ANKIT GUPTA
Data Analyst | Prompt Engineer

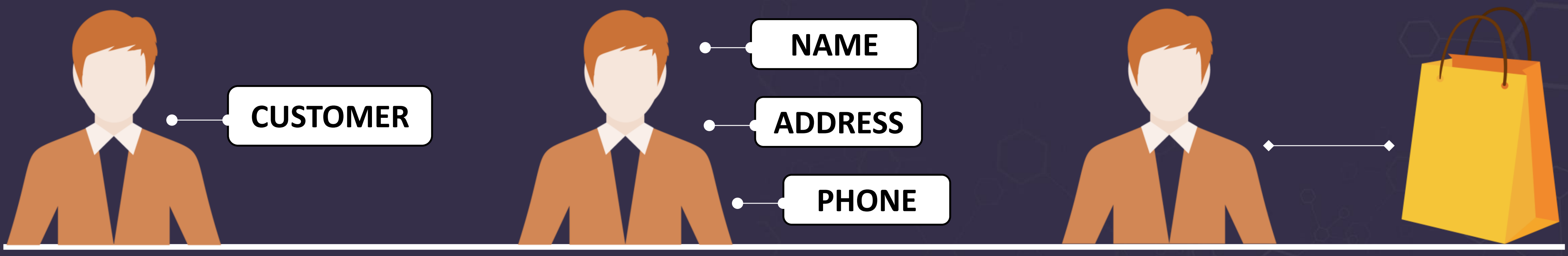
DATA MODELING

THE FOUNDATION OF DATA

DATA MODELLING

It is the process of creating a visual representation of an information system or a database. This representation is known as a data model, which describes the structure, relationships, and constraints of the data stored within the system. Data models serve as blueprints for designing and implementing databases, ensuring that data is organized, consistent, and easily accessible.

COMPONENTS OF DATA MODELING:



CUSTOMER

NAME

ADDRESS

PHONE

ENTITIES

ATTRIBUTES

RELATIONSHIPS

COMPONENTS OF DATA MODELING

A DATA MODEL CONSISTS OF 3 COMPONENTS:

1. Entities:

Entities represent the objects or things in a system about which we want to store data. They are typically nouns and can be tangible or intangible. For instance, in a business, an entity could be "Customer" or "Product."

Examples of Entities in a Business Context:

ENTITY	DESCRIPTION
Customer	Represents individuals or companies purchasing products or services.
Product	Represents items or services offered by the business.
Order	Represents transactions between the business and customers.
Employee	Represents individuals working for the business.
Supplier	Represents entities providing goods or services to the business.

COMPONENTS OF DATA MODELING

Power BI Implementation:

In Power BI, entities are represented by tables.

Examples:

- **Customer Table:** Contains records of each customer.
- **Product Table:** Contains records of each product

COMPONENTS OF DATA MODELING

2. Attributes:

Attributes are the specific pieces of information we want to store about an entity. They define the properties of an entity and are often thought of as the columns in a table. For a "Customer," attributes could include name, address, and phone number.

Examples of Attributes for Entities:

Customer’s Attributes

- Name
- Address
- Phone Number
- Email

Product’s Attributes

- Product ID
- Product Name
- Price
- Category

Order’s Attributes

- Order ID
- Order Date
- Quantity
- Total Amount

COMPONENTS OF DATA MODELING

Power BI Implementation:

In Power BI, Attributes are the columns within a table.

Examples:

- **Customer Table:**
  - Customer_ID (Primary Key)
  - Customer Name
  - Address
  - Phone Number
  - Email

COMPONENTS OF DATA MODELING

3. Relationships:

Relationships define how entities are connected or related to each other. They establish the link between different tables & enable us to perform comprehensive analysis & reporting. For example, a "Customer" can have a relationship with an "Order."

Types of Relationships:

1. One-to-one Relationship
2. One-to-many Relationship
3. Many-to-one Relationship
4. Many-to-many Relationship

TYPES OF RELATIONSHIPS

1. One-to-One Relationship:

A one-to-one relationship exists when each record in Table A relates to one and only one record in Table B, and vice versa.

Examples:

Customer & CustomerDetails Table: Each customer has one detailed profile.

