

# **POWER BI**

## **PL-300**

### **PART -2**

**Question 22:**

You have data in a Microsoft Excel worksheet as shown in the following table.

	A	B	C
1	SKU	price	discount
2	P00001	100	0.08
3	P00002	150	0.03
4	P00003	130	#DIV/0!
5	P00004	200	0.06
6	P00005	80	#NAME?
7	P00006	350	#N/A
8	P00007	100	#NULL!
9	P00008	200	0.05
10	P00009	135	#NUM!
11	P00010	90	#REF!
12	P00011	120	#VALUE!

You need to use Power Query to clean and transform the dataset. The solution must meet the following requirements:

- If the **Discount** column returns an error, a **discount of 0.05** must be used.
- **All rows of data must be maintained.**
- **Administrative effort must be minimized.**

What should you do in Power Query Editor?

**Options:**

- Select **Replace Errors**.
- Edit the query in the **Query Errors** group.
- Select **Remove Errors**.
- Select **Keep Errors**.

**Answer:**

- A. Select Replace Errors.**

**Solution Explanation:**

- "**Replace Errors**" allows you to replace error values with a specified default value (**0.05 in this case**) while keeping all rows in the dataset.

### Question 23:

You have two Microsoft Excel workbooks in a **Microsoft OneDrive folder**.

Each workbook contains a table named **Sales** with the same data structure.

You plan to use **Power BI** to combine both **Sales** tables into a **single table** and create **visuals** based on this combined data.

The solution must ensure that you can **publish a separate report and dataset**.

**Requirement:**

Select the appropriate **storage mode** for:

- **Report file**
- **Dataset file**

**Storage Mode Options:**

1. Import
2. DirectQuery
3. Dual Mode
4. Live Connection

**Answer:**

- Dataset File: Import Mode**  
 **Report File: DirectQuery Mode**

**Solution Explanation:**

- **Import Mode for Dataset File** ensures that **data from both Excel files is loaded into Power BI**, allowing better performance and data transformation capabilities.
- **DirectQuery Mode for Report File** enables the report to connect to the dataset **without reloading the data**, ensuring real-time updates when the dataset is refreshed.

### Question 24:

You use **Power Query** to import two tables named **Order Header** and **Order Details** from an **Azure SQL database**.

The **Order Header** table relates to the **Order Details** table through a column named **Order ID** in both tables.

You need to **combine** these tables into a **single query** that contains the **unique columns** of each table.

**Options:**

- A. Merge queries
- B. Combine files
- C. Append queries

**Answer:**

- A. Merge queries

**Question 25:**

You have a **PBIX** file that imports data from a **Microsoft Excel** data source stored in a **file share** on a **local network**.

You are notified that the **Excel data source was moved to a new location**.

You need to update the **PBIX file** to use the **new location**.

**Options:**

- A. From the **Datasets settings** of the **Power BI service**, configure the **data source credentials**.
- B. From the **Data source settings** in **Power BI Desktop**, configure the **file path**.
- C. From **Current File** in **Power BI Desktop**, configure the **Data Load settings**.
- D. From **Power Query Editor**, use the **formula bar** to configure the **file path** for the applied step.
- E. From **Advanced Editor** in **Power Query Editor**, configure the **file path** in the **M code**.

**Answer:**

- B. From the **Data source settings** in **Power BI Desktop**, configure the **file path**.
- D. From **Power Query Editor**, use the **formula bar** to **configure the file path for the applied step**.
- E. From **Advanced Editor** in **Power Query Editor**, configure the **file path** in the **M code**.

---

**Solution Explanation:**

When a **data source moves to a new location**, you need to **update the file path** in **Power BI Desktop**. There are three ways to achieve this:

**1 Data Source Settings (Option B) – Best Approach:**

- Go to "File" > "Options and settings" > "Data source settings".
- Select the old file path and click "Change Source".
- Browse to the new file location and update it.
- This method is simple and recommended for managing external data sources.

## 2 Formula Bar in Power Query Editor (Option D):

- Open "Power Query Editor".
- Select the "Source" step in the Applied Steps pane.
- Update the file path directly in the formula bar.
- Example: Change Excel.Workbook(File.Contents("C:\OldPath\File.xlsx")) to Excel.Workbook(File.Contents("C:\NewPath\File.xlsx")).

## 3 Advanced Editor in Power Query Editor (Option E):

- Open "Power Query Editor".
- Click on "Advanced Editor".
- Locate and update the file path manually in the M code.
- This method gives more control but requires knowledge of **M Language**.

### Question 26:

You have a **folder** that contains **50 JSON files**.

