

DUMPS ARENA

Designing and Implementing Enterprise-Scale Analytics Solutions Using Microsoft Azure and Microsoft Power BI

Microsoft DP-500

Version Demo

Total Demo Questions: 10

Total Premium Questions: 113

Buy Premium PDF

<https://dumpsarena.co>

sales@dumpsarena.co

sales@dumpsarena.co
dumpsarena.co

Topic Break Down

Topic	No. of Questions
Topic 1, Litware, Inc.	6
Topic 2, Contoso, Ltd	8
Topic 3, Misc. Questions	99
Total	113

QUESTION NO: 1 - (DRAG DROP)

You are configuring Azure Synapse Analytics pools to support the Azure Active Directory groups shown in the following table.

Name	Requirement
Group1	Analyze data to create and train machine learning models in Synapse Analytics.
Group2	Execute complex queries with multiple joins against relational data. Results will be exported by using PolyBase.
Group3	Query and load data from Apache Parquet files stored in Azure Data Lake Storage Gen2. Costs must be based on the amount of data processed.

Which type of pool should each group use? To answer, drag the appropriate pool types to the groups. Each pool type 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:

Explanation:

Box 1: Apache Spark pool

An Apache Spark pool provides open-source big data compute capabilities. After you've created an Apache Spark pool in your Synapse workspace, data can be loaded, modeled, processed, and distributed for faster analytic insight.

Box 2: Dedicated SQL Pool

Dedicated SQL Pool - Data is stored in relational tables

Box 3: Serverless SQL pool

Serverless SQL pool - Cost is incurred for the data processed per query

Reference: <https://docs.microsoft.com/en-us/azure/synapse-analytics/quickstart-create-apache-spark-pool-portal>

<https://www.royalcyber.com/blog/data-services/dedicated-sql-pool-vs-serverless-sql/>

QUESTION NO: 2

You have a Power BI report that contains one visual.

You need to provide users with the ability to change the visual type without affecting the view for other users.

What should you do?

- A. From Report setting, select Personalize visuals.
- B. From Tabular Editor, create a new perspective.
- C. From the Bookmarks pane, select Focus mode, and then select Add.
- D. From Visual options in Report settings, select Use the modern visual header with updated styling options.

ANSWER: A

Explanation:

Enable personalization in a report

You can enable the feature either in Power BI Desktop or the Power BI service. You can also enable it in embedded reports.

To enable the feature in Power BI Desktop, go to File > Options and settings > Options > Current file > Report settings. Make sure Personalize visuals is turned on.

QUESTION NO: 3

You have a 2-GB Power BI dataset.

You need to ensure that you can redeploy the dataset by using Tabular Editor. The solution must minimize how long it will take to apply changes to the dataset from powerbi.com.

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

NOTE: Each correct selection is worth one point

- A. Enable service principal authentication for read-only admin APIs.

- B. Turn on Large dataset storage format.
- C. Connect the target workspace to an Azure Data Lake Storage Gen2 account.
- D. Enable XMLA read-write.

ANSWER: B D

Explanation:

Optimize datasets for write operations by enabling large models

When using the XMLA endpoint for dataset management with write operations, it's recommended you enable the dataset for large models. This reduces the overhead of write operations, which can make them considerably faster. For datasets over 1 GB in size (after compression), the difference can be significant.

Tabular Editor supports Azure Analysis Services and Power BI Premium Datasets through XMLA read/write.

Note: Tabular Editor - An open-source tool for creating, maintaining, and managing tabular models using an intuitive, lightweight editor. A hierarchical view shows all objects in your tabular model. Objects are organized by display folders with support for multi-select property editing and DAX syntax highlighting. XMLA read-only is required for query operations. Read-write is required for metadata operations.

Reference: <https://docs.microsoft.com/en-us/power-bi/enterprise/service-premium-connect-tools>

<https://tabulareditor.github.io/>

QUESTION NO: 4

After you answer a question in this section, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.