TABLE (A) : CUSTOMER TABLE

CustomerID	CustomerName	Address	PhoneNumber
1	John Doe	123 Elm St.	123-456-7890
2	Jane Smith	456 Oak St.	987-654-3210
3	Alice Brown	789 Pine St.	555-555-5555

TYPES OF RELATIONSHIPS

Examples:

TABLE (B) : CUSTOMER DETAILS TABLE			
CustomerID	**DateOfBirth**	**MembershipStatus**	**Preferences**
1	1990-01-01	Gold	Vegetarian
2	1985-05-15	Silver	Vegan
3	1978-07-22	Platinum	Gluten-Free

Power BI Implementation:

Relationship As One -to- One

- Customer Table: CustomerID (Primary Key)
- CustomerDetails Table: CustomerID (Foreign Key, Unique)

Customer.CustomerID 1 ——— 1 CustomerDetails.CustomerID

TYPES OF RELATIONSHIPS

2. One-to-Many Relationship:

A one-to-many relationship exists when each record in Table A can relate to zero, one, or many records in Table B, but each record in Table B relates to one and only one record in Table A.

Examples:

Customer & **Order:** One customer can place multiple orders.

TABLE (A) : CUSTOMER TABLE

CustomerID	CustomerName	Address	PhoneNumber	Email
1	John Doe	123 Elm St.	123-456-7890	john@example.com
2	Jane Smith	456 Oak St.	987-654-3210	jane@example.com
3	Alice Brown	789 Pine St.	555-555-5555	alice@example.com

TYPES OF RELATIONSHIPS

Examples:

TABLE (B) : ORDER TABLE			
OrderID	**OrderDate**	**CustomerID**	**TotalAmount**
101	2024-01-01	1	100.00
102	2024-02-01	1	150.00
103	2024-01-15	2	200.00
104	2024-03-01	3	250.00

Power BI Implementation:

Relationship As One -to- Many

- Customer Table: CustomerID (Primary Key)
- Order Table: CustomerID (Foreign Key)

Customer.CustomerID ●1 ———— *● Order.CustomerID

TYPES OF RELATIONSHIPS

3. Many-to-One Relationship:

A many-to-one relationship is essentially the inverse of a one-to-many relationship. Each record in Table A can relate to one and only one record in Table B, but each record in Table B can relate to zero, one, or many records in Table A.

Examples:

Order & **Customer:** Each order is placed by one customer, but a customer can place multiple orders.

TABLE (A) : CUSTOMER TABLE				
CustomerID	**CustomerName**	**Address**	**PhoneNumber**	**Email**
1	John Doe	123 Elm St.	123-456-7890	john@example.com
2	Jane Smith	456 Oak St.	987-654-3210	jane@example.com
3	Alice Brown	789 Pine St.	555-555-5555	alice@example.com

TYPES OF RELATIONSHIPS

Examples:

TABLE (B) : ORDER TABLE

OrderID	OrderDate	CustomerID	TotalAmount
101	2024-01-01	1	100.00
102	2024-02-01	1	150.00
103	2024-01-15	2	200.00
104	2024-03-01	3	250.00

Power BI Implementation:

Relationship As Many -to- One

- Order Table: CustomerID (Foreign Key)
- Customer Table: CustomerID (Primary Key)

Order.CustomerID * —— 1 Customer.CustomerID

TYPES OF RELATIONSHIPS

4. Many-to-Many Relationship:

A many-to-many relationship exists when each record in Table A can relate to zero, one, or many records in Table B, and vice versa. This is typically managed through an intermediary or junction table.

Examples:

Order & **Product:** Each order can contain multiple products, and each product can appear in multiple orders.

TABLE (A) : ORDER TABLE

OrderID	OrderDate	CustomerID	TotalAmount
101	2024-01-01	1	100.00
102	2024-02-01	1	150.00
103	2024-01-15	2	200.00
104	2024-03-01	3	250.00

TYPES OF RELATIONSHIPS

Examples:

TABLE (B) : PRODUCT TABLE			
ProductID	**ProductName**	**Price**	**Category**
201	Widget A	10.00	Gadgets
202	Widget B	20.00	Gadgets
203	Widget C	30.00	Widgets

TABLE (C) : ORDER DETAILS TABLE			
OrderID	**ProductID**	**Quantity**	**Total**
101	201	5	50.00
101	202	2	40.00
102	203	3	90.00
103	201	10	100.00
104	202	5	100.00

Power BI Implementation:

Relationship As Many -to- Many

- Order Table: OrderID (Primary Key)
- Product Table: ProductID (Primary Key)
- OrderDetails Table: OrderID (Foreign Key), ProductID (Foreign Key)