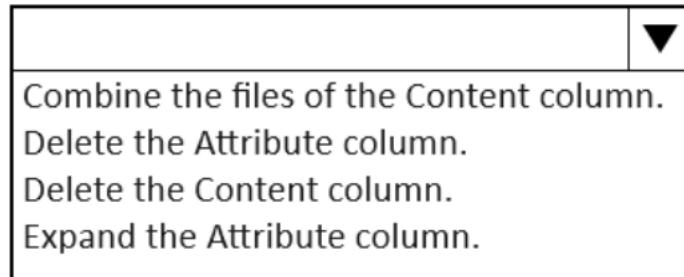
You need to use **Power BI Desktop** to make the **metadata** of the files available as a **single dataset**.

The solution must **NOT store the data** of the JSON files.

**Data source type:**



**Transformation:**



**Answer:**

**Data Source: Folder**

**Transformation: Delete the content columns**

**Explanation:**

### **1 Choosing the Correct Data Source – "Folder"**

- Since the JSON files are stored in a folder, you should use the "Folder" data source in Power BI.
- Steps:
  - In Power BI Desktop, go to "Get Data" > "Folder".
  - Select the folder containing the JSON files.
  - Power BI will list file metadata (file name, extension, date modified, etc.).
  - Since the requirement is to extract metadata only, this is the correct choice.

### **2 Selecting the Correct Transformation – "Delete the content columns"**

- By default, Power BI loads all files and includes a Content column that stores JSON data.
- Since we need only metadata and NOT the file content, we must delete the Content column.
- Steps:
  - In Power Query Editor, remove the "Content" column to avoid storing JSON data.
  - Keep metadata columns such as "Name", "Date modified", and "Size".

### Question 27:

You have a Power BI semantic model that contains the data sources shown in the following table.

Name	Description
Employee review data	Contains sensitive information Must <b>NOT</b> be folded into any other data sources
Sales opportunities	Contains less sensitive information Must only be available internally

You need to configure the privacy levels of the data sources.

What should you configure for each data source? To answer, select the appropriate options in the answer area.

NOTE: Each correct answer is worth one point.

Employee review data:

- 1. None
- 2. Organizational
- 3. Public
- 4. Private

Sales opportunities:

- 1. None
- 2. Organizational
- 3. Public
- 4. Private

Answer:

- Employee Review Data: Private
- Sales Opportunities: Organizational

Explanation:

1 Employee Review Data →  Private

- Contains sensitive information (e.g., personal employee details, performance reviews).
- Privacy Level: Private ensures that this data source remains isolated and is not combined with other sources to prevent data leaks.

**2 Sales Opportunities →  Organizational**

- Contains less sensitive information but must be accessible internally only.
- Privacy Level: Organizational ensures that data is shared only within the organization and not outside.

**Question 28:**

You plan to use Power BI Desktop to create a bug tracking dashboard that will pull data from Analytics in Azure DevOps.

From Power BI Desktop, you need to configure a data connector to authenticate to Azure DevOps. The solution must meet the following requirements:

- Use Analytics views.
- Filter data from the cloud.

Which connector should you use?

Options:

- A. OData queries
- B. Azure DevOps (Boards only)
- C. Azure DevOps Server (Boards only)
- D. OData Feed

Answer:

D. OData Feed

Solution Explanation:

- Azure DevOps Analytics views are exposed as OData feeds, making the OData Feed connector the best choice for connecting Power BI to Azure DevOps.
- OData Feed allows filtering data from the cloud, meeting the requirement for fetching selective bug tracking data.
- "Azure DevOps (Boards only)" and "Azure DevOps Server (Boards only)" do not support Analytics views and would not provide full access to the required data.
- "OData queries" is incorrect because Power BI requires the OData Feed connector to establish the connection properly.
- Using OData Feed enables real-time and flexible data retrieval from Azure DevOps Analytics views, making it the ideal choice for bug tracking dashboards.

### Question 29:

You use Power Query Editor to preview the data shown in the following exhibit.

Column1	Column2	Column3	Column4
<ul style="list-style-type: none"><li>● Valid 82%</li><li>● Error 0%</li><li>● Empty 18%</li></ul>  10 distinct, 9 unique	<ul style="list-style-type: none"><li>● Valid 82%</li><li>● Error 0%</li><li>● Empty 18%</li></ul>  10 distinct, 9 unique	<ul style="list-style-type: none"><li>● Valid 82%</li><li>● Error 0%</li><li>● Empty 18%</li></ul>  10 distinct, 9 unique	<ul style="list-style-type: none"><li>● Valid 82%</li><li>● Error 0%</li><li>● Empty 18%</li></ul>  8 distinct, 5 unique
1			
2			
3 metric_order	metric	actual	goal
4 1	Project Percent Complete	55	60
5 2	On-Time Task Completion	97	100
6 3	Promised Requirements Met	92	100
7 5	Costs	1,570,250	1,580,000
8 4	Team Utilization Rate	110	95
9 6	Customer Satisfaction Index	78	90
10 7	Team Satisfaction Index	91	90
11 8	Post-Deployment Support Hours	100	85

You confirm that the data will always start on row 3, and row 3 will always contain the column names.