You are using an Azure Synapse Analytics serverless SQL pool to query a collection of Apache Parquet files by using automatic schema inference. The files contain more than 40 million rows of UTF-8-encoded business names, survey names, and participant counts. The database is configured to use the default collation.

The queries use open row set and infer the schema shown in the following table.

name	system_type_name	max_length
businessName	varchar(8000)	8000
surveyName	varchar(8000)	8000
participants	int	4

You need to recommend changes to the queries to reduce I/O reads and tempdb usage.

Solution: You recommend using openrowset with to explicitly define the collation for businessName and surveyName as Latim_Generai_100_BiN2_UTF8.

Does this meet the goal?

- A. Yes
- B. No

ANSWER: A**Explanation:**

Query Parquet files using serverless SQL pool in Azure Synapse Analytics.

Important

Ensure you are using a UTF-8 database collation (for example Latin1_General_100_BIN2_UTF8) because string values in PARQUET files are encoded using UTF-8 encoding. A mismatch between the text encoding in the PARQUET file and the collation may cause unexpected conversion errors. You can easily change the default collation of the current database using the following T-SQL statement: alter database current collate Latin1_General_100_BIN2_UTF8'.

Note: If you use the Latin1_General_100_BIN2_UTF8 collation you will get an additional performance boost compared to the other collations. The Latin1_General_100_BIN2_UTF8 collation is compatible with parquet string sorting rules. The SQL pool is able to eliminate some parts of the parquet files that will not contain data needed in the queries (file/column-segment pruning). If you use other collations, all data from the parquet files will be loaded into Synapse SQL and the filtering is happening within the SQL process. The Latin1_General_100_BIN2_UTF8 collation has additional performance optimization that works only for parquet and CosmosDB. The downside is that you lose fine-grained comparison rules like case insensitivity.

Reference: <https://docs.microsoft.com/en-us/azure/synapse-analytics/sql/query-parquet-files>

QUESTION NO: 5

You are optimizing a dataflow in a Power BI Premium capacity. The dataflow performs multiple joins. You need to reduce the load time of the dataflow.

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

- A. Reduce the memory assigned to the dataflows.
- B. Execute non-foldable operations before foldable operations.
- C. Execute foldable operations before non-foldable operations.
- D. Place the ingestion operations and transformation operations in a single dataflow.
- E. Place the ingestion operations and transformation operations in separate dataflows.

ANSWER: C E**Explanation:**

Using the compute engine to improve performance

Take the following steps to enable workloads trigger the compute engine, and always improve performance:

For computed and linked entities in the same workspace:

Ensure you perform the operations that fold, such as merges, joins, conversion, and others.

For ingestion focus on getting the data into the storage as fast as possible, using filters only if they reduce the overall dataset size. It's best practice to keep your transformation logic separate from this step, and allow the engine to focus on the initial

gathering of ingredients. Next, separate your transformation and business logic into a separate dataflow in the same workspace, using linked or computed entities; doing so allows for the engine to activate and accelerate your computations. In our analogy, it's like food preparation in the kitchen: food preparation is typically a separate and distinct step from gathering your raw ingredients, and a pre-requisite for putting the food in the oven. Similarly, your logic needs to be prepared separately before it can take advantage of the compute engine.

Reference: <https://docs.microsoft.com/en-us/power-bi/transform-model/dataflows/dataflows-premium-workload-configuration>

QUESTION NO: 6 - (HOTSPOT)

You are configuring an aggregation table as shown in the following exhibit.

