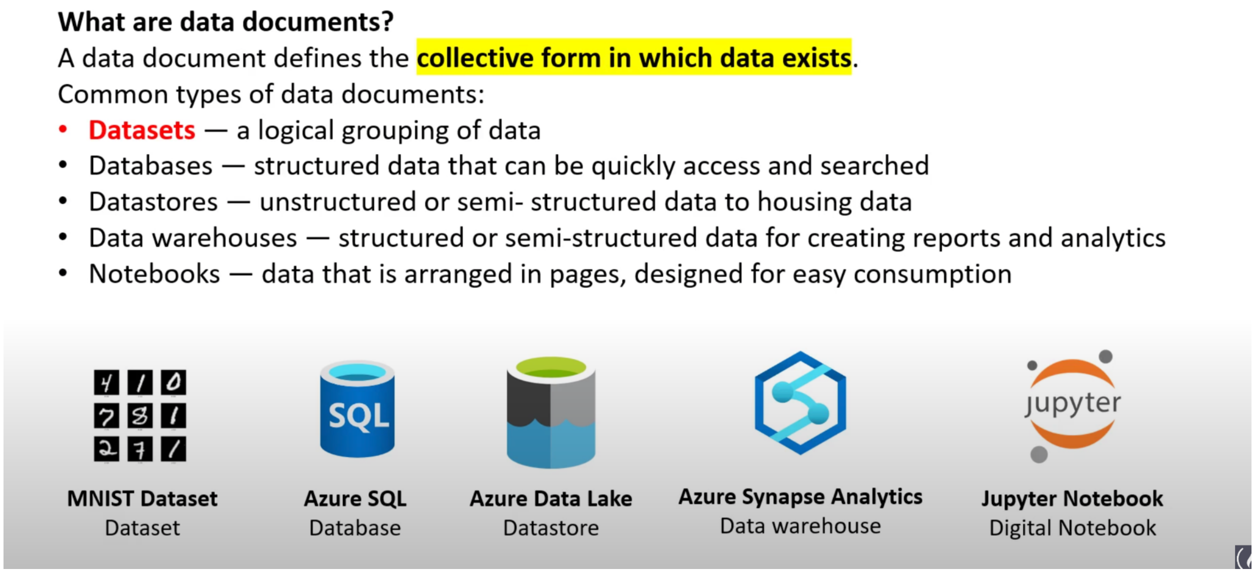
Describe core data concepts (25—30%)

Describe ways to represent data

Data is units of information such as a collection of facts, numbers, descriptions, and observations.

Data structures in which this data is organized often represent ***entities*** that are important to an organization (such as customers, products, sales orders, and so on).

Each entity typically has one or more ***attributes***, or characteristics (for example, a customer might have a name, an address, a phone number, and so on).

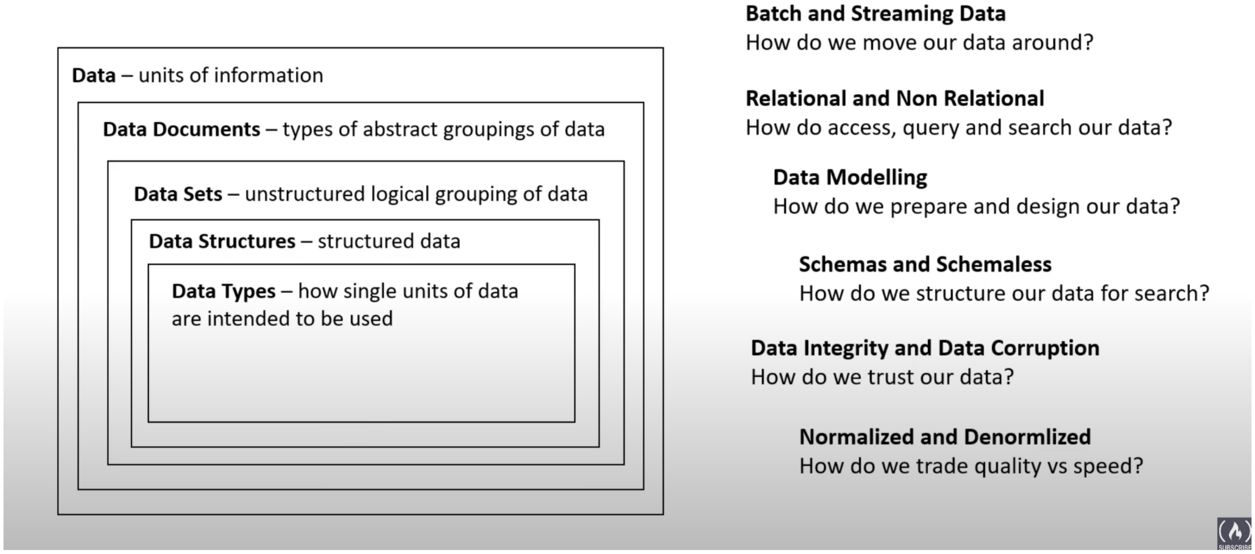


What are data documents?

A data document defines the **collective form in which data exists**.