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

**NOTE: Each correct selection is worth one point.**

#### Answer Area

To configure the query to start on row 3, select [answer choice]

Keep top rows
Remove errors
Remove top rows

To use the values in row 3 as the column names, select [answer choice]

Rename
Replace values
Use first row as headers

**Answer:**

- 1. Remove top rows
- 2. Use first row as headers

**Question 30:**

**Question:**

You have a data source that contains a column with **case-sensitive data**.

You have a **Power BI semantic model in DirectQuery mode**.

After connecting to the model, you notice **undefined values and errors**.

You need to resolve the issue.

**Solution:** You implicitly convert the values into the required type.

Does this meet the goal?

**Options:**

- A. Yes
- B. No

**Answer:**

- B. No

**Solution Explanation:**

- **Implicit type conversion** does **not** address case sensitivity issues.
- Power BI, especially in **DirectQuery mode**, relies on the **underlying database engine** for processing queries. Some databases (like SQL Server) **treat values as case-insensitive by default**, while others (like PostgreSQL) are case-sensitive.
- **Case sensitivity issues** can cause **undefined values and errors** when data is queried, especially if the database treats case variations as different values.
- The correct solution would be to **ensure consistent case handling** by either **applying transformations in Power Query or modifying the database collation settings** to avoid mismatches.
- Since **implicit conversion does not resolve case sensitivity issues**, this solution **does not meet the goal**.

**Question 31:**

You have a data source that contains a column with **case-sensitive data**.

You have a **Power BI semantic model in DirectQuery mode**.

After connecting to the model, you notice **undefined values and errors**.

You need to resolve the issue.

**Solution:** You change the **semantic model mode**.

Does this meet the goal?

**Options:**

- A. Yes
- B. No

**Answer:**

 B. No

**Solution Explanation:**

- **Changing the semantic model mode** (e.g., from **DirectQuery** to **Import mode**) does not directly address case sensitivity issues.
- The issue arises due to **case-sensitive data handling in the data source**, which can cause mismatches when queried in **Power BI**.
- Even if you **switch from DirectQuery to Import mode**, the underlying data issue **remains unresolved**, leading to the same **undefined values and errors**.
- The correct approach would be to **apply transformations in Power Query**, such as **changing the case of text columns (uppercase/lowercase)** or ensuring **consistent collation settings in the database**.
- Since **changing the semantic model mode does not fix case sensitivity problems**, this solution **does not meet the goal**.

### Question 32:

You have a data source that contains a column with **case-sensitive data**.

You have a **Power BI semantic model in DirectQuery mode**.

After connecting to the model, you notice **undefined values and errors**.

You need to resolve the issue.

**Solution:** You **normalize casing in the source query or Power Query Editor**.

Does this meet the goal?

**Options:**

- A. Yes
- B. No

**Answer:**

A. Yes

**Solution Explanation:**

- The issue arises because **Power BI may treat case-sensitive values differently from the data source**, leading to **errors or undefined values** when querying data.
- By **normalizing casing** (e.g., converting all text values to **uppercase or lowercase**) in either **the source query (SQL) or Power Query Editor**, you ensure **consistent data matching**, eliminating undefined values and errors.
- This approach **directly addresses the root cause** of the issue by ensuring that **Power BI queries return consistent results**, regardless of case sensitivity.
- Since **this method effectively resolves the issue**, the solution **meets the goal**.

**Question 33:**

You have a data source that contains a column with **case-sensitive data**.

You have a **Power BI semantic model in DirectQuery mode**.

After connecting to the model, you notice **undefined values and errors**.

You need to resolve the issue.

**Solution:** You add an index key and normalize casing in the data source.

Does this meet the goal?

**Options:**

- A. Yes
- B. No

**Answer:**

A. Yes

**Solution Explanation:**

- **Issue:** Power BI may treat **case-sensitive** values inconsistently, leading to **undefined values and errors** when querying data in **DirectQuery mode**.
- **Adding an index key** provides a **unique identifier** for each row, ensuring consistency when retrieving and joining data.
- **Normalizing casing** (e.g., converting text to **uppercase or lowercase**) in the **data source** ensures **case-insensitive** matching and prevents discrepancies.

- This solution effectively addresses the issue by ensuring that Power BI queries return consistent results, regardless of case sensitivity.
- Since both adding an index key and normalizing casing help resolve the problem, this solution meets the goal.

### Question 34:

#### Question:

You have a **Microsoft Excel file** stored in a **Microsoft OneDrive** folder.

The file must be **imported into a Power BI semantic model**.

You need to ensure that the **semantic model can be refreshed in PowerBI.com**.

Which two connectors can you use to connect to the file?

