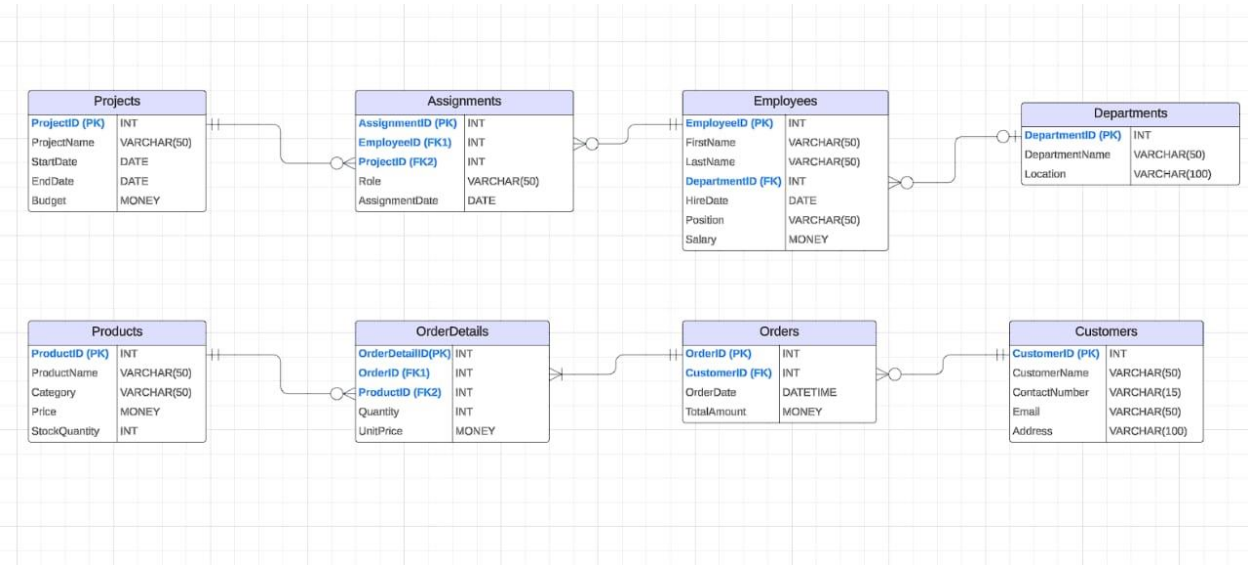
Database Documentation

Introduction

Purpose: This database is designed to manage various aspects of a company's operations, including projects, employees, departments, products, orders, and customers.
Overview: The database consists of interconnected tables that store data related to projects, assignments, employees, departments, products, order details, orders, and customers.

Database Structure

ERD



Tables and Columns

- ## Schema

staff	sales
- projects - assignments - employees - department	- customer - orders - orderDetails - products

1. ## Projects

- **Table Name**: `Projects`
- **Description:** Stores information about company projects.

- **Columns:**
  - `ProjectID` (INT, Primary Key): Unique identifier for the project.
  - `ProjectName` (VARCHAR(50)): Name of the project.
  - `StartDate` (DATE): Start date of the project.
  - `EndDate` (DATE): End date of the project.
  - `Budget` (MONEY): Budget allocated for the project.

2. Assignments

- **Table Name:** `Assignments`
- **Description**: Links employees to projects.
- **Columns:**
  - `AssignmentID` (INT, Primary Key): Unique identifier for the assignment.
  - `EmployeeID` (INT, Foreign Key): References `Employees(EmployeeID)`.
  - `ProjectID` (INT, Foreign Key): References `Projects(ProjectID)`.
  - `Role` (VARCHAR(50)): Role of the employee in the project.
  - `AssignmentDate` (DATE): Date of assignment.

3. Employees

- **Table Name:** `Employees`
- **Description**: Stores information about employees.
- **Columns:**
  - `EmployeeID` (INT, Primary Key): Unique identifier for the employee.
  - `FirstName` (VARCHAR(50)): First name of the employee.
  - `LastName` (VARCHAR(50)): Last name of the employee.
  - `DepartmentID` (INT, Foreign Key): References `Departments(DepartmentID)`.
  - `HireDate` (DATE): Date of hire.
  - `Position` (VARCHAR(50)): Position of the employee.
  - `Salary` (MONEY): Salary of the employee.

4. Departments

- **Table Name:** `Departments`
- **Description:** Stores information about departments within the company.
- **Columns:**
  - `DepartmentID` (INT, Primary Key): Unique identifier for the department.
  - `DepartmentName` (VARCHAR(50)): Name of the department.
  - `Location` (VARCHAR(100)): Location of the department.

5. Products

- **Table Name:** `Products`
- **Description:** Stores information about products offered by the company.
- **Columns:**
  - `ProductID` (INT, Primary Key): Unique identifier for the product.
  - `ProductName` (VARCHAR(50)): Name of the product.
  - `Category` (VARCHAR(50)): Category of the product.
  - `Price` (MONEY): Price of the product.
  - `StockQuantity` (INT): Quantity of the product in stock.

6. ## OrderDetails

- **Table Name:** `OrderDetails`
- **Description:** Stores details of orders.
- **Columns:**
  - `OrderDetailID` (INT, Primary Key): Unique identifier for the order detail.
  - `OrderID` (INT, Foreign Key): References `Orders(OrderID)`.
  - `ProductID` (INT, Foreign Key): References `Products(ProductID)`.
  - `Quantity` (INT): Quantity of the product ordered.
  - `UnitPrice` (MONEY): Price per unit of the product.

7. ## Orders

- **Table Name**: `Orders`
- **Description:** Stores information about customer orders.
- **Columns:**
  - `OrderID` (INT, Primary Key): Unique identifier for the order.
  - `CustomerID` (INT, Foreign Key): References `Customers(CustomerID)`.
  - `OrderDate` (DATETIME): Date and time of the order.
  - `TotalAmount` (MONEY): Total amount of the order.

8. ## Customers

- **Table Name:** `Customers`
- **Description:** Stores information about customers.
- **Columns:**
  - `CustomerID` (INT, Primary Key): Unique identifier for the customer.
  - `CustomerName` (VARCHAR(50)): Name of the customer.
  - `ContactNumber` (VARCHAR(15)): Contact number of the customer.
  - `Email` (VARCHAR(50)): Email address of the customer.
  - `Address` (VARCHAR(100)): Address of the customer.

Relationships

- `Projects` are linked to `Assignments` via `ProjectID`.
- `Employees` are linked to `Assignments` via `EmployeeID`.
- `Departments` are linked to `Employees` via `DepartmentID`.
- `Products` are linked to `OrderDetails` via `ProductID`.
- `Orders` are linked to `OrderDetails` via `OrderID`.
- `Customers` are linked to `Orders` via `CustomerID`.

Indexes

• **Primary Keys:** Ensure uniqueness of records in each table.
• **Foreign Keys**: Maintain referential integrity between tables.