PassQueen
a.k.a dumpsqueen.com

Implementing Analytics Solutions Using Microsoft Fabric

Microsoft DP-600

Version Demo

Total Demo Questions: 24

Total Premium Questions: 240

Buy Premium PDF

https://passqueen.com

support@passqueen.com

Topic Break Down

Topic	No. of Questions
Topic 1, Implement and manage an analytics solution	121
Topic 2, Ingest and transform data	88
Topic 3, Monitor and optimize an analytics solution	31
Total	240

QUESTION NO: 1

Which type of data store should you recommend in the AnalyticsPOC workspace?

A. a data lake

B. a warehouse

C. a lakehouse

D. an external Hive metaStore

ANSWER: C

Explanation:

A lakehouse is the appropriate recommendation for the AnalyticsPOC workspace because it provides a unified Fabric data store for both data engineering and analytical workloads. A Fabric lakehouse stores files in OneLake and organizes managed tables using the Delta Lake format, which provides reliable, transactional tables suitable for preparing and governing analytical data. It also includes an automatically generated SQL analytics endpoint, allowing users to query lakehouse tables with T-SQL and build reporting solutions without copying the data into a separate platform.

This combination is particularly well suited to a proof-of-concept analytics workspace: data can be ingested, transformed with notebooks or Spark jobs, queried through SQL, and used directly by semantic models and Power BI reports. The lakehouse therefore supports an end-to-end analytics workflow while retaining open data formats and a single logical copy of the data in OneLake. Microsoft describes Fabric lakehouses as a platform that combines the scalable storage of a data lake with the querying capabilities needed for business intelligence and analytics. See Microsoft Fabric lakehouse overview and SQL analytics endpoint documentation.

QUESTION NO: 2

You have a Fabric tenant that contains a workspace named Workspace^ Workspace is assigned to a Fabric capacity.

You need to recommend a solution to provide users with the ability to create and publish custom Direct Lake semantic models by using external tools. The solution must follow the principle of least privilege.

Which three actions in the Fabric Admin portal should you include in the recommendation? Each correct answer presents part of the solution.

NOTE: Each correct answer is worth one point.

A. From the Tenant settings, set Allow XMLA Endpoints and Analyze in Excel with on-premises datasets to Enabled

B. From the Tenant settings, set Allow Azure Active Directory guest users to access Microsoft Fabric to Enabled

C. From the Tenant settings, select Users can edit data models in the Power BI service.

D. From the Capacity settings, set XMLA Endpoint to Read Write

E. From the Tenant settings, set Users can create Fabric items to Enabled

F. From the Tenant settings, enable Publish to Web

ANSWER: A C D

Explanation:

Custom Direct Lake semantic models can be authored and deployed through XMLA-compatible external tools, such as Tabular Editor. The tenant setting *Allow XMLA Endpoints and Analyze in Excel with on-premises datasets* permits use of XMLA endpoints for the applicable users. To meet least-privilege requirements, the setting should be assigned to a specific security group containing only the required model authors, rather than enabled for the entire organization.

Users can edit data models in the Power BI service must also be enabled for those authorized users. This tenant-level control permits semantic-model editing activities and should likewise be scoped to the smallest suitable group.

The capacity XMLA endpoint must be configured as *Read Write*. Read-write XMLA access is what permits an external tool to publish metadata and model changes to a semantic model in a workspace assigned to that capacity. In addition to these administrative settings, users need appropriate workspace permissions, such as Member or Contributor as appropriate for the publishing workflow. Microsoft documents XMLA endpoint configuration and external-tool connectivity at Connect to semantic models with XMLA endpoints and documents Fabric tenant settings at Tenant settings in Microsoft Fabric.

QUESTION NO: 3

You have a Fabric workspace named Workspace 1 that contains a dataflow named Dataflow1. Dataflow1 has a query that returns 2.000 rows. You view the query in Power Query as shown in the following exhibit.

What can you identify about the pickupLongitude column?

A. The column has duplicate values.

B. All the table rows are profiled.

C. The column has missing values.

D. There are 935 values that occur only once.

ANSWER: A

Explanation:

The column has duplicate values. Power Query's column distribution profiling shows both the total count of non-error, non-empty values assessed and the number of distinct values in the profiled data. In the exhibit, the pickupLongitude column has a count of 1,000 and a distinct count of 935. Since the number of distinct values is lower than the total value count, at least some longitude values must repeat. In other words, 1,000 observed values cannot be represented by only 935 unique values unless duplicate occurrences are present.

The query itself returns 2,000 rows, but Power Query profiling commonly evaluates the first 1,000 rows by default unless the profiling setting is changed to use the entire data set. Therefore, the displayed counts establish duplication within the profiled sample, which is sufficient to identify duplicate values in pickupLongitude. The Column distribution view is specifically intended to expose statistics such as distinct and unique counts, helping identify repetition and cardinality characteristics before data is loaded or transformed. For details, see Data profiling tools in Power Query and Power Query column distribution.

QUESTION NO: 4

You have a Fabric warehouse that contains a fact table named FactSales. You need to improve the performance of queries that commonly filter FactSales by OrderDate and aggregate sales by ProductKey.

Which two actions should you perform? Each correct answer presents part of the solution.

NOTE: Each correct answer is worth one point.

A. Create statistics on OrderDate.

B. Create statistics on ProductKey.

C. Create a clustered columnstore index on FactSales.

D. Create a trigger on FactSales.

E. Create a primary XML index on FactSales.

ANSWER: A B C

Explanation:

Creating statistics on the columns used in predicates and joins helps the warehouse query optimizer estimate row counts and select more efficient query plans. In this scenario, statistics on **OrderDate** and **ProductKey** provide useful information for date filtering and product-based aggregation or joins.

Creating a **clustered columnstore index** on the large fact table is also appropriate for an analytical workload. Columnstore storage is optimized for scanning and aggregating large volumes of fact data, and it can reduce I/O for queries that access only selected columns. A rowstore index is generally more suitable for selective transactional lookups than broad analytical scans. For more information, see Tables in Microsoft Fabric Data Warehouse and Columnstore indexes overview.