#### Options:

- A. Web
- B. Excel Workbook
- C. Folder
- D. Text/CSV
- E. SharePoint Folder

#### Answer:

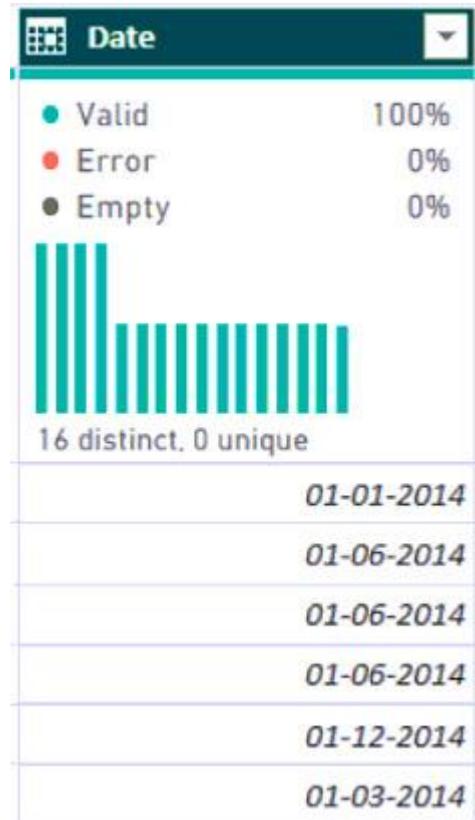
- A. Web
- E. SharePoint Folder

#### Solution Explanation:

- Power BI service (PowerBI.com) needs cloud-based access to refresh data sources like OneDrive files.
- Option A (Web):
  - When connecting to OneDrive, using the **Web connector** with the direct **OneDrive URL** ensures **automatic refresh** in Power BI.
  - Power BI can fetch the latest file version from OneDrive without requiring manual updates.
- Option E (SharePoint Folder):
  - OneDrive is built on **SharePoint Online**, so connecting via **SharePoint Folder** also allows **scheduled refresh** in Power BI.
  - This method works well for **multiple files stored in a folder**.

### Question 35:

You use **Power Query Editor** to preview a column named **Date**.



You need to **change the Date column to contain only the year** while **minimizing administrative effort**.

What should you do?

**Options:**

- A. Split the column by delimiter
- B. Split the column by number of characters
- C. Extract the text after the delimiter
- D. Transform the column to contain only the year

**Answer:**

- D. Transform the column to contain only the year

**Solution Explanation:**

- **Power Query provides built-in date transformation options** that allow users to extract specific date parts.

- Choosing "Transform → Date → Year" will directly extract the **year from the Date column**, making it the most efficient method.

### Question 36:

You are designing the data model for a Power BI semantic model.

You have the following tables in the star schema.

Name	Description
Date	Contains one row for each day from the last five years: Each row contains attributes for the year, quarter, month, week of the year, and day of the week. Date is the unique identifier of a row.
Patient	Contains one row per patient: Each row contains attributes for the patient key, patient source ID, first name, last name, date of birth, gender, address, city, state, and country. Patient key is the unique identifier of a row.
Test	Contains one row per test: Each row contains attributes for the test key, test source ID, type, and name. The test key is the unique identifier of a row.
Test Result	Contains one row per administered test: Each row contains attributes for the date the test was administered, test key, patient key, result value, and comments.

Which table is the fact table of the star schema, and which column in the Patient table is the surrogate key of the star schema? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

#### Answer Area

Fact table:

Date

Patient

Test

Test Result

Surrogate key:

Date of birth

Last name

Patient key

Patient source ID

**Answer:**

- A. FACT TABLE: TEST RESULT
- B SURROGATE KEY: PATIENT KEY

**Question 37:**

You use **Power BI Desktop** to import two tables named **Customer** and **Contacts**.

The **Customer** table contains:

• Customer_Name
• Customer ID
• Website

The **Contacts** table contains:

• Contact ID
• Contact Email
• Contact Name
• Customer Name

A **web-based contact form** is used to fill the **Contacts** table, and the **data is not sanitized**.

You need to **merge** the **Customer** and **Contacts** tables.

**Options:**

- A. Disable fuzzy matching
- B. Enable fuzzy matching
- C. Set Join Kind to Left Outer

**Answer:**

- B. Enable fuzzy matching

**Solution Explanation:**

- The **Customer Name** column is present in both tables, but **since the data is not sanitized**, there may be **spelling variations, extra spaces, or formatting differences**.
- **Fuzzy matching** allows Power BI to **match similar text values** that might have slight differences, which is useful when **merging tables with inconsistent text data**.

**Question 38:**

You are using Microsoft Power BI Desktop to profile data in Power Query Editor.

Table data is displayed as shown in the following exhibit.

