**Power BI Assignment 2**

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

Natural Queries in Power BI refer to the use of natural language to interact with data and generate visualizations. Instead of using technical queries or complex programming languages, users can simply ask questions about their data in a conversational manner, similar to how they would ask a question in everyday language. Power BI's Q&A (Question and Answer) feature allows users to explore data using natural language and obtain real-time visualizations as answers.

**Advantages of Natural Queries in Power BI:**

**Ease of Use:** Natural Queries make data exploration accessible to all users, regardless of their technical expertise. Non-technical users can easily ask questions and get insights without needing to know the underlying data structures or query languages.

**Rapid Insights:** Natural Queries enable quick access to insights. Users can directly ask questions and receive immediate visualizations, eliminating the need for manual data manipulation or complex report building.

**Interactive Exploration:** Users can iteratively refine their questions and explore data from different angles in real-time. The interactive nature of natural language queries allows for an exploratory and flexible data analysis process.

**Empowerment of Business Users:** Business users can independently access and analyse data without relying on IT or data analysts to build custom reports. This empowerment fosters a data-driven culture within the organization.

**Better Decision-Making:** By enabling users to gain rapid insights into their data, natural language queries support data-driven decision-making processes. Users can make informed decisions based on real-time data visualizations.

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

The Web Front End (WFE) cluster is one of the two main clusters in the Power BI service architecture. The WFE cluster is responsible for handling initial connections and authentication to the Power BI service. Once authenticated, the Back-End cluster handles all subsequent user interactions.

The WFE cluster consists of a number of web servers that are deployed in Azure App Service Environments. These web servers are responsible for serving the initial HTML page content to users, as well as providing pointers to CDN content that is used to render the site in the browser.

The WFE cluster also handles authentication for the Power BI service. This is done using Azure Active Directory (Azure AD). When a user attempts to connect to the Power BI service, their credentials are checked against Azure AD. If the credentials are valid, the user is authenticated and is allowed to access the Power BI service.

The WFE cluster is a critical part of the Power BI service architecture. It is responsible for handling the initial connection and authentication for users, which is essential for ensuring the security and integrity of the service.

Here are some of the benefits of using a WFE cluster in the Power BI service architecture:

**Scalability:** The WFE cluster can be scaled horizontally to handle more users. This is done by adding more web servers to the cluster.

Performance: The WFE cluster can be optimized for performance. This is done by using a variety of techniques, such as caching and load balancing.

**Security:** The WFE cluster can be secured using Azure AD. This ensures that only authorized users can access the Power BI service.

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

The Power BI Service Architecture is designed to handle various tasks and processes efficiently to deliver outstanding business intelligence solutions. It is built on Microsoft's cloud computing infrastructure and platform, Azure. The architecture of the Power BI service is based on two main clusters: The Web Front End (WFE) cluster and the Back-End cluster.

**Back-End Cluster:**

The Back-End cluster is a crucial component of the Power BI Service Architecture. Once a user is authenticated by the Web Front End (WFE) cluster, the Back-End takes over and handles all subsequent user interactions with the Power BI service.

Responsibilities of the Back-End cluster include:

**Data Processing:** The Back-End cluster is responsible for processing the user's requests and retrieving data from various data sources. It performs data transformations and manipulations to prepare the data for visualization.

**Report and Dashboard Generation:** After processing the data, the Back-End cluster generates reports and dashboards based on the user's input and preferences. These reports and dashboards are interactive and dynamic, allowing users to explore data visually.

**Data Storage:** The Back-End cluster manages data storage using Azure services such as Azure BLOB and Azure SQL Database. It ensures that the data is stored securely and efficiently for quick access during user interactions.

**User Management:** The Back-End cluster uses Azure Active Directory (Azure AD) to store and manage user identities. It handles user authentication and ensures that only authorized users can access specific data and reports.

**Performance and Scalability:** The Back-End cluster is designed to provide optimal performance and scalability to handle a large number of users and data processing requests simultaneously. This ensures a smooth and responsive user experience.