Common types of data documents:

- **Datasets** — a logical grouping of data
- Databases — structured data that can be quickly access and searched
- Datastores — unstructured or semi- structured data to housing data
- Data warehouses — structured or semi-structured data for creating reports and analytics
- Notebooks — data that is arranged in pages, designed for easy consumption



Data – units of information

Data Documents – types of abstract groupings of data

Data Sets – unstructured logical grouping of data

Data Structures – structured data

Data Types – how single units of data are intended to be used

Batch and Streaming Data
How do we move our data around?

Relational and Non Relational
How do access, query and search our data?

Data Modelling
How do we prepare and design our data?

Schemas and Schemaless
How do we structure our data for search?

Data Integrity and Data Corruption
How do we trust our data?

Normalized and Denormlized
How do we trade quality vs speed?

Schema is a formal language that describes the structure of data in a database.
Query is a request for data results(reads) all to perform operations such as inserting, updating, and deleting data (writes)

- **Describe features of structured data**

Structured data is data that has a relationship(schema), is easy to browse and easy to search. The most common structured data is **tabular data.**
Postgres, MySQL, Azure SQL, and Azure Synapse Analytics are examples of a structured database.

Schema is a formal language that describes the structure of data in a database.

- **Describe features of semi-structured**

Semi-structured data is information that has some structure but allows for some relationships between entity instances. (Common format JSON)
Azure Tables, Azure Cosmos DB, Mongo DB, and Apache Cassandra for storing semi-structured data.

- **Describe features of unstructured data**

  Documents, images, audio, video data, and binary files might not have a specific structure. This kind of data is referred to as ***unstructured*** **data**.
  Azure Blob storage, Azure Files, and Azure Data Lake for storing unstructured data.

Identify options for data storage

- **Describe common formats for data files**

  Delimited Text Files
  The most common format for delimited data is comma-separated values (CSV) in which fields are separated by commas, and rows are terminated by a carriage return / new line.
  Other common formats include tab-separated values (TSV) and space-delimited (in which tabs or spaces are used to separate fields), and fixed-width data in which each field is allocated a fixed number of characters.

  JSON
  JSON is a ubiquitous format in which a hierarchical document schema is used to define data entities (objects) that have multiple attributes. Each attribute might be an object (or a collection of objects); making JSON a flexible format that's good for both structured and semi-structured data.

  XML
  XML is a human-readable data format that was popular in the 1990s and 2000s. It's largely been superseded by the less verbose JSON format, but there are still some systems that use XML to represent data.

  BLOB (Binary Large Object)

Ultimately, all files are stored as binary data (1's and 0's), but in the human-readable formats discussed above, the bytes of binary data are mapped to printable characters
Common types of data stored as binary include images, video, audio, and application-specific documents.

- **Describe types of databases**

Relational Databases

Relational databases are commonly used to store and query structured data. The data is stored in tables that represent entities, such as customers, products, or sales orders. Each instance of an entity is assigned a *primary key* that uniquely identifies it, and these keys are used to reference the entity instance in other tables.
The tables are managed and queried using Structured Query Language (SQL)

Non-relational Databases

Non-relational databases are data management systems that don't apply a relational schema to the data. Non-relational databases are often referred to as NoSQL databases, even though some support a variant of the SQL language.

There are four common types of Non-relational databases commonly in use:

- Key-value databases
- Document databases
- Column family databases
- Graph databases

Describe common data workloads

- **Describe features of transactional workloads**

A transactional data processing system is the primary function of business computing. A transactional system records *transactions* that encapsulate specific events that the organization wants to track.

Transactional systems are often high-volume, sometimes handling many millions of transactions in a single day. The data being processed has to be accessible very quickly. The work performed by transactional systems is often referred to as **Online Transactional Processing (OLTP)**.

OLTP systems enforce transactions that support so-called **ACID** semantics:

- **Atomicity** – each transaction is treated as a single unit, which succeeds completely or fails completely.
- **Consistency** – transactions can only take the data in the database from one valid state to another.
- **Isolation** – concurrent transactions cannot interfere with one another and must result in a consistent database state.
- **Durability** – when a transaction has been committed, it will remain committed.

- **Describe features of analytical workloads**

Analytical data processing typically uses read-only (or read-*mostly*) systems that store vast volumes of historical data or business metrics.

- Data files may be stored in a central data lake for analysis.
- **An extract, transform, and load (ETL)** process copies data from files and OLTP databases into a data warehouse that is optimized for read activity.
- Data in the data warehouse may be aggregated and loaded into an online **analytical processing (OLAP)** model, or *cube*.
- The data in the data lake, data warehouse, and analytical model can be queried to produce reports, visualizations, and dashboards.

Data lakes are common in large-scale data analytical processing scenarios, where a large volume of file-based data must be collected and analyzed. (big data)

Data warehouses are an established way to store data in a relational schema that is optimized for read operations – primarily queries to support reporting and data visualization. (analytic workloads)

Data warehouses perform aggregation. Aggregation is grouping data to find a total or average.

