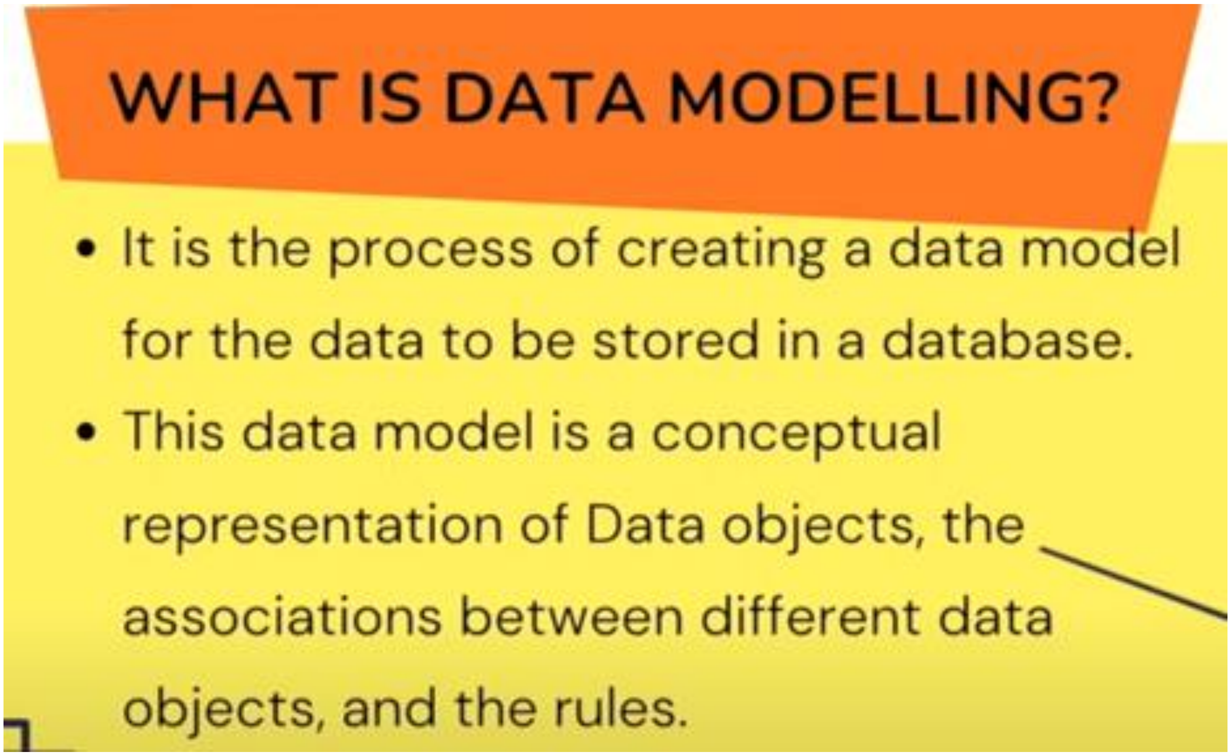
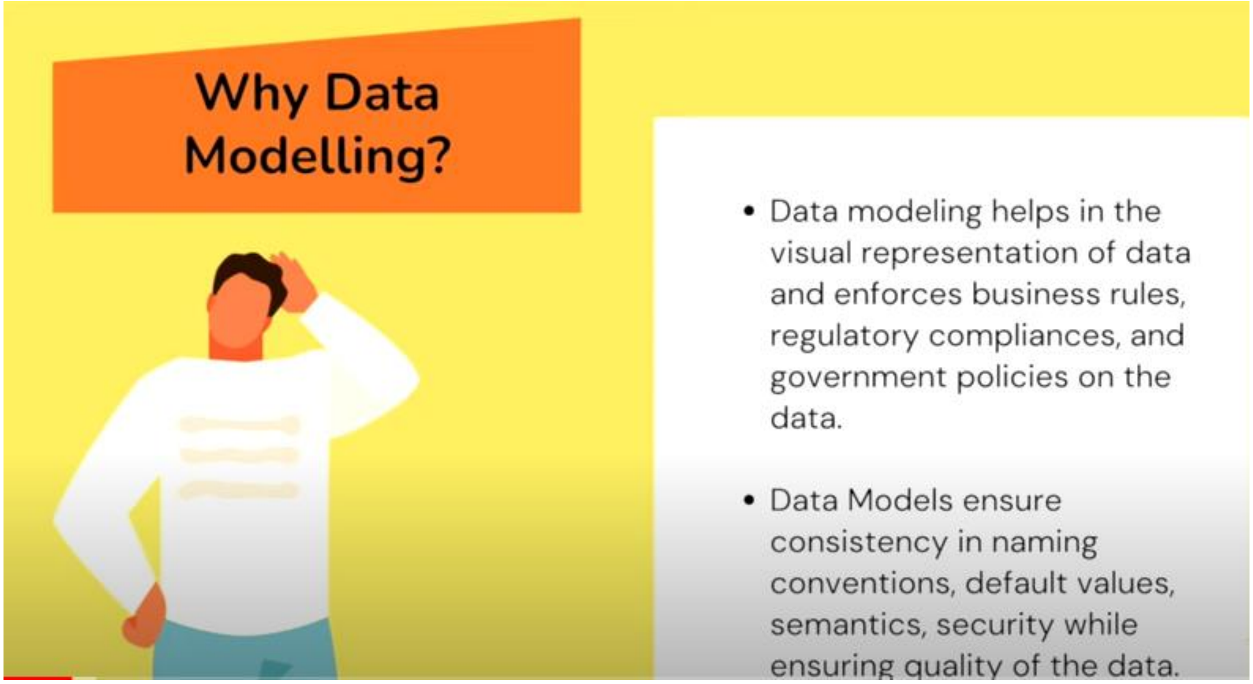


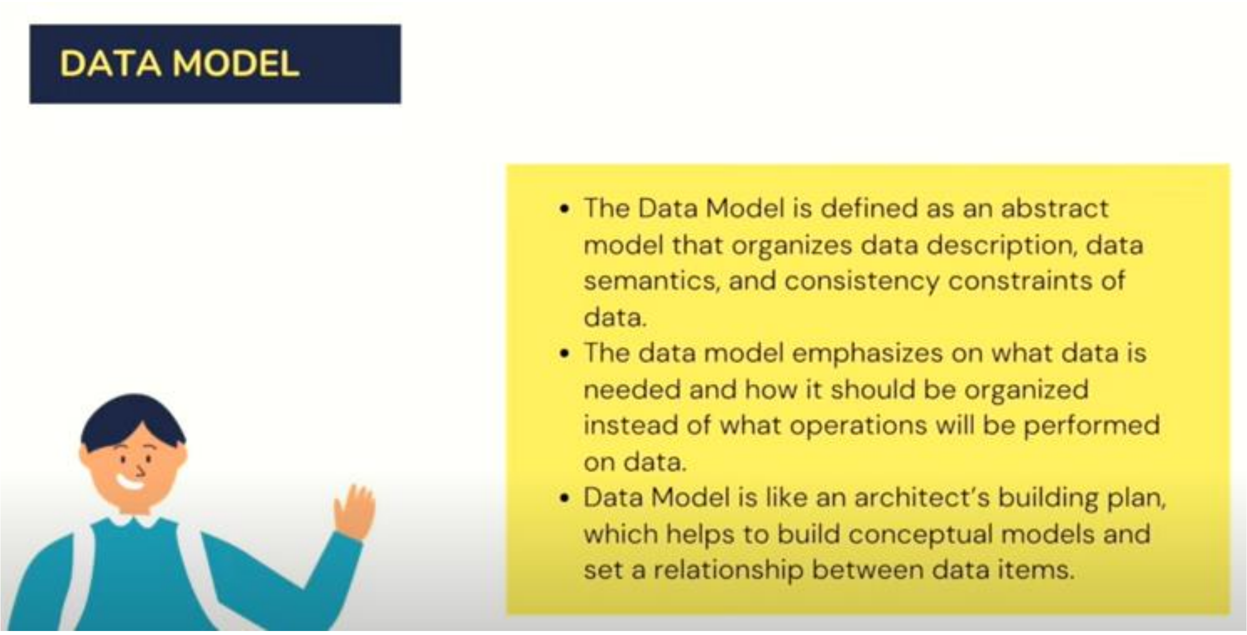
WHAT IS DATA MODELLING?

- It is the process of creating a data model for the data to be stored in a database.
- This data model is a conceptual representation of Data objects, the associations between different data objects, and the rules.



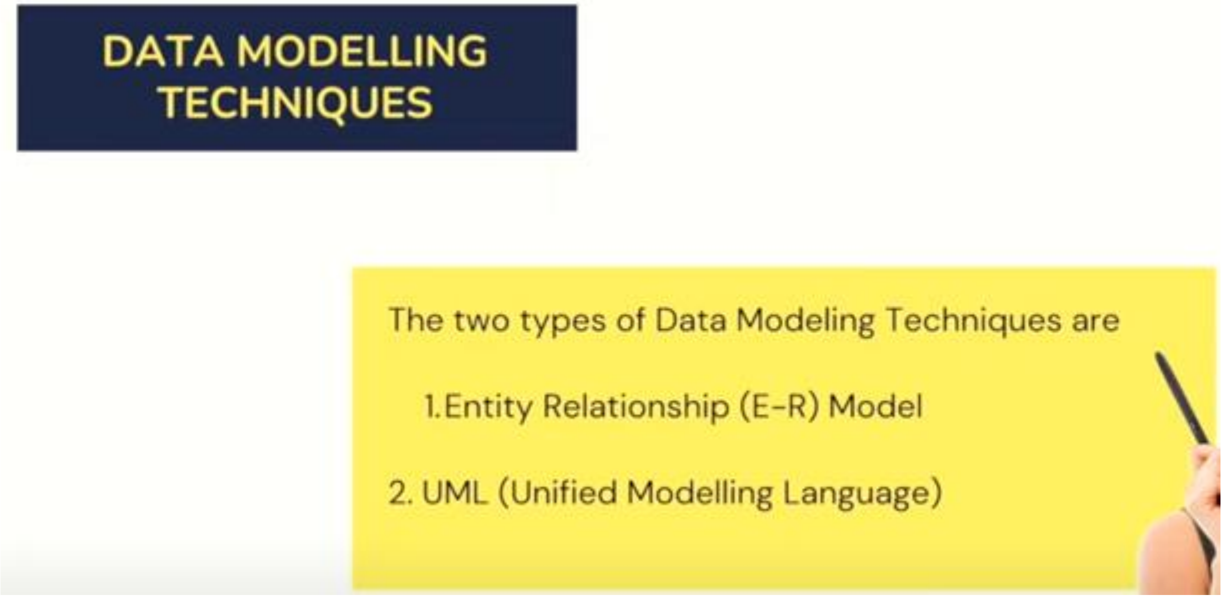
Why Data Modelling?

- Data modeling helps in the visual representation of data and enforces business rules, regulatory compliances, and government policies on the data.
- Data Models ensure consistency in naming conventions, default values, semantics, security while ensuring quality of the data.



DATA MODEL

- The Data Model is defined as an abstract model that organizes data description, data semantics, and consistency constraints of data.
- The data model emphasizes on what data is needed and how it should be organized instead of what operations will be performed on data.
- Data Model is like an architect's building plan, which helps to build conceptual models and set a relationship between data items.

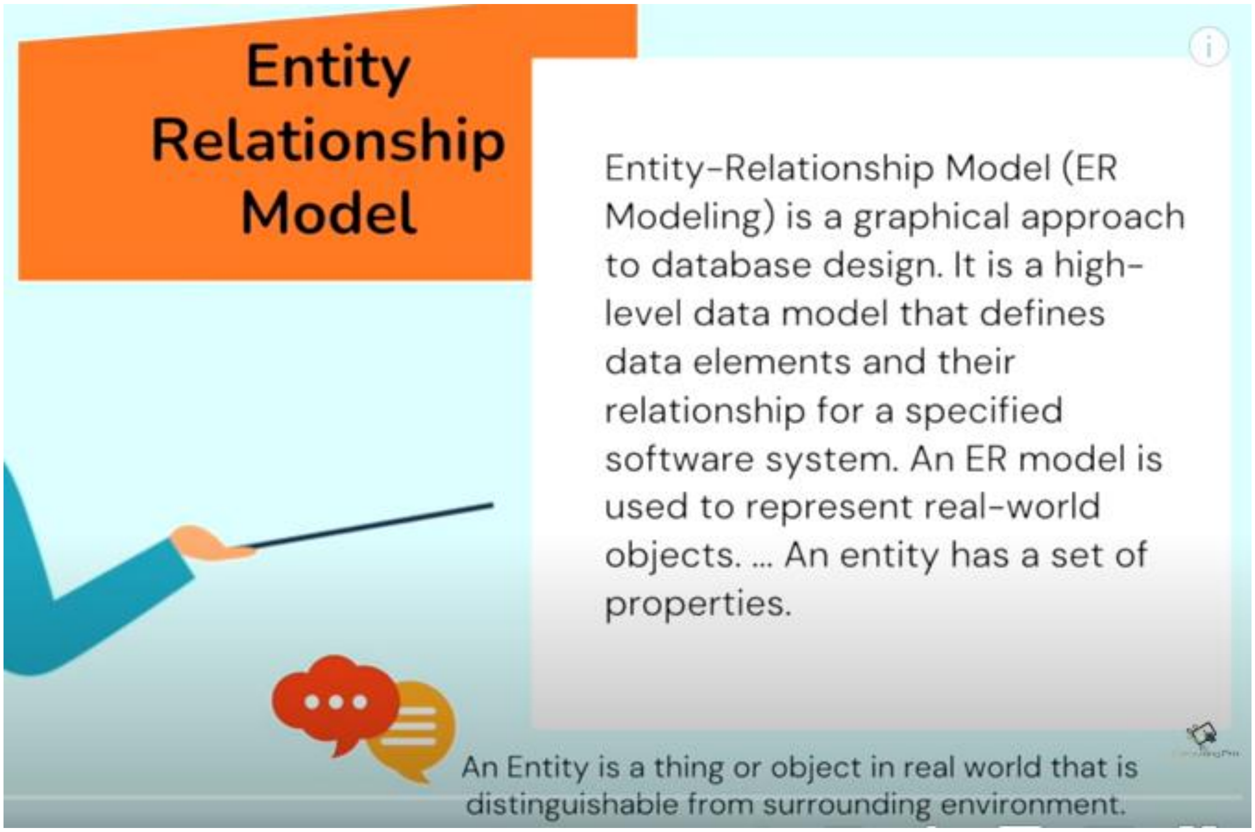


DATA MODELLING TECHNIQUES

The two types of Data Modeling Techniques are

1. Entity Relationship (E-R) Model

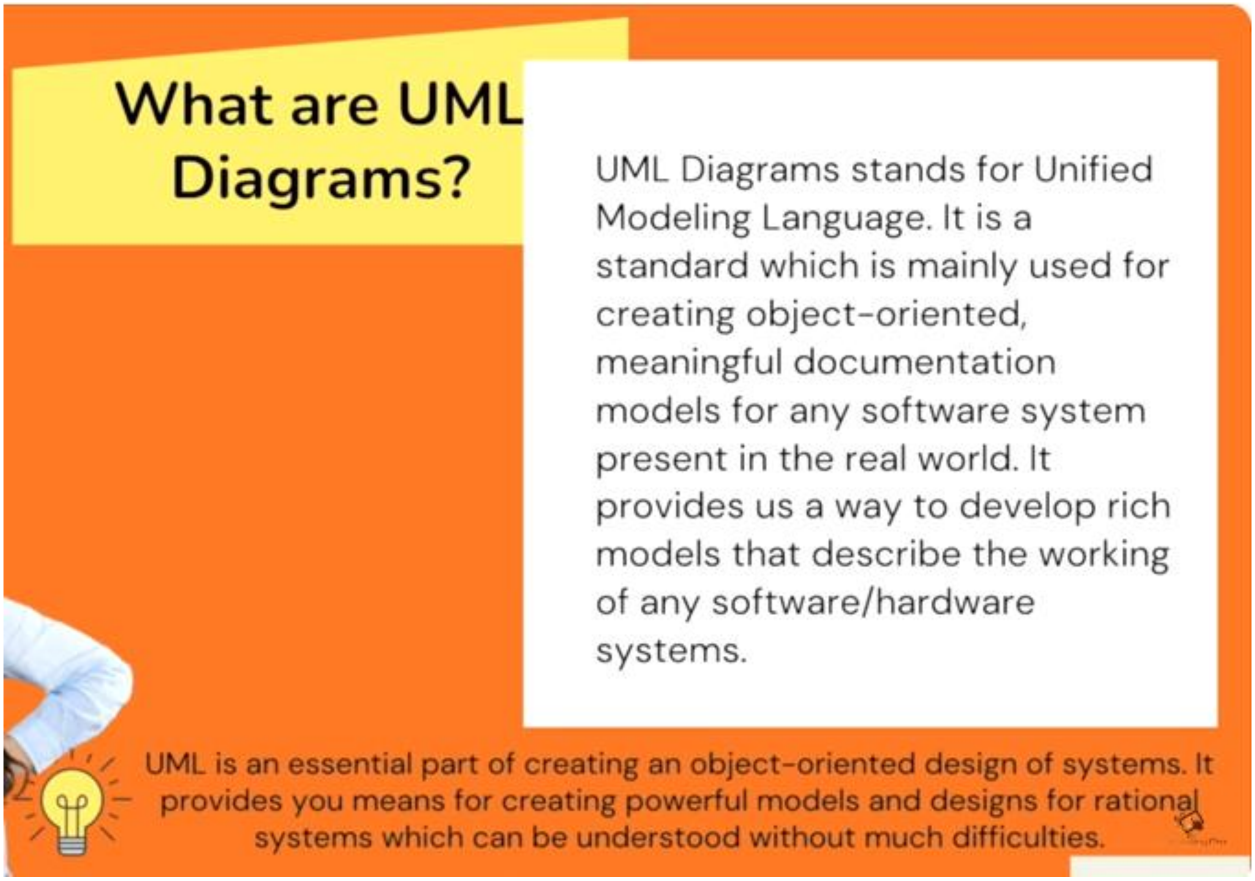
2. UML (Unified Modelling Language)



Entity Relationship Model

Entity-Relationship Model (ER Modeling) is a graphical approach to database design. It is a high-level data model that defines data elements and their relationship for a specified software system. An ER model is used to represent real-world objects. … An entity has a set of properties.