Manage aggregations

Aggregations accelerate query performance to unlock big-data sets. [Learn more](#)

Aggregation table: FactSales(Agg) | Precedence: 0

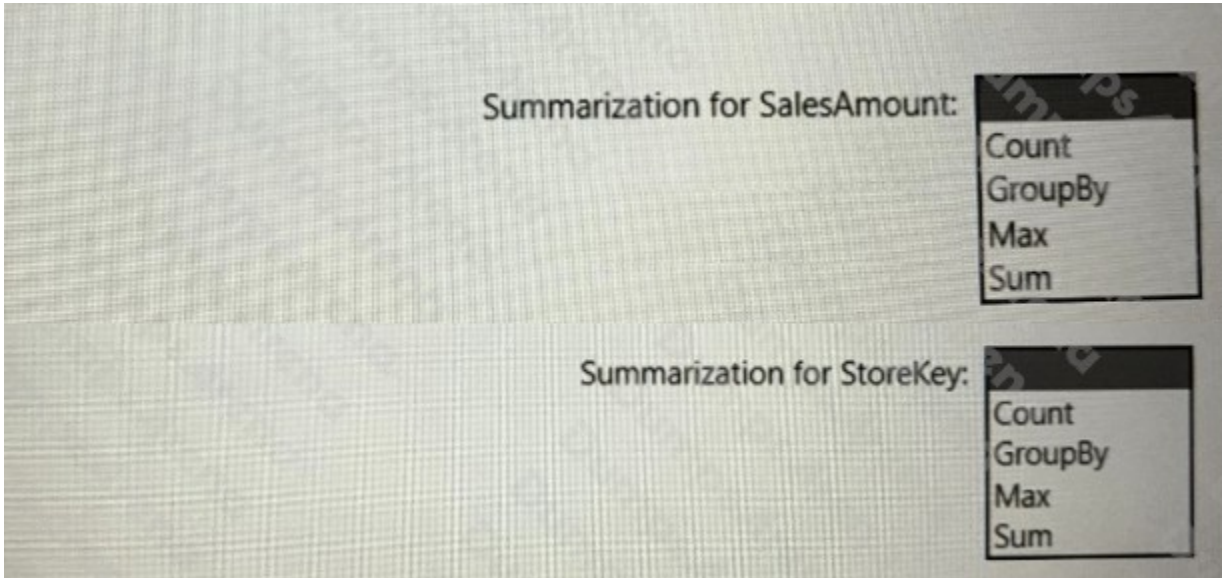
Column	Summarization	Aggregation	Distribution
ProductKey	Select Summarizatio...		
PromotionKey	Select Summarizatio...		
SalesAmount	Select Summarizatio...		
SalesQuantity	Select Summarizatio...		
StoreKey	Select Summarizatio...		
TotalCost	Select Summarizatio...		

The detail table is named FactSales and the aggregation table is named FactSales(Agg).

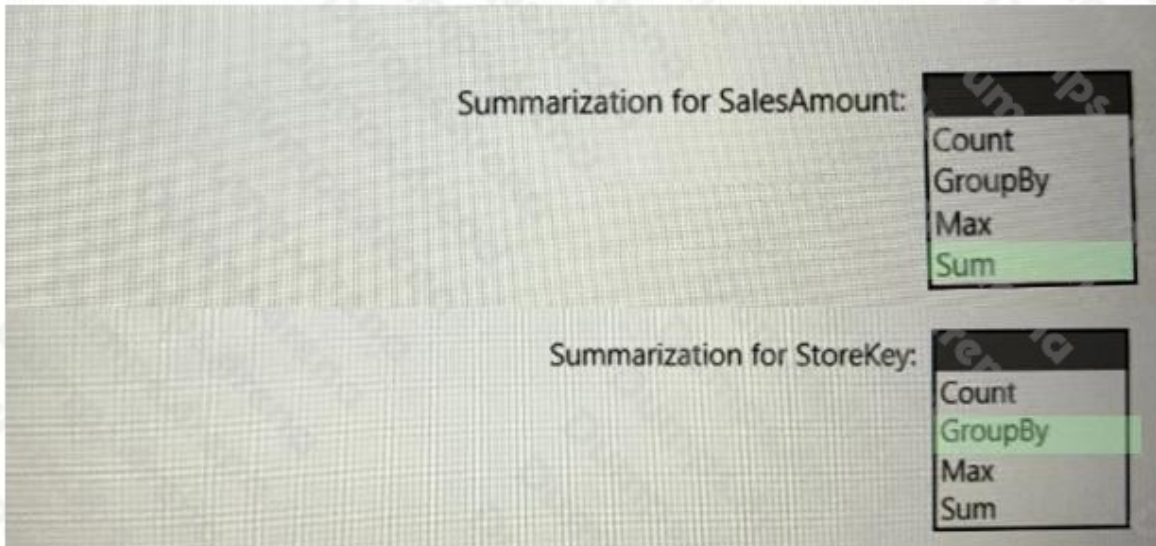
You need to aggregate SalesAmount for each store.