Identify roles and responsibilities for data workloads

- **Describe responsibilities for database administrators**

  Configures and maintains a databases
  - Database management
  - Manage security, granting user access
  - Backups
  - Monitors performance

  Common tools:
  - Azure Data Studio
  - SQL Server Management Studio (SSMS)
  - Azure Portal/CLI

- **Describe responsibilities for data engineers**

  Design and implement data tasks
  - Database pipelines and process
  - Data ingestion storage
  - Prepare data for analytics/ analytical processing

  Common tools:
  - Azure Synapse Studio

  - SQL
  - Azure CLI

- **Describe responsibilities for data analysts**

  Analyze business data
  - Provides insight into the data
  - Visual reporting
  - Modeling data for analysis
  - Combines data for visualization and analysis

  Common tools:
  - Power BI Desktop/ Portal/ Services/ Report builder

- **Most commonly used data services for modern transactional and analytical solutions.**
  - **Azure SQL Database** – a fully managed PaaS database hosted in Azure
  - **Azure SQL Managed Instance** – a hosted instance of SQL Server with automated maintenance, which allows a more flexible configuration than Azure SQL DB but with more administrative responsibility for the owner.
  - **Azure SQL VM** – a virtual machine with an installation of SQL Server, allowing maximum configurability with full management responsibility.
  - **Azure Database for MySQL** - a simple-to-use open-source database management system that is commonly used in *Linux*, *Apache*, *MySQL*, and *PHP* (LAMP) stack apps.
  - **Azure Database for MariaDB** - a newer database management system, created by the original developers of MySQL. The database engine has since been rewritten and optimized to improve performance. MariaDB offers compatibility with Oracle Database (another popular commercial database management system).
  - **Azure Database for PostgreSQL** - a hybrid relational-object database. You can store data in relational tables, but a PostgreSQL database also enables you to store custom data types, with their own non-relational properties.

- **Azure Cosmos DB** is a global-scale non-relational (*NoSQL*) database system that supports multiple application programming interfaces (APIs), enabling you to store and manage data as JSON documents, key-value pairs, column families, and graphs.
- **Azure Storage**
  - Blob containers - scalable, cost-effective storage for binary files.
  - File shares - network file shares such as you typically find in corporate networks.
  - Tables - key-value storage for applications that need to read and write data values quickly.
- **Azure Data Factory** is an Azure service that enables you to define and schedule data pipelines to transfer and transform data. You can integrate your pipelines with other Azure services, enabling you to ingest data from cloud data stores, process the data using cloud-based computing, and persist the results in another data store.

  Azure Data Factory is used by data engineers to build *extract*, *transform*, and *load* (ETL) solutions that populate analytical data stores with data from transactional systems across the organization.

- **Azure Synapse Analytics** is a comprehensive, unified data analytics solution that provides a single service interface for multiple analytical capabilities, including:
  - Pipelines - based on the same technology as Azure Data Factory.
  - SQL - a highly scalable SQL database engine, optimized for data warehouse workloads.
  - Apache Spark - an open-source distributed data processing system that supports multiple programming languages and APIs, including Java, Scala, Python, and SQL.
  - Azure Synapse Data Explorer - a high-performance data analytics solution optimized for real-time querying of log and telemetry data using Kusto Query Language (KQL).
- **Azure Databricks** is an Azure-integrated version of the popular Databricks platform, which combines the Apache Spark data processing

platform with SQL database semantics and an integrated management interface to enable large-scale data analytics.

- **Azure HDInsight** is an Azure service that provides Azure-hosted clusters for popular Apache open-source big data processing technologies, including
    - Apache Spark - open-source unified analytics engine for big data and machine learning.
    - Apache Hadoop - a distributed system that uses *MapReduce* jobs to process large volumes of data efficiently across multiple cluster nodes. MapReduce jobs can be written in Java or abstracted by interfaces such as Apache Hive - a SQL-based API that runs on Hadoop.
    - Apache HBase - an open-source system for large-scale NoSQL data storage and querying.
    - Apache Kafka - a message broker for data stream processing.
    - Apache Storm - an open-source system for real-time data processing through a topology of *spouts* and *bolts*.
- **Azure Stream Analytics** is a real-time stream processing engine that captures a stream of data from input, applies a query to extract and manipulate data from the input stream, and writes the results to output for analysis or further processing.
- **Azure Data Explorer** is a standalone service that offers the same high-performance querying of log and telemetry data as the Azure Synapse Data Explorer runtime in Azure Synapse Analytics.
- **Microsoft Purview** provides a solution for enterprise-wide data governance and discoverability. You can use Microsoft Purview to create a map of your data and track data lineage across multiple data sources and systems, enabling you to find trustworthy data for analysis and reporting.
- **Microsoft Power BI** is a platform for analytical data modeling and reporting that data analysts can use to create and share interactive data visualizations.

Identify considerations for relational data on Azure (20—25%)

Describe relational concepts

- **Identify features of relational data**

  In a relational database, you model collections of entities from the real world as ***tables***. A table contains rows, and each row represents a single instance of an entity.

  Relational tables are a format for structured data, and each row in a table has the same columns. Each column stores data of a specific data type.

  Primary Key

  Foreign Key

- **Describe normalization and why it is used**

  Normalization is a term used by database professionals for a schema design process that minimizes data duplication and enforces data integrity.

  - Separate each *entity* into its own table.
  - Separate each discrete *attribute* into its own column.
  - Uniquely identify each entity instance (row) using a *primary key*.
  - Use *foreign key* columns to link related entities.