QUESTION NO: 5

You have a Fabric workspace named Workspace1 that contains a warehouse named Warehouse1 and a lakehouse named Lakehouse1. Warehouse 1 contains a table named Table!. Lakehouse! contains a delta table named Table2.

You need to persist data from Table! to Table2. The solution must use a low-code interface. What should you do?

A. Use a notebook to read the data from Table2 and insert the data into Tabid.

B. Use a notebook to read the data from Table! and insert the data into Table2.

C. Use a Dataflow Gen2 to read the data from Table! and insert the data into Table2.

D. Use a stored procedure to read the data from Table2 and insert the data into Tabid.

ANSWER: C

Explanation:

Use a Dataflow Gen2 to read the data from Table! and insert the data into Table2. Dataflow Gen2 is Fabric's low-code data-integration experience and uses the Power Query visual authoring interface. It can connect to supported Fabric data sources, apply transformations through the graphical interface, and load the resulting data to a configured destination without requiring the author to write a notebook or procedural SQL code. In this scenario, the warehouse table is the source and the Delta table in the lakehouse is the target, so a Dataflow Gen2 provides the appropriate extract, transform, and load pattern while meeting the low-code requirement. The dataflow can be configured with a Warehouse connector for the source query and a Lakehouse destination that writes to the target table. It can also be refreshed on demand or orchestrated as part of a Fabric pipeline when recurring persistence is required. Microsoft documents Dataflow Gen2 as a Power Query-based tool for ingesting and transforming data into Fabric destinations, including lakehouses. See Dataflow Gen2 overview and Dataflow Gen2 data destinations.

QUESTION NO: 6

You need to implement the date dimension in the data store. The solution must meet the technical requirements.

What are two ways to achieve the goal? Each correct answer presents a complete solution.

NOTE: Each correct selection is worth one point.

A. Populate the date dimension table by using a dataflow.

B. Populate the date dimension table by using a Stored procedure activity in a pipeline.

C. Populate the date dimension view by using T-SQL.

D. Populate the date dimension table by using a Copy activity in a pipeline.

ANSWER: A B

Explanation:

Populate the date dimension table by using a dataflow is a complete solution because a Dataflow Gen2 can create and transform data with Power Query, then load the resulting rows into a Fabric destination. A date range can be generated programmatically and enriched with required calendar attributes, such as year, quarter, month, week, day, and fiscal-period columns. This makes the date dimension repeatable and maintainable without depending on a source application to provide calendar records.

Populate the date dimension table by using a Stored procedure activity in a pipeline is also a complete solution. A stored procedure can use T-SQL to generate the required date sequence and insert or refresh the dimension table in the data store. The pipeline activity provides orchestration, allowing the procedure to run on a schedule or as part of a broader warehouse-load workflow. This approach is especially appropriate when the dimensional model and its data-loading logic are managed in SQL. Both methods produce a persisted date dimension table that can support relationships and analytical filtering in the semantic model. For implementation details, see Dataflow Gen2 overview and Execute a stored procedure from a pipeline.

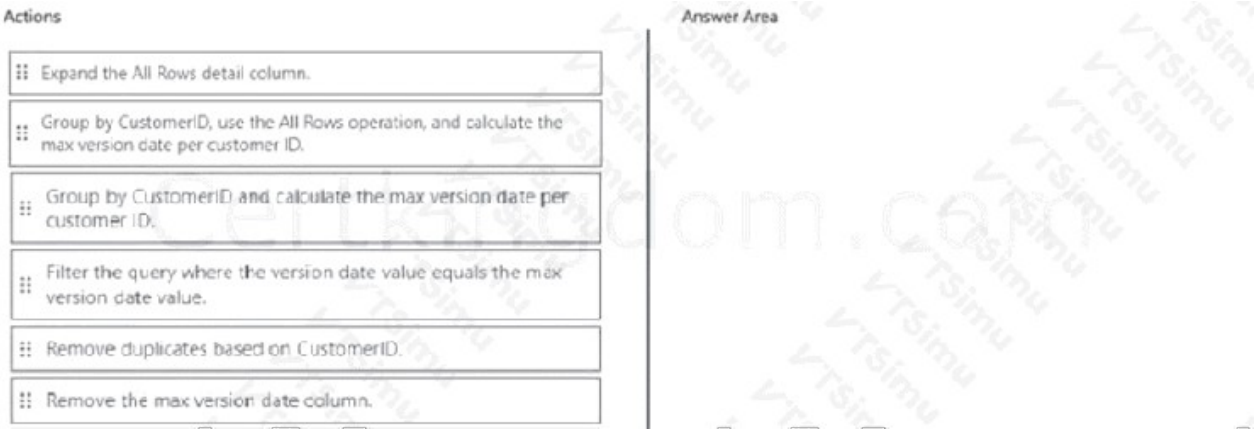
QUESTION NO: 7 - (DRAG DROP)

You have a Fabric workspace that contains a Dataflow Gen2 query. The query returns the following data.

