**About this dataset:**

The "Northwind" database is designed to centralize and streamline the management of various entities involved in product management. By having a well-structured database, the system can efficiently handle operations such as inventory management, order processing, customer relationship management, and supplier coordination.

**Entities and Their Descriptions:**

**1. Products**

**Table Name:** Products

**Description:** This table contains information about all the products available in the inventory. Each product is uniquely identified and described with attributes such as:

* **ProductID:** Unique identifier for each product. (Primary Key).
* **CategoryID:** Key linking to the Categories table. (Foreign Key)**.**
* **ProductName:** Name of the product.
* **SupplierID:** Identifier for the supplier of the product. (Foreign Key)
* **UnitPrice:** Price per unit of the product.
* **UnitsInStock:** Number of units currently in stock.
* **UnitsOnOrder:** Number of units currently on order.
* **QuantityPerUnit:** Packaging details, e.g., "24 - 12 oz bottles."
* **ReorderLevel:** Minimum stock level to trigger reorder.
* **Discontinued:** Boolean indicator if the product is no longer available.

**2. Customers**

**Table Name:** Customers

**Description**:This table holds information about customers, including personal and contact details.

* **CustomerID:** Unique identifier for each customer. (Primary Key)
* **CompanyName:** Name of the customer's company.
* **ContactName:** Name of the contact person.
* **ContactTitle:** Title of the contact person.
* **Address:** Street No. Customer.
* **City:** City where the customer is located.
* **Region:** Region where the customer is located.
* **Country**: Country where the customer is located.
* **Phone:** Contact phone number.
* **Fax:** Contact fax number.
* **PostalCode:** Customer’s postal address.

**3. Employees**

**Table Name:** Employees

**Description:** This table stores information about employees working within the company, including personal and contact details.

* **EmployeeID:** Unique identifier for each employee. (Primary Key)
* **FirstName:** First name of the employee.
* **LastName:** Last name of the employee.
* **Title:** Job title of the employee.
* **Address:** Address of the employee.
* **City:** City where the employee is located.
* **Country:** Country where the employee is located.
* **Region:** The region or state where the employee is located.
* **HomePhone:** Home phone number.
* **HireDate:** Date when the employee was hired.
* **BirthDate:** The employee's date of birth.
* **Note:** Additional notes or comments about the employee.
* **TitleOfCourtesy:** The courtesy title used for the employee, such as Mr., Ms., or Dr.
* **PostalCode:** The postal code of the employee's address.
* **Extension:** The extension number for the employee's office phone.
* **Photo:** Binary Image for the Employee.
* **Photo Path:** Link for the employee’s photo.
* **ReportsTo:** Identifier of the supervisor of the employee.

**4. Orders**

**Table Name:** Orders

**Description:** This table tracks orders placed by customers, including details about order dates, shipping information, and the employees who handled the orders.

* **OrderID:** Unique identifier for each order. (Primary Key)
* **CustomerID**: Key linking to the Customers table. (Foreign key)
* **EmployeeID:** Key linking to the Employees table. (Foreign key)
* **ShipVia:** Key linking to the Shippers table. (Foreign key)
* **OrderDate:** Date when the order was placed.
* **RequiredDate:** Date by which the order is required.
* **ShippedDate:** Date when the order was shipped.
* **Freight:** Cost of shipping.
* **ShipName:** Name of the Customer purchasing the order.
* **ShipAddress:** Shipping address for the order.
* **ShipCity:** City where the order is shipped to.
* **ShipCountry:** Country where the order is shipped to.
* **ShipRegion:** The region or state where the order is shipped to.
* **ShipPostalCode**: The postal code of the order's destination.
* **ModifiedDate**: The last date in which the record is updated or inserted.

**5. Shippers**

**Table Name:** Shippers

**Description:** This table contains information about the companies responsible for shipping orders to customers.

* **ShipperID**: Unique identifier for each shipper. (Primary Key)
* **CompanyName**: Name of the shipping company.
* **Phone**: Contact phone number.

**6. Suppliers**

**Table Name:** Suppliers

**Description:** This table includes details about the suppliers who provide products for the company. Information includes contact details and company information.

* **SupplierID**: Unique identifier for each supplier. (Primary Key)
* **CompanyName**: Name of the supplier's company.
* **ContactName**: Name of the contact person.
* **ContactTitle**: Title of the contact person.
* **Address**: Address of the supplier.
* **City**: City where the supplier is located.
* **Country**: Country where the supplier is located.
* **Region:** The region or state where the employee is located.
* **Phone**: Contact phone number.
* **Fax:** Contact fax number.
* **PostalCode:** Supplier’s postal address.
* **HomePage:** The URL of the supplier's website.

**7. Categories**

**Table Name:** Categories

**Description:** This table categorizes all products into distinct groups, making it easier to organize and manage the inventory.

* **CategoryID**: Unique identifier for each category. (Primary Key)
* **CategoryName**: Name of the category.
* **Description**: Detailed description of the category.

**8. Territories:**

**Table Name:** Territories

**Description:** This table contains information about geographical areas where the company's business operations are carried out. It helps in organizing and managing business activities within different regions.

* **TerritoryID**: Unique identifier for each territory. (Primary Key)
* **RegionID**: Key linking to the Region table. (Foreign Key)
* **TerritoryDescription:** Description or name of the territory.

**9. Regions:**

**Table Name:** Region

**Description:** This table contains information about the larger geographical regions that encompass multiple territories. It helps in organizing territories into broader areas for better management and reporting.

* **RegionID**: Unique identifier for each region (Primary Key)
* **RegionDescription:** Description or name of the Region.

**Entities Relationships:**

1. **Orders and Customers**
   * **Relationship**: One-to-Many
   * **Description**: Many orders can be placed by a customer, represented by the CustomerID foreign key in the Orders table.
2. **Orders and Shippers**
   * **Relationship**: One-to-Many
   * **Description**: Many orders can be shipped by a shipper, represented by the ShipperID foreign key in the Orders table.
3. **Orders and Employees**
   * **Relationship**: One-to-Many
   * **Description**: Many orders can be processed by an employee, represented by the EmployeeID foreign key in the Orders table.
4. **Orders and Products**
   * **Relationship**: Many-to-Many
   * **Description**: Many orders can have many products. This relationship is established through the OrderID foreign key in the OrderDetails table, linking to the Orders table, and the ProductID foreign key linking to the Products table.
5. **Categories and Products**
   * **Relationship**: One-to-Many
   * **Description**: Each category can have multiple products. This relationship is represented by the CategoryID primary key in the Categories table, which is referenced in the Products table.
6. **Suppliers and Products**
   * **Relationship**: One-to-Many
   * **Description**: Each supplier can supply multiple products. This relationship is represented by the SupplierID primary key in the Suppliers table, which is referenced in the Products table.
7. **Employee and Employee**
   * **Relationship**: One-to-Many
   * **Description**: Many employees can report to one manager. This recursive relationship is represented by the EmployeeID primary key in the Employees table.
8. **Employee and Territory**
   * **Relationship**: Many-to-Many
   * **Description**: Each employee can be associated with multiple territories, and each territory can have multiple employees. This relationship is typically implemented using a junction table, often called EmployeeTerritories. This table contains foreign keys linking EmployeeID from the Employees table and TerritoryID from the Territories table.
9. **Region and Territory**
   * **Relationship**: One-to-Many
   * **Description**: Each territory belongs to one region, and each region can encompass multiple territories. This relationship is represented by the RegionID foreign key in the Territories table.