- **Identify common structured query language (SQL) statements**

SQL stands for *Structured Query Language* and is used to communicate with a relational database. It's the standard language for relational database management systems. SQL commands:

- Data Definition Language (DDL): create, alter, drop, rename
- Data Control Language (DCL): grant, deny, revoke
- Data Manipulation Language (DML): select, insert, update, delete

- **Identify common database objects**
  - A **view** is a virtual table based on the results of a SELECT query. You can think of a view as a window on specified rows in one or more underlying tables.
  - A **stored procedure** defines SQL statements that can be run on command. Stored procedures are used to encapsulate programmatic logic in a database for actions that applications need to perform when working with data. You can define a stored procedure with parameters to create a flexible solution for common actions that might need to be applied to data based on a specific key or criteria.
  - An **index** helps you search for data in a table. It improves the speed of the read by storing the same or partial redundant data.

Describe relational Azure data services

- **Describe the Azure SQL family of products including Azure SQL Database, Azure SQL**

  Azure SQL is a collective term for a family of Microsoft SQL Server-based database services in Azure.

  - **SQL Server on Azure VMs**: A virtual machine running in Azure with an installation of SQL Server. (IaaS) Lift and shift migration.

    - **Azure SQL Managed Instance**(PaaS): provides near-100% compatibility with on-premises SQL Server instances while abstracting the underlying hardware and operating system.
    - **Azure SQL Database**(PaaS): A fully managed, highly scalable database service that is designed for the cloud. This service includes the core database-level capabilities of on-premises SQL Server and is a good option when you need to create a new application in the cloud.

      You create a managed database server in the cloud and then deploy your databases on this server.

        - ○ **Single Database**: enables you to quickly set up and run a single SQL Server database. You create and run a database server in the cloud, and you access your database through this server.
        - ○ **Elastic Pool**: Default multiple databases that can share the same resources, such as memory, data storage space, and processing power through multiple-tenancy.

    □ Automatically updates
    □ Scalability
    □ High availability guarantees(99.995%)
    □ Advanced threat protection

    - **Azure SQL Edge**: A SQL engine that is optimized for Internet-of-things (IoT) scenarios that need to work with streaming time-series data.

- **Managed Instance, and SQL Server on Azure Virtual Machines**
    - ○ **Azure SQL Managed instance** effectively runs a fully controllable instance of SQL Server in the cloud. You can install multiple databases on the same instance.

      Managed instances depend on other Azure services such as Azure Storage for backups, Azure Event Hubs for telemetry, Azure Active Directory for authentication, Azure Key Vault for Transparent Data

Encryption (TDE), and a couple of Azure platform services that provide security and supportability features. The managed instances make connections to these services.

- □ enables a system administrator to spend less time on administrative tasks
- □ operating system and database management system software installation and patching
- □ resizing and configuration
- □ backups, database replication
- □ high availability configuration

○ **SQL Server on Virtual Machines(IaaS)** enables you to use full versions of SQL Server in the Cloud without having to manage any on-premises hardware.

SQL Server running on an Azure virtual machine effectively replicates the database running on real on-premises hardware. Migrating from the system running on-premises to an Azure virtual machine is no different than moving the databases from one on-premises server to another.

- □ combination of on-premises and cloud-hosted deployments

- **Identify Azure database services for open-source database systems**

Azure data services are available for other popular relational database systems. The primary reason for these services is to enable organizations that use them in on-premises apps to move to Azure quickly, without making significant changes to their applications.

Azure Database for MySQL is a PaaS implementation of MySQL in the Azure cloud, based on the MySQL Community Edition.

Azure Database for MariaDB is an implementation of the MariaDB database management system adapted to run in Azure. It's based on the MariaDB Community Edition.

Azure Database for PostgreSQL to run a PaaS implementation of PostgreSQL in the Azure Cloud. This service provides the same availability, performance, scaling, security, and administrative benefits as the MySQL service.

Describe considerations for working with non-relational data on Azure (15—20%)

Describe the capabilities of Azure storage

- **Describe Azure Blob storage**

  Azure Blob Storage is a service that enables you to store massive amounts of unstructured data as binary large objects, or *blobs*, in the cloud.

  In an Azure storage account, you store blobs in *containers*.

  - **Block blobs:** handled as a set of blocks. (100 MB)

    The block is the smallest amount of data that can be read or written as an individual unit. Block blobs are best used to store discrete, large, binary objects that change infrequently.

  - **Page blobs**: organized as a collection of fixed-size 512-byte pages. A page blob is optimized to support random read and write operations. (store virtual hard drive for VM)

    - **Append blobs**: a block blob optimized to support append operations. You can only add blocks to the end of an append blob; updating or deleting existing blocks isn't supported. (logging data from VM)

- **Describe Azure File storage**

  Azure file is a fully managed file share in the cloud. A file share is a centralized server for storage(VM) that allows multiple connections.

  Azure Files is essentially a way to create cloud-based network shares, such as you typically find in on-premises organizations to make documents and other files available to multiple users. By hosting file shares in Azure, organizations can eliminate hardware costs and maintenance overhead, and benefit from high availability and scalable cloud storage for files. (up to 100 TB)

  AzCopy

  Azure File Sync

  Azure File Storage offers two performance tiers. The Standard tier uses hard disk-based hardware in a datacenter, and the Premium tier uses solid-state disks.