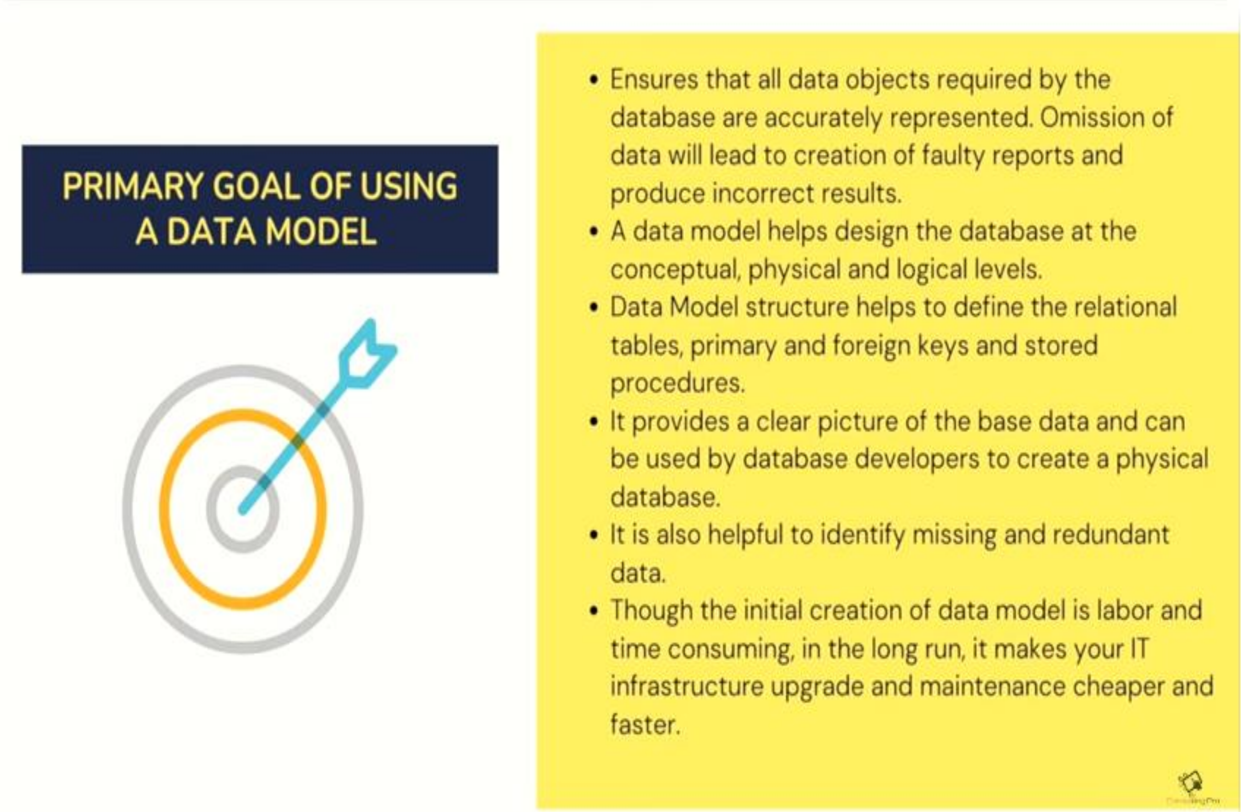
An Entity is a thing or object in real world that is distinguishable from surrounding environment.



What are UML Diagrams?

UML Diagrams stands for Unified Modeling Language. It is a standard which is mainly used for creating object-oriented, meaningful documentation models for any software system present in the real world. It provides us a way to develop rich models that describe the working of any software/hardware systems.

UML is an essential part of creating an object-oriented design of systems. It provides you means for creating powerful models and designs for rational systems which can be understood without much difficulties.

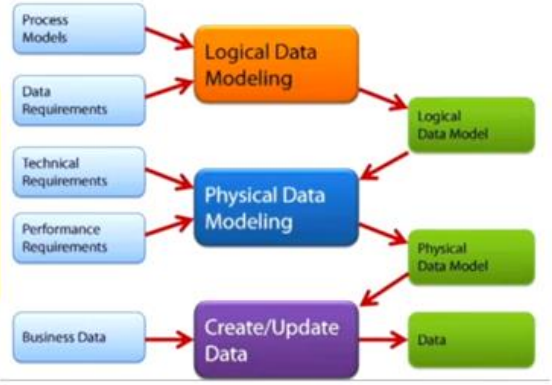


PRIMARY GOAL OF USING A DATA MODEL

- Ensures that all data objects required by the database are accurately represented. Omission of data will lead to creation of faulty reports and produce incorrect results.
- A data model helps design the database at the conceptual, physical and logical levels.
- Data Model structure helps to define the relational tables, primary and foreign keys and stored procedures.
- It provides a clear picture of the base data and can be used by database developers to create a physical database.
- It is also helpful to identify missing and redundant data.
- Though the initial creation of data model is labor and time consuming, in the long run, it makes your IT infrastructure upgrade and maintenance cheaper and faster.

TYPES OF DATA MODELS

- **Conceptual Data Model**
- **Logical Data Model**
- **Physical Data Model**



Process Models

Data Requirements

Logical Data Modeling

Logical Data Model

Technical Requirements

Performance Requirements

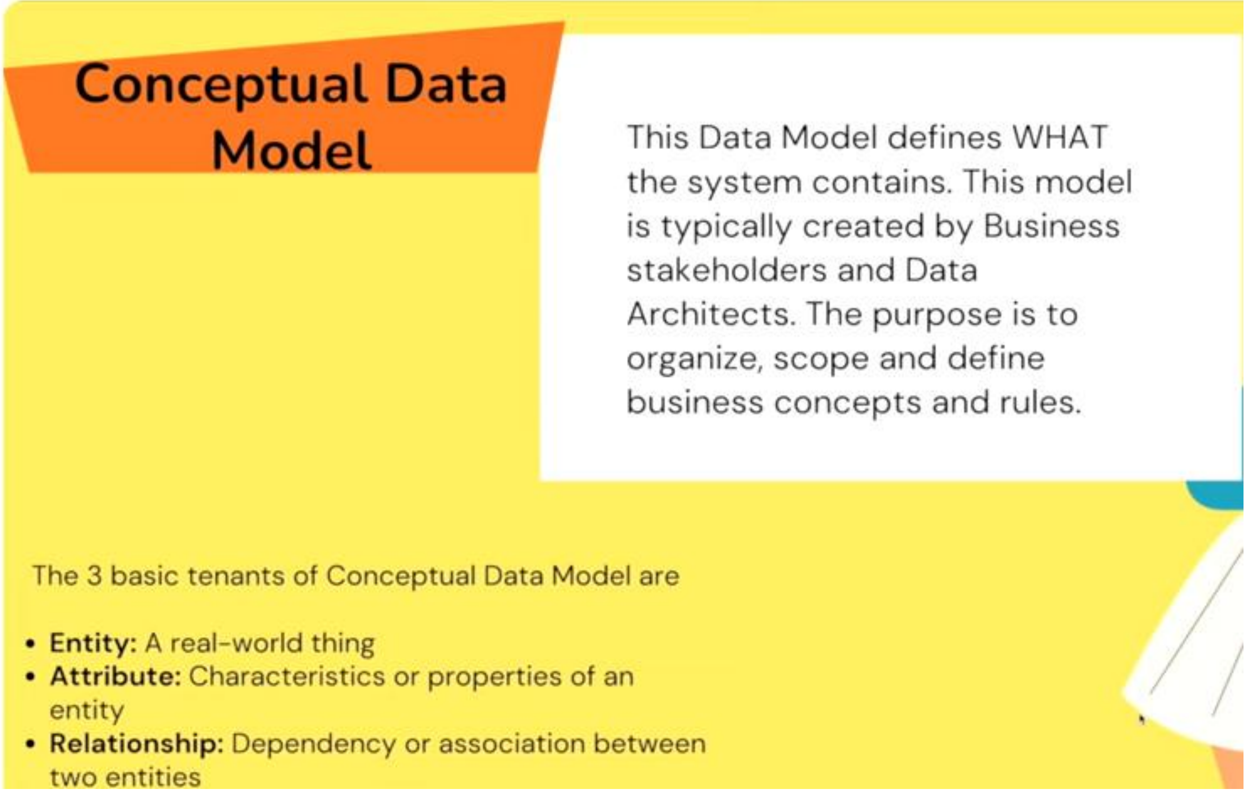
Physical Data Modeling

Physical Data Model

Business Data

Create/Update Data

Data

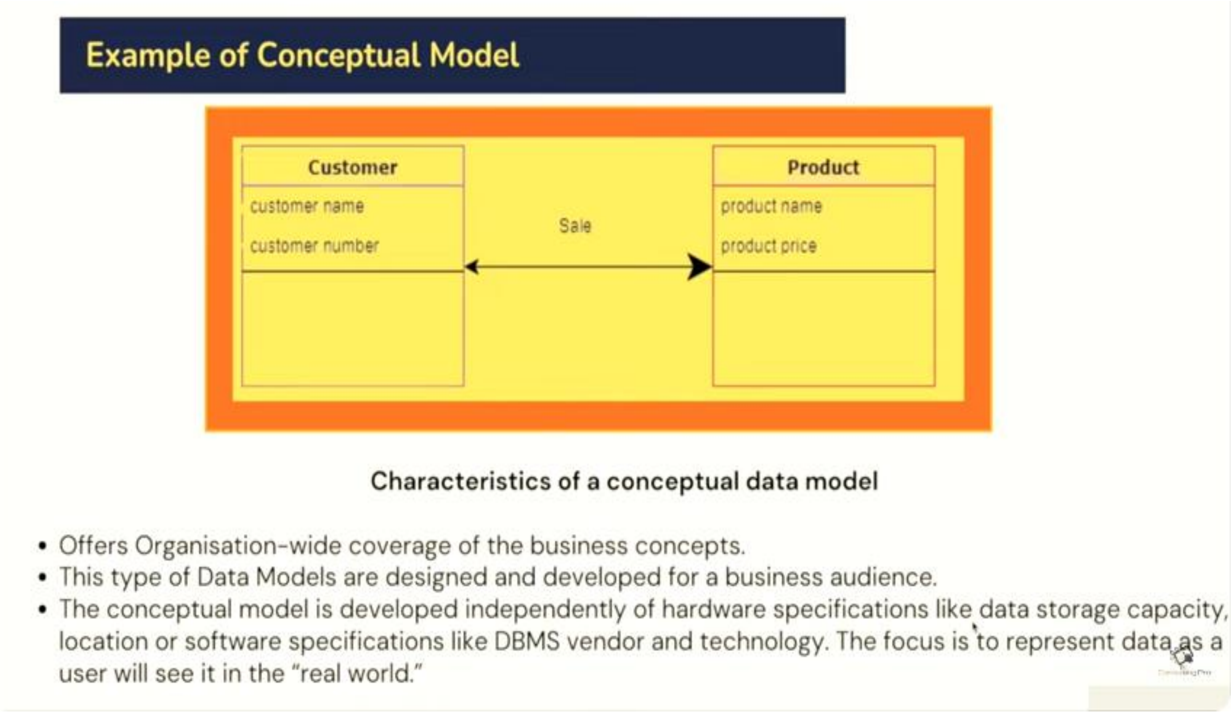


Conceptual Data Model

This Data Model defines WHAT the system contains. This model is typically created by Business stakeholders and Data Architects. The purpose is to organize, scope and define business concepts and rules.

The 3 basic tenants of Conceptual Data Model are

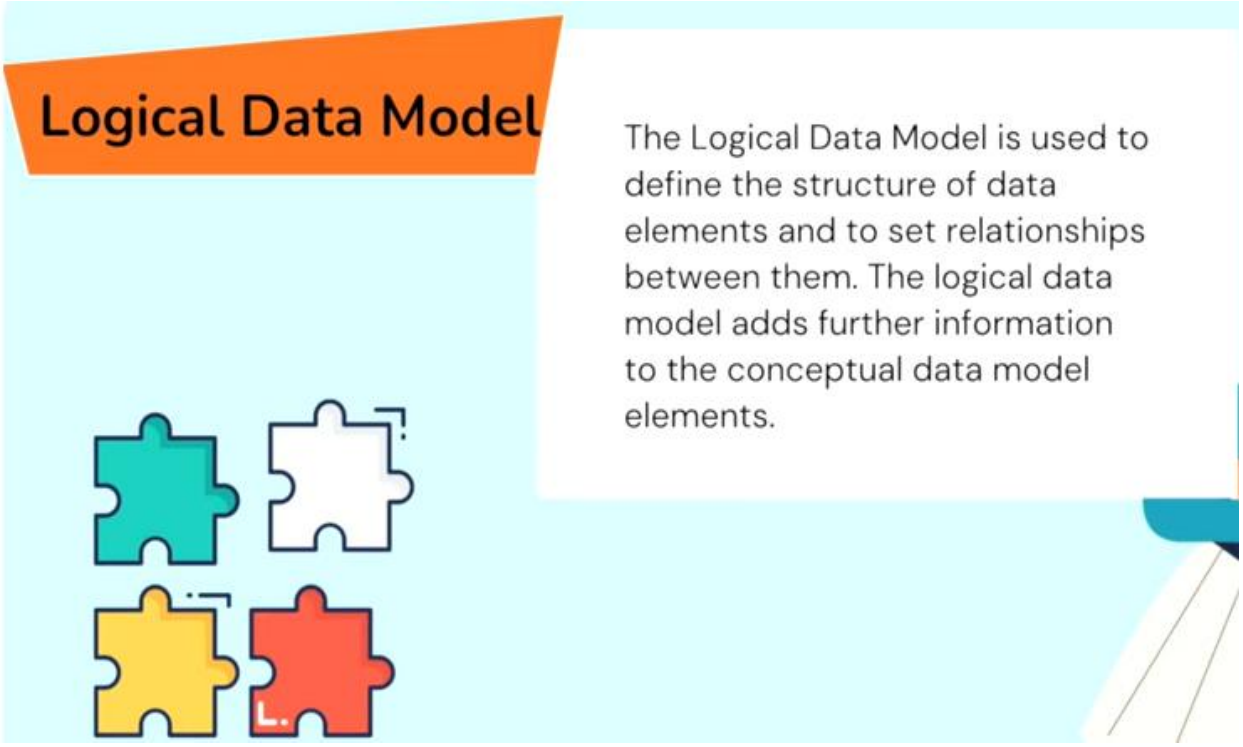
- **Entity:** A real-world thing
- **Attribute:** Characteristics or properties of an entity
- **Relationship:** Dependency or association between two entities