Order.OrderID ●1———*● OrderDetails.OrderID

Product.ProductID ●1———*● OrderDetails.ProductID

TYPES OF DATA MODELLING

- Conceptual
- Logical
- Physical

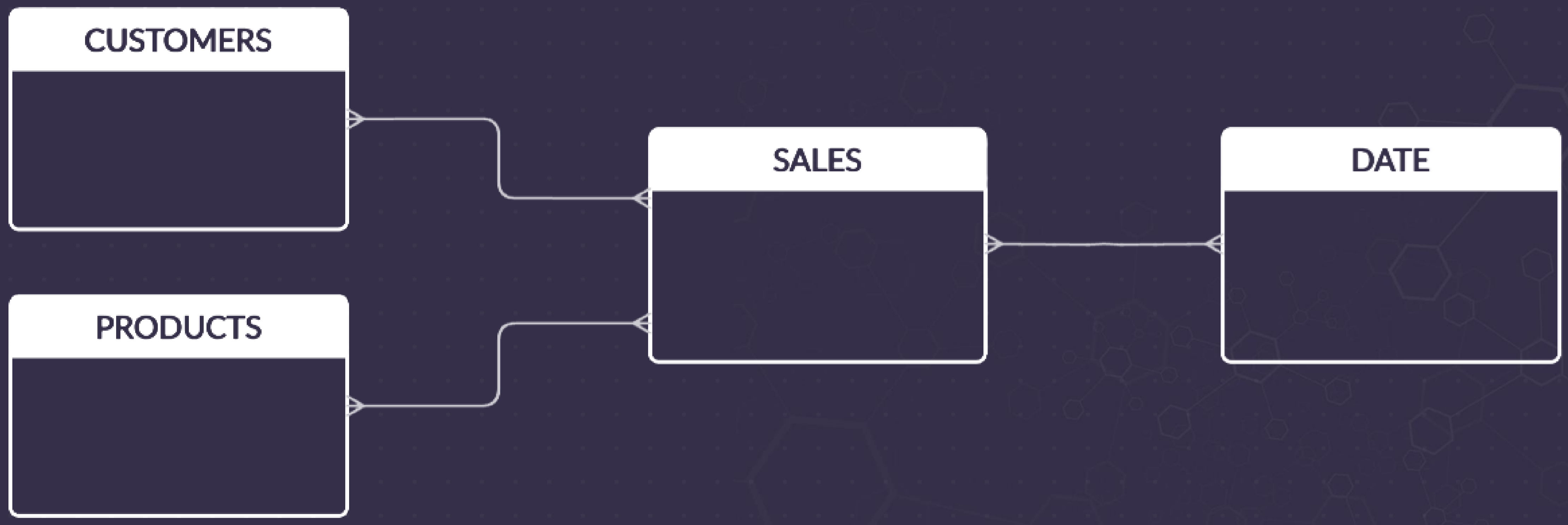
TYPES OF DATA MODELLING

1. CONCEPTUAL DATA MODEL:

High-level view, abstract design that defines the overall structure and relationships of data entities without going into technical details.

Examples:

Creating a conceptual model for a retail business with entities like Customers, Products, Date, Orders, and Sales.