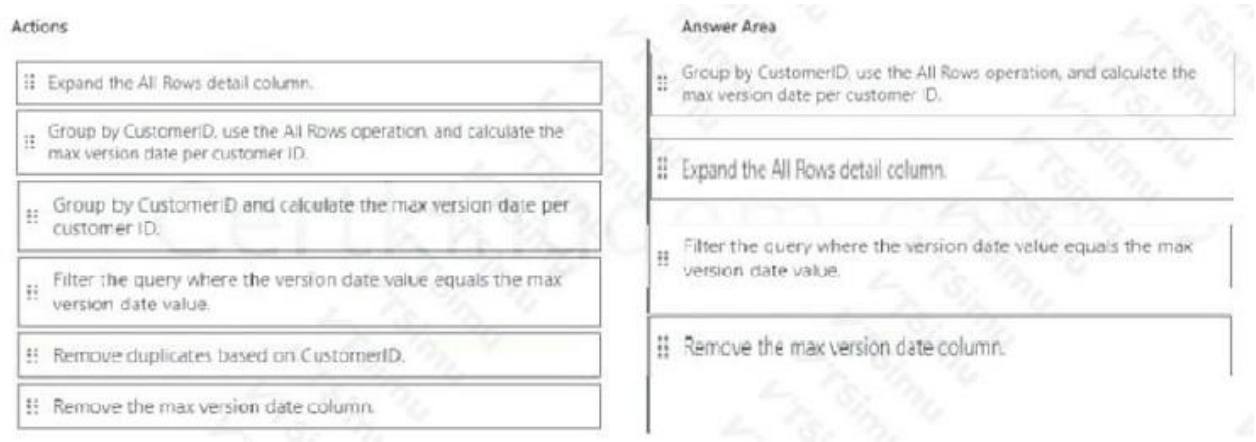
Column name	Data type	Description
CustomerID	Whole number	Contains unique customer IDs
CustomerName	Text	Contains customer names
VersionDate	Date/Time	Contains record timestamps

You need to filter the results to ensure that only the latest version of each customer's record is retained. The solution must ensure that no new columns are loaded to the semantic model.

Which four actions should you perform in sequence in Power Query Editor? To answer, move the appropriate actions from the list of actions to the answer area and arrange them in the correct order.



ANSWER:



Explanation:

Use Power Query grouping to calculate the latest date without discarding the customer-record detail that must ultimately be retained. Grouping by **CustomerID** with the **All Rows** operation creates a nested table containing the source rows for each customer. At the same time, calculate the maximum **VersionDate** for that customer. This provides both the complete detail needed for the final result and a per-customer value that identifies which record is current.

Next, expand the All Rows detail column so the nested customer records are available again as normal query columns. Filter the expanded rows so that **VersionDate** equals the calculated maximum version-date value. At that point, the query contains the latest dated record for every CustomerID while preserving the original fields from the source data. Finally, remove the calculated maximum version-date column. It was necessary as a temporary comparison value during filtering, but removing it before the dataflow output prevents that helper field from being loaded to the semantic model.

This approach uses the Group By capability to keep row-level details through an aggregation, rather than reducing the result to an aggregated customer-level table. Microsoft documents that grouping can create an **All Rows** result containing the rows for each group, which can subsequently be expanded and transformed. See Group rows in Power Query and Expand a column containing an associated table.

QUESTION NO: 8

You have a Fabric tenant that contains a semantic model.

You need to prevent report creators from populating visuals by using implicit measures.

What are two tools that you can use to achieve the goal? Each correct answer presents a complete

solution.

NOTE: Each correct answer is worth one point.

A. Microsoft Power BI Desktop

B. Tabular Editor

C. Microsoft SQL Server Management Studio (SSMS)

D. DAX Studio

ANSWER: A B

Explanation:

Microsoft Power BI Desktop and Tabular Editor can both set the semantic-model property that discourages implicit measures. Implicit measures are aggregations that Power BI automatically creates when a numeric column is placed in a visual, such as an automatically generated sum. Setting the model to discourage implicit measures ensures report creators use explicitly authored DAX measures instead, which provides more consistent calculation logic, naming, formatting, and governance across reports that use the semantic model.

In Microsoft Power BI Desktop, this setting is available in the semantic model's properties and is saved with the model when it is published. Tabular Editor can also edit the semantic model metadata directly, including the `DiscourageImplicitMeasures` property. This makes it particularly useful when managing model settings through external tooling or metadata-based development workflows. After the property is enabled and the updated model is published, report authors consuming that semantic model are guided away from automatic aggregations and toward the approved explicit measures.

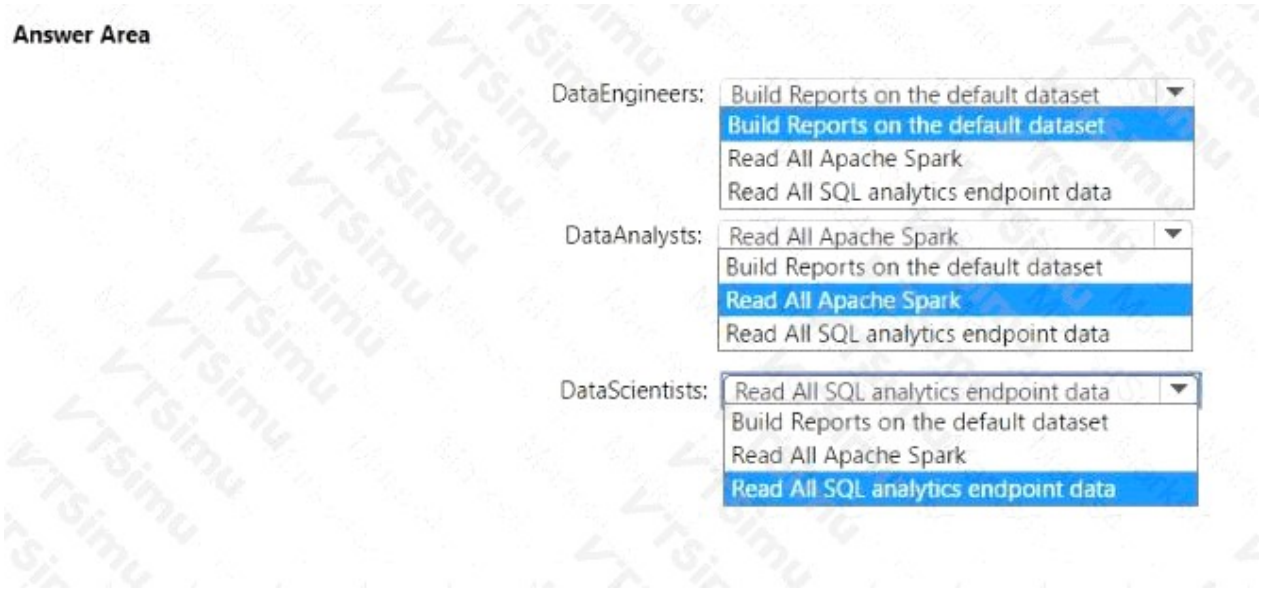
Microsoft documents the behavior and configuration of this semantic-model setting in Implicit measures in Power BI Desktop. For metadata editing through the external tool, see the Tabular Editor documentation.