Example of Conceptual Model

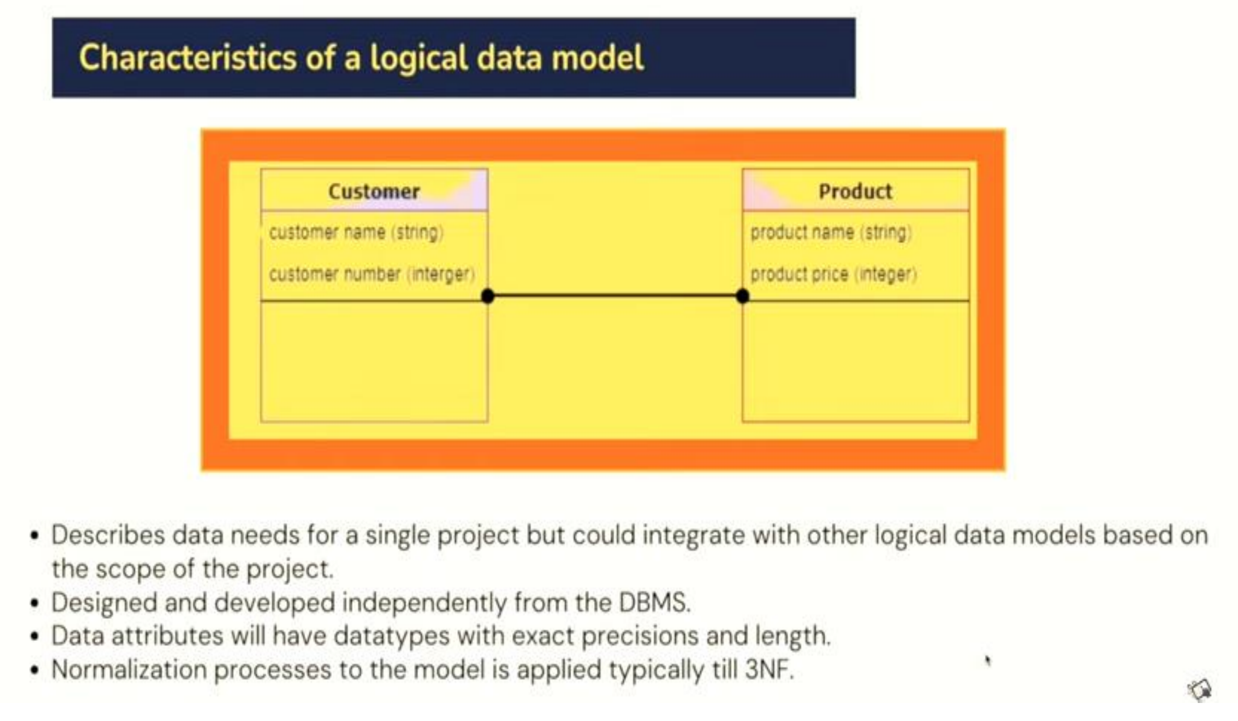
Characteristics of a conceptual data model

- Offers Organisation-wide coverage of the business concepts.
- This type of Data Models are designed and developed for a business audience.
- The conceptual model is developed independently of hardware specifications like data storage capacity, location or software specifications like DBMS vendor and technology. The focus is to represent data as a user will see it in the "real world."



Logical Data Model

The Logical Data Model is used to define the structure of data elements and to set relationships between them. The logical data model adds further information to the conceptual data model elements.

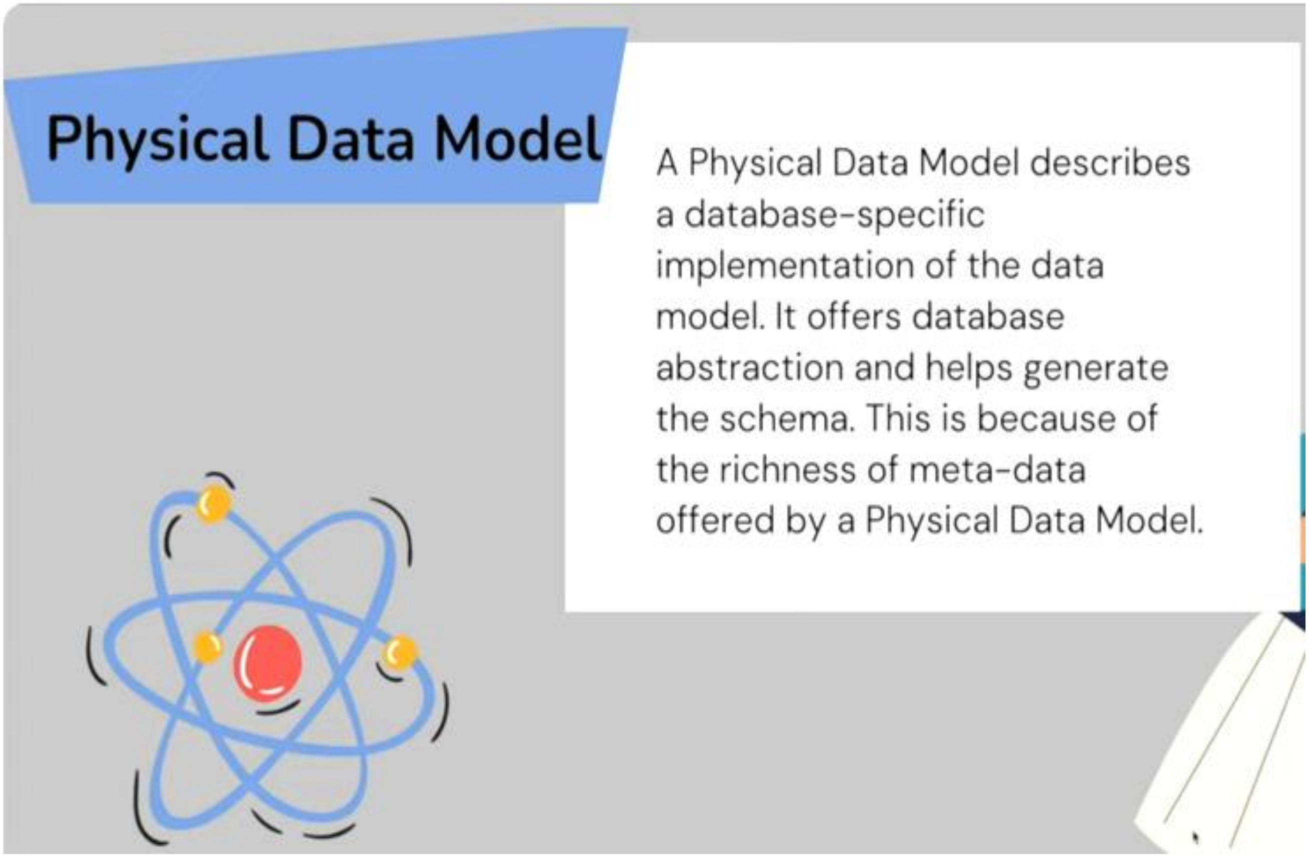


Characteristics of a logical data model

Customer
customer name (string)
customer number (interger)

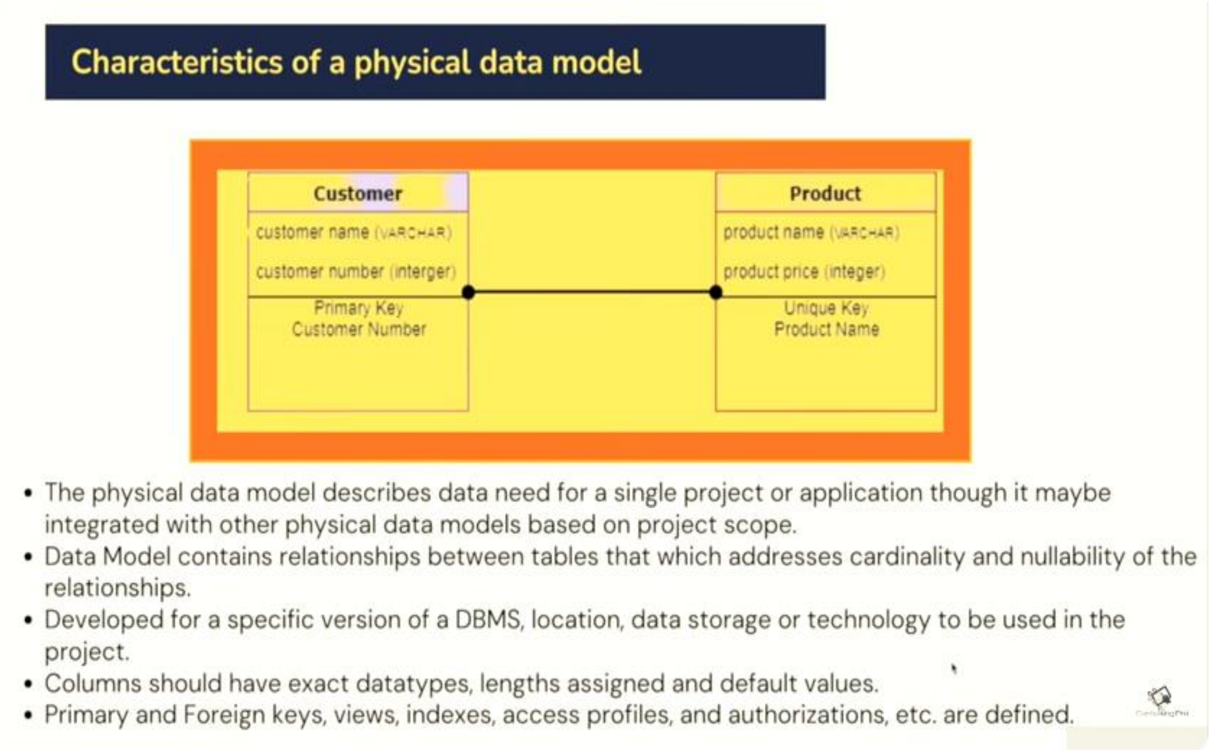
Product
product name (string)
product price (integer)

- Describes data needs for a single project but could integrate with other logical data models based on the scope of the project.
- Designed and developed independently from the DBMS.
- Data attributes will have datatypes with exact precisions and length.
- Normalization processes to the model is applied typically till 3NF.



Physical Data Model

A Physical Data Model describes a database-specific implementation of the data model. It offers database abstraction and helps generate the schema. This is because of the richness of meta-data offered by a Physical Data Model.



Characteristics of a physical data model

Customer

customer name (VARCHAR)

customer number (interger)

Primary Key
Customer Number

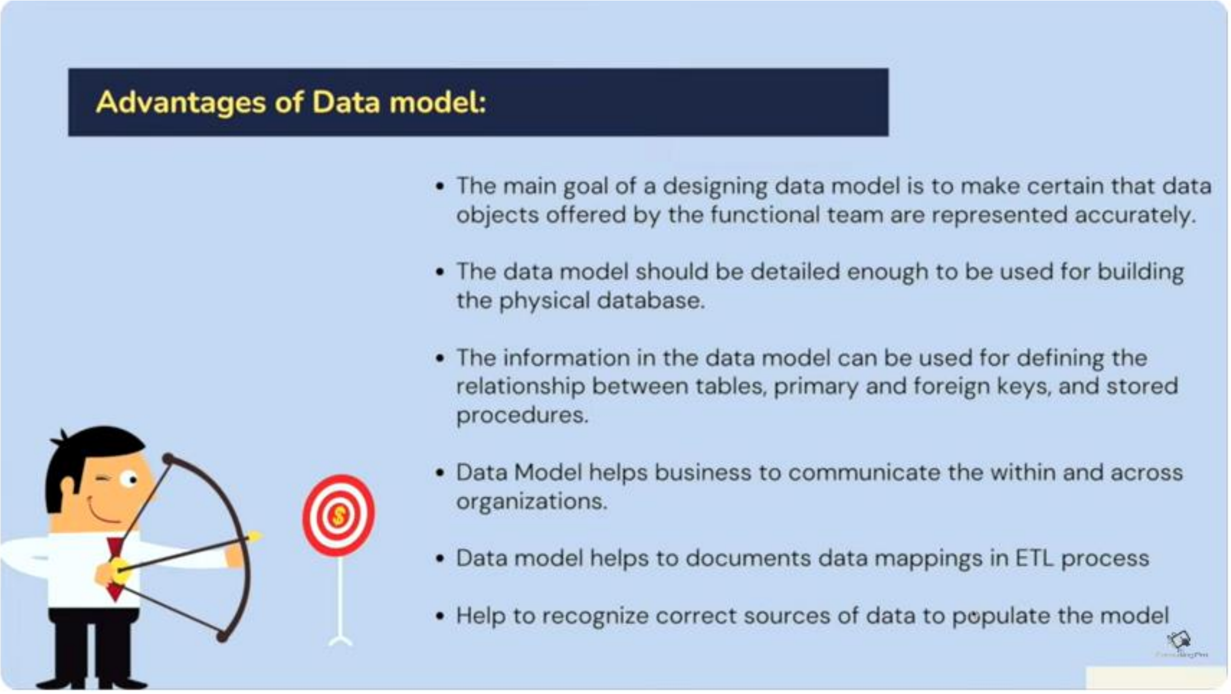
Product

product name (VARCHAR)

product price (integer)

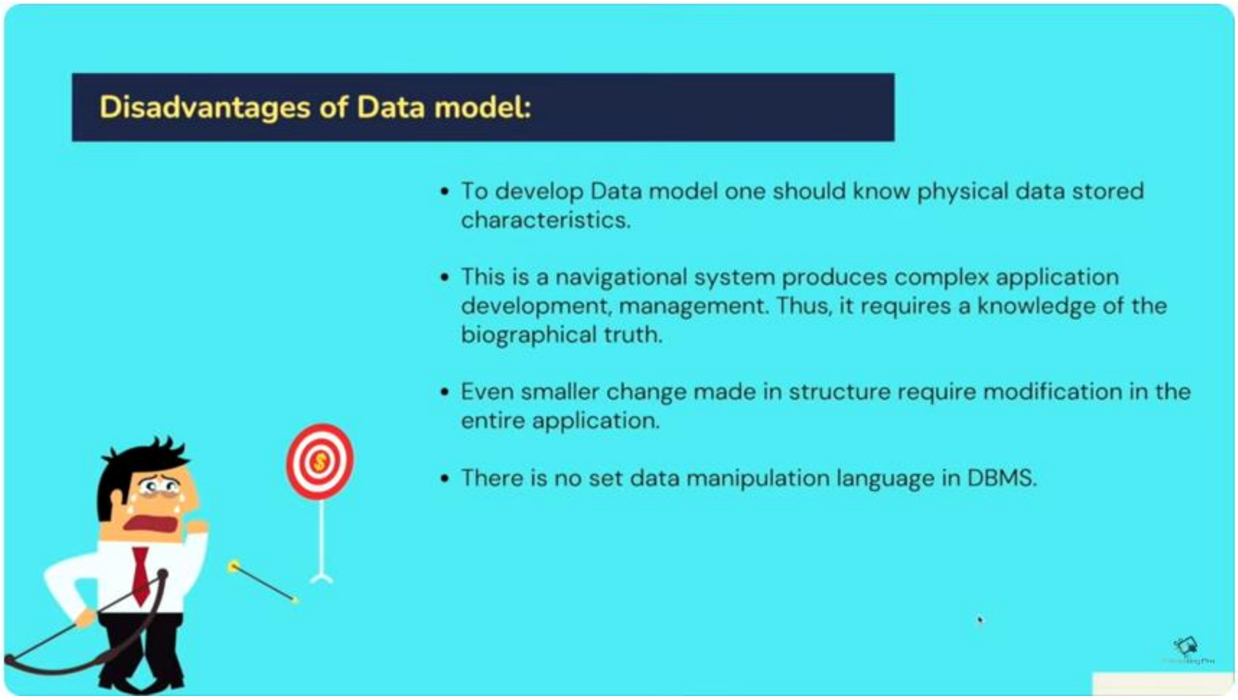
Unique Key
Product Name

- The physical data model describes data need for a single project or application though it maybe integrated with other physical data models based on project scope.
- Data Model contains relationships between tables that which addresses cardinality and nullability of the relationships.
- Developed for a specific version of a DBMS, location, data storage or technology to be used in the project.
- Columns should have exact datatypes, lengths assigned and default values.
- Primary and Foreign keys, views, indexes, access profiles, and authorizations, etc. are defined.