CUSTOMERS

SALES

DATE

PRODUCTS

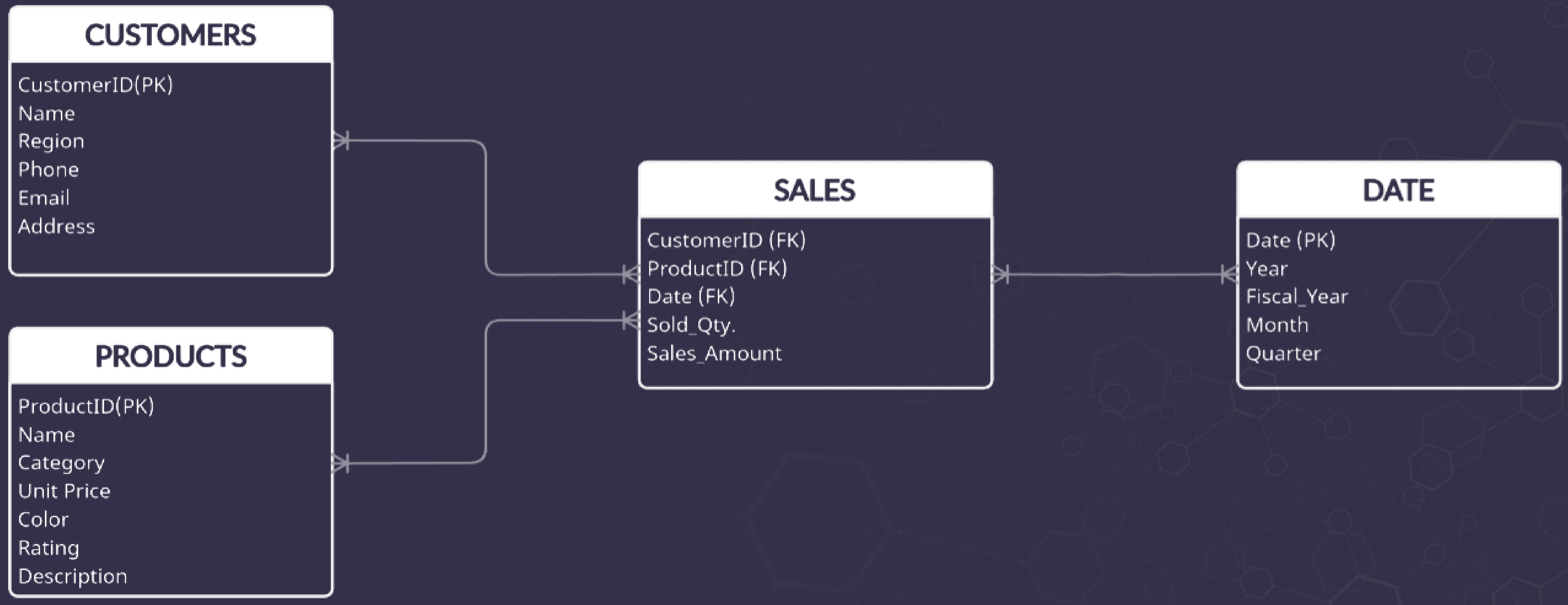
TYPES OF DATA MODELLING

2. LOGICAL DATA MODEL:

More detailed than conceptual models, defining entities, attributes, and relationships but still independent of physical implementation.

Examples:

Adding details to the retail model such as CustomerID, ProductID, OrderDate, and defining relationships between Customers and Sales.