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

The ASP.NET component in the Power BI service architecture is responsible for rendering Power BI reports in the browser. It takes the data from the Presentation Role and renders it in the browser using ASP.NET.

The ASP.NET component is a critical part of the Power BI service architecture. It ensures that Power BI reports are rendered correctly in the browser, and it also provides a number of features, such as caching and security.

Here are some of the benefits of using an ASP.NET component in the Power BI service architecture:

**Scalability:** The ASP.NET component can be scaled horizontally to handle more users. This is done by adding more web servers to the cluster.

**Performance:** The ASP.NET component can be optimized for performance. This is done by using a variety of techniques, such as caching and load balancing.

**Security:** The ASP.NET component can be secured using Azure AD. This ensures that only authorized users can access the Power BI service.

Overall, the ASP.NET component is a critical part of the Power BI service architecture. It provides a number of benefits, such as scalability, performance, and security.

Here are some additional details about the ASP.NET component in the Power BI service architecture:

* The ASP.NET component is written in C# and is hosted on Azure App Service Environments.
* The ASP.NET component uses the Power BI JavaScript API to render Power BI reports in the browser.
* The ASP.NET component also provides a number of features, such as caching and security.

The ASP.NET component is a powerful tool that can be used to render Power BI reports in the browser. It is scalable, performant, and secure, making it a valuable asset for any Power BI deployment.

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

**Data import, Data transformation, Modelling, Reporting, Server Deployment, Convert Models, Cost**

Below is a comparison of Microsoft Excel and Power BI Desktop on the following features:

**Data import**

**Excel:** Excel can import data from a wide variety of data sources, including CSV files, text files, and databases.

**Power BI Desktop:** Power BI Desktop can import data from even more data sources than Excel, including Azure Data Lake Storage, Azure SQL Database, and Amazon Redshift.

**Data transformation**

**Excel:** Excel has a wide variety of data transformation tools, including filters, pivot tables, and macros.

**Power BI Desktop:** Power BI Desktop has even more data transformation tools than Excel, including advanced filtering, DAX, and the Power Query Editor.

**Modelling**

**Excel:** Excel can be used to create simple data models, but it is not as powerful as Power BI Desktop for modelling.

Power BI Desktop: Power BI Desktop is a powerful data modelling tool that can be used to create complex data models.

**Reporting**

**Excel:** Excel can be used to create simple reports, but it is not as powerful as Power BI Desktop for reporting.

**Power BI Desktop:** Power BI Desktop is a powerful reporting tool that can be used to create interactive, dashboard-style reports.

**Server deployment**

**Excel:** Excel reports can be published to SharePoint or a web server, but this is not a native feature of Excel.

**Power BI Desktop:** Power BI Desktop reports can be published to the Power BI service, which is a cloud-based service that allows you to share reports with others.

**Convert Models**

**Excel:** Excel does not have a native way to convert models to other formats.

**Power BI Desktop:** Power BI Desktop can convert models to a variety of formats, including XLSX, PBIX, and JSON.

**Cost**

**Excel:** Excel is a free product that is included with Microsoft Office.

**Power BI Desktop:** Power BI Desktop is a paid product that costs $9.99 per month per user.

Overall, Power BI Desktop is a more powerful tool than Excel for data analysis, reporting, and modelling. However, Excel is a free product, while Power BI Desktop is a paid product.

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

Below are the 20 data sources supported by Power BI Desktop:

* SQL Server
* Microsoft Access Database Engine (ACE)
* Azure Data Explorer (Kusto)
* SharePoint Online List
* Excel Workbook
* SharePoint Folder
* Web
* OData Feed
* SQL Server Analysis Services (SSAS) Database
* SQL Server Analysis Services (SSAS) Tabular Model
* Power BI Dataflows
* JSON
* XML
* Folder
* PDF
* Dynamics 365 Business Central
* Google Analytics
* Salesforce
* SharePoint Server
* MySQL Database