Which type of summarization should you use for SalesAmount and StoreKey? To answer, select the appropriate options in the answer area,

NOTE: Each correct selection is worth one point.



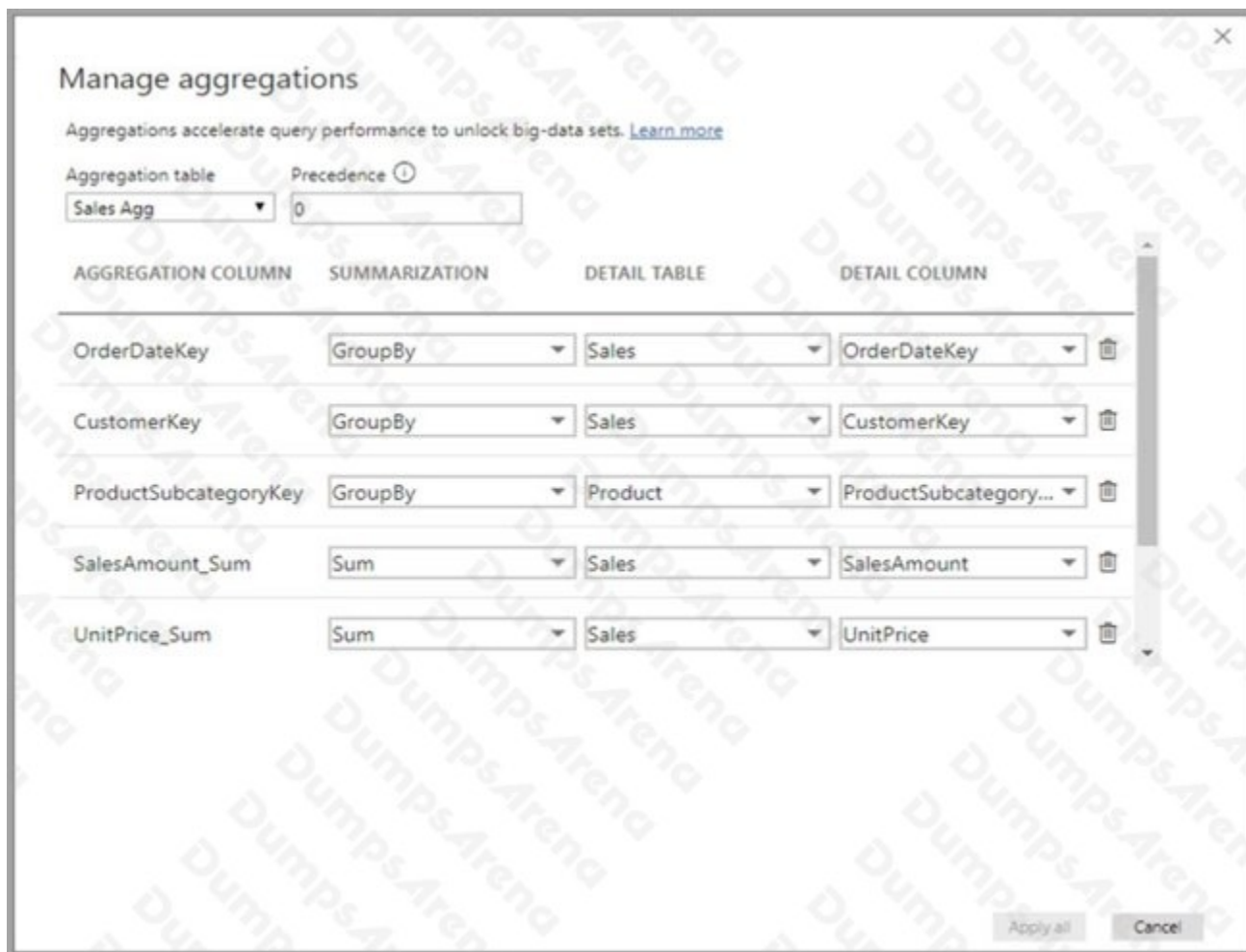
ANSWER:



