inventory management system

INVENTORY MANAGEMENT SYSTEM

1. **Description**: An inventory management database system is a computerized tool that helps businesses track and manage inventory levels, from raw materials to finished goods. It provides features like inventory tracking, sales tracking, reporting, alerts for low stock, and integration with other business systems (e.g., accounting, order management). Benefits include improved inventory accuracy, reduced costs, better customer service, and increased efficiency through automation. When choosing a system, consider your business size, budget, and specific needs to find the right solution.
2. ERD:

[Product] ----< Belongs to >---- [Category]

| |

| |

| |

<Stored in> <Supplied by>

| |

| |

[Inventory] ----< Located in >---- [Location]

| |

| |

<Includes> <Places>

| |

| |

[PurchaseOrderDetail] ----< Part of >---- [PurchaseOrder]

| |

| |

<Includes> <Places>

| |

| |

[SalesOrderDetail] ----< Part of >---- [SalesOrder]

| |

| |

<Includes> <Places>

| |

| |

[Customer] ------------------------------

1. LDM:

**LDM Diagram Structure:**

* **Entities** are represented as tables.
* **Attributes** are listed as columns within the tables.
* **Primary Keys (PK)** are underlined and uniquely identify each record.
* **Foreign Keys (FK)** establish relationships between tables.
* **Relationships** are represented using lines connecting the tables.

[Product] --------------------- [Category]

| ProductID (PK) | CategoryID (PK)

| Name | CategoryName

| Description | Description

| SKU |

| UnitPrice |

| CategoryID (FK) |

[Inventory] ------------------- [Location]

| InventoryID (PK) | LocationID (PK)

| ProductID (FK) | LocationName

| LocationID (FK) | Address

| Quantity | ContactInfo

| ReorderLevel |

[Supplier] -------------------- [PurchaseOrder]

| SupplierID (PK) | PurchaseOrderID (PK)

| SupplierName | SupplierID (FK)

| ContactInfo | OrderDate

| Address | ExpectedDeliveryDate

| TotalAmount

[PurchaseOrderDetail] --------- [Product]

| PurchaseOrderID (FK) | ProductID (PK)

| ProductID (FK) | Name

| Quantity | Description

| UnitPrice | SKU

| UnitPrice

| CategoryID (FK)

[Customer] -------------------- [SalesOrder]

| CustomerID (PK) | SalesOrderID (PK)

| CustomerName | CustomerID (FK)

| ContactInfo | OrderDate

| Address | TotalAmount

[SalesOrderDetail] ------------ [Product]

| SalesOrderID (FK) | ProductID (PK)

| ProductID (FK) | Name

| Quantity | Description

| UnitPrice | SKU

| UnitPrice

| CategoryID (FK)

**Key Features of the LDM:**

1. **Normalization**:
   * The database is normalized to eliminate redundancy and ensure data integrity.
   * Typically follows **3NF (Third Normal Form)**.
2. **Primary and Foreign Keys**:
   * Ensures unique identification of records and establishes relationships between tables.
3. **Scalability**:
   * Designed to handle growing data volumes and business needs.
4. **Data Integrity**:
   * Constraints (e.g., NOT NULL, UNIQUE) ensure accurate and consistent data.
5. **Efficient Querying**:
   * Optimized for fast data retrieval and reporting.
6. PDM:

