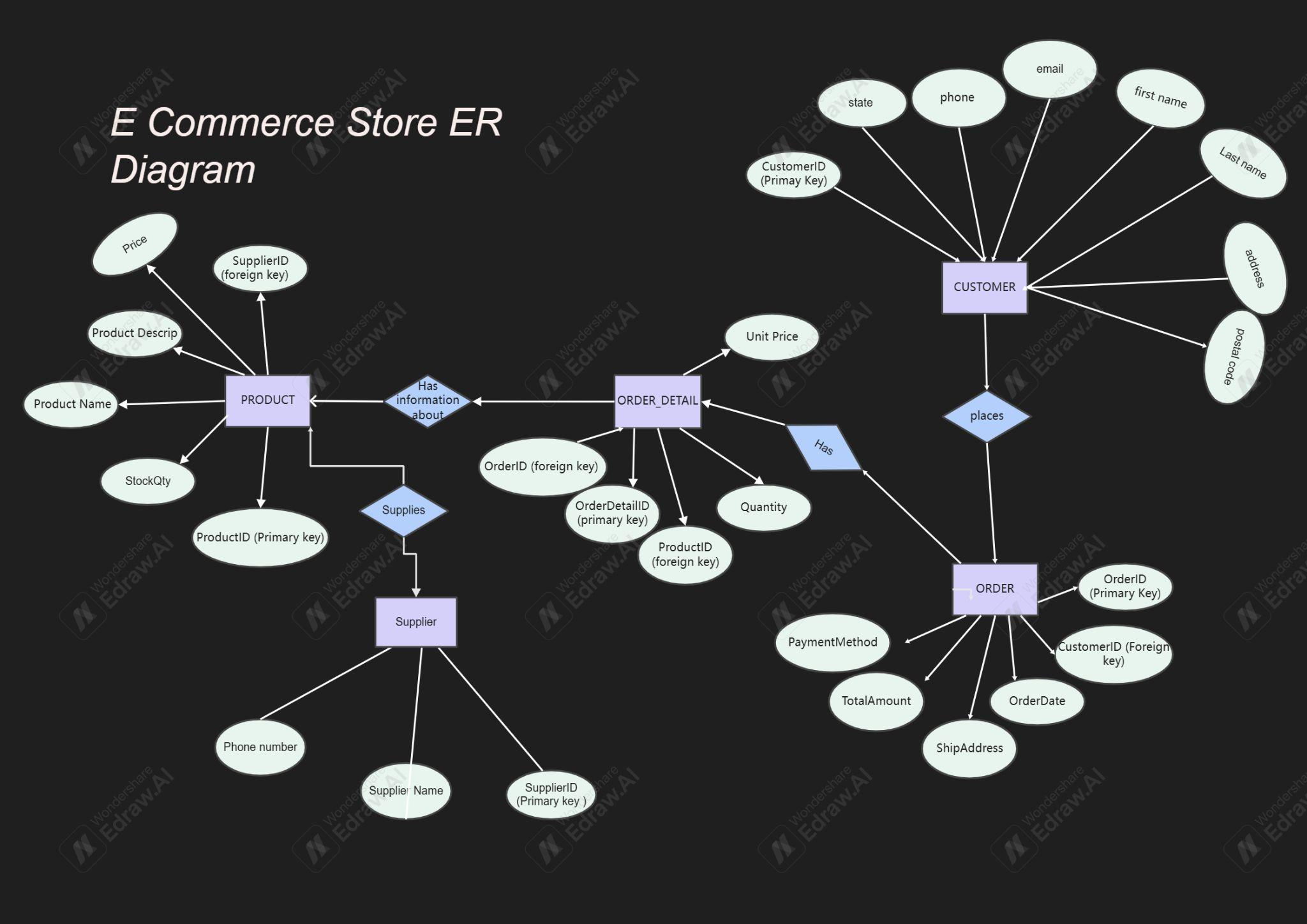
### **For designing the data base i have taken Following Entities and Attributes:**

