**Power BI Assignment 2**

1. **Explain the advantages of Natural Queries in PowerBi with an example?**

=> Natural Queries in Power BI refer to the ability to interact with data using natural language queries or questions instead of writing complex queries or using specific programming syntax. It enables users to ask questions in a conversational manner, and Power BI interprets the intent of the question to generate relevant visualizations and insights. This feature offers several advantages, making data exploration more accessible and intuitive for a broader range of users, including non-technical individuals.

Example:

Let's say you are a sales manager, and you are using Power BI to analyze sales data. Instead of writing a traditional query or navigating through different menus, you can use Natural Queries to ask questions about your sales data.

For instance, you can ask, "What were our sales in the last quarter by region?" Power BI will interpret your question and generate a bar chart or a table showing sales figures for each region during the last quarter.

Following up on the previous question, you can ask, "Show me the top 5 products in the Southeast region." Power BI will refine the previous visualization, focusing on the Southeast region and displaying the top-selling products for that specific area.

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

=> The Web Front End (WFE) cluster is a critical component of the Power BI Service architecture that facilitates user access to Power BI content and ensures a smooth, secure, and responsive experience for users across different devices and locations. By handling user interactions, load balancing, security, and rendering, the WFE cluster helps deliver the full capabilities of Power BI to end-users in an efficient and user-friendly manner.

1. **Explain Back End cluster from Power BI Service Architecture?**

=> In the Power BI Service architecture, the Back End cluster is a crucial component responsible for managing data processing, storage, and other backend services. It works behind the scenes to handle tasks such as data loading, data modeling, report processing, and dataset management. The Back End cluster works in conjunction with the Web Front End (WFE) cluster to deliver a comprehensive and efficient Power BI experience to users.

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

=>

In the Power BI Service Architecture, ASP.NET components play a significant role in the Web Front End (WFE) cluster. ASP.NET is a web development framework developed by Microsoft, and it is used to build the web applications and services that power the Power BI Service.

Here's how ASP.NET components contribute to the Power BI Service Architecture:

* **Web Application Development**: ASP.NET is used to develop the web applications and services that form the core of the Power BI Service. It provides a robust and scalable framework for building interactive and responsive web applications that users can access through their web browsers and mobile devices.
* **User Interface**: ASP.NET allows developers to create the user interface (UI) elements of the Power BI Service, such as the navigation menus, buttons, filters, and interactive elements. It enables the dynamic rendering of Power BI reports and dashboards, making them accessible and visually appealing to users.
* **Data Processing and Interaction**: ASP.NET components handle user interactions in the Power BI Service. When users interact with the reports and dashboards, such as applying filters or drilling down into data, ASP.NET processes these interactions and triggers data queries to fetch the relevant information from the Back End cluster.
* **Authentication and Security**: ASP.NET provides a robust authentication and authorization framework that ensures secure access to the Power BI Service. It manages user authentication, allowing users to log in using their credentials and ensuring that only authorized users can access specific content and features.
* **Session Management**: ASP.NET manages user sessions, keeping track of user activities and maintaining session state during interactions with the Power BI Service. This ensures a consistent and seamless user experience as users navigate through different pages and reports.

1. **Compare Microsoft Excel and PowerBi Desktop on the following features:**

**Data import**

**Data transformation**

**Modeling**

**Reporting**

**Server Deployment**

**Convert Models**

**Cost**

=>

1. **Data Import:**

Microsoft Excel:

* Excel allows users to import data from various sources, such as CSV files, databases, web sources, and other Excel workbooks.
* Importing data in Excel is relatively straightforward, but it may require additional manual steps to clean and format the data.

Power BI Desktop:

* Power BI Desktop supports a wide range of data sources, including databases, cloud services, files, web sources, and many other online services.
* Power BI Desktop provides a more streamlined data import process with built-in data transformation capabilities.

2. **Data Transformation:**

Microsoft Excel:

* Excel provides basic data transformation features, such as sorting, filtering, and basic formulas.

Power BI Desktop:

* Power BI Desktop offers powerful data transformation capabilities, including advanced query editor tools for data cleansing, shaping, and transformation.

3. **Modeling:**

Microsoft Excel:

* Excel allows users to create basic data models and relationships between tables using the Power Pivot feature.

Power BI Desktop:

* Power BI Desktop provides advanced data modeling capabilities, including defining relationships between tables, creating hierarchies, and implementing calculated columns and measures using DAX.

4. **Reporting:**

Microsoft Excel:

* Excel enables users to create basic reports and charts using its built-in charting tools.
* Excel reports may lack the interactive and visually appealing features of Power BI reports.

Power BI Desktop:

* Power BI Desktop excels in creating interactive and visually rich reports and dashboards with various visualization options and interactive elements.
* Power BI reports offer more advanced data exploration and drill-down capabilities.

5. **Server Deployment:**

Microsoft Excel:

* Excel files are typically shared through email or file-sharing platforms. Collaborating on Excel files can be challenging, and real-time updates are not supported.

Power BI Desktop:

* Power BI reports are designed for server deployment on the Power BI service, allowing for centralized sharing and collaboration among users with real-time updates.

6. **Convert Models:**

Microsoft Excel:

* Excel models can be converted into Power BI datasets using the Power Query Editor, but this process may require some additional steps to optimize the model for Power BI.

Power BI Desktop:

* Power BI Desktop models can be directly published to the Power BI service for sharing and collaboration.

7. **Cost:**

Microsoft Excel:

* Excel is part of the Microsoft Office suite and is generally included in various Office 365 plans or can be purchased separately as a standalone application.

Power BI Desktop:

* Power BI Desktop is free to download and use.

1. **List 20 data sources supported by Power Bi desktop.**

=>

* 1. Excel workbook (XLSX)
  2. CSV (Comma-Separated Values) files
  3. SQL Server Database
  4. SQL Database (Azure SQL Database)
  5. MySQL Database
  6. PostgreSQL Database
  7. Oracle Database
  8. Microsoft Access Database
  9. SharePoint List
  10. Web (HTML) data sources
  11. SharePoint Online List
  12. OneDrive for Business
  13. Google Analytics
  14. Dynamics 365 (CRM) Online
  15. Salesforce
  16. JSON (JavaScript Object Notation) files and REST APIs
  17. Hadoop (HDFS)
  18. SAP HANA Database
  19. SharePoint Folder
  20. PDF files (using Power BI dataflows)