Explanation:

Box 1: Sum

The Manage aggregations dialog shows a row for each column in the table, where you can specify the aggregation behavior. In the following example, queries to the Sales detail table are internally redirected to the Sales Agg aggregation table.



Box 2: GroupBy

Reference: <https://docs.microsoft.com/en-us/power-bi/transform-model/aggregations-advanced>

QUESTION NO: 7

You have a Power BI workspace that contains one dataset and four reports that connect to the dataset. The dataset uses Import storage mode and contains the following data sources:

- A CSV file in an Azure Storage account
- An Azure Database for PostgreSQL database

You plan to use deployment pipelines to promote the content from development to test to production. There will be different data source locations for each stage. What should you include in the deployment pipeline to ensure that the appropriate data source locations are used during each stage?

- A. parameter rules
- B. selective deployment
- C. auto-binding across pipelines

D. data source rules

ANSWER: A**Explanation:**

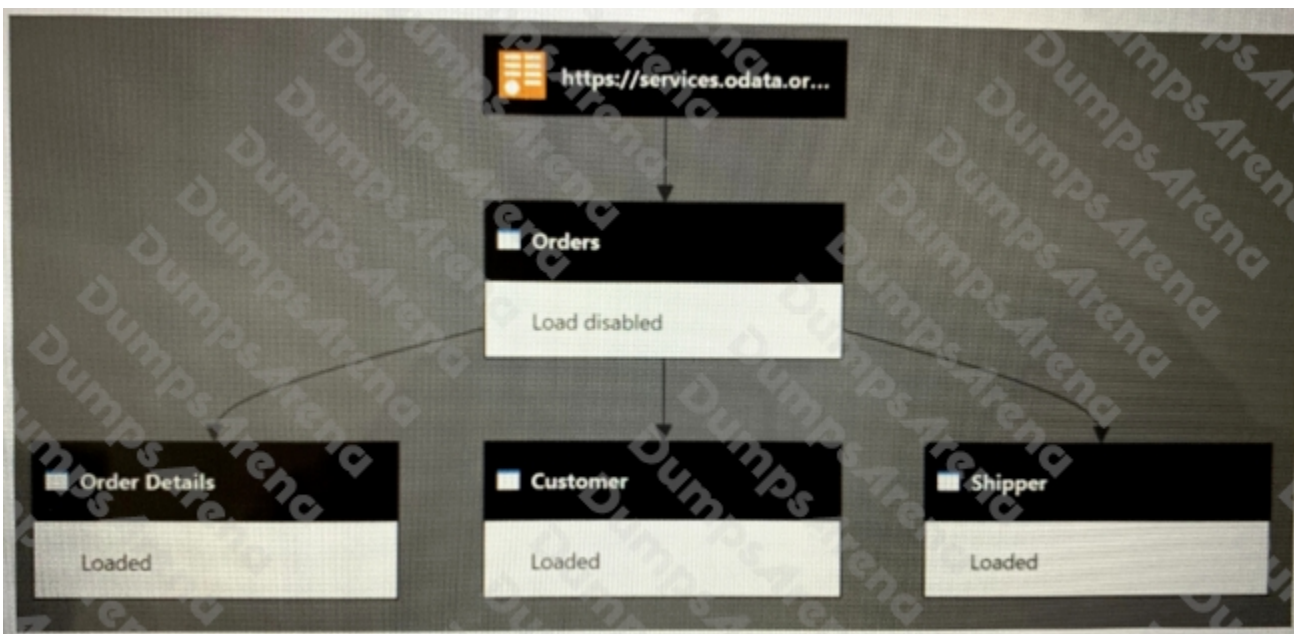
Note: Create deployment rules

When working in a deployment pipeline, different stages may have different configurations. For example, each stage can have different databases or different query parameters. The development stage might query sample data from the database, while the test and production stages query the entire database.

