**Power BI Assignment 2**

Q1. Explain the advantages of Natural Queries in Power BI with an example?

Ans: Natural Language Queries (NLQ) in Power BI allow users to interact with data and generate insights using everyday language instead of writing complex queries or formulas. Here are some advantages of Natural Queries in Power BI:

1. Ease of Use: Natural Queries make it easier for non-technical users to access and analyze data. Instead of learning SQL or other query languages, users can simply type or speak their questions in plain language.
2. Accessibility: Natural Queries make data analysis accessible to a wider audience within an organization. Business users who may not have a technical background can still explore data and gain insights without relying on data analysts or IT professionals.
3. Faster Insights: With Natural Queries, users can quickly ask questions and receive immediate responses. This enables faster exploration of data and helps users discover insights more efficiently.
4. Increased Interactivity: Natural Queries promote a more interactive and conversational approach to data analysis. Users can ask follow-up questions, refine queries, and dynamically explore data by having a back-and-forth conversation with the system.

Example:

Let's say you have a Power BI report with sales data for a retail business. Using Natural Queries, you can ask questions such as:

* "What were the total sales for each product category last month?"
* "Show me a bar chart of sales by region for the current quarter."
* "Compare sales performance of Product A and Product B by month."
* "What are the top-selling products in the last six months?"
* "Visualize the trend of sales by day of the week."

Power BI will understand the natural language queries and generate the corresponding visualizations or reports based on the requested information. This allows users to explore and analyze data in a more intuitive and conversational manner, without the need for writing complex queries or interacting with the data using technical terms.

Q2. Explain Web Front End(WFE) cluster from Power BI Service Architecture?

Ans: In the Power BI service architecture, a Web Front End (WFE) cluster is a critical component that handles the user interface and communication between the user's browser and the Power BI service. The WFE cluster is responsible for rendering and serving the Power BI web interface, dashboards, reports, and other user-facing elements.

Here are key points to understand about the Web Front End (WFE) cluster in the Power BI Service architecture:

1. User Interface Rendering: The WFE cluster plays a crucial role in rendering the Power BI user interface in the user's web browser. It generates and serves the HTML, CSS, and JavaScript code required for displaying the Power BI dashboards, reports, and other interactive elements.
2. Load Balancing: The WFE cluster operates in a load-balanced configuration to distribute user requests evenly across multiple servers. This ensures scalability, high availability, and efficient handling of user traffic in the Power BI service.
3. Authentication and Authorization: The WFE cluster handles user authentication and authorization processes. When a user accesses the Power BI service, the WFE cluster verifies the user's credentials, manages access permissions, and enforces security measures to protect the data and ensure appropriate user access levels.
4. Communication with Backend Services: The WFE cluster communicates with various backend services to retrieve and process the requested data. This includes accessing the Power BI data model, interacting with data storage systems, and retrieving the required reports and visualizations to be displayed in the user interface.
5. Session Management: The WFE cluster manages user sessions, ensuring that user interactions are maintained consistently across different requests and actions. It tracks user activities, maintains state information, and enables smooth navigation and interaction within the Power BI service.
6. Performance Optimization: The WFE cluster incorporates performance optimization techniques to enhance the user experience. It employs caching mechanisms, compresses data for efficient transmission, and applies techniques like minification and bundling of web assets to reduce load times and improve responsiveness.
7. Scalability and Redundancy: The WFE cluster is designed for scalability and redundancy. Additional WFE servers can be added to handle increased user loads, and redundant configurations ensure high availability and fault tolerance in case of server failures.

Q3. Explain Back End cluster from Power BI Service Architecture?

Ans: In the Power BI service architecture, the Back End cluster is a key component responsible for handling the data processing, storage, and management tasks behind the scenes. It supports the functionality of the Power BI service by managing data connections, data modelling, report processing, and other back-end operations.

Here are key points to understand about the Back End cluster in the Power BI Service architecture:

1. Data Processing and Storage: The Back End cluster handles data processing tasks such as data ingestion, data transformation, and data modelling. It can connect to various data sources, extract data, apply transformations, and store the data in a suitable format for efficient querying and analysis.
2. Data Modelling and Calculation: The Back End cluster is responsible for creating and managing the data model that underlies Power BI reports and visualizations. It enables users to define relationships between data tables, create calculated measures and columns, and optimize the data structure for efficient querying and analysis.
3. Report and Dashboard Rendering: Once the data is processed and the data model is created, the Back End cluster facilitates the rendering of reports, dashboards, and visualizations. It retrieves the required data and generates the visual elements to be displayed in the Power BI user interface.
4. Query Processing and Optimization: When users interact with Power BI reports and visualizations, the Back End cluster handles the query processing and optimization. It translates user queries into optimized SQL or other query languages, retrieves the required data from the data sources, and applies caching and other optimization techniques to enhance query performance.
5. Security and Access Control: The Back End cluster enforces security measures and access control to protect the data and ensure appropriate user access levels. It manages user authentication, authorization, and data encryption to maintain the confidentiality and integrity of the data stored and processed within the Power BI service.
6. Metadata Management: The Back End cluster maintains metadata about data sources, data models, reports, and other artifacts in the Power BI service. It tracks dependencies, manages versioning, and enables efficient metadata retrieval for various operations within the service.
7. Scalability and Redundancy: The Back End cluster is designed to scale horizontally to handle increasing workloads and user demands. It can distribute data processing tasks across multiple servers and provide redundancy to ensure high availability and fault tolerance.