**Tables and Columns:**

1. **Product**:
   * **ProductID** (INT, Primary Key, Auto Increment)
   * Name (VARCHAR(100), NOT NULL)
   * Description (TEXT, NULL)
   * SKU (VARCHAR(50), UNIQUE, NOT NULL)
   * UnitPrice (DECIMAL(10, 2), NOT NULL)
   * **CategoryID** (INT, Foreign Key to Category)
2. **Category**:
   * **CategoryID** (INT, Primary Key, Auto Increment)
   * CategoryName (VARCHAR(100), NOT NULL)
   * Description (TEXT, NULL)
3. **Inventory**:
   * **InventoryID** (INT, Primary Key, Auto Increment)
   * **ProductID** (INT, Foreign Key to Product)
   * **LocationID** (INT, Foreign Key to Location)
   * Quantity (INT, NOT NULL)
   * ReorderLevel (INT, NOT NULL)
4. **Location**:
   * **LocationID** (INT, Primary Key, Auto Increment)
   * LocationName (VARCHAR(100), NOT NULL)
   * Address (VARCHAR(255), NOT NULL)
   * ContactInfo (VARCHAR(100), NULL)
5. **Supplier**:
   * **SupplierID** (INT, Primary Key, Auto Increment)
   * SupplierName (VARCHAR(100), NOT NULL)
   * ContactInfo (VARCHAR(100), NOT NULL)
   * Address (VARCHAR(255), NULL)
6. **PurchaseOrder**:
   * **PurchaseOrderID** (INT, Primary Key, Auto Increment)
   * **SupplierID** (INT, Foreign Key to Supplier)
   * OrderDate (DATETIME, NOT NULL)
   * ExpectedDeliveryDate (DATETIME, NULL)
   * TotalAmount (DECIMAL(10, 2), NOT NULL)
7. **PurchaseOrderDetail**:
   * **PurchaseOrderID** (INT, Foreign Key to PurchaseOrder)
   * **ProductID** (INT, Foreign Key to Product)
   * Quantity (INT, NOT NULL)
   * UnitPrice (DECIMAL(10, 2), NOT NULL)
8. **SalesOrder**:
   * **SalesOrderID** (INT, Primary Key, Auto Increment)
   * **CustomerID** (INT, Foreign Key to Customer)
   * OrderDate (DATETIME, NOT NULL)
   * TotalAmount (DECIMAL(10, 2), NOT NULL)
9. **SalesOrderDetail**:
   * **SalesOrderID** (INT, Foreign Key to SalesOrder)
   * **ProductID** (INT, Foreign Key to Product)
   * Quantity (INT, NOT NULL)
   * UnitPrice (DECIMAL(10, 2), NOT NULL)
10. **Customer**:
    * **CustomerID** (INT, Primary Key, Auto Increment)
    * CustomerName (VARCHAR(100), NOT NULL)
    * ContactInfo (VARCHAR(100), NOT NULL)
    * Address (VARCHAR(255), NULL)

**Constraints:**

1. **Primary Keys**:
   * Each table has a primary key to uniquely identify records (e.g., ProductID, CategoryID, InventoryID).
2. **Foreign Keys**:
   * Relationships between tables are enforced using foreign keys (e.g., CategoryID in Product references Category).
3. **Unique Constraints**:
   * Ensures no duplicate values in specific columns (e.g., SKU in Product).
4. **NOT NULL Constraints**:
   * Ensures mandatory fields are always populated (e.g., Name in Product).
5. **Check Constraints**:
   * Validates data (e.g., Quantity in Inventory must be greater than or equal to 0).

**Indexes:**

1. **Primary Key Indexes**:
   * Automatically created for primary keys (e.g., ProductID in Product).
2. **Foreign Key Indexes**:
   * Improves join performance (e.g., CategoryID in Product).
3. **Custom Indexes**:
   * Created on frequently searched columns (e.g., SKU in Product).

5.DATA DICTIONARY:

**Tables and Columns:**

**1. Product**

| **Column Name** | **Data Type** | **Constraints** | **Description** |
| --- | --- | --- | --- |
| ProductID | INT | Primary Key, Auto Increment | Unique identifier for each product. |
| Name | VARCHAR(100) | NOT NULL | Name of the product. |
| Description | TEXT | NULL | Description of the product. |
| SKU | VARCHAR(50) | UNIQUE, NOT NULL | Stock Keeping Unit (unique identifier). |
| UnitPrice | DECIMAL(10, 2) | NOT NULL | Price per unit of the product. |
| CategoryID | INT | Foreign Key (Category) | Category to which the product belongs. |

**2. Category**

| **Column Name** | **Data Type** | **Constraints** | **Description** |
| --- | --- | --- | --- |
| CategoryID | INT | Primary Key, Auto Increment | Unique identifier for each category. |
| CategoryName | VARCHAR(100) | NOT NULL | Name of the category. |
| Description | TEXT | NULL | Description of the category. |

**3. Inventory**

| **Column Name** | **Data Type** | **Constraints** | **Description** |
| --- | --- | --- | --- |
| InventoryID | INT | Primary Key, Auto Increment | Unique identifier for inventory records. |
| ProductID | INT | Foreign Key (Product) | Product in the inventory. |
| LocationID | INT | Foreign Key (Location) | Location where the product is stored. |
| Quantity | INT | NOT NULL, CHECK (>= 0) | Quantity of the product in stock. |
| ReorderLevel | INT | NOT NULL | Minimum stock level before reordering. |