When you deploy content between pipeline stages, configuring deployment rules enables you to allow changes to content, while keeping some settings intact. For example, if you want a dataset in a production stage to point to a production database, you can define a rule for this. The rule is defined in the production stage, under the appropriate dataset. Once the rule is defined, content deployed from test to production, will inherit the value as defined in the deployment rule, and will always apply as long as the rule is unchanged and valid.

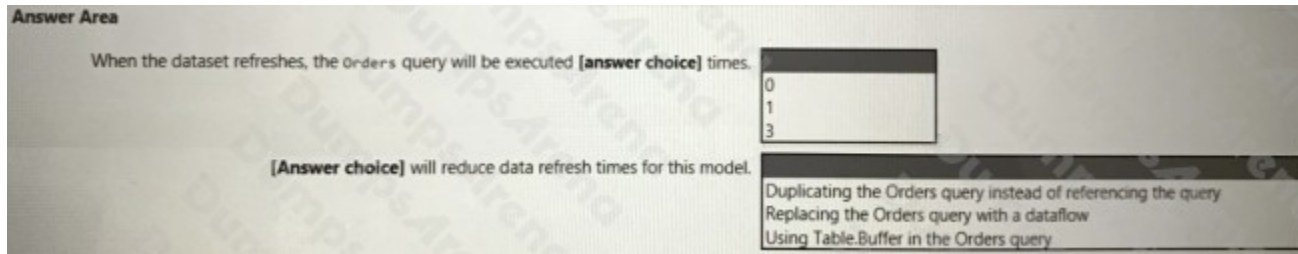
QUESTION NO: 8 - (HOTSPOT)

You have a Power BI dataset that has the query dependencies shown in the following exhibit.

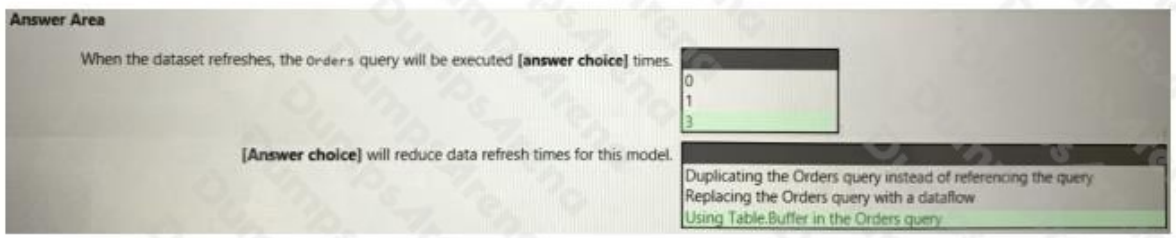


Use the drop-down menus to select the answer choice that completes each statement based on the information presented in the graphic.

NOTE: Each correct selection is worth one point.



ANSWER:



Explanation:

Box 1: 3

Power Query doesn't start at the first query and work down, it starts at the bottom (last) query and works backwards, so 3 tables from 1 will cause it to process that first source table 3 times.

Box 2: Using Table.Buffer in the Orders query

Table.Buffer buffers a table in memory, isolating it from external changes during evaluation. Buffering is shallow. It forces the evaluation of any scalar cell values, but leaves non-scalar values (records, lists, tables, and so on) as-is.

Note that using this function might or might not make your queries run faster. In some cases, it can make your queries run more slowly due to the added cost of reading all the data and storing it in memory, as well as the fact that buffering prevents downstream folding.

Example 1

Load all the rows of a SQL table into memory, so that any downstream operations will no longer be able to query the SQL server.