ABC 123 _id	ABC 123 date	data.Entries.menuAmount
1	2024-02-07	01/01/1900
1	2024-02-07	02/01/1900
1	2024-02-07	Error

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 Area**

Before you can transform the date column to show only the day, you must

[answer choice]:

- change the data type.
- use the Parse command on the column.
- use the Replace Values command on the column.

To fix the error displayed for the data.Entries.menuAmount column, you must

[answer choice]:

- change the data type.
- rename the column.
- use Conditional Formatting.

**Answer:**

A. USE THE PARSE COMMAND ON THE COLUMN

B. CHANGE THE DATA TYPE

**Question 38:**

Column name	Data type	Description
PurchaseID	Text	Contains a unique ID for each order
CustomerID	Text	Contains a unique ID for each customer
PurchaseDateTime	Date/Time/Timezone	Contains the date and time that each order occurred
Region	Text	Contains the region where each order occurred
TotalAmount	Decimal number	Contains the total cost of each order

You have a **Power BI semantic model (Model1)** containing a **Sales** table with **10 million records**.

The **related report** displays **weekly sales per region**.

You need to **minimize the size of Model1**.

## Answer Area

CustomerID:

Change the data type to Binary. Remove any duplicate values. Remove the column.
---

PurchaseDateTime:

Change the data type to Duration. Remove the column. Split the column into separate date and time columns.
--

Answer:

- A. REMOVE THE COLUMN
- B. SPLIT THE COLUMN INTO SEPARATE DATE AND TIME COLUMNS

## Question 39:

You have a **Power BI Desktop report (Report1)** that uses an **Azure SQL database** as a **data source**.

A user named **User1** plans to **create a new report** using the **same data source** as Report1.

You need to **simplify the connection** to the **data source** for User1.

Options:

- A. PBIDS
- B. XLSX
- C. PBIT
- D. PBIX

Answer:

- A. PBIDS

### Solution Explanation:

#### 1. PBIDS (Power BI Data Source file)

- A PBIDS file is designed to **simplify data source connections** for users.
- It **stores connection details** (e.g., server name, database name) but **does not include reports or visuals**.
- When **User1 opens the PBIDS file**, Power BI automatically sets up the connection, requiring only **authentication**.
- **Best for ensuring users connect easily to a shared data source.**

### Question 40:

You have a **Power BI semantic model**.

You discover that the **semantic model contains values** that display as **errors**.

You need to **use data profiling features in Power Query to preview the data and identify the issues**.

#### Options:

**To gain insight into the number of errors:**

- A. Column Distribution
- B. Column Profile
- C. Column Quality
- D. Query Dependencies

**To resolve the errors:**

- A. Remove Errors
- B. Replace Errors
- C. Keep Errors
- D. Filter Errors

#### Answer:

- To gain insight into the number of errors: C. Column Quality
- To resolve the errors: B. Replace Errors

### Solution Explanation:

#### 1. To gain insight into the number of errors:

- The **Column Quality** feature in **Power Query** helps identify **data issues**, such as **valid, empty, and error values**.
- It provides a **quick overview of the percentage of errors in each column**, making it easier to detect **problematic data**.

2. **To resolve the errors:**

- **Replace Errors** allows you to **substitute error values** with a default value, such as **null, a specific number, or text**.
- This is useful when you **don't want to remove the entire row** but need to handle errors gracefully.

**Question 41:**

You plan to create a **Power BI semantic model** named **Model1** that will contain data from an **Azure SQL database (DB1)**.

**Requirement:**

- Model1 must show **updated data within two minutes** of data being updated in **DB1**.

**Options:**

- A. **DirectQuery**
- B. **Live connection**
- C. **Import**

**Answer:**

A. **DirectQuery**

**Solution Explanation:**

1. **DirectQuery** is the best choice because:

- It **fetches data in real-time** from **DB1** every time a user interacts with the report.
- The **data is not stored** in Power BI; instead, it is **queried directly** from the database when needed.
- This ensures that **Model1 always reflects the latest data**, meeting the "**within two minutes**" requirement.

2. **Why not "Live connection"?**

- **Live Connection** is used for **SQL Server Analysis Services (SSAS)** but **not for Azure SQL databases**.

**Question 42:**

You need to create a **semantic model** in **Power BI Desktop** that meets the following requirements:

- The model must contain a **table named Orders** with **one row per order**, including the **total amount per order**.
- The orders must be **filtered** by **selected CustomerID**.
- Users must **select the CustomerID** from a list.
- The **list of customers** must come from an **OData source**.