  To connect to the file share in the cloud:

    - **Server Message Block (SMB)** file sharing is commonly used across multiple operating systems (Windows, Linux, macOS).
    - **Network File System (NFS)** shares are used by some Linux and macOS versions. To create an NFS share, you must use a premium tier storage account and create and configure a virtual network through which access to the share can be controlled.

- **Describe Azure Table storage**

  Azure Table Storage is a NoSQL storage solution that makes use of tables containing *key/value* data items. Each item is represented by a row that contains columns for the data fields that need to be stored.

  An Azure Table enables you to store semi-structured data. Data in Azure Table storage is usually denormalized, with each row holding the entire data for a logical entity.

  To help ensure fast access, Azure Table Storage splits a table into partitions. Partitioning is a mechanism for grouping related rows, based on common property or partition key. Rows that share the same partition key will be stored together.

Describe the capabilities and features of Azure Cosmos DB

Azure Cosmos DB(PaaS) is a service for fully managed NoSQL databases that are design to scale and high performance.

- **Identify use cases for Azure Cosmos DB**

  Cosmos DB is a highly scalable database management system. Cosmos DB automatically allocates space in a container for your partitions. Indexes are created and maintained automatically.

  - ***IoT and telematics***: These systems typically ingest large amounts of data in frequent bursts of activity. Cosmos DB can accept and store this information quickly.
  - ***Retail and marketing*****:** Microsoft uses Cosmos DB for its own e-commerce platforms that run as part of Windows Store and

Xbox Live. It's also used in the retail industry for storing catalog data and for event sourcing in order processing pipelines.

  - ***Gaming***: Games often require single-millisecond latencies for reads and write to provide an engaging in-game experience. A game database needs to be fast and be able to handle massive spikes in request rates during new game launches and feature updates.
  - ***Web and mobile applications***: well suited for modeling social interactions, integrating with third-party services, and for building rich personalized experiences.

- **Describe Azure Cosmos DB APIs**

  An ***API*** is an ***Application Programming Interface***. Database management systems provide a set of APIs that developers can use to write programs that need to access data.

  Azure Cosmos DB supports multiple APIs, enabling developers to easily migrate data from commonly used NoSQL stores and apply their existing programming skills. When you provision a new Cosmos DB instance, you select the API that you want to use.

  - **Core (SQL) API**

    The native API in Cosmos DB manages data in JSON **document format**, and despite being a NoSQL data storage solution, uses SQL syntax to work with the data.

    SELECT, WHERE, FROM, ORDER, BY, SUM

    Usage: product catalog

  - **Mongo DB API**

    MongoDB is a popular open-source database in which data is stored in **Binary JSON (BSON) format**. The Azure Cosmos DB

MongoDB API enables developers to use MongoDB client libraries and code to work with data in Azure Cosmos DB.

Usage: import historical order data

- **Table API**

  The Table API is used to work with data in **key-value tables**, similar to Azure Table Storage. The Azure Cosmos DB Table.

  API offers greater scalability and performance than Azure Table Storage.

  Usage: store IoT data

- **Cassandra API**

  The Cassandra API is compatible with Apache Cassandra, which is a popular open-source database that uses a **column-family storage structure**. Column families are tables, similar to those in a relational database, with the exception that every row doesn't need to have the same columns.

  Usage: Web analytics

- **Gremlin API**

  The Gremlin API is used with data in a **graph structure**; in which entities are defined as *vertices* that form nodes in connected graphs. Nodes are connected by *edges* that represent relationships, like this:

  Usage: a recommendation engine

Describe an analytics workload on Azure (25—30%)

Describe common elements of large-scale analytics

Data warehouse architecture

- **Data ingestion and processing** – data from one or more transactional data stores, files, real-time streams, or other sources is loaded into a data lake or a relational data warehouse.

  The load(transform) operation usually involves ETL, an ELT process in which the data is cleaned, filtered, and restructured for analysis.

  ETL: transform data from one data store to another, doesn't work with a data lake.

  ELT: transformations done at the target data store, works with data lakes, more common in cloud services

- **Analytical data store** – data stores for large-scale analytics include relational *data warehouses*, file-system-based *data lakes*, and hybrid architectures that combine features of data warehouses and data lakes.
- **Analytical data model**- described as *cubes*, in which numeric data values are aggregated across one or more dimensions
- **Data visualization** – data analysts consume data from analytical models, and directly from analytical stores to create reports, dashboards, and other visualizations.

- ### Describe considerations for data ingestion and processing

On Azure, large-scale data ingestion is best implemented by creating *pipelines* that orchestrate ETL processes. You can create and run pipelines using Azure Data Factory, or you can use the same pipeline engine in Azure Synapse Analytics if you want to manage all of the components of your data warehousing solution in a unified workspace.

Pipelines consist of one or more *activities* that operate on data. An input dataset provides the source data, and activities can be defined as a data flow that incrementally manipulates the data until an output dataset is produced.

Pipelines use *linked services* to load and process data – enabling you to use the right technology for each step of the workflow.

- **Describe options for analytical data stores**

  There are two common types of analytical data store.

  - A ***data warehouse*** is a relational database in which the data is stored in a schema that is optimized for data analytics rather than transactional workloads.

    The data from a transactional store is transformed into a schema in numeric values stored in central *fact* tables, which are related to one or more *dimension* tables that represent entities.