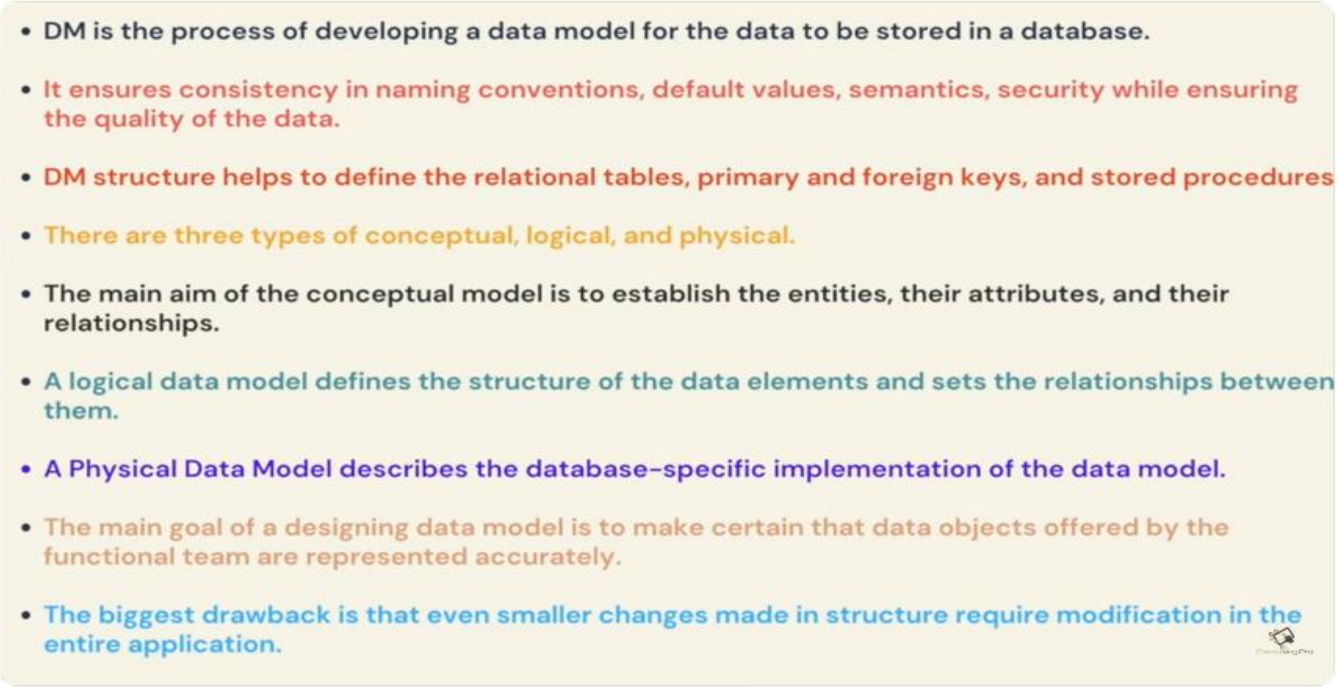
Advantages of Data model:

- The main goal of a designing data model is to make certain that data objects offered by the functional team are represented accurately.
- The data model should be detailed enough to be used for building the physical database.
- The information in the data model can be used for defining the relationship between tables, primary and foreign keys, and stored procedures.
- Data Model helps business to communicate the within and across organizations.
- Data model helps to documents data mappings in ETL process
- Help to recognize correct sources of data to populate the model

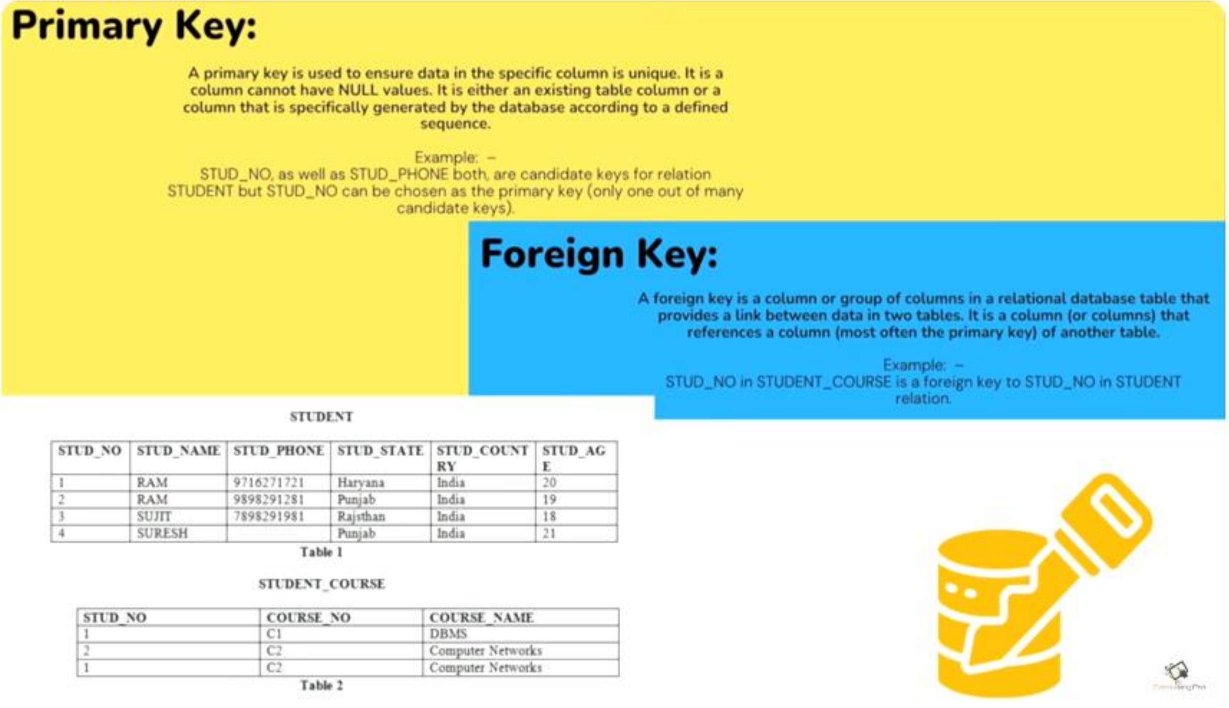


Disadvantages of Data model:

- To develop Data model one should know physical data stored characteristics.
- This is a navigational system produces complex application development, management. Thus, it requires a knowledge of the biographical truth.
- Even smaller change made in structure require modification in the entire application.
- There is no set data manipulation language in DBMS.



- DM is the process of developing a data model for the data to be stored in a database.
- It ensures consistency in naming conventions, default values, semantics, security while ensuring the quality of the data.
- DM structure helps to define the relational tables, primary and foreign keys, and stored procedures
- There are three types of conceptual, logical, and physical.
- The main aim of the conceptual model is to establish the entities, their attributes, and their relationships.
- A logical data model defines the structure of the data elements and sets the relationships between them.
- A Physical Data Model describes the database-specific implementation of the data model.
- The main goal of a designing data model is to make certain that data objects offered by the functional team are represented accurately.
- The biggest drawback is that even smaller changes made in structure require modification in the entire application.



Primary Key:

A primary key is used to ensure data in the specific column is unique. It is a column cannot have NULL values. It is either an existing table column or a column that is specifically generated by the database according to a defined sequence.

Example: –
STUD_NO, as well as STUD_PHONE both, are candidate keys for relation STUDENT but STUD_NO can be chosen as the primary key (only one out of many candidate keys).

Foreign Key:

A foreign key is a column or group of columns in a relational database table that provides a link between data in two tables. It is a column (or columns) that references a column (most often the primary key) of another table.

Example: –
STUD_NO in STUDENT_COURSE is a foreign key to STUD_NO in STUDENT relation.

STUDENT

STUD_NO	STUD_NAME	STUD_PHONE	STUD_STATE	STUD_COUNTRY	STUD_AGE
1	RAM	9716271721	Haryana	India	20
2	RAM	9898291281	Punjab	India	19
3	SUJIT	7898291981	Rajsthan	India	18
4	SURESH		Punjab	India	21

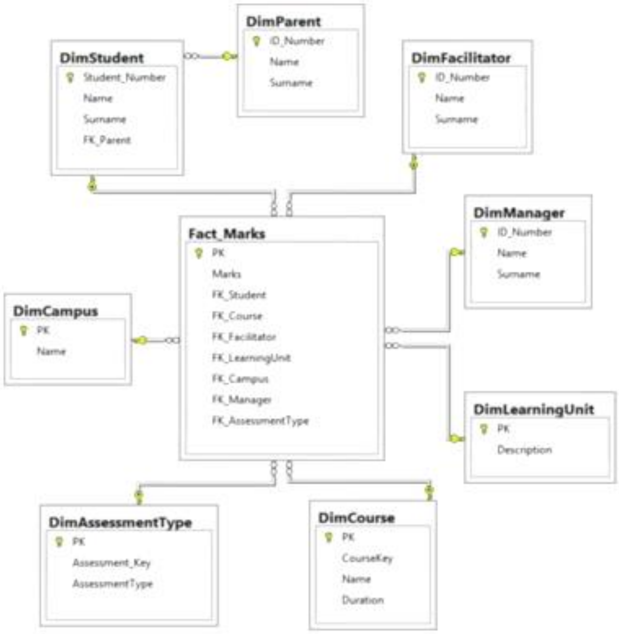
Table 1

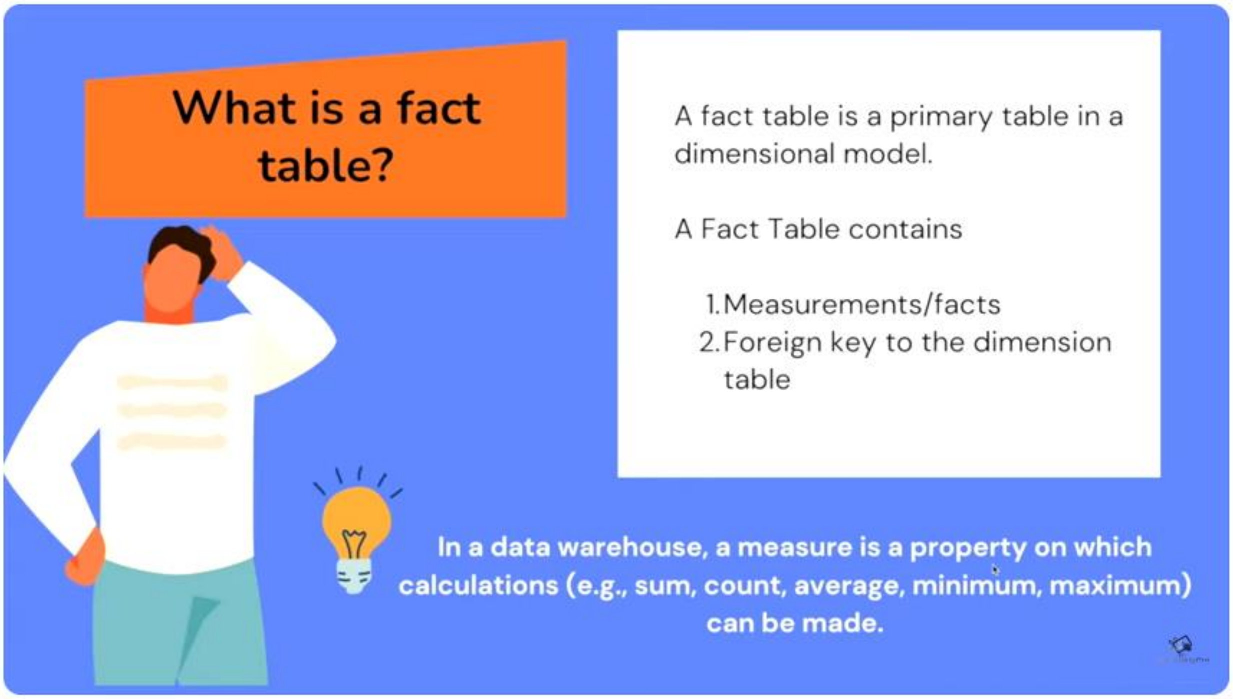
STUDENT_COURSE

STUD_NO	COURSE_NO	COURSE_NAME
1	C1	DBMS
2	C2	Computer Networks
1	C2	Computer Networks

Table 2

Fact Table and Dimension Table





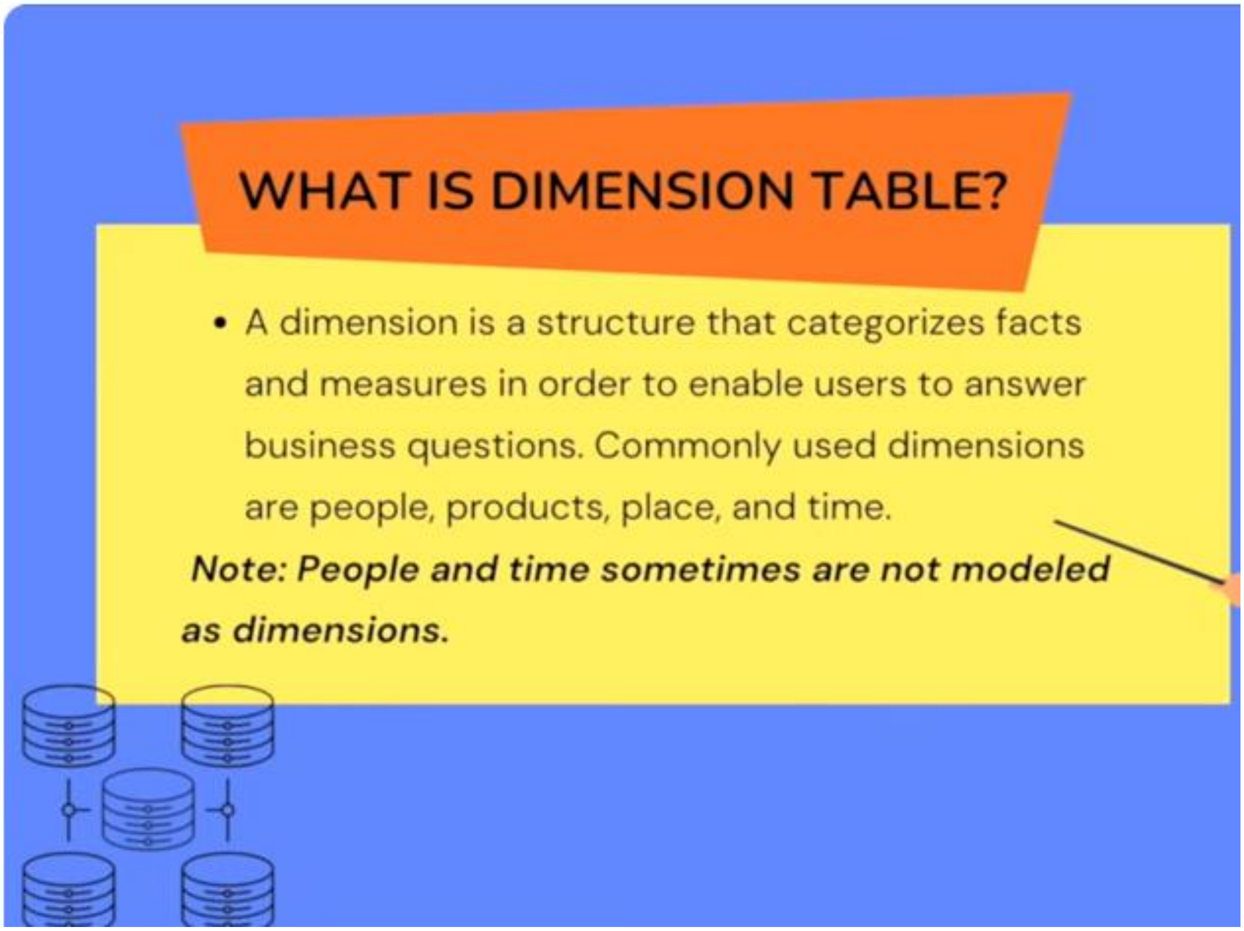
What is a fact table?