QUESTION NO: 9 - (HOTSPOT)

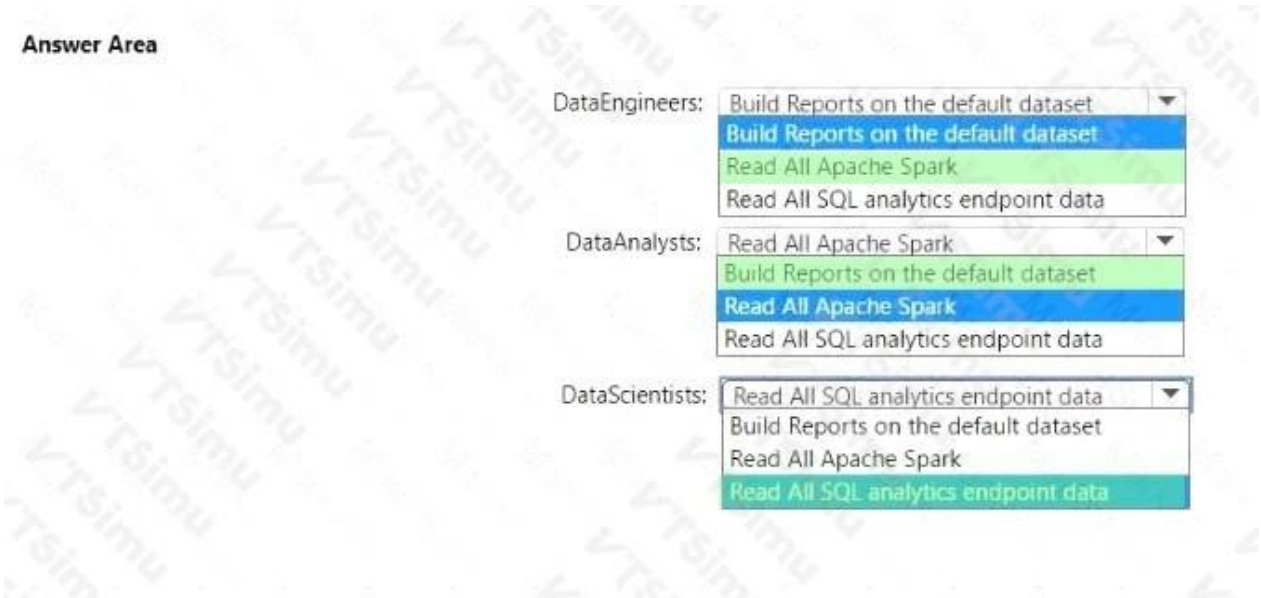
You to need assign permissions for the data store in the AnalyticsPOC workspace. The solution must meet the security requirements.

Which additional permissions should you assign when you share the data store? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.



ANSWER:



Explanation:

Share permissions on a Microsoft Fabric lakehouse are designed to grant access through a particular consumption experience without broadly changing workspace-level access. The correct assignments give each team access through the interface that supports its intended work while keeping permissions scoped to the shared data store.

DataEngineers should receive **Read All Apache Spark**. This grants read access to lakehouse tables from Apache Spark workloads, including notebooks and Spark-based engineering processes. It supports engineering activities that read and transform data using Fabric's Spark runtime and the lakehouse's Spark-facing data access path.

DataAnalysts should receive **Build Reports on the default dataset**. A lakehouse can expose a default semantic model for reporting. This permission enables analysts to create Power BI reports from that semantic model, allowing reporting and self-service analysis to be based on the governed model associated with the shared data store.

DataScientists should receive **Read All SQL analytics endpoint data**. The SQL analytics endpoint provides a SQL-based read interface over lakehouse data. Granting this permission lets the data science team read the relevant data through that endpoint for SQL-oriented exploration, data retrieval, and analytical workflows.

These item-level sharing permissions are additive and are separate from Fabric workspace roles. Microsoft documents lakehouse sharing permissions and the access experiences they enable in Share a lakehouse in Microsoft Fabric. For details on the reporting model created for a lakehouse, see Default Power BI semantic models in Microsoft Fabric. The SQL access surface is described in SQL analytics endpoint of a lakehouse.

QUESTION NO: 10

You have a Fabric workspace named Workspace1 that is assigned to a newly created Fabric capacity named Capacity1.

You create a semantic model named Model1 and deploy Model1 to Workspace1.

You need to publish changes to Model1 directly from Tabular Editor.

What should you do?

A. For Workspace1, enable Git integration.

B. For Workspace1, create a managed private endpoint.

C. For Model1, enable external sharing.

D. For Capacity1, set XMLA Endpoint to Read Write.

ANSWER: D

Explanation:

For Capacity1, set XMLA Endpoint to Read Write. Tabular Editor is an external Analysis Services modeling tool that connects to Power BI and Fabric semantic models through the XMLA endpoint. While an XMLA endpoint supports metadata and query operations, publishing model changes—such as modifying tables, measures, relationships, perspectives, roles, or other tabular metadata—requires write access. Setting the capacity-level XMLA Endpoint property to Read Write enables write operations against semantic models hosted in workspaces assigned to that capacity.

After this capacity setting is enabled, an authorized user can connect Tabular Editor to the workspace XMLA endpoint and deploy changes to Model1. The user must also have sufficient workspace permissions to modify the semantic model. XMLA write operations are available for semantic models in workspaces on an eligible Fabric or Power BI Premium capacity; they are configured in the capacity settings rather than as a property of the individual workspace or semantic model. This configuration supports professional semantic-model development workflows while retaining the model in the Fabric workspace.