    A data warehouse is a great choice when you have transactional data that can be organized into a structured schema of tables, and you want to use SQL to query them.

  - A ***data lake*** is a file store, usually on a distributed file system for high-performance data access.

    Data lakes are great for supporting a mix of structured, semi-structured, and even unstructured data that you want to analyze without the need for schema enforcement when the data is written to the store.

  - You can use a hybrid approach that combines features of data lakes and data warehouses in a *lake database* or *data lakehouse*.

- **Describe Azure services for data warehousing, including Azure Synapse Analytics, Azure Databricks, Azure HDInsight, and Azure Data Factory**

  On Azure, there are three main services that you can use to implement a large-scale analytical store.

  - **Azure Synapse Analytics** is a unified, end-to-end solution for large-scale data analytics. It brings together multiple technologies and capabilities, enabling you to combine the data integrity and reliability of a scalable, high-performance SQL Server-based relational **data warehouse** with the flexibility of a **data lake** and open-source Apache Spark.

    It also includes native support for log and telemetry analytics with Azure Synapse Data Explorer pools, as well as built-in data pipelines for data ingestion and transformation. All Azure Synapse Analytics services can be managed through a single, interactive user interface called Azure Synapse Studio, which includes the ability to create interactive notebooks in which Spark code and markdown content can be combined. Synapse Analytics is a great choice when you want to create a single, unified analytics solution on Azure.

  - **Azure Databricks** is a comprehensive data analytics solution built on Apache Spark and offers native SQL capabilities as well as workload-optimized Spark clusters for data analytics and data science.

    Databricks provides an interactive user interface through which the system can be managed and data can be explored in interactive notebooks.

    Azure Databricks for your analytical store if you want to use existing expertise with the platform or if you need to operate in a multi-cloud environment or support a cloud-portable solution. (Big Data, Machine Learning)

- **Azure HDInsight(PaaS)** is an Azure service that supports multiple open-source data analytics cluster types. Although not as user-friendly as Azure Synapse Analytics and Azure Databricks, it can be a suitable option if your analytics solution relies on multiple open-source frameworks or if you need to migrate an existing on-premises Hadoop-based solution to the cloud.
- **Azure Data Factory** is a managed service for ETL, ELT and data integration. Create data driven workflows for orchestrating data movement and transforming data at scale.

Describe consideration for real-time data analytics

- **Describe the difference between batch and streaming data**

Batch processing, in which multiple data records are collected and stored before being processed together in a single operation.

Advantages of batch processing:

- Large volumes of data can be processed at a convenient time.
- It can be scheduled to run at a time when computers or systems might otherwise be idle, such as overnight, or during off-peak hours.

Stream processing, in which a source of data is constantly monitored and processed in real-time as new data events occur.

    - ***Data scope*****:** Batch processing can process all the data in the dataset. Stream processing typically only has access to the most recent data received, or within a rolling time window (the last 30 seconds, for example).
    - ***Data size*****:** Batch processing is suitable for handling large datasets efficiently. Stream processing is intended for individual records or *micro batches* consisting of few records.
    - ***Performance*****:** *Latency* is the time taken for the data to be received and processed. The latency for batch processing is typically a few hours. Stream processing typically occurs immediately, with latency in the order of seconds or milliseconds.
    - ***Analysis*****:** You typically use batch processing to perform complex analytics. Stream processing is used for simple response functions, aggregates, or calculations such as rolling averages.

- **Describe technologies for real-time analytics including Azure Stream Analytics, Azure Synapse Data Explorer, and Spark structured streaming**
    - **Azure Stream Analytics**: (PaaS) is a service for complex event processing and analysis of streaming data. Stream Analytics is used to:
        - ○ Ingest data from an *input*, such as an Azure event hub, Azure IoT Hub, or Azure Storage blob container.
        - ○ Process the data by using a *query* to select, project, and aggregate data values.
        - ○ Write the results to an *output*, such as Azure Data Lake Gen 2, Azure SQL Database, Azure Synapse Analytics, Azure Functions, Azure event hub, Microsoft Power BI, or others.
    - **Spark Structured Streaming**: An open-source library that enables you to develop complex streaming solutions on Apache Spark-based services, including Azure Synapse Analytics, Azure Databricks, and Azure HDInsight.

Spark Structured Streaming library, which provides an application programming interface (API) for ingesting, processing, and outputting results from perpetual streams of data.

Spark Structured Streaming is built on a ubiquitous structure in Spark called a *dataframe*, which encapsulates a table of data.

You use the Spark Structured Streaming API to read data from a real-time data source, such as a Kafka hub, a file store, or a network port, into a "boundless" dataframe that is continually populated with new data from the stream.

You then define a query on the dataframe that selects, projects, or aggregates the data - often in temporal windows. The results of the query generate another dataframe, which can be persisted for analysis or further processing.

- **Azure Data Explorer**: A high-performance database and analytics service that is optimized for ingesting and querying batch or streaming data with a time-series element, and which can be used as a standalone Azure service or as an Azure Synapse Data Explorer runtime in an Azure Synapse Analytics workspace.

  You can use the service as the output for analyzing large volumes of diverse data from data sources such as websites, applications, IoT devices, and more.

  Data is ingested into Data Explorer through one or more connectors or by writing a minimal amount of code.