A fact table is a primary table in a dimensional model.

A Fact Table contains

1. Measurements/facts
2. Foreign key to the dimension table

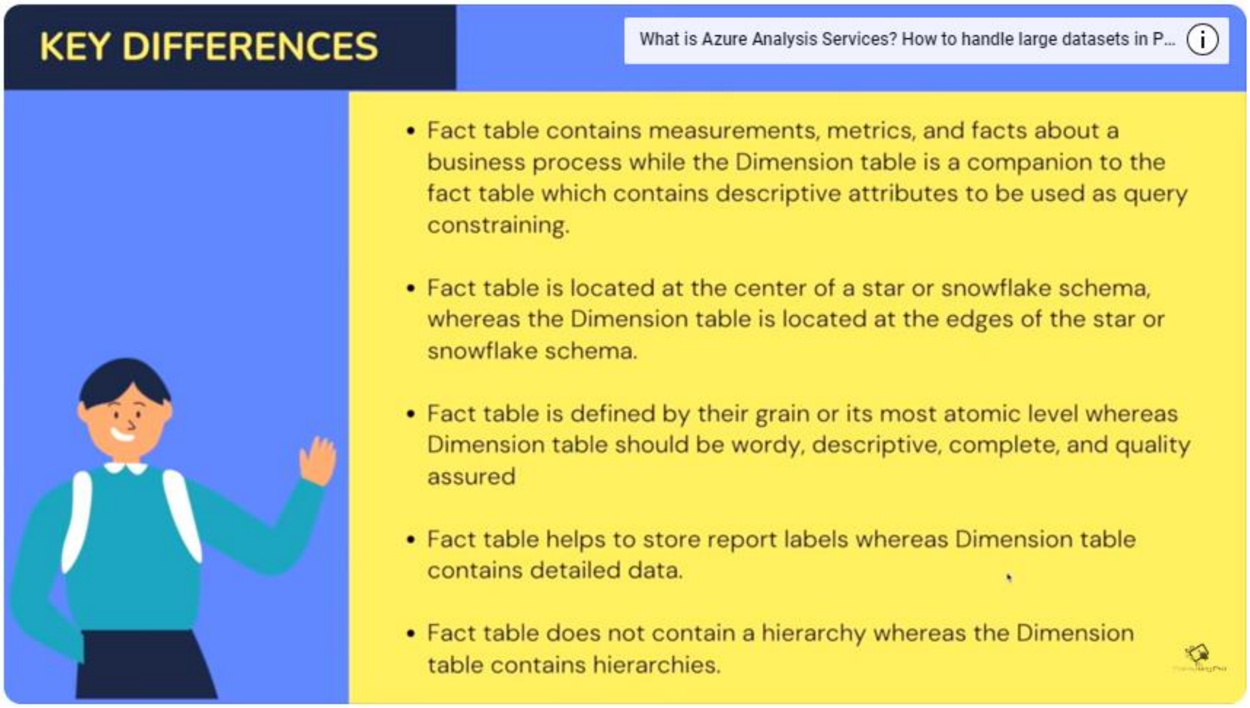
In a data warehouse, a measure is a property on which calculations (e.g., sum, count, average, minimum, maximum) can be made.



WHAT IS DIMENSION TABLE?

- A dimension is a structure that categorizes facts and measures in order to enable users to answer business questions. Commonly used dimensions are people, products, place, and time.

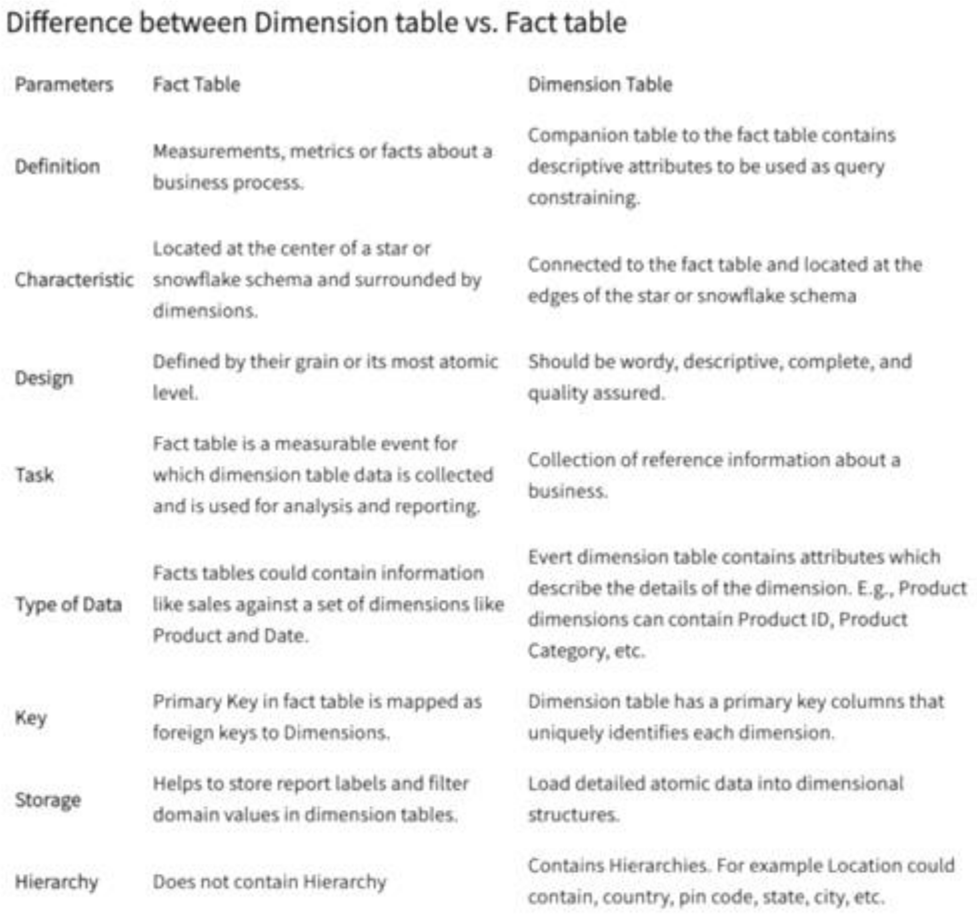
Note: People and time sometimes are not modeled as dimensions.



KEY DIFFERENCES

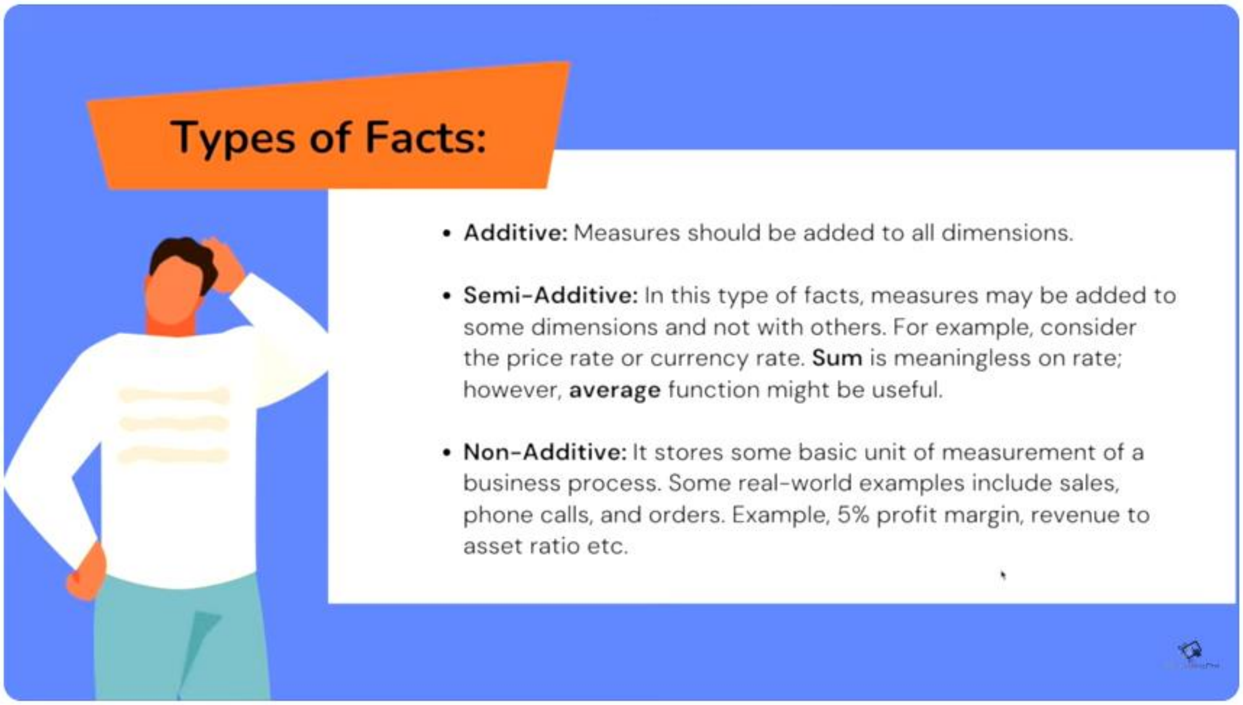
What is Azure Analysis Services? How to handle large datasets in P...

- Fact table contains measurements, metrics, and facts about a business process while the Dimension table is a companion to the fact table which contains descriptive attributes to be used as query constraining.
- Fact table is located at the center of a star or snowflake schema, whereas the Dimension table is located at the edges of the star or snowflake schema.
- Fact table is defined by their grain or its most atomic level whereas Dimension table should be wordy, descriptive, complete, and quality assured
- Fact table helps to store report labels whereas Dimension table contains detailed data.
- Fact table does not contain a hierarchy whereas the Dimension table contains hierarchies.



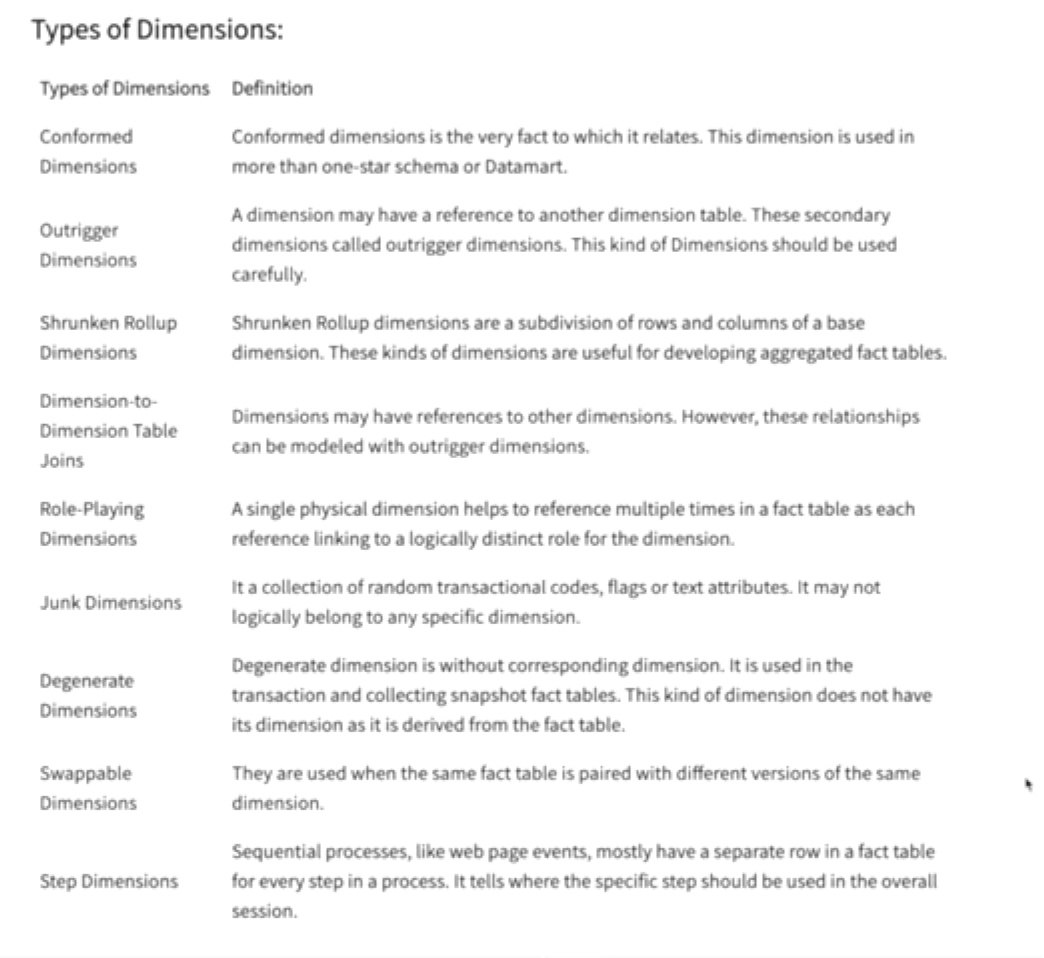
Difference between Dimension table vs. Fact table

Parameters	Fact Table	Dimension Table
Definition	Measurements, metrics or facts about a business process.	Companion table to the fact table contains descriptive attributes to be used as query constraining.
Characteristic	Located at the center of a star or snowflake schema and surrounded by dimensions.	Connected to the fact table and located at the edges of the star or snowflake schema
Design	Defined by their grain or its most atomic level.	Should be wordy, descriptive, complete, and quality assured.
Task	Fact table is a measurable event for which dimension table data is collected and is used for analysis and reporting.	Collection of reference information about a business.
Type of Data	Facts tables could contain information like sales against a set of dimensions like Product and Date.	Evert dimension table contains attributes which describe the details of the dimension. E.g., Product dimensions can contain Product ID, Product Category, etc.
Key	Primary Key in fact table is mapped as foreign keys to Dimensions.	Dimension table has a primary key columns that uniquely identifies each dimension.
Storage	Helps to store report labels and filter domain values in dimension tables.	Load detailed atomic data into dimensional structures.
Hierarchy	Does not contain Hierarchy	Contains Hierarchies. For example Location could contain, country, pin code, state, city, etc.



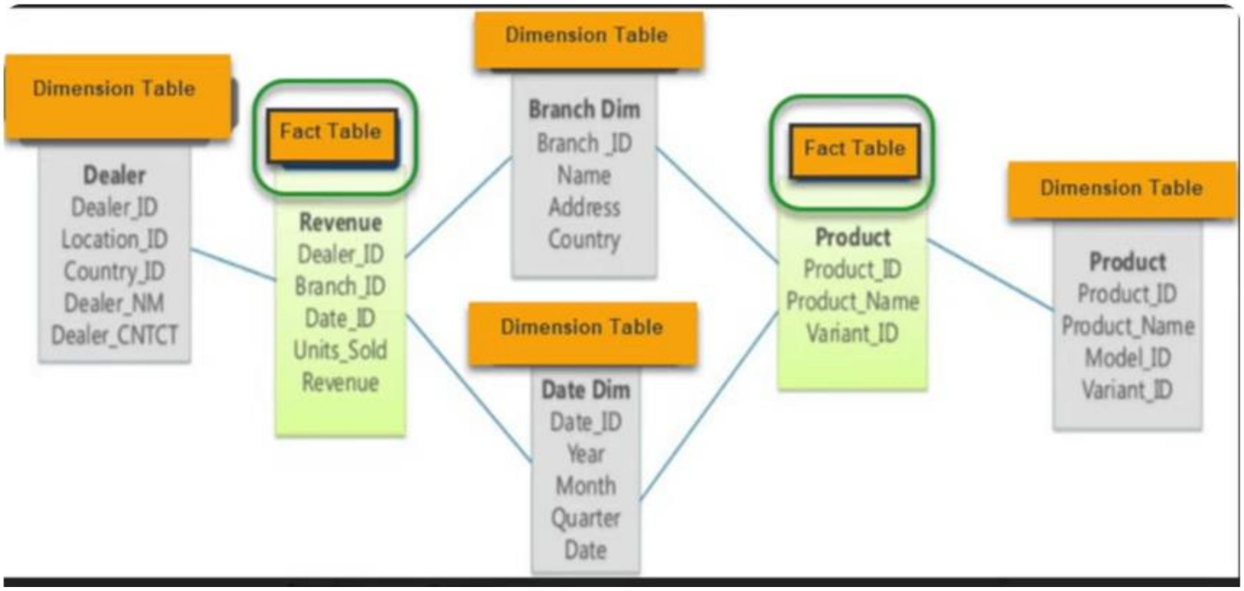
Types of Facts:

- **Additive:** Measures should be added to all dimensions.
- **Semi-Additive:** In this type of facts, measures may be added to some dimensions and not with others. For example, consider the price rate or currency rate. **Sum** is meaningless on rate; however, **average** function might be useful.
- **Non-Additive:** It stores some basic unit of measurement of a business process. Some real-world examples include sales, phone calls, and orders. Example, 5% profit margin, revenue to asset ratio etc.