CUSTOMERS

CustomerID(PK)
Name
Region
Phone
Email
Address

PRODUCTS

ProductID(PK)
Name
Category
Unit Price
Color
Rating
Description

SALES

CustomerID (FK)
ProductID (FK)
Date (FK)
Sold_Qty.
Sales_Amount

DATE

Date (PK)
Year
Fiscal_Year
Month
Quarter

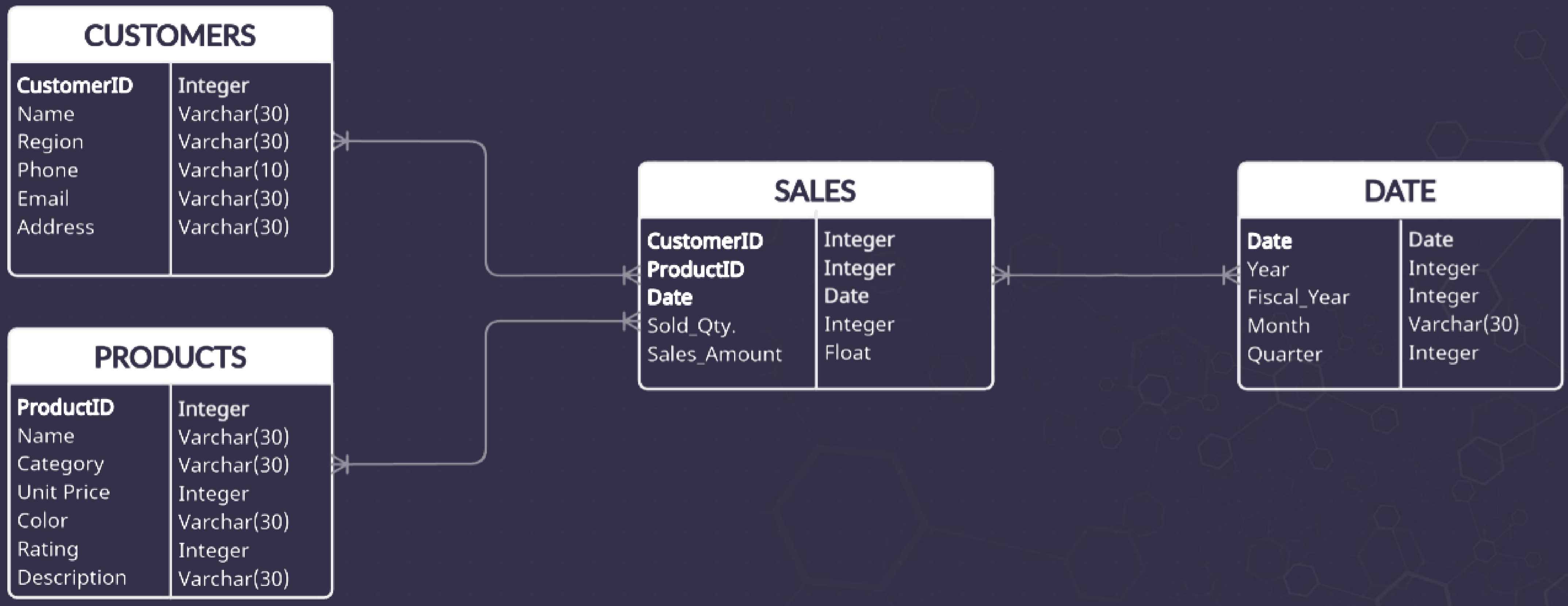
TYPES OF DATA MODELLING

3. PHYSICAL DATA MODEL:

It specifies how the data will be stored, the implementation of the logical model into a specific database system, including table structures, columns, data types, indexes, and constraints.

Examples:

Creating tables based on the logical model, specifying data types (e.g., INT, VARCHAR), and defining indexes for optimization.



CUSTOMERS

CustomerID	Integer
Name	Varchar(30)
Region	Varchar(30)
Phone	Varchar(10)
Email	Varchar(30)
Address	Varchar(30)

PRODUCTS

ProductID	Integer
Name	Varchar(30)
Category	Varchar(30)
Unit Price	Integer
Color	Varchar(30)
Rating	Integer
Description	Varchar(30)

SALES

CustomerID	Integer
ProductID	Integer
Date	Date
Sold_Qty.	Integer
Sales_Amount	Float

DATE

Date	Date
Year	Integer
Fiscal_Year	Integer
Month	Varchar(30)
Quarter	Integer

TABLE

Tables are the core structures used in databases to store and organize data. Each table consists of rows & columns, where each column represents an attribute, & each row represents a record. Tables are fundamental components in databases, used to store & organize data efficiently.

Types of Tables:

In data modeling, tables are primarily categorized into two types:

1. Fact Tables

2. Dimension Tables

TYPES OF TABLES

1. FACT TABLE :

Fact tables store quantitative data for analysis and often contain measurable, numerical data. These tables are typically used to store transactional data and are designed to capture the metrics or facts of a business process.

Characteristics:

- Contain quantitative data (facts) for analysis.
- Have a composite key made up of foreign keys from dimension tables.
- Store aggregated data (e.g., sales, revenue, quantities).

TYPES OF TABLES

Example:

A sales fact table might include data such as:

SALES				
SaleID	DateID	ProductID	CustomerID	Amount
1001	2024-01-01	201	1	500.00
1002	2024-01-02	202	2	300.00
1003	2024-01-03	203	1	700.00

- **SaleID:** Unique identifier for each sale (primary key).
- **DateID:** (Foreign key) referencing a date dimension table.
- **ProductID:** (Foreign key) referencing a product dimension table.
- **CustomerID:** (Foreign key) referencing a customer dimension table.
- **Amount:** The total sale amount.

TYPES OF TABLES

2. DIMENSION TABLE :

Dimension tables consist of attributes that describe the objects of a fact table. Each dimension table includes a primary key that uniquely identifies each record and using this key dimension table associates with fact tables. It store descriptive, textual, or categorical data related to dimensions of the facts.

Characteristics:

- Contain descriptive attributes.
- Serve as reference information for fact tables.
- Usually have a single primary key that is referenced by fact tables.

TYPES OF TABLES

Example:

A product dimension table might include:

PRODUCT			
ProductID	**ProductName**	**Price**	**Category**
201	Widget A	10.00	Gadgets
202	Widget B	20.00	Gadgets
203	Widget C	30.00	Widgets

- **ProductID:** Unique identifier for each product (primary key).
- **ProductName:** The name of the product
- **Price:** The price of the product.
- **Category:** The category the product belongs to.