  To query Data Explorer tables, you can use **Kusto Query Language (KQL)**, a language that is specifically optimized for fast read performance – particularly with telemetry data that includes a timestamp attribute.

The following services are commonly used to ingest data for stream processing on Azure:

- **Azure Event Hubs**: A data ingestion service that you can use to manage queues of event data, ensuring that each event is processed in order, exactly once.
- **Azure IoT Hub**: A data ingestion service that is similar to Azure Event Hubs, but optimized for managing event data from *Internet-of-things* (IoT) devices.
- **Azure Data Lake Store Gen 2**: A highly scalable storage service that is often used in *batch processing* scenarios, but which can also be used as a source of streaming data.
- **Apache Kafka**: An open-source data ingestion solution that is commonly used together with Apache Spark. You can use Azure HDInsight to create a Kafka cluster. (open source streaming platform)

Describe data visualization in Microsoft Power BI

- ### Identify capabilities of Power BI

Microsoft Power BI(SaaS) is to build interactive data visualizations for business users.

A typical workflow for creating a data visualization solution starts with **Power BI Desktop**, you can import data from a wide range of data sources, combine and organize the data from these sources in an analytics data model, and create reports that contain interactive visualizations of the data.

After you've created data models and reports, you can publish them to the **Power BI service**. You can use the service to schedule refreshes of the data sources on which your reports are based, and to share reports with other users. You can also define **dashboards** and apps that combine related reports in a single, easy-to-consume location.

There is also a **Power BI phone app**.

- **Describe features of data models in Power BI**

  Models based on related tables of data and define the numeric values that you want to analyze or report, the model forms a multidimensional structure, which is commonly referred to as a *cube.*

  Dimension tables represent the entities by which you want to aggregate numeric measures. Each entity is represented by a row with a unique key value. The remaining columns represent attributes of an entity.

  The numeric measures that will be aggregated by the various dimensions in the model are stored in *Fact* tables. Each row in a fact table represents a recorded event that has numeric measures associated with it.

  This type of schema, where a fact table is related to one or more dimension tables, is referred to as a **star schema**. (there are five dimensions related to a single fact table)

  Creation of attribute ***hierarchies*** that enable you to quickly *drill up* or *drill down* to find aggregated values at different levels in a hierarchical dimension.

  You can use Power BI to define an analytical model from tables of data, which can be imported from one or more data sources. You can then use the data modeling interface on the Model tab of Power BI Desktop to define your analytical model by creating relationships between fact and dimension tables, defining hierarchies, setting data types and display formats for fields in the tables, and managing other properties of your data that help define a rich model for analysis.

- **Identify appropriate visualizations for data**

Tables and text are often the simplest way to communicate data.

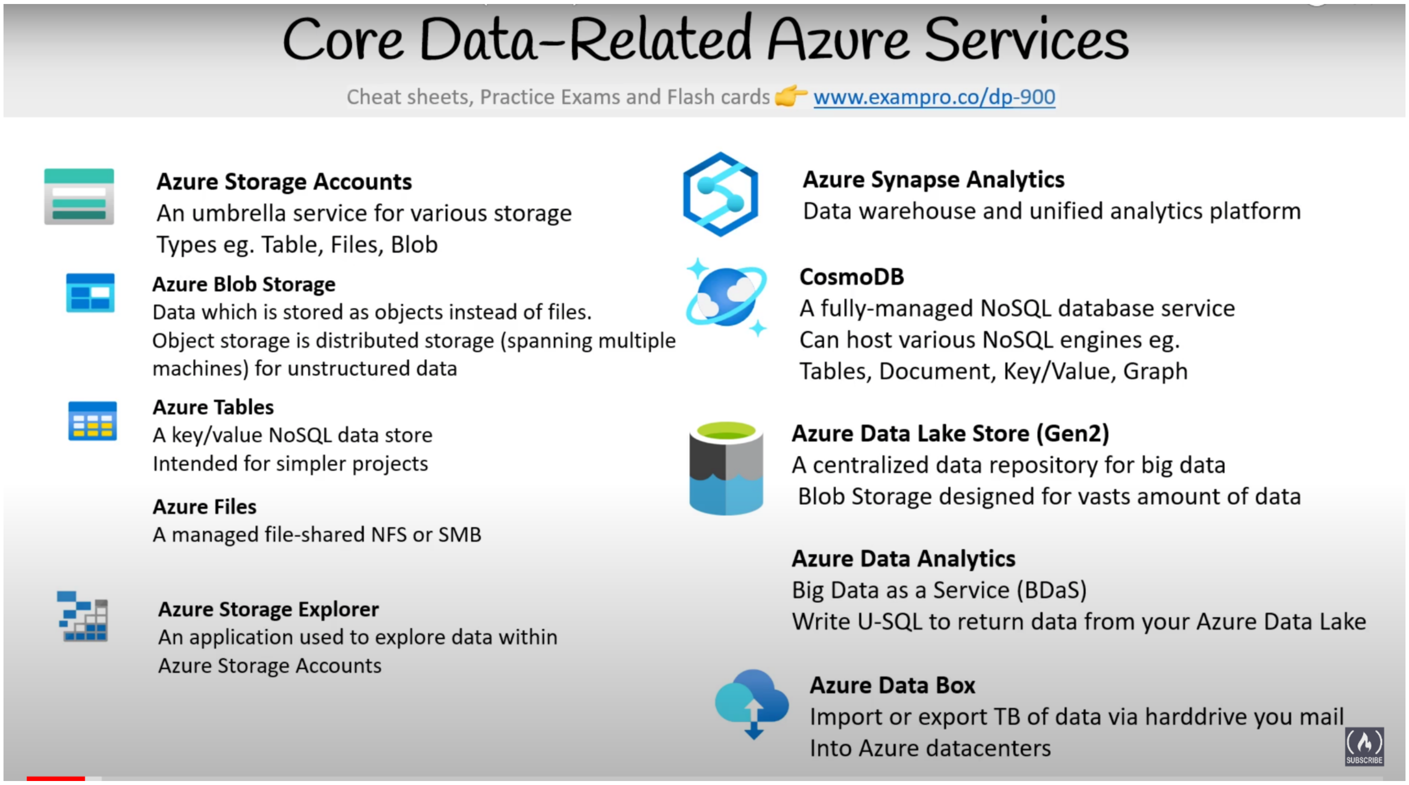
Bar and column charts are a good way to visually compare numeric values for discrete categories.