For implementation details and requirements, see Connect to semantic models with the XMLA endpoint and Manage Microsoft Fabric and Power BI capacities.

QUESTION NO: 11

You have a Fabric tenant that contains a workspace named Workspace1. Workspace1 uses Pro license mode and contains a semantic model named Model1. You need to ensure that Modell supports XMLA connections. Which setting should you modify?

A. Enable Cache for Shortcuts

B. License mode

C. Enforce strict access control for all data connection types

D. Users can edit data models in the Power BI service

ANSWER: B

Explanation:

License mode is the setting to modify because XMLA endpoint connectivity for a Power BI semantic model depends on the workspace being assigned to a capacity-backed or Premium Per User license mode, rather than remaining in Pro mode. XMLA provides a standard endpoint through which supported client tools and applications can connect to semantic models for operations such as querying metadata, managing models, and—where the assigned capacity and tenant settings permit it—read/write operations. A Pro workspace does not provide the XMLA endpoint capabilities required for these connections. Move Workspace1 to an appropriate supported mode, such as Premium Per User or a Fabric/Premium capacity that meets the XMLA requirements. After assignment, the semantic model can be addressed through the workspace XMLA endpoint; any applicable tenant and capacity XMLA configuration must also allow the intended access level. This is a workspace licensing/capacity decision, so the relevant workspace setting is License mode. Microsoft documents XMLA endpoint requirements and supported client connectivity in Connect to Power BI semantic models with the XMLA endpoint. For details on assigning a workspace to the available licensing modes, see Workspaces in Microsoft Fabric.

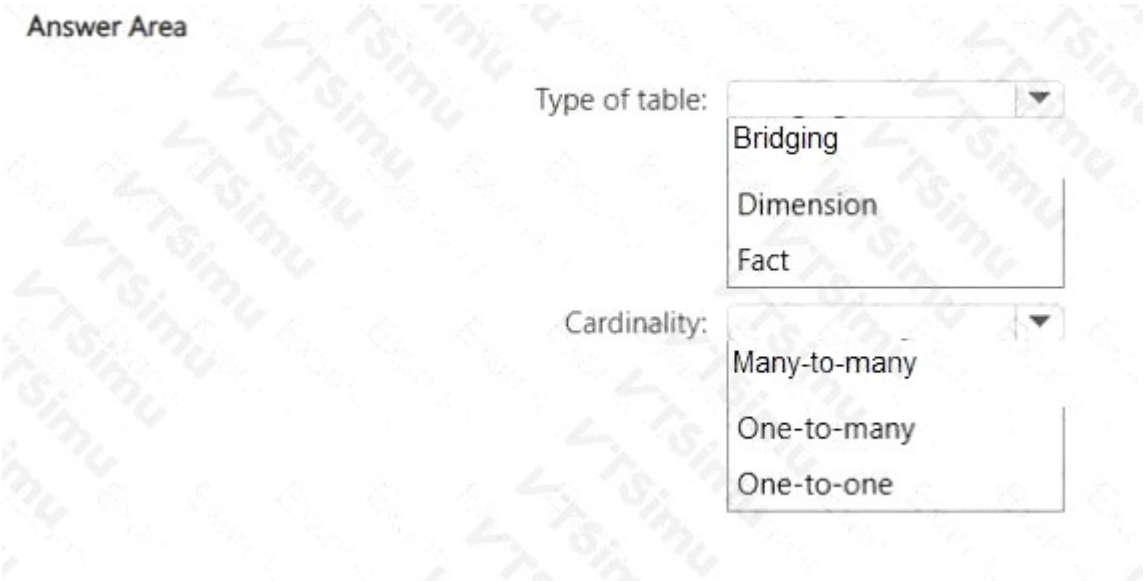
QUESTION NO: 12 - (HOTSPOT)

You have a Fabric tenant that contains a semantic model named Modell. Model! contains the following tables.

Name	Description
Sales	Contains three columns named SalesDate, StoreID, and ProductID
Product	Contains a column named ProductID
Store	Contains a column named StoreID
Campaign	Contains six columns named CampaignDate, CampaignID, ProductID, StoreID, Name, and Description

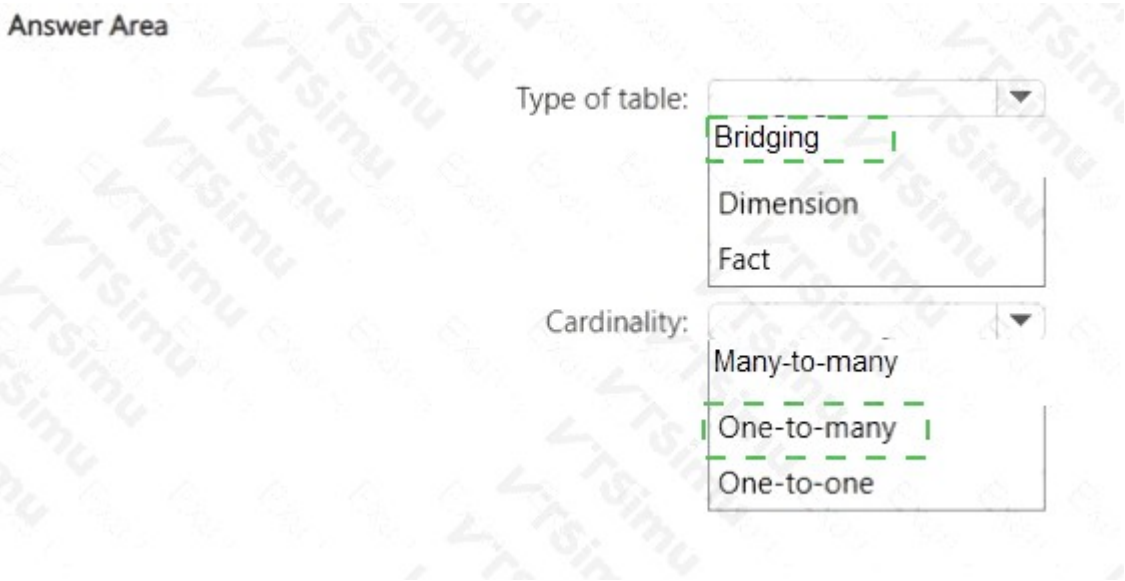
Product and Store can be part of multiple campaigns, and multiple active campaigns can exist simultaneously. You plan to create a report named Report1 that will display sales made after a campaign for a given store. You need to add a table to Modell and configure cardinality for the tables relationships. What should you configure? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.