**Options:**

- A. An Orders query that has a filter on CustomerID
- B. A Customers query that has a filter on CustomerID
- C. An Orders query that has a single column containing a list of customers
- D. A Customers query that has a single column containing a list of customer IDs
- E. A parameter for CustomerID that uses a query to populate the suggested values
- F. A parameter for CustomerID that uses manually entered values to populate the suggested values

**Answer:**

- A. An Orders query that has a filter on CustomerID
- D. A Customers query that has a single column containing a list of customer IDs
- E. A parameter for CustomerID that uses a query to populate the suggested values

---

**Solution Explanation:**

1. **An Orders query that has a filter on CustomerID (Option A)**
  - The **Orders table** should include **one row per order**, displaying the **total amount per order**.
  - To meet the requirement of **filtering by selected CustomerID**, we must apply a **filter** on the Orders table.
2. **A Customers query that has a single column containing a list of customer IDs (Option D)**
  - Since users must select **CustomerID from a list**, we need to create a **Customers query** that provides **only a list of customer IDs**.
  - This list will be used to create a **parameter** for filtering orders.
3. **A parameter for CustomerID that uses a query to populate the suggested values (Option E)**
  - Instead of manually entering values (**Option F is incorrect**), we should create a **parameter** that dynamically pulls values from the **Customers query**.
  - This ensures that the list of available **CustomerIDs** is always updated based on the OData source.

### Question 43:

You are creating a report in Power BI Desktop.

You load a data extract that includes a free text field named col1.

You need to analyze the **frequency distribution of the string lengths** in col1. The solution must **not affect the size of the model**.

#### Options:

- A. In the report, add a DAX calculated column that calculates the length of col1.
- B. In the report, add a DAX function that calculates the average length of col1.
- C. From Power Query Editor, add a column that calculates the length of col1.
- D. From Power Query Editor, change the distribution for the Column Profile to group by length for col1.

#### Answer:

D. From Power Query Editor, change the distribution for the Column Profile to group by length for col1.

- **Column Profiling** in Power Query **does not store extra data** in the model.
- It provides a **visual frequency distribution** of the **string lengths** without adding a new column.
- **Best choice** for analyzing the data without increasing model size.

### Question 44:

#### Question:

You have a collection of reports for the HR department of your company. The datasets use **row-level security (RLS)**. The company has multiple **sales regions**, and each region has an **HR manager**.

You need to ensure that **HR managers can only access data from their region** while being **prevented from changing the layout of reports**.

#### Options:

- A. Publish the reports in an app and grant the HR managers access permission.
- B. Create a new workspace, copy the datasets and reports, and add the HR managers as members of the workspace.
- C. Publish the reports to a different workspace other than the one hosting the datasets.

D. Add the HR managers as members of the existing workspace that hosts the reports and the datasets.

**Answer:**

A. Publish the reports in an app and grant the HR managers access permission.

**Solution Explanation:**

Publishing the reports in a **Power BI app** ensures that **HR managers can only view the reports** without modifying them. The **RLS (Row-Level Security) rules** will apply, restricting access to data based on each HR manager's region. Unlike adding them as **workspace members (Option D)**, which might allow them to edit reports, the **app limits their access to viewing only**. Additionally, this approach maintains a **single dataset**, reducing duplication and maintenance overhead compared to creating a new workspace (Option B).

### Question 45:

You need to provide a user with the ability to **add members to a workspace**. The solution must follow the **principle of least privilege**.

**Options:**

- A. Viewer
- B. Admin
- C. Contributor
- D. Member

**Answer:**

D. Member

**Solution Explanation:**

The **Member** role allows users to **add members to a workspace** while ensuring they don't have full administrative rights, which would be provided by the **Admin** role.

The **Contributor** role allows a user to modify the content within the workspace but doesn't provide the ability to manage workspace membership.

The **Viewer** role only allows viewing content without any modification rights.

Using the **Member** role strikes the right balance by granting the necessary permission to add users but not full control over all workspace settings. This adheres to the **principle of least privilege** by providing the minimum required permissions.

### Question 46:

You have a Power BI query named **Sales** that imports the columns shown in the following table. Users only use the date part of the **Sales\_Date** field. Only rows with a **Status** of "Finished" are used in analysis.

Name	Description	Sample value
ID	A unique value that represents a sale	10253
Sale_Date	Sales date A column to extract the date of the sale	2021-11-23T09:53:00
Customer_ID	Represents a unique customer ID number	13158
Delivery_Time	Elapsed delivery time in hours Can contain null values	51.52
Status	Sales status Contains only the following two values: Finished and Canceled	Finished
Canceled_Date	Cancellation date and time Can contain null values	2021-11-24T14:11:23

You need to reduce the load times of the query without affecting the analysis.

Which **two actions** achieve this goal? Each correct answer presents a complete solution.

**Options:**

- A. Remove the rows in which Sales[Status] has a value of Canceled.
- B. Remove Sales[Sales\_Date].
- C. Change the data type of Sales[Delivery\_Time] to Integer.
- D. Split Sales[Sales\_Date] into separate date and time columns.
- E. Remove Sales[Canceled Date].

**Answer:**

- A. Remove the rows in which Sales[Status] has a value of Canceled.
- E. Remove Sales[Canceled Date].