Types of Dimensions:

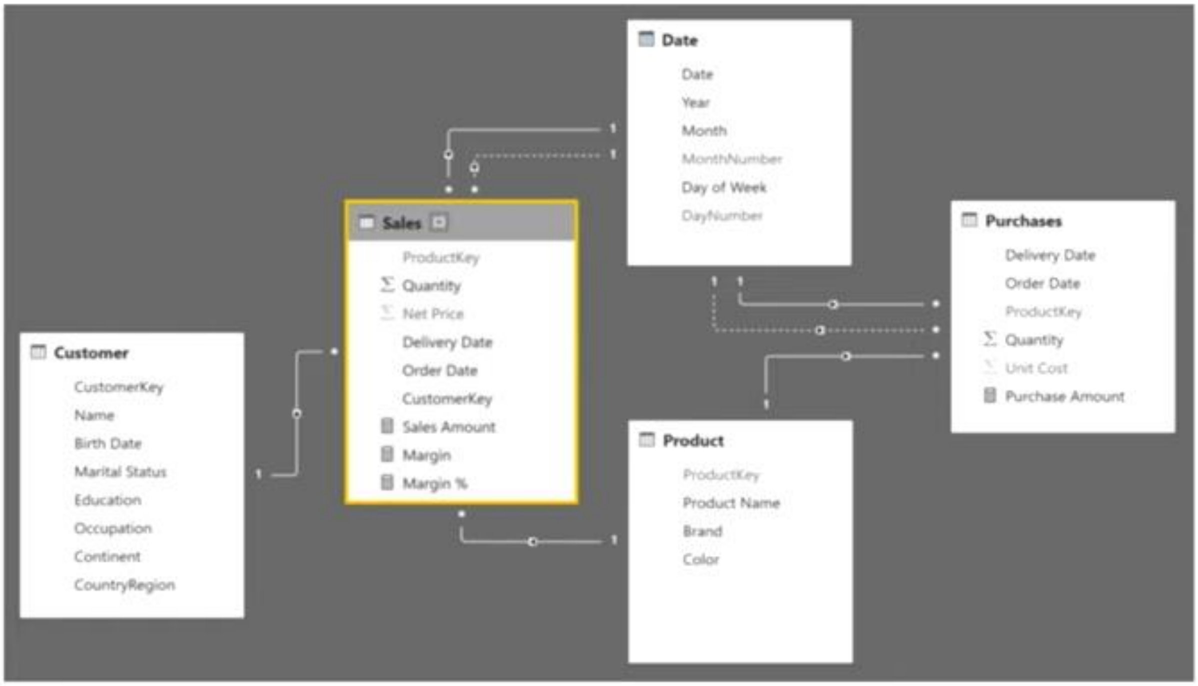
Types of Dimensions	Definition
Conformed Dimensions	Conformed dimensions is the very fact to which it relates. This dimension is used in more than one-star schema or Datamart.
Outrigger Dimensions	A dimension may have a reference to another dimension table. These secondary dimensions called outrigger dimensions. This kind of Dimensions should be used carefully.
Shrunken Rollup Dimensions	Shrunken Rollup dimensions are a subdivision of rows and columns of a base dimension. These kinds of dimensions are useful for developing aggregated fact tables.
Dimension-to-Dimension Table Joins	Dimensions may have references to other dimensions. However, these relationships can be modeled with outrigger dimensions.
Role-Playing Dimensions	A single physical dimension helps to reference multiple times in a fact table as each reference linking to a logically distinct role for the dimension.
Junk Dimensions	It a collection of random transactional codes, flags or text attributes. It may not logically belong to any specific dimension.
Degenerate Dimensions	Degenerate dimension is without corresponding dimension. It is used in the transaction and collecting snapshot fact tables. This kind of dimension does not have its dimension as it is derived from the fact table.
Swappable Dimensions	They are used when the same fact table is paired with different versions of the same dimension.
Step Dimensions	Sequential processes, like web page events, mostly have a separate row in a fact table for every step in a process. It tells where the specific step should be used in the overall session.



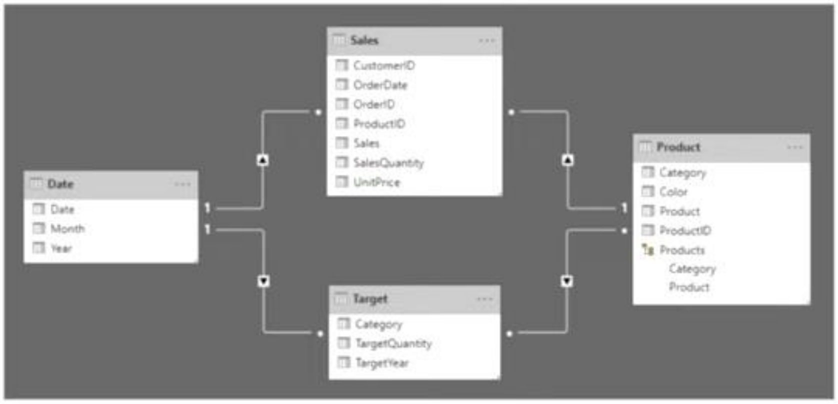
What is a relationship in Database?

A relationship, in the context of databases, is a situation that exists between two relational database tables when one table has a foreign key that references the primary key of the other table. Relationships allow relational databases to split and store data in different tables, while linking disparate data items.

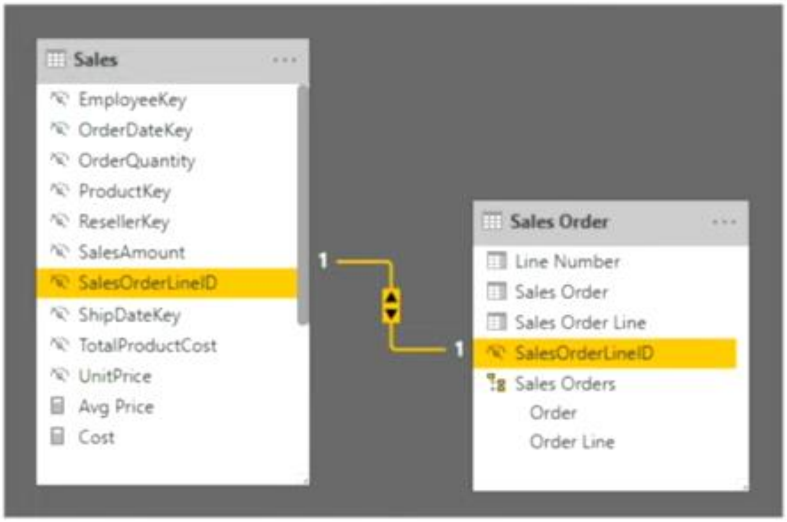
A relational database is a type of database that stores and provides access to data points that are related to one another. Relational databases consist of tables, columns, rows, keys, and relationship between tables.



One to Many (1 : M) Relationship



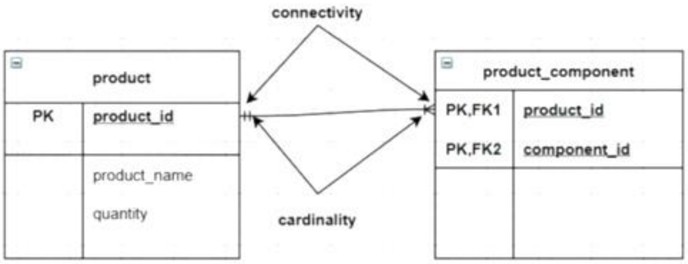
One to one (1:1) Relationship



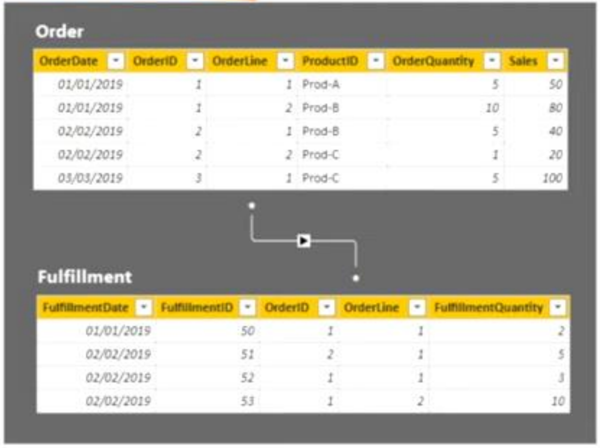
Cardinality and Connectivity

Cardinality describes the relationship between two data tables by expressing the minimum and maximum number of entity occurrences associated with one occurrence of a related entity. Either there is a relationship or not.

Connectivity is the relationship between two tables. It can be one to one or one to many.



Many to Many (M : N) Relationship



Order

OrderDate	OrderID	OrderLine	ProductID	OrderQuantity	Sales
01/01/2019	1	1	Prod-A	5	50
01/01/2019	1	2	Prod-B	10	80
02/02/2019	2	1	Prod-B	5	40
02/02/2019	2	2	Prod-C	1	20
03/03/2019	3	1	Prod-C	5	100

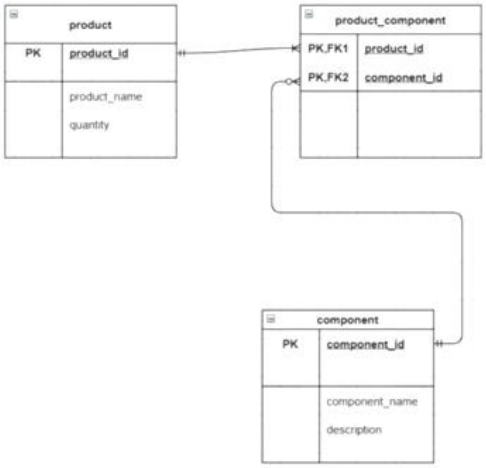
Fulfillment

FulfillmentDate	FulfillmentID	OrderID	OrderLine	FulfillmentQuantity
01/01/2019	50	1	1	2
02/02/2019	51	2	1	5
02/02/2019	52	1	1	3
02/02/2019	53	1	2	10

Many to many relationship causes duplications in database. Duplications cause false results from queries.

How to prevent duplication?

The solution is creating a totally new table. It's called as bridge table or join table.



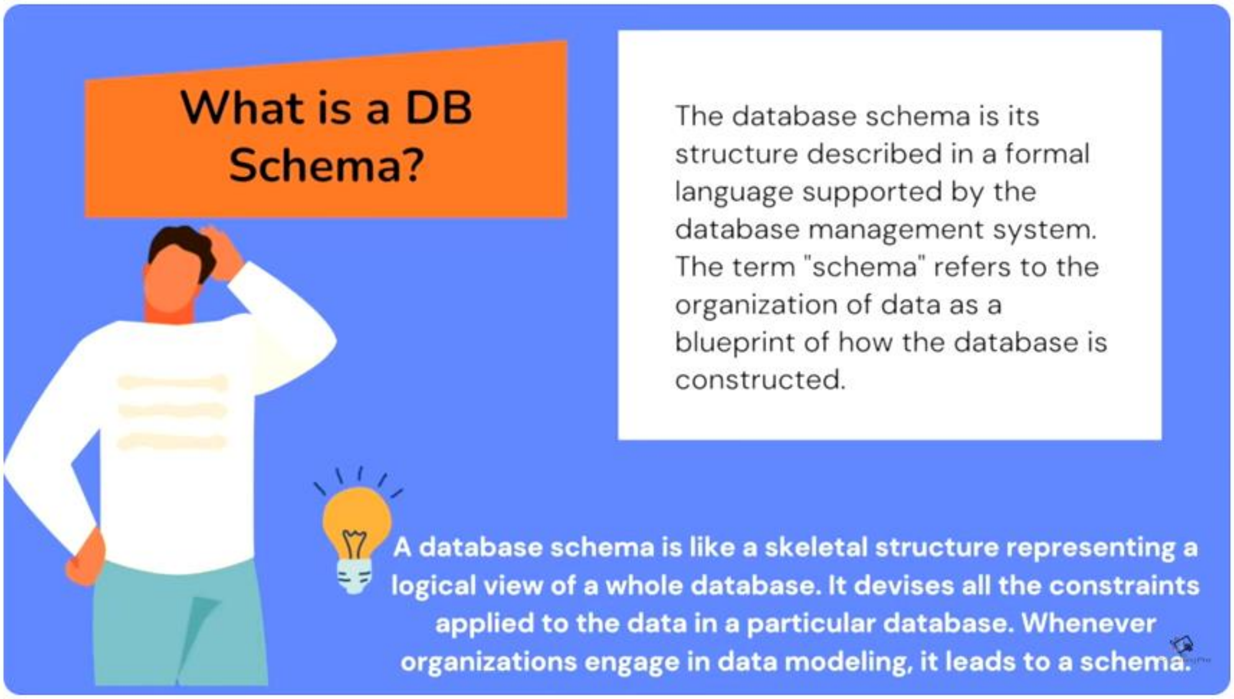
Bridge table consist of "product_id" , "component_id" from related tables' Primary Keys. "product_id" and "component_id" are both Primary and Foreign Keys the bridge table. Also, new columns can be added that are not available in the product table and component table.

Entity-Relationship (ER) Diagram helps to make and understand Entity Relationship (ER) Data Model.

Reference: medium.com

Summary

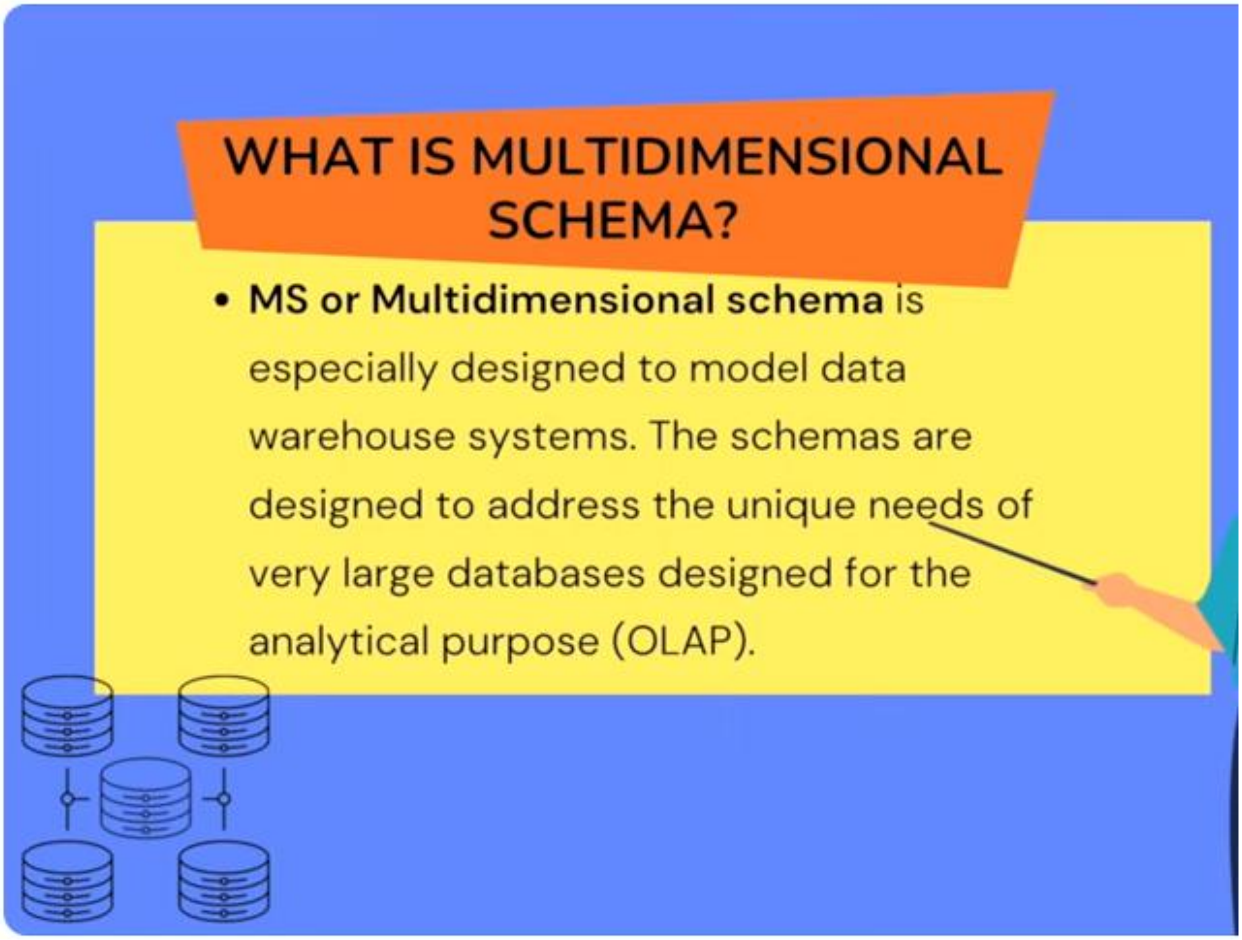
- Relational database has the ability to create meaningful information by joining tables.
- Joining tables help to understand the relationship between the data and tables. Besides, relational databases eliminate data redundancy.
- Relational database has three relationship between tables.
- Many to many relationship is harder to understand than other relationships.
- The bridge table makes the database and relationship easy to understand, and it prevents data redundancy.



