENAME
    DATEPART
AND Input.Time = LastInWindow.LastEventTime
```

Box 1: MAX The first step on the query finds the maximum time stamp in 10-minute windows, that is the time stamp of the last event for that window. The second step joins the results of the first query with the original stream to find the event that match the last time stamps in each window.

Query: WITH LastInWindow AS (

```
SELECT MAX(Time) AS LastEventTime FROM Input TIMESTAMP BY Time GROUP BY TumblingWindow(minute, 10)
)
```

```
SELECT Input.License_plate, Input.Make, Input.Time
```

```
FROM Input TIMESTAMP BY Time INNER JOIN LastInWindow ON DATEDIFF(minute, Input, LastInWindow)
BETWEEN 0 AND 10 AND Input.Time = LastInWindow.LastEventTime
```

Box 2: TumblingWindow

Tumbling windows are a series of fixed-sized, non-overlapping and contiguous time intervals.

Box 3: DATEDIFF

DATEDIFF is a date-specific function that compares and returns the time difference between two DateTime fields, for more information, refer to date functions.

Reference:

<https://docs.microsoft.com/en-us/stream-analytics-query/tumbling-window-azure-stream-analytics>

QUESTION 4

You have an Azure subscription that contains an Azure SQL database named DB1 and a storage account named storage1. The storage1 account contains a file named File1.txt. File1.txt contains the names of selected tables in DB1. You need to use an Azure Synapse pipeline to copy data from the selected tables in DB1 to the files in storage1. The solution must meet the following requirements:

1.

The Copy activity in the pipeline must be parameterized to use the data in File1.txt to identify the source and destination of the copy.

2.

Copy activities must occur in parallel as often as possible.

Which two pipeline activities should you include in the pipeline? Each correct answer presents part of the solution. NOTE: Each correct selection is worth one point.

- A. If Condition
- B. ForEach
- C. Lookup
- D. Get Metadata

Correct Answer: BC

Lookup: This is a control activity that retrieves a dataset from any of the supported data sources and makes it available for use by subsequent activities in the pipeline. You can use a Lookup activity to read File1.txt from storage1 and store its content as an array variable1. ForEach: This is a control activity that iterates over a collection and executes specified activities in a loop. You can use a ForEach activity to loop over the array variable from the Lookup activity and pass each table name as a parameter to a Copy activity that copies data from DB1 to storage11.

QUESTION 5**DRAG DROP**

You need to ensure that the Twitter feed data can be analyzed in the dedicated SQL pool. The solution must meet the customer sentiment analytic requirements.

Which three Transact-SQL DDL commands should you run in sequence? To answer, move the appropriate commands from the list of commands to the answer area and arrange them in the correct order.

NOTE: More than one order of answer choices is correct. You will receive credit for any of the correct orders you select.

Select and Place:

Commands

- CREATE EXTERNAL DATA SOURCE
- CREATE EXTERNAL FILE FORMAT
- CREATE EXTERNAL TABLE
- CREATE EXTERNAL TABLE AS SELECT
- CREATE DATABASE SCOPED CREDENTIAL

Answer Area

Correct Answer:

Commands

-
-
- CREATE EXTERNAL TABLE
-
- CREATE DATABASE SCOPED CREDENTIAL

Answer Area

- CREATE EXTERNAL DATA SOURCE
- CREATE EXTERNAL FILE FORMAT
- CREATE EXTERNAL TABLE AS SELECT

Scenario: Allow Contoso users to use PolyBase in an Azure Synapse Analytics dedicated SQL pool to query the content of the data records that host the Twitter feeds. Data must be protected by using row-level security (RLS). The users must

be authenticated by using their own Azure AD credentials.

Box 1: CREATE EXTERNAL DATA SOURCE

External data sources are used to connect to storage accounts.

Box 2: CREATE EXTERNAL FILE FORMAT

CREATE EXTERNAL FILE FORMAT creates an external file format object that defines external data stored in Azure Blob Storage or Azure Data Lake Storage. Creating an external file format is a prerequisite for creating an external table.

Box 3: CREATE EXTERNAL TABLE AS SELECT

When used in conjunction with the CREATE TABLE AS SELECT statement, selecting from an external table imports data into a table within the SQL pool. In addition to the COPY statement, external tables are useful for loading data.

Incorrect Answers:

CREATE EXTERNAL TABLE

The CREATE EXTERNAL TABLE command creates an external table for Synapse SQL to access data stored in Azure Blob Storage or Azure Data Lake Storage.

Reference:

<https://docs.microsoft.com/en-us/azure/synapse-analytics/sql/develop-tables-external-tables>

[DP-203 VCE Dumps](#)

[DP-203 Exam Questions](#)

[DP-203 Braindumps](#)