**WAREHOUSE** (<u>WarehouseID</u>, WarehouseStreet, WarehouseCity, WarehouseState, WarehouseCountry, WarehouseZip, PhoneNumber, LastUpdatedBy, LastUpdated)

**SUPPLIER** (<u>SupplierID</u>, SupplierStreet, SupplierCity, SupplierState, SupplierCountry, SupplierZip, EmailAddress, PhoneNumber, CompanyName, LastUpdatedBy, LastUpdated)

**SHIPMENT** (<u>ShipmentID