Q4. What ASP.NET component does in Power BI Service Architecture?

Ans: The ASP.NET component plays a crucial role in handling the web application framework and infrastructure that powers the Power BI service. ASP.NET is a web development framework developed by Microsoft, and it is used extensively in the Power BI service to handle the user interface, web requests, and server-side processing.

Here are key points to understand about the ASP.NET component in the Power BI Service architecture:

1. Web Application Framework: ASP.NET provides the framework for building and deploying web applications within the Power BI service. It offers a robust set of tools, libraries, and functionalities for creating dynamic and interactive web interfaces.
2. User Interface Rendering: The ASP.NET component handles the rendering of the user interface elements in the Power BI service. It generates HTML, CSS, and JavaScript code that define the layout, styling, and interactivity of the Power BI web interface, including dashboards, reports, and other visualizations.
3. Server-Side Processing: ASP.NET processes incoming web requests from users and performs server-side operations. It interacts with other components in the Power BI service architecture, such as the Back End cluster and data storage systems, to retrieve and process data, execute business logic, and generate dynamic content to be displayed in the user interface.
4. State Management: ASP.NET manages the state of web applications within the Power BI service. It tracks user sessions, maintains user-specific data and settings, and enables the preservation of application state across multiple requests.
5. Authentication and Authorization: ASP.NET handles user authentication and authorization processes in the Power BI service. It integrates with authentication systems, such as Azure Active Directory (AAD), to verify user credentials, manage access permissions, and enforce security measures.
6. Request Routing and Load Balancing: ASP.NET manages request routing and load balancing to distribute incoming user requests across multiple servers in the Web Front End (WFE) cluster. This ensures scalability, high availability, and efficient utilization of server resources.
7. Performance Optimization: ASP.NET incorporates performance optimization techniques to enhance the responsiveness and efficiency of the Power BI service. It utilizes caching mechanisms, minimizes network latency, and applies various optimization strategies to deliver a smooth and fast user experience.

Q5. Compare Microsoft Excel and Power BI Desktop on the following features:

Data import

Data transformation

Modelling

Reporting

Server Deployment

Convert Models

Cost

Ans: Comparing Microsoft Excel and Power BI Desktop on various features:

1. Data Import:
   * Excel: Excel allows importing data from various sources such as databases, files, web, and more. It supports importing data using built-in connectors, ODBC, OLE DB, and Power Query.
   * Power BI Desktop: Power BI Desktop offers extensive data import capabilities, including connectivity to a wide range of data sources such as databases, cloud services, files, and more. It provides seamless integration with Power Query for data extraction and transformation.
2. Data Transformation:
   * Excel: Excel provides basic data transformation capabilities through built-in functions, formulas, and tools like Power Query. However, it may require manual steps and formula writing for complex transformations.
   * Power BI Desktop: Power BI Desktop offers advanced data transformation capabilities through Power Query Editor. It provides a visual interface with a wide range of data transformation options, transformations, and query steps to clean, reshape, and combine data from multiple sources.
3. Modelling:
   * Excel: Excel allows basic data modelling using features like tables, relationships, and calculated columns. It supports PivotTables and Pivot Charts for analysis.
   * Power BI Desktop: Power BI Desktop offers robust data modelling capabilities. It supports advanced modelling features such as creating relationships, defining hierarchies, creating measures using DAX (Data Analysis Expressions), and handling complex data models with large volumes of data.
4. Reporting:
   * Excel: Excel provides reporting capabilities through features like PivotTables, charts, and conditional formatting. It allows users to create interactive reports and dashboards.
   * Power BI Desktop: Power BI Desktop is specifically designed for creating interactive and visually appealing reports and dashboards. It provides a wide range of visualizations, customization options, drill-down capabilities, and interactive filters to create engaging reports.
5. Server Deployment:
   * Excel: Excel files can be shared through email or stored on shared drives. Collaborative editing and real-time sharing of Excel files are limited.
   * Power BI Desktop: Power BI Desktop reports can be published to the Power BI service, where they can be accessed and shared with others. The Power BI service provides collaboration features, sharing options, and the ability to schedule data refreshes.
6. Convert Models:
   * Excel: Excel models can be converted into Power BI models by exporting data and recreating the model structure in Power BI Desktop.
   * Power BI Desktop: Power BI Desktop models can be easily converted into Power BI service models by publishing the reports and datasets to the Power BI service.
7. Cost:
   * Excel: Excel is part of the Microsoft Office suite, and its cost is typically included in the Office package or available on a standalone basis.
   * Power BI Desktop: Power BI Desktop is free to download and use for creating reports and dashboards. However, there may be costs associated with using the Power BI service for sharing and collaboration, depending on the licensing and subscription model.

Q6. List 20 data sources supported by Power Bi desktop.

Ans: Here is a list of 20 common data sources that are supported:

1. Excel workbook (XLSX, XLS)
2. CSV (Comma-separated values) files
3. SQL Server Database
4. Azure SQL Database
5. Oracle Database
6. MySQL Database
7. PostgreSQL Database
8. SharePoint Lists
9. Dynamics 365
10. Salesforce
11. Google Analytics
12. Azure Data Lake Storage
13. Azure Blob Storage
14. SharePoint Online
15. Web API (RESTful services)
16. JSON files
17. XML files
18. Hadoop Distributed File System (HDFS)
19. OData feeds
20. Azure Cosmos DB