

Assignment: 1

1) What is Power Query, and what role does it play in Power BI?

Ans: Power Query is a data connection technology that makes it possible to discover, connect, and merge data from many sources. In Power BI, it is the foremost data extraction, transformation, and loading ETL tool. It makes it possible to clean, transform, and get data ready before loading it into Power BI for analysis and visualization.

2) How do you connect to different data sources using Power Query?

Ans: Power Query has a simple interface for connecting to a vast number of data sources, such as:

- Excel, CSV, and text files
- Databases such as SQL Server, MySQL, and PostgreSQL
- Web services and APIs
- Online services like SharePoint, Salesforce, and Google Analytics
- Cloud platforms such as Azure and Microsoft Exchange

3) Can you explain the process of data transformation in Power Query?

Ans: Power Query data transformation consists of the following steps:

- Loading Data: Load data from different sources.
- Cleaning Data: Eliminate errors, duplicates, or unwanted information.
- Shaping Data: Convert data types, split or concatenate columns, and pivot/unpivot.
- Combining Data: Append or join queries to combine data.
- Enhancing Data: Create calculated columns, conditional rules, or custom formulas.
- Finalizing: Execute transformations and load data into Power BI for analysis.

4) What are some common data transformation operations you can perform in Power Query?

Ans: Power Query offers a variety of transformation operations like Filtering rows, Sorting data, Changing data types, Removing or renaming columns, Grouping data, pivoting and unpivoting columns, merging and appending queries and adding custom or conditional columns.

5) How do you handle missing or duplicate data in Power Query?

Ans: Power Query provides tools to manage missing or duplicate data:

- **Handling Missing Data:**
 - Use the "Remove Empty" option to eliminate rows with null values.

- Replace missing values with default values using the "Replace Values" feature.

- **Handling Duplicate Data:**

- Identify duplicates using the Remove Duplicates" option.
- Group data to aggregate duplicates based on specific columns.

6) Explain the difference between merging and appending queries in Power Query.

Ans: **Merging Queries:** Combines columns from two queries based on a related column, similar to SQL JOIN operations. This is useful when we need to enrich data by adding related information from another table.

Appending Queries: Stacks rows from multiple queries into a single query. UNION ALL operations. This is ideal for consolidating data from similar datasets.

7) What are conditional columns, and how do you create them in Power Query?

Ans: Conditional columns allow us to create new columns based on logical conditions applied to existing columns. To create a conditional column:

- Go to the "Add Column" tab and select "Conditional Column."
- Define the condition
- Specify the output value when the condition is true.
- Add more conditions Else clause when none of the conditions are met.

8) Describe the purpose of index columns in Power Query.

Ans: Index columns add a sequential number to each row, which can be useful for:

- Tracking row positions.
- Creating unique identifiers.
- Sorting or referencing rows in subsequent transformations.

To add an index column:

- Go to the "Add Column" tab and select "Index Column."
- Choose to start from 0 or 1, or define custom starting points and increments.

9) How can you create a calendar table using Power Query?

Ans: Following steps to be followed :

- In Power BI Desktop, go to the Home tab and click on Transform Data to open the Power Query Editor.

- In the Power Query Editor, click on Home -> New Source -> Blank Query.
- With the new query selected, click on Advanced Editor in the Home tab.
- Enter M Code for Calendar Table.
- Apply Changes and close and apply.

10) What are some best practices for optimizing data loading and transformation in Power Query?

Ans: In order to do so we need to follow several steps :

- Filter and Clean Data Early.
- Leverage Query Folding.
- Optimize Joins and Merges.
- Modularize Complex Queries.
- Utilize Parameters and Variables.
- Optimize Data Loading Settings.
- Monitor and Analyze Performance.