ANSWER:

Answer



Answer:

Explanation:

The added table should be a **bridging** table, with relationships configured as **one-to-many**. A bridge is appropriate because campaign participation is inherently an association between entities rather than a simple descriptive attribute of either Product or Store. The Campaign data records campaign information alongside ProductID and StoreID, and the scenario explicitly allows each product and each store to occur in multiple campaigns. It also allows several campaigns to be active at the same time. This means campaign-related rows can legitimately repeat a product, a store, or a product/store combination over time and across campaign instances.

A bridging design preserves those multiple associations and gives the model a reliable way to evaluate sales in the context of campaign activity. The bridge can retain the campaign identifier and campaign date while connecting the relevant store and product context. Sales can then be analyzed according to the campaign context required by the report, including determining sales occurring after a campaign date for a specified store. The relationships from the entity tables that contain unique identifiers to the bridge should be one-to-many because one product or store can have many associated campaign rows, while each bridge row identifies one related product and store association.

This approach follows Microsoft guidance for modeling many-to-many business associations using a bridge table and one-to-many relationships from dimensions into that bridge. It maintains clear filter propagation and avoids treating repeated campaign associations as though they were unique dimension values. See Many-to-many relationship guidance in Power BI and Understand star schema and the importance for Power BI.

QUESTION NO: 13

You need to create a data loading pattern for a Type 1 slowly changing dimension (SCD). Which two actions should you include in the process? Each correct answer presents part of the solution.

NOTE: Each correct answer is worth one point.

A. Update rows when the non-key attributes have changed.

B. Insert new rows when the natural key exists in the dimension table, and the non-key attribute values have changed.

C. Update the effective end date of rows when the non-key attribute values have changed.

D. Insert new records when the natural key is a new value in the table.

ANSWER: A D

Explanation:

A Type 1 slowly changing dimension retains only the current version of each dimension member. The loading process identifies a member by its natural (business) key and compares incoming descriptive, non-key attribute values with the existing dimension row. When those attribute values have changed, the existing row is updated in place. This overwrites the

prior descriptive values, so the dimension remains a current-state representation and does not preserve historical versions of the member.

The process must also accommodate dimension members that have not previously been loaded. When an incoming natural key does not match any existing dimension row, the process inserts a new record for that member. Together, updating matched members with changed attributes and inserting members with new natural keys provides the standard Type 1 dimension loading pattern.

Microsoft describes Type 1 dimensions as overwriting changed attribute values without retaining history. The SSIS Slowly Changing Dimension guidance likewise identifies updating the changing attribute in the existing dimension record as the Type 1 behavior. See Microsoft guidance on slowly changing dimensions and Slowly Changing Dimension transformation documentation.

QUESTION NO: 14

You have a Fabric workspace that contains a lakehouse named Lakehouse1.

You need to enable a business analyst to create reports by using the default semantic model of Lakehouse1. The analyst must be able to create reports but must NOT be able to modify the lakehouse or its tables.

Which two actions should you perform? Each correct answer presents part of the solution.

NOTE: Each correct answer is worth one point.

A. Assign the Viewer role for the workspace to the analyst.

B. Grant Build permission on the default semantic model to the analyst.

C. Assign the Contributor role for the workspace to the analyst.

D. Assign the Member role for the workspace to the analyst.

E. Grant Write permission on the lakehouse to the analyst.

ANSWER: A B

Explanation:

Assigning the analyst the **Viewer** role for the workspace provides read-only access to workspace content without allowing the analyst to create, edit, or delete lakehouse items. This prevents the analyst from modifying Lakehouse1 and its tables.

Granting **Build** permission on the default semantic model enables the analyst to create new reports that use the semantic model. Build permission is required for users to create content based on an existing semantic model when they do not have a workspace role that already grants equivalent access. Together, Viewer access and Build permission satisfy the reporting requirement while preserving the least-privileged boundary. For more information, see Roles in workspaces and Build permission for shared semantic models.

QUESTION NO: 15 - (SIMULATION)

You are implementing a medallion architecture in a single Fabric workspace.

You have a lakehouse that contains the Bronze and Silver layers and a warehouse that contains the Gold layer.

You create the items required to populate the layers as shown in the following table.

You need to ensure that the layers are populated daily in sequential order such that Silver is populated only after Bronze is complete, and Gold is populated only after Silver is complete. The solution must minimize development effort and complexity.

What should you use to execute each set of items? To answer, drag the appropriate options to the correct items. Each option 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.

ANSWER: See the explanation for the answer

Explanation:

Use a Fabric data pipeline to orchestrate all three layers. Create a single scheduled pipeline and configure activity dependencies so that each activity starts only after the preceding activity succeeds.

Bronze: Use a **Copy data** activity to run the Bronze ingestion pipeline(s).
Silver: Use a **Dataflow** activity to run the Silver dataflow(s), with a `Succeeded` dependency on the Bronze activity.
Gold: Use a **Stored procedure** activity to execute the Gold warehouse stored procedure(s), with a `Succeeded` dependency on the Silver activity.

Schedule the orchestration pipeline to run daily. This provides Bronze → Silver → Gold sequencing without implementing custom scheduling or dependency logic in notebooks/scripts, and ensures Gold does not execute unless Silver succeeds and Silver does not execute unless Bronze succeeds.

QUESTION NO: 16

You have a Fabric tenant that contains customer churn data stored as Parquet files in OneLake. The

data contains details about customer demographics and product usage.