Line charts can also be used to compare categorized values and are useful when you need to examine trends, often over time.

Pie charts are often used in business reports to visually compare categorized values as proportions of a total.

Scatter plots are useful when you want to compare two numeric measures and identify a relationship or correlation between them.

Maps are a great way to visually compare values for different geographic areas or locations.



Core Data-Related Azure Services

Cheat sheets, Practice Exams and Flash cards 👉 www.exampro.co/dp-900

Azure Storage Accounts
An umbrella service for various storage
Types eg. Table, Files, Blob

Azure Blob Storage
Data which is stored as objects instead of files.
Object storage is distributed storage (spanning multiple machines) for unstructured data

Azure Tables
A key/value NoSQL data store
Intended for simpler projects

Azure Files
A managed file-shared NFS or SMB

Azure Storage Explorer
An application used to explore data within
Azure Storage Accounts

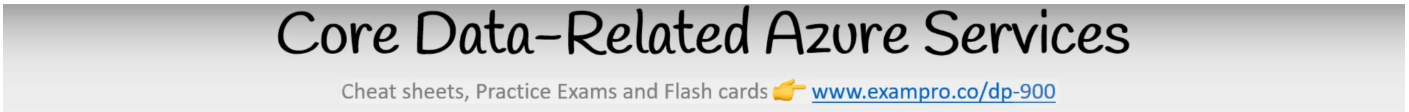
Azure Synapse Analytics
Data warehouse and unified analytics platform

CosmoDB
A fully-managed NoSQL database service
Can host various NoSQL engines eg.
Tables, Document, Key/Value, Graph

Azure Data Lake Store (Gen2)
A centralized data repository for big data
Blob Storage designed for vasts amount of data

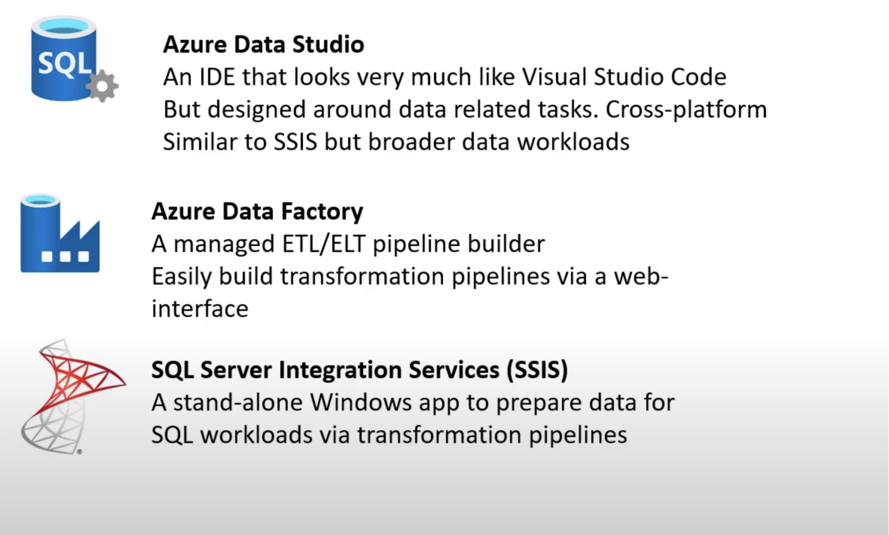
Azure Data Analytics
Big Data as a Service (BDaS)
Write U-SQL to return data from your Azure Data Lake

Azure Data Box
Import or export TB of data via harddrive you mail
Into Azure datacenters



Core Data-Related Azure Services

Cheat sheets, Practice Exams and Flash cards 👉 www.exampro.co/dp-900



Azure Data Studio
An IDE that looks very much like Visual Studio Code
But designed around data related tasks. Cross-platform
Similar to SSIS but broader data workloads

Azure Data Factory
A managed ETL/ELT pipeline builder
Easily build transformation pipelines via a web-interface

SQL Server Integration Services (SSIS)
A stand-alone Windows app to prepare data for
SQL workloads via transformation pipelines