What is a DB Schema?

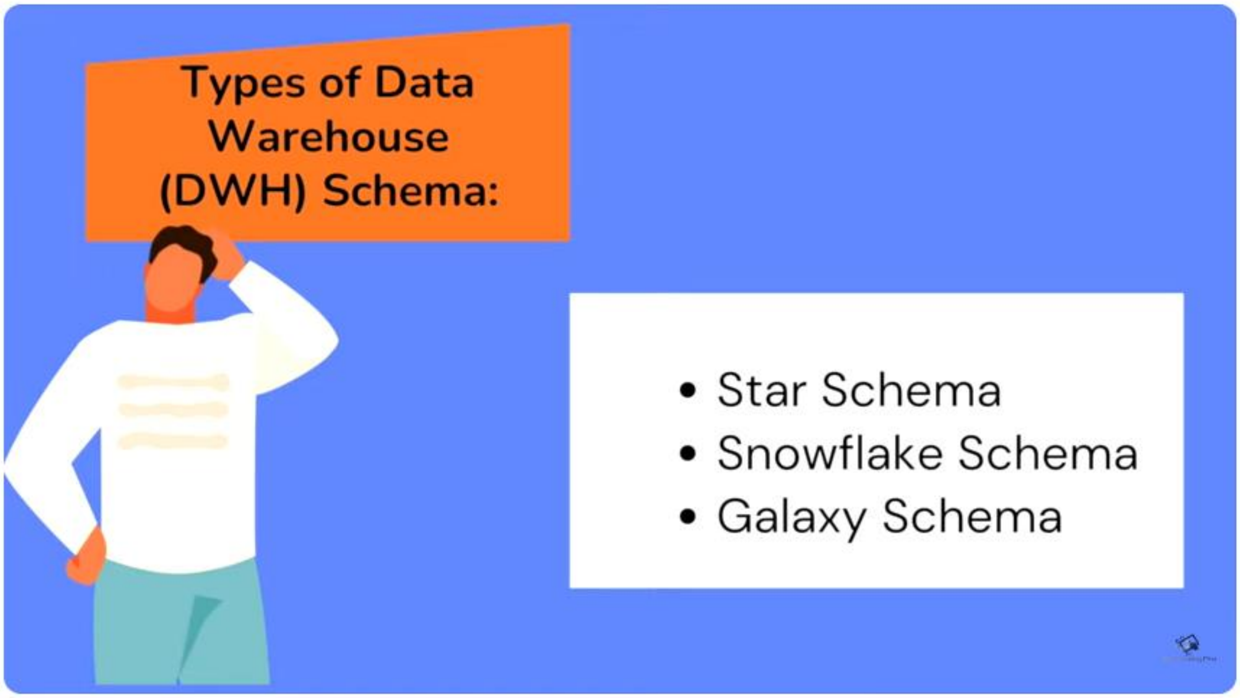
The database schema is its structure described in a formal language supported by the database management system. The term "schema" refers to the organization of data as a blueprint of how the database is constructed.

A database schema is like a skeletal structure representing a logical view of a whole database. It devises all the constraints applied to the data in a particular database. Whenever organizations engage in data modeling, it leads to a schema.



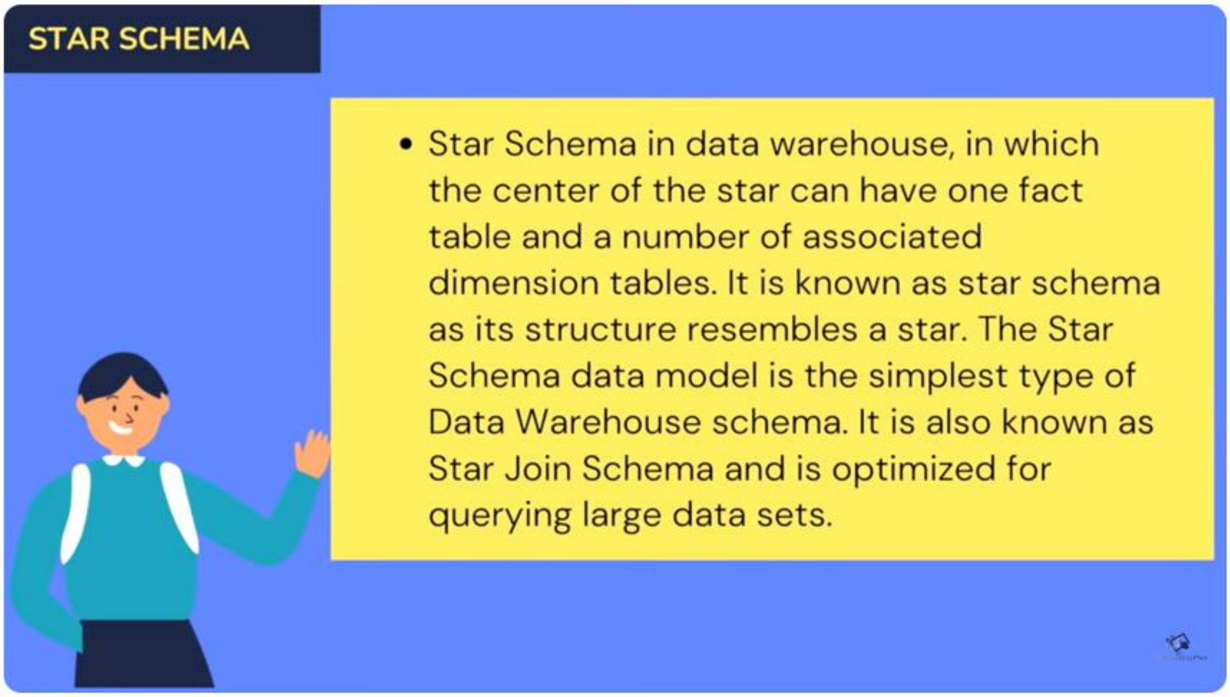
WHAT IS MULTIDIMENSIONAL SCHEMA?

- **MS or Multidimensional schema** is especially designed to model data warehouse systems. The schemas are designed to address the unique needs of very large databases designed for the analytical purpose (OLAP).



Types of Data Warehouse (DWH) Schema:

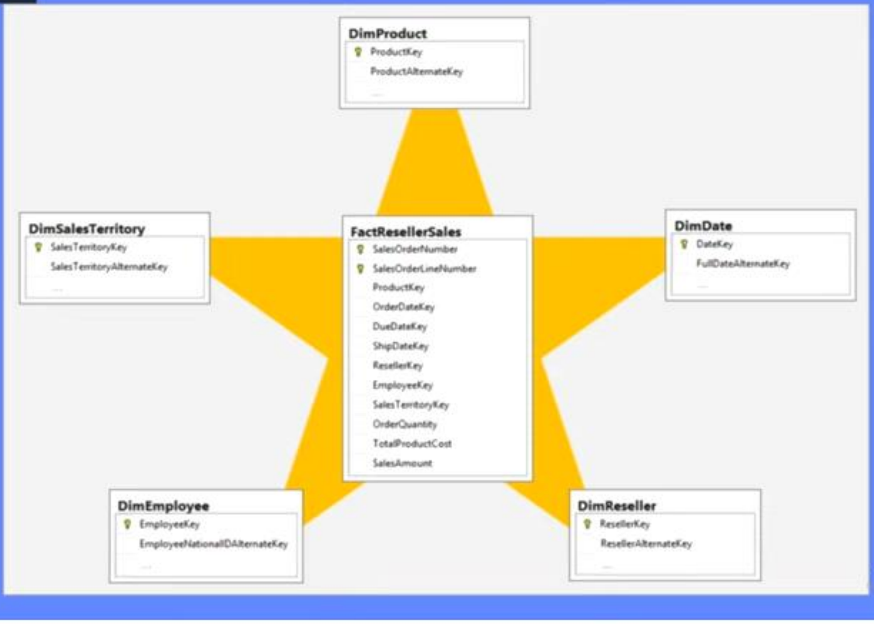
- Star Schema
- Snowflake Schema
- Galaxy Schema



STAR SCHEMA

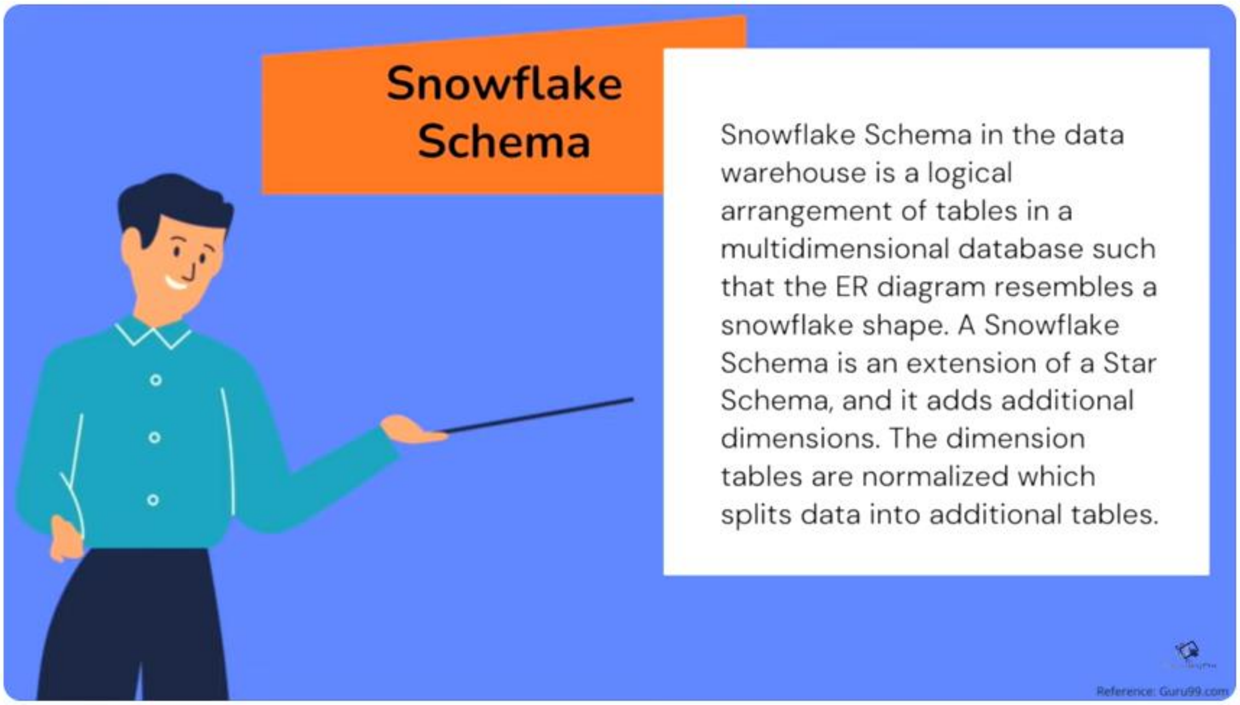
- Star Schema in data warehouse, in which the center of the star can have one fact table and a number of associated dimension tables. It is known as star schema as its structure resembles a star. The Star Schema data model is the simplest type of Data Warehouse schema. It is also known as Star Join Schema and is optimized for querying large data sets.

STAR SCHEMA



CHARACTERISTICS OF STAR SCHEMA:

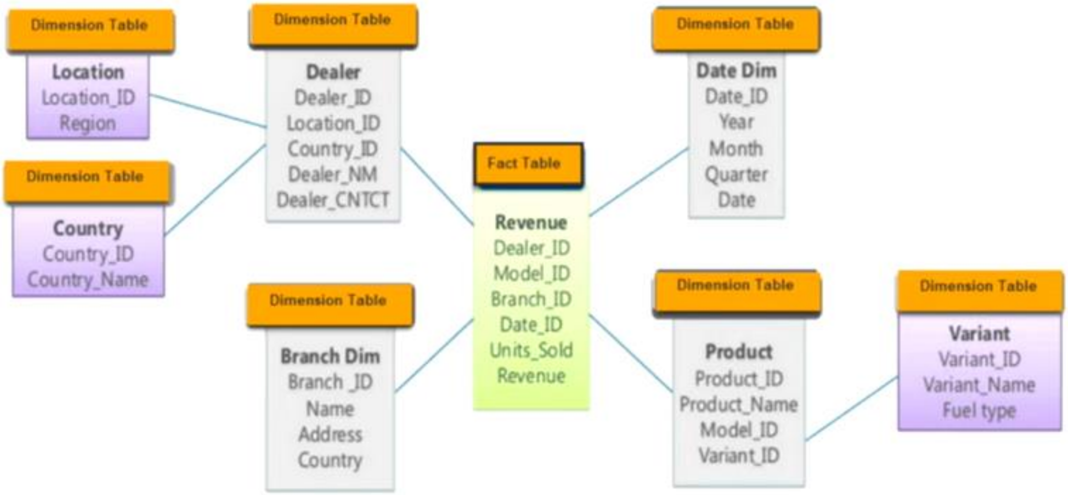
- Every dimension in a star schema is represented with the only one-dimension table.
- The dimension table should contain the set of attributes.
- The dimension table is joined to the fact table using a foreign key
- The dimension table are not joined to each other
- Fact table would contain key and measure
- The Star schema is easy to understand and provides optimal disk usage.
- The dimension tables are not normalized. For instance, in the previous image, Product_ID does not have Product lookup table as an OLTP design would have.
- The schema is widely supported by BI Tools



Snowflake Schema

Snowflake Schema in the data warehouse is a logical arrangement of tables in a multidimensional database such that the ER diagram resembles a snowflake shape. A Snowflake Schema is an extension of a Star Schema, and it adds additional dimensions. The dimension tables are normalized which splits data into additional tables.