1. **Customers**:
   * CustomerID (Primary key)
   * FirstName
   * LastName
   * Email
   * Phone
   * Address
   * State
   * PostalCode
2. **Products**:
   * ProductID (Primary Key )
   * ProductName
   * ProductDescription
   * Price
   * StockQuantity
   * SupplierID (Foreign Key)
3. **Orders**:
   * OrderID (Primary Key)
   * CustomerID (Foreign key)
   * OrderDate
   * ShippingAddress
   * TotalAmount
   * PaymentMethod
4. **OrderDetails**:
   * OrderDetailID (Primary key)
   * OrderID (Foreign Key)
   * ProductID (FK)
   * Quantity
   * UnitPrice
5. **Suppliers**:
   * SupplierID (Primary Key)
   * SupplierName
   * PhoneNumber

**ER DIAGRAM**



### **Entities and Relationships:**

1. **Customers**:
   * **Attributes**: CustomerID (PK), FirstName, LastName, Email, Phone, Address, State, PostalCode
   * **Primary Key** : CustomerID
   * **Relationships**:
     + **One-to-Many** relationship with **Orders** (One customer can place many orders).
2. **Products**:
   * **Attributes**: ProductID (PK), ProductName, ProductDescription, Price, StockQuantity
   * **Primary Key (PK)**: ProductID
   * **Relationships**:
     + **Many-to-One** relationship with **Suppliers** (Each product is supplied by one supplier).
     + **Many-to-Many** relationship with **Orders** through the **OrderDetails** table (A product can be part of many orders, and an order can contain multiple products).
3. **Orders**:
   * **Attributes**: OrderID (PK), CustomerID (FK), OrderDate, ShippingAddress, TotalAmount, PaymentMethod
   * **Primary Key (PK)**: OrderID
   * **Foreign Key (FK)**: CustomerID (references **Customers**)
   * **Relationships**:
     + **Many-to-One** relationship with **Customers** (Many orders can be placed by one customer).
     + **Many-to-Many** relationship with **Products** through **OrderDetails** (One order can contain many products).
4. **OrderDetails**:
   * **Attributes**: OrderDetailID (PK), OrderID (FK), ProductID (FK), Quantity, UnitPrice
   * **Primary Key (PK)**: OrderDetailID
   * **Foreign Keys (FK)**: OrderID (references **Orders**), ProductID (references **Products**)
   * **Relationships**:
     + **Many-to-One** relationship with **Orders** (One order can have multiple products).
     + **Many-to-One** relationship with **Products** (A product can appear in many orders).
5. **Suppliers**:
   * **Attributes**: SupplierID (PK), SupplierName, PhoneNumber
   * **Primary Key (PK)**: SupplierID
   * **Relationships**:
     + **One-to-Many** relationship with **Products** (A supplier can supply many products).

### 

### 

### 

### 

### 

### 

### 

### 

### 

### **Explaining Normalisation**

### **Normalization Explanation in Brief**

#### **1st Normal Form (1NF):**

A table is in **1NF** if:

* All columns have atomic values (indivisible).
* Each row is unique, identified by a **Primary Key (PK)**.
* Each column has a unique name.

**How 1NF is satisfied**:

* All attributes are atomic (e.g., the **Customers** table has separate fields for FirstName, LastName, Email, etc.).
* Each table has a **Primary Key** (e.g., CustomerID, ProductID, OrderID), ensuring uniqueness and no duplicate rows.

#### **2nd Normal Form (2NF):**

A table is in **2NF** if:

* It is in **1NF**.
* It has no **partial dependencies**, meaning non-key attributes must depend on the entire primary key, not just part of it (relevant for composite primary keys).

**How 2NF is satisfied**:

* **OrderDetails** has a composite primary key (OrderID, ProductID). Both Quantity and UnitPrice depend on both keys, not just one part of them.
* All other tables have single-column primary keys, and all non-key attributes depend entirely on the primary key (e.g., in **Customers**, FirstName, LastName depend on CustomerID).

#### **3rd Normal Form (3NF):**

A table is in **3NF** if:

* It is in **2NF**.
* There are no **transitive dependencies** (non-key attributes should not depend on other non-key attributes).

**How 3NF is satisfied**:

* In **Customers**, FirstName doesn’t depend on LastName, nor does Phone depend on Email. All non-key attributes depend only on CustomerID.
* Similarly, in **Products**, **Orders**, and **Suppliers**, all non-key attributes depend directly on their respective primary keys, and there are no indirect dependencies.