Usage

```
let
Source = Sql.Database("SomeSQLServer", "MyDb"),
MyTable = Source[{"Item="MyTable"}][Data],
BufferMyTable = Table.Buffer(dbo_MyTable)
in
BufferMyTable
```

Output

table

Reference: <https://radacad.com/performance-tip-for-power-bi-enable-load-sucks-memory-up>

<https://docs.microsoft.com/en-us/powerquery-m/table-buffer>

QUESTION NO: 9

You have a Power BI dataset that uses DirectQuery against an Azure SQL database.

Multiple reports use the dataset.

A database administrator reports that too many queries are being sent from Power BI to the database.

You need to reduce the number of queries sent to the database. The solution must meet the following requirements:

- DirectQuery must continue to be used.
- Visual interactions in all the reports must remain as they are configured currently.
- Consumers of the reports must only be allowed to apply filters from the Filter pane.

Which two settings should you select? Each correct answer presents part of the solution.

NOTE: Each correct selection is worth one point.

- A. Disabling cross highlighting/filtering by default
- B. Add a single Apply button to the filter pane to apply changes at once
- C. Add an Apply button to each slicer to apply changes when you're ready
- D. Add Apply buttons to all basic filters to apply changes when you're ready
- E. Ignore the Privacy Levels and potentially improve performance

ANSWER: B C**Explanation:**

Reduce queries

Reduce the number of queries sent by Power BI using the Query reduction settings. For slicers, select the “Add an Apply button to each slicer to apply changes when you’re ready” option. For filters, select “Add a single Apply button to the filter pane to apply changes at once (preview).”

Reference: <https://maqsoftware.com/insights/power-bi-best-practices>

QUESTION NO: 10 - (DRAG DROP)

You have a Power BI dataset. The dataset contains data that is updated frequently.

You need to improve the performance of the dataset by using incremental refreshes.

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

Actions**Answer Area**

Define the incremental refresh policy for the table.

Enable query caching.

Publish the model to the Power BI service.

Create RangeStart and RangeEnd parameters.

Use the Power BI REST API to post a message to /refreshes.

Apply a custom Date/Time filter to the data.



ANSWER:

Actions

- Define the incremental refresh policy for the table.
- Enable query caching.
- Publish the model to the Power BI service.
- Create RangeStart and RangeEnd parameters.
- Use the Power BI REST API to post a message to /refreshes.
- Apply a custom Date/Time filter to the data.

Answer Area

- Create RangeStart and RangeEnd parameters.
- Apply a custom Date/Time filter to the data.
- Define the incremental refresh policy for the table.
- Publish the model to the Power BI service.

Explanation:

- Create RangeStart and RangeEnd parameters.
- Apply a custom Date/Time filter to the data.
- Define the incremental refresh policy for the table.
- Publish the model to the Power BI service.

Step 1: Create RangeStart and RangeEnd parameters.

Create parameters

In this task, use Power Query Editor to create RangeStart and RangeEnd parameters with default values. The default values apply only when filtering the data to be loaded into the model in Power BI Desktop. The values you enter should include only

a small amount of the most recent data from your data source. When published to the service, these values are overridden by the incremental refresh policy.

Step 2: Apply a custom Date/Time filter to the data.

Filter data

With RangeStart and RangeEnd parameters defined, apply a filter based on conditions in the RangeStart and RangeEnd parameters.

Before continuing with this task, verify your source table has a date column of Date/Time data type.

Step 3: Define the incremental refresh policy for the table.

Define policy

After you've defined RangeStart and RangeEnd parameters, and filtered data based on those parameters, you define an incremental refresh policy. The policy is applied only after the model is published to the service and a manual or scheduled refresh operation is performed.

Step 4: Publish the model to the Power BI service.

Save and publish to the service

When your RangeStart and RangeEnd parameters, filtering, and refresh policy settings are complete, be sure to save your model, and then publish to the service.

Reference: <https://docs.microsoft.com/en-us/power-bi/connect-data/incremental-refresh-configure>