Reference: Guru99.com



Dimension Table
Location
Location_ID
Region

Dimension Table
Country
Country_ID
Country_Name

Dimension Table
Dealer
Dealer_ID
Location_ID
Country_ID
Dealer_NM
Dealer_CNTCT

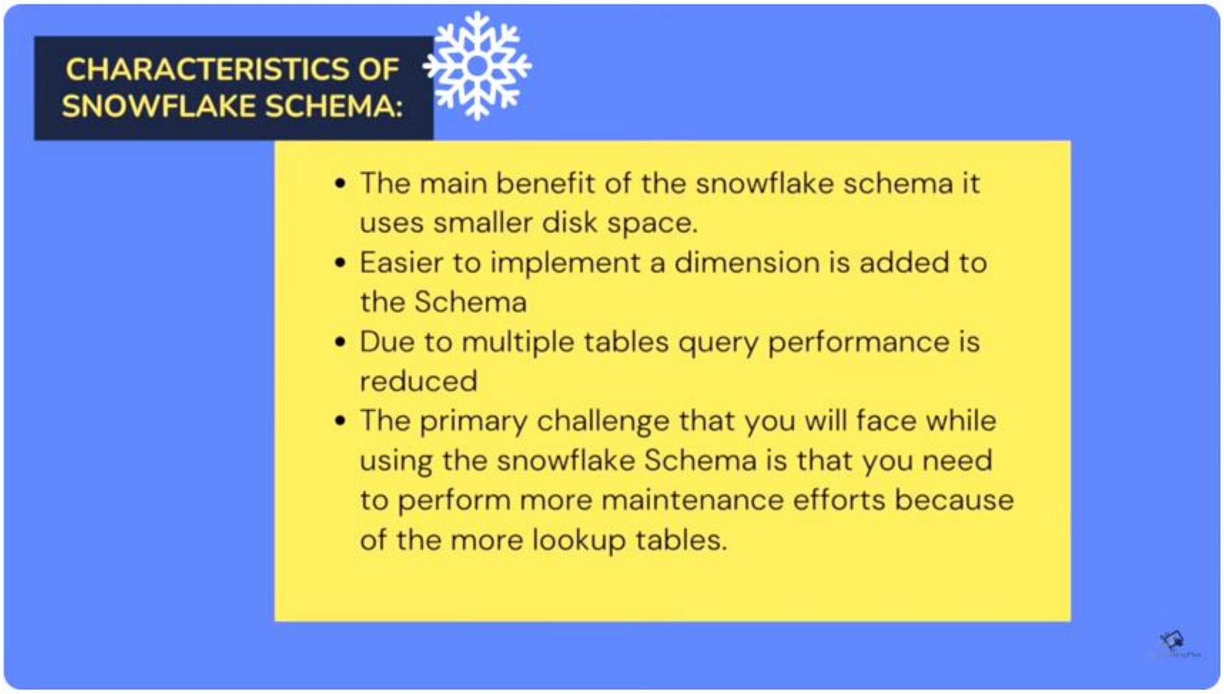
Dimension Table
Branch Dim
Branch_ID
Name
Address
Country

Fact Table
Revenue
Dealer_ID
Model_ID
Branch_ID
Date_ID
Units_Sold
Revenue

Dimension Table
Date Dim
Date_ID
Year
Month
Quarter
Date

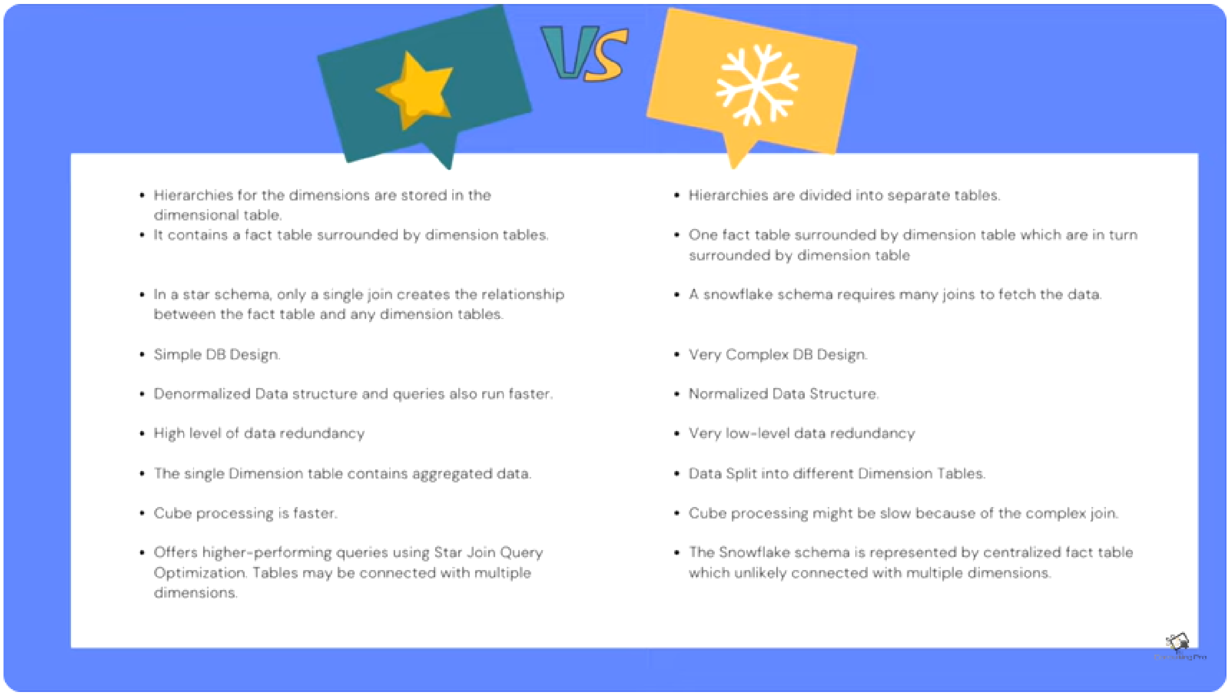
Dimension Table
Product
Product_ID
Product_Name
Model_ID
Variant_ID

Dimension Table
Variant
Variant_ID
Variant_Name
Fuel type

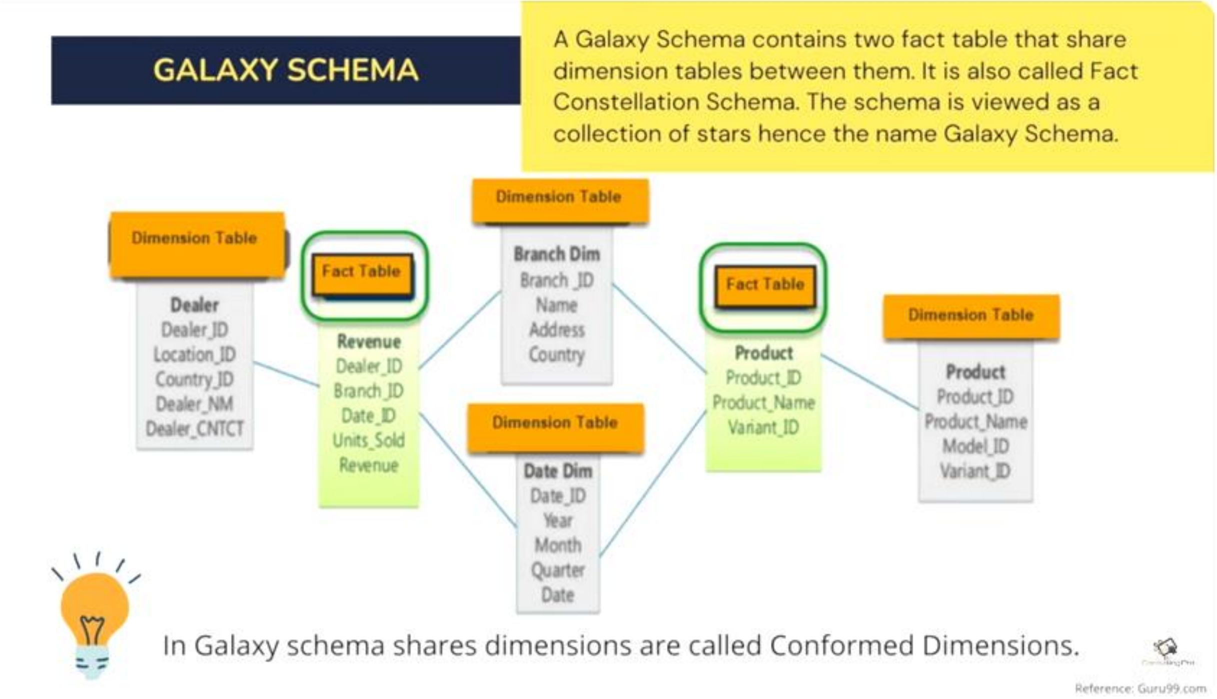


CHARACTERISTICS OF SNOWFLAKE SCHEMA:

- The main benefit of the snowflake schema it uses smaller disk space.
- Easier to implement a dimension is added to the Schema
- Due to multiple tables query performance is reduced
- The primary challenge that you will face while using the snowflake Schema is that you need to perform more maintenance efforts because of the more lookup tables.



★	❄
Hierarchies for the dimensions are stored in the dimensional table. It contains a fact table surrounded by dimension tables.	Hierarchies are divided into separate tables. One fact table surrounded by dimension table which are in turn surrounded by dimension table
In a star schema, only a single join creates the relationship between the fact table and any dimension tables.	A snowflake schema requires many joins to fetch the data.
Simple DB Design.	Very Complex DB Design.
Denormalized Data structure and queries also run faster.	Normalized Data Structure.
High level of data redundancy	Very low-level data redundancy
The single Dimension table contains aggregated data.	Data Split into different Dimension Tables.
Cube processing is faster.	Cube processing might be slow because of the complex join.
Offers higher-performing queries using Star Join Query Optimization. Tables may be connected with multiple dimensions.	The Snowflake schema is represented by centralized fact table which unlikely connected with multiple dimensions.



GALAXY SCHEMA

A Galaxy Schema contains two fact table that share dimension tables between them. It is also called Fact Constellation Schema. The schema is viewed as a collection of stars hence the name Galaxy Schema.

Dimension Table

Dealer
Dealer_ID
Location_ID
Country_ID
Dealer_NM
Dealer_CNTCT

Fact Table

Revenue
Dealer_ID
Branch_ID
Date_ID
Units_Sold
Revenue

Dimension Table

Branch Dim
Branch_ID
Name
Address
Country

Dimension Table

Date Dim
Date_ID
Year
Month
Quarter
Date

Fact Table

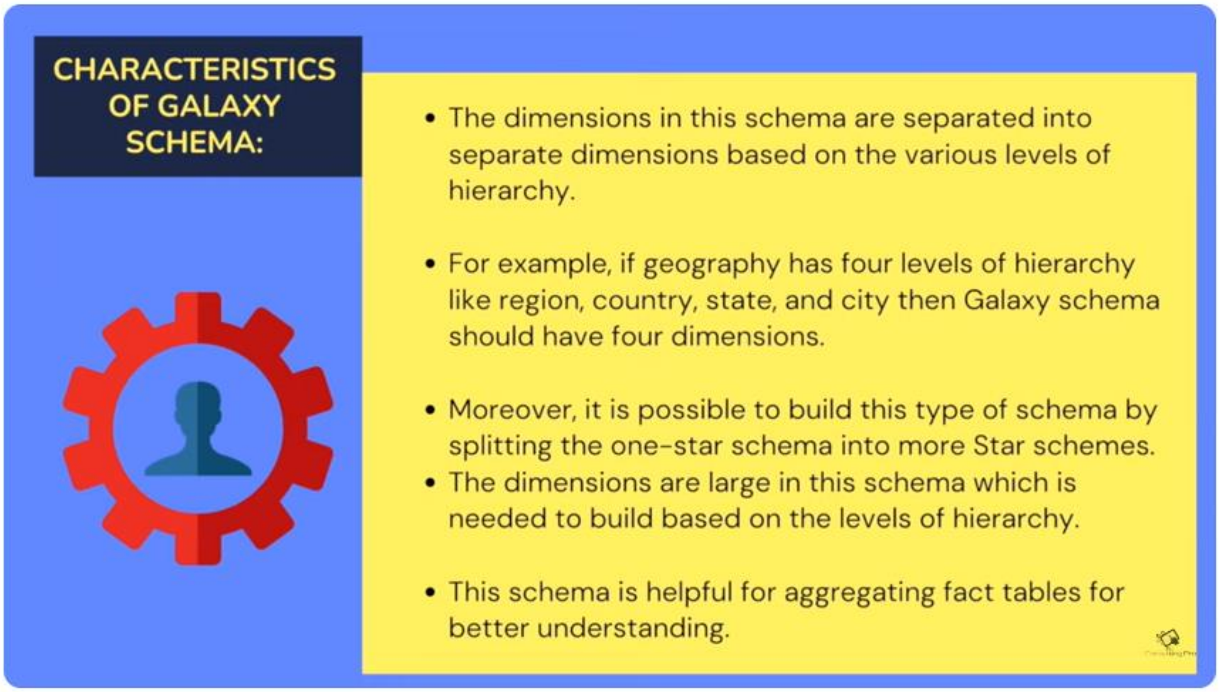
Product
Product_ID
Product_Name
Variant_ID

Dimension Table

Product
Product_ID
Product_Name
Model_ID
Variant_ID

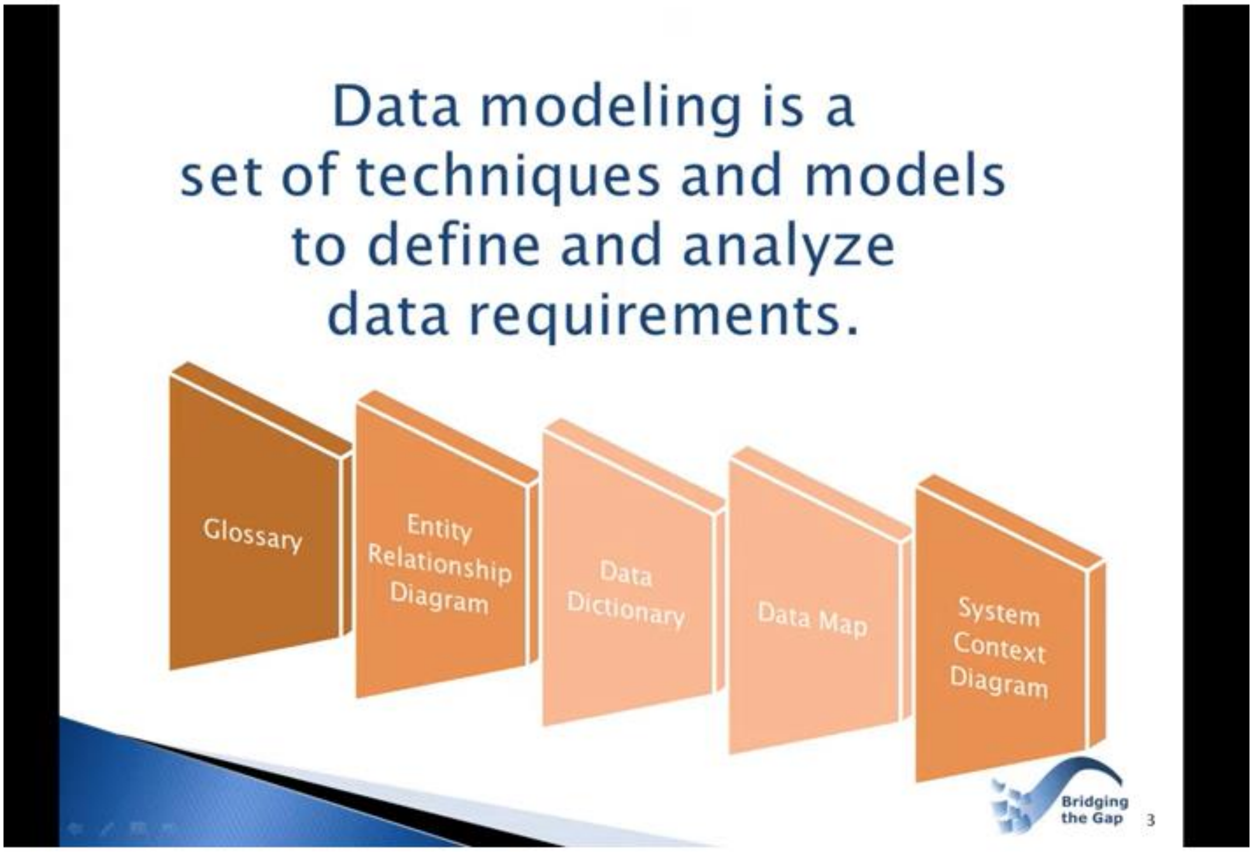
In Galaxy schema shares dimensions are called Conformed Dimensions.

Reference: Guru99.com



CHARACTERISTICS OF GALAXY SCHEMA:

- The dimensions in this schema are separated into separate dimensions based on the various levels of hierarchy.
- For example, if geography has four levels of hierarchy like region, country, state, and city then Galaxy schema should have four dimensions.
- Moreover, it is possible to build this type of schema by splitting the one-star schema into more Star schemes.
- The dimensions are large in this schema which is needed to build based on the levels of hierarchy.
- This schema is helpful for aggregating fact tables for better understanding.



Data modeling is a set of techniques and models to define and analyze data requirements.

Glossary

Entity Relationship Diagram

Data Dictionary

Data Map

System Context Diagram

Bridging the Gap