You create a Fabric notebook to read the data into a Spark DataFrame. You then create column charts in the notebook that show the distribution of retained customers as compared to lost customers

based on geography, the number of products purchased, age. and customer tenure. Which type of analytics are you performing?

A. prescriptive

B. diagnostic

C. descriptive

D. predictive

ANSWER: C

Explanation:

descriptive is correct because the notebook uses existing customer churn data to summarize and visualize observed outcomes. The column charts compare the distribution of retained and lost customers across geography, number of products purchased, age, and customer tenure. This presents patterns in the data and answers questions about what has occurred or what the current data shows, without producing a forecast, recommendation, or causal conclusion.

In Microsoft Fabric, a notebook can use Spark to load Parquet files from OneLake into a DataFrame and generate charts for exploratory analysis. Those capabilities support examining and communicating data distributions. In this case, the analysis remains descriptive because the results are aggregations and visual comparisons of historical customer status by demographic and usage-related attributes. A churn prediction model is not trained or applied; instead, the notebook reports the observed relationship between customer segments and churn outcomes.

Microsoft describes descriptive analytics as the process of using historical data to understand and summarize what happened. Visualizations such as column charts are a common way to communicate those summaries to stakeholders. For more information, see Analytics types in Power BI and How to use Microsoft Fabric notebooks.

QUESTION NO: 17

You have a Fabric tenant that contains a warehouse.

A user discovers that a report that usually takes two minutes to render has been running for 45 minutes and has still not rendered.

You need to identify what is preventing the report query from completing.

Which dynamic management view (DMV) should you use?

A. sys.dm_exec_requests

B. sys.dn_.exec._sessions

C. sys.dm._exec._connections

D. sys.dm_pdw_exec_requests

ANSWER: A

Explanation:

`sys.dm_exec_requests` is the appropriate DMV for investigating an active report query that is taking unexpectedly long to finish. It exposes information about requests currently executing in the Fabric Warehouse, including the request status, start time, elapsed time, command, session ID, and details that help identify waits or blocking. In particular, the request data can be correlated with session information and used to determine whether the query is waiting on another session, consuming resources, or otherwise stalled during execution. This makes it the starting point for diagnosing why a currently running report query has not completed.

For a report that has been rendering for 45 minutes, query this DMV while the report is still active and locate the long-running request by its session, user, command text, or elapsed duration. Review the request's current state and blocking-related information to identify the immediate condition preventing completion. Microsoft documents `sys.dm_exec_requests` as a Warehouse DMV for monitoring active requests and query execution activity. See Monitor connections, sessions, and requests using DMVs in Fabric Data Warehouse and sys.dm_exec_requests (Transact-SQL).

QUESTION NO: 18

You have a Fabric eventstream that receives telemetry events from IoT devices.

You need to transform the event data before it is stored. The solution must remove events that have a null DeviceId and calculate the average Temperature value by device for each five-minute window.

Which two eventstream transformations should you use? Each correct answer presents part of the solution.

NOTE: Each correct answer is worth one point.

A. Filter

B. Aggregate

C. Manage fields

D. Union

E. Expand

ANSWER: A B

Explanation:

A **Filter** transformation can remove events for which `DeviceId` is null. Applying this transformation upstream prevents incomplete events from reaching downstream processing and destinations.

An **Aggregate** transformation supports calculation of summary values over a defined window. Configuring it to group by DeviceId, use a five-minute window, and calculate the average of Temperature produces the required device-level telemetry summary. Eventstreams provide no-code transformations for real-time event processing before data is routed to

destinations. For more information, see Process event data by using eventstream transformations and Microsoft Fabric eventstreams.

QUESTION NO: 19

You have a Fabric eventstream that receives temperature readings from manufacturing equipment.

You need to notify an operations team in Microsoft Teams when the temperature of any device exceeds a defined threshold.

What should you create?

A. a Data Activator reflex

B. a deployment pipeline

C. a Dataflow Gen2 dataflow

D. a semantic model calculation group

ANSWER: A

Explanation:

A **Data Activator reflex** is appropriate because it can monitor incoming event data, evaluate conditions, and initiate actions when the conditions are met. A reflex can identify events in which a temperature value exceeds the configured threshold and send a notification to the operations team.

Data Activator is designed for event-driven monitoring scenarios in Fabric. It evaluates data as it changes, which is preferable to requiring users to manually inspect a report or wait for a scheduled pipeline execution. See What is Data Activator?.

QUESTION NO: 20

You need to recommend a solution to prepare the tenant for the Po

C.

Which two actions should you recommend performing from the Fabric Admin portal? Each correct answer presents part of the solution.

NOTE: Each correct answer is worth one point.

A. Enable the Users can try Microsoft Fabric paid features option for specific security groups.

B. Enable the Allow Azure Active Directory guest users to access Microsoft Fabric option for specific security groups.

C. Enable the Users can create Fabric items option and exclude specific security groups.

D. Enable the Users can try Microsoft Fabric paid features option for the entire organization.

E. Enable the Users can create Fabric items option for specific security groups.

ANSWER: A E

Explanation:

Enable the *Users can try Microsoft Fabric paid features* option for specific security groups so that only the PoC participants can start and use a Fabric trial. This supports evaluation of Fabric capabilities that require paid capacity while maintaining controlled access during the proof of concept. Scoping the setting to designated Microsoft Entra security groups is appropriate tenant governance: administrators can grant trial eligibility to the evaluation team without exposing trial creation and paid Fabric experiences to all users in the tenant.

Enable the *Users can create Fabric items* option for specific security groups so that the same controlled PoC population can create the Fabric artifacts needed to implement and test the solution. Fabric items include workload artifacts such as lakehouses, notebooks, pipelines, semantic models, and reports; permitting item creation only for the groups participating in the PoC helps keep the trial environment governed and prevents uncontrolled experimentation elsewhere in the tenant. Both settings are managed as Fabric tenant settings in the Admin portal and can be applied to the whole organization or restricted to selected security groups. Microsoft recommends using security-group scoping when access should be limited to a defined set of users. See Microsoft Fabric trial administration and Microsoft Fabric tenant settings.