**Question 47:**

You build a report to analyze customer transactions from a database that contains the tables shown in the following table.

Table name	Column name
Customer	CustomerID (primary key)
	Name
	State
	Email
Transaction	TransactionID (primary key)
	CustomerID (foreign key)
	Date
	Amount

You import the tables.

Which relationship should you use to link the tables?

- A. one-to-many from Transaction to Customer
- B. one-to-one between Customer and Transaction
- C. many-to-many between Customer and Transaction
- D. one-to-many from Customer to Transaction

**Answer:**

D. one-to-many from Customer to Transaction

### Question 48:

You have a custom connector that returns ID, From, To, Subject, Body, and Has Attachments for every email sent during the past year. More than 10 million records are returned.

You build a report analyzing the internal networks of employees based on whom they send emails to.

You need to prevent report recipients from reading the analyzed emails. The solution must minimize the model size.

**Options:**

- A. From Model view, set the Subject and Body columns to Hidden.
- B. Remove the Subject and Body columns during the import.
- C. Implement row-level security (RLS) so that the report recipients can only see results based on the emails they sent.

**Answer:**

 **B. Remove the Subject and Body columns during the import.**

**Solution Explanation:**

**1. Remove the Subject and Body columns during the import (B):**

Since the objective is to minimize the model size and prevent recipients from reading email content, removing these columns entirely during the import is the most effective solution. This way, you eliminate the unnecessary data and minimize the size of the model without keeping any email content visible in the report.

**2. Why the other options are not correct:**

- **C. Implement row-level security (RLS) so that the report recipients can only see results based on the emails they sent:**

Row-level security helps restrict what data users can access based on their role, but it doesn't minimize the model size. Additionally, RLS wouldn't prevent recipients from accessing or reading the Subject and Body columns, which is a key requirement here.

**Question 49:**

You create a Power BI dataset that contains the table shown in the following exhibit.

Business Unit	⋮
Cost Center	
Headcount	
ID	
Name	
<a href="#">Collapse ^</a>	

You need to make the table available as an organizational data type in Microsoft Excel. How should you configure the properties of the table? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

Hot Area:

## Answer Area

Row label:

	▼
Cost Center	
Headcount	
ID	
Name	

Key column:

	▼
Cost Center	
Headcount	
ID	
Name	

Is featured table:

	▼
No	
Yes	

Answer:

- A. ROW LEBEL: COST CENTER
- B. KEY COLUMN: ID
- C. IS FEATURED TABLE: NO

## Question 50:

You have the Power BI model shown in the following exhibit.



A manager can represent only a single country.

You need to use row-level security (RLS) to meet the following requirements:

☞ The managers must only see the data of their respective country.

☞ The number of RLS roles must be minimized.

Which two actions should you perform? Each correct answer presents a complete

**solution.**

**Options:**

- A. Create a single role that filters Country[Manager\_Email] by using the USERNAME DAX function.
- B. Create a single role that filters Country[Manager\_Email] by using the USEROBJECTID DAX function.
- C. For the relationship between Purchase Detail and Purchase, select Apply security filter in both directions.
- D. Create one role for each country.
- E. For the relationship between Purchase and Purchase Detail, change the Cross filter direction to Single.

**Answer:**

- A. Create a single role that filters Country[Manager\_Email] by using the USERNAME DAX function.
- B. Create a single role that filters Country[Manager\_Email] by using the USEROBJECTID DAX function.

**Solution Explanation:**

**1. A. Create a single role that filters Country[Manager\_Email] by using the USERNAME DAX function:**

This is an efficient solution that allows you to filter the data based on the email of the logged-in user (which is typically used to identify the manager). By using the USERNAME() DAX function, you can create a single role that ensures managers only see data for their respective country without creating multiple roles.

**2. B. Create a single role that filters Country[Manager\_Email] by using the USEROBJECTID DAX function:**

The USEROBJECTID() function is another way to identify the user, especially in a more secure environment where the email might not be used. This method also allows creating a single role to restrict data based on the manager's unique ID.

**3. Why the other options are not correct:**

**o C. For the relationship between Purchase Detail and Purchase, select Apply security filter in both directions:**

This option could impact the data model's performance and the ability to apply filters correctly. For RLS with minimized roles, it's better to avoid unnecessary filter propagation.