**4. Location**

| **Column Name** | **Data Type** | **Constraints** | **Description** |
| --- | --- | --- | --- |
| LocationID | INT | Primary Key, Auto Increment | Unique identifier for each location. |
| LocationName | VARCHAR(100) | NOT NULL | Name of the location (e.g., warehouse). |
| Address | VARCHAR(255) | NOT NULL | Address of the location. |
| ContactInfo | VARCHAR(100) | NULL | Contact information for the location. |

**5. Supplier**

| **Column Name** | **Data Type** | **Constraints** | **Description** |
| --- | --- | --- | --- |
| SupplierID | INT | Primary Key, Auto Increment | Unique identifier for each supplier. |
| SupplierName | VARCHAR(100) | NOT NULL | Name of the supplier. |
| ContactInfo | VARCHAR(100) | NOT NULL | Contact information for the supplier. |
| Address | VARCHAR(255) | NULL | Address of the supplier. |

**6. PurchaseOrder**

| **Column Name** | **Data Type** | **Constraints** | **Description** |
| --- | --- | --- | --- |
| PurchaseOrderID | INT | Primary Key, Auto Increment | Unique identifier for each purchase order. |
| SupplierID | INT | Foreign Key (Supplier) | Supplier for the purchase order. |
| OrderDate | DATETIME | NOT NULL | Date the order was placed. |
| ExpectedDeliveryDate | DATETIME | NULL | Expected delivery date. |
| TotalAmount | DECIMAL(10, 2) | NOT NULL | Total amount of the purchase order. |

**7. PurchaseOrderDetail**

| **Column Name** | **Data Type** | **Constraints** | **Description** |
| --- | --- | --- | --- |
| PurchaseOrderID | INT | Foreign Key (PurchaseOrder) | Purchase order associated with the detail. |
| ProductID | INT | Foreign Key (Product) | Product in the purchase order. |
| Quantity | INT | NOT NULL | Quantity of the product ordered. |
| UnitPrice | DECIMAL(10, 2) | NOT NULL | Price per unit of the product. |

**8. SalesOrder**

| **Column Name** | **Data Type** | **Constraints** | **Description** |
| --- | --- | --- | --- |
| SalesOrderID | INT | Primary Key, Auto Increment | Unique identifier for each sales order. |
| CustomerID | INT | Foreign Key (Customer) | Customer who placed the order. |
| OrderDate | DATETIME | NOT NULL | Date the order was placed. |
| TotalAmount | DECIMAL(10, 2) | NOT NULL | Total amount of the sales order. |

**9. SalesOrderDetail**

| **Column Name** | **Data Type** | **Constraints** | **Description** |
| --- | --- | --- | --- |
| SalesOrderID | INT | Foreign Key (SalesOrder) | Sales order associated with the detail. |
| ProductID | INT | Foreign Key (Product) | Product in the sales order. |
| Quantity | INT | NOT NULL | Quantity of the product sold. |
| UnitPrice | DECIMAL(10, 2) | NOT NULL | Price per unit of the product. |

**10. Customer**

| **Column Name** | **Data Type** | **Constraints** | **Description** |
| --- | --- | --- | --- |
| CustomerID | INT | Primary Key, Auto Increment | Unique identifier for each customer. |
| CustomerName | VARCHAR(100) | NOT NULL | Name of the customer. |
| ContactInfo | VARCHAR(100) | NOT NULL | Contact information for the customer. |
| Address | VARCHAR(255) | NULL | Address of the customer. |

**Relationships:**

1. **Product** ↔ **Category**: A product belongs to one category.
2. **Product** ↔ **Inventory**: A product can be stored in multiple locations.
3. **Inventory** ↔ **Location**: Inventory is stored in specific locations.
4. **Supplier** ↔ **PurchaseOrder**: A supplier can have multiple purchase orders.
5. **PurchaseOrder** ↔ **PurchaseOrderDetail**: A purchase order can include multiple products.
6. **Customer** ↔ **SalesOrder**: A customer can place multiple sales orders.
7. **SalesOrder** ↔ **SalesOrderDetail**: A sales order can include multiple products.

**Key Features of the Data Dictionary:**

1. **Detailed Metadata**:
   * Provides a clear description of each table and column.
2. **Constraints**:
   * Specifies primary keys, foreign keys, unique constraints, and NOT NULL constraints.
3. **Data Types**:
   * Defines the type of data stored in each column (e.g., INT, VARCHAR, DECIMAL).
4. **Relationships**:
   * Describes how tables are related to each other.