QUESTION NO: 21 - (DRAG DROP)

You are implementing two dimension tables named Customers and Products in a Fabric warehouse.

You need to use slowly changing dimension (SCO) to manage the versioning of dat

a. The solution must meet the requirements shown in the following table.

Table	Change action
Customers	Create a new version of the row.
Products	Overwrite the existing value in the latest row.



ANSWER:



Explanation:

Customers should use Type 2 because its required behavior is to create a new version of a dimension row when the tracked customer data changes. A Type 2 slowly changing dimension preserves the historical state of the dimension by retaining the earlier row and adding a new row for the changed version. Implementations commonly identify the active version with effective-date columns, expiration-date columns, or a current-row indicator. This pattern allows reports to relate facts to the customer attributes that were valid at the time represented by those facts, which is the key reason to use a versioned dimension. Microsoft's dimensional modeling guidance describes Type 2 as adding a new dimension row when a change occurs so historical analysis remains possible. See Slowly changing dimensions in Microsoft Fabric.

Products should use Type 1 because the requirement is to overwrite the existing value in the latest row. Type 1 applies the incoming changed attribute directly to the existing product dimension record. There is one current row for that product, and its former attribute value is not retained as a separate historical version. This is appropriate when the warehouse should present the corrected or current product attribute consistently, rather than analyze the earlier value over time. Microsoft's

Fabric guidance identifies Type 1 as the approach that updates an existing dimension row and does not preserve the prior attribute value. The mappings are therefore Customers to Type 2 and Products to Type 1.

QUESTION NO: 22

You have a Fabric tenant that contains a lakehouse. You plan to use a visual query to merge two tables.

You need to ensure that the query returns all the rows that are present in both tables. Which type of join should you use?

A. left outer

B. right anti

C. full outer

D. left anti

E. right outer

F. inner

ANSWER: F

Explanation:

The correct join type is **inner**. An inner join returns only records for which the specified join key has a matching value in both input tables. Therefore, its result represents the intersection of the two tables based on the selected matching columns: every returned row is present on both sides of the merge. This precisely meets the requirement to return rows that are present in both tables.

In a Fabric lakehouse visual query, you choose the join type while defining the relationship between tables and identify the columns used to match records. The selected **inner** join retains matching records, allowing the query result to combine related attributes from each table while limiting output to shared key values. This is useful, for example, when combining transactional records with corresponding reference or dimension data only where a valid relationship exists.

Microsoft's SQL join guidance describes an inner join as returning rows when the join predicate is satisfied by both tables. The Fabric visual query experience provides a graphical method for constructing these query relationships. See Microsoft documentation on joins and Microsoft documentation on the Fabric visual query editor.

QUESTION NO: 23

You have a Microsoft Power BI semantic model.

You need to identify any surrogate key columns in the model that have the Summarize By property set to a value other than to None. The solution must minimize effort.

What should you use?

A. DAX Formatter in DAX Studio

B. Model view in Microsoft Power BI Desktop

C. Model explorer in Microsoft Power BI Desktop

D. Best Practice Analyzer in Tabular Editor

ANSWER: D

Explanation:

Best Practice Analyzer in Tabular Editor is the appropriate tool because it can evaluate metadata across the entire semantic model at once, rather than requiring manual inspection of each column's properties. A Best Practice Analyzer rule can identify columns that participate in relationships, which commonly represent surrogate-key columns in a dimensional model, and test whether their *SummarizeBy* property is set to *None*. The results are presented as a consolidated list of rule violations, making it straightforward to find every affected column and review or correct the setting.

This approach minimizes effort because the validation is automated and repeatable. The same rule can be run again after model changes, including after new tables, relationships, or columns are added. Tabular Editor's Best Practice Analyzer supports rules expressed against tabular-object metadata and includes rules that help detect inappropriate aggregation behavior for key-related columns. In a Power BI semantic model, setting key columns not to summarize prevents accidental aggregation and helps ensure that fields behave correctly when used in reports.

For details, see the Tabular Editor Best Practice Analyzer documentation and Microsoft's guidance on using Tabular Editor with Power BI Desktop.

QUESTION NO: 24

Your company has a finance department.

You have a Fabric tenant, an Azure Storage account named storagel, and a Microsoft Entra group named Groupl. Groupl contains the users in the finance department.

You need to create a new workspace named Workspacel in the tenant. The solution must meet the following requirements:

Ensure that the finance department users can create and edit items in Workspace"!. Ensure that Workspacel can securely access storagel to read and write data.

Ensure that you are the only admin of Workspacel. Minimize administrative effort.

You create Workspacel.

Which two actions should you perform next? Each correct answer presents part of the solution. NOTE: Each correct selection is worth one point.

A. Assign the Contributor role to Groupl.

B. Assign the Contributor role to each finance department user.

C. Assign the Admin role to yourself.

D. Create a workspace identity.

ANSWER: A D

Explanation:

Assign the Contributor role to Groupl. A Fabric workspace role can be assigned to a Microsoft Entra security group, so the finance users inherit access through their existing group membership. The Contributor workspace role permits members to create and edit workspace content. Using Groupl avoids individual role assignments and keeps administration low: future finance staffing changes are managed by changing the group membership rather than repeatedly updating the workspace. The workspace creator is an administrator by default, so this assignment does not require adding another workspace administrator.

Create a workspace identity. A Fabric workspace identity is a managed identity associated with the workspace, allowing Fabric items to authenticate to supported Azure resources without embedding account keys, connection secrets, or user credentials. After it is created, the workspace identity can be granted an appropriate Azure RBAC data-plane role on storagel, such as Storage Blob Data Contributor where blob read/write operations are required. This provides a secure, centrally managed identity for the workspace's Azure Storage access while preserving the requirement that no additional user is made a workspace admin. See Microsoft Fabric workspace roles and Microsoft Fabric workspace identity.