**o D. Create one role for each country:**

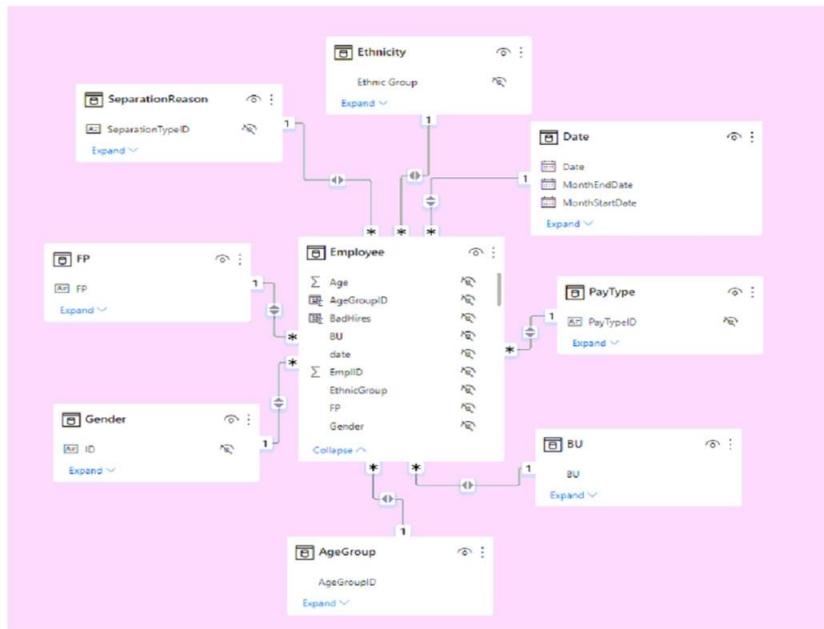
Creating a separate role for each country would not minimize the number of RLS roles and would require significant administrative effort. This contradicts the requirement to minimize roles.

- E. For the relationship between Purchase and Purchase Detail, change the Cross filter direction to Single:

Changing the filter direction to single could lead to unintended filtering behavior and doesn't directly address the RLS requirements for managing managers' data access based on country.

### Question 51:

You have a Power BI imported dataset that contains the data model 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.

Hot Area:

#### Answer Area

Changing the [answer choke] setting of the relationships will improve report query performance.

▼
Cardinality
Cross filter direction
Assume Referential Integrity

The data model is organized into a [answer choice].

▼
star schema
snowflake schema
denormalized table

Answer:

- A. Assume Referential Integrity
- B. Star schema

**Box 1: Assume Referential Integrity**

When connecting to a data source using DirectQuery, you can use the Assume Referential Integrity selection to enable running more efficient queries against your data source. This feature has a few requirements of the underlying data, and it is only available when using DirectQuery.

Note: The following **requirements are necessary for Assume referential integrity to work properly:**

1. Data in the From column in the relationship is never Null or blank
2. For each value in the From column, there is a corresponding value in the To column

**Box 2: Star schema -**

Star schema is a mature modeling approach widely adopted by relational data warehouses. It requires modelers to classify their model tables as either dimension or fact.

Generally, dimension tables contain a relatively small number of rows. Fact tables, on the other hand, can contain a very large number of rows and continue to grow over time.

**Question 52:**

You have a Power BI model that contains a table named Sales and a related date table. Sales contains a measure named Total Sales.

You need to create a measure that calculates the total sales from the equivalent month of the previous year.

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

**Hot Area:**

Sales Previous Year =

▼
CALCULATE
EVALUATE
SUM
SUMX

[Total Sales],

▼
DATESMTD
PARALLELPERIOD
SAMEPERIODLASTYEAR
TOTALMTD

▼
[Date]
'Date' [Date]
'Date' [Month]

)  
)

**Answer:**

- A. CALCULATE
- B. PARALLELPERIOD
- C. DATE[MONTH]

**Question 53:**

You plan to create a report that will display sales data from the last year for multiple regions. You need to restrict access to individual rows of the data on a per region-basis by using roles. Which four actions should you perform in sequence? To answer, move the appropriate actions from the list of actions to the answer area and arrange them in the correct order.

**Actions**

- Publish the report.
- Assign users to the role.
- Add a filter to the report.
- Create a role definition.
- Import the data to Power BI Desktop.



**Answer Area**

- Import the data to Power BI Desktop.
- Create a role definition.
- Assign users to the role.
- Publish the report.

You can define roles and rules within Power BI Desktop. When you publish to Power BI, it also publishes the role definitions.

To define security roles, follow these steps.

- 1. Import data into your Power BI Desktop report (Step 1)**
2. From the Modeling tab, select Manage Roles.
- 3. From the Manage roles window, select Create. (Step 2)**
4. Under Roles, provide a name for the role.
5. Under Tables, select the table to which you want to apply a DAX rule.
6. In the Table filter DAX expression box, enter the DAX expressions. This expression returns a value of true or false. For example: [Entity ID] = "Value"(Step 3)

7. After you've created the DAX expression, select the checkmark above the expression box to validate the expression.

8. Select Save.

**Step 3: Assign Users to the role.**

You can't assign users to a role within Power BI Desktop. You assign them in the Power BI service.

After you've created your roles, test the results of the roles within Power BI Desktop.

**Step 4: Publish the report.**

Now that you're done validating the roles in Power BI Desktop, go ahead and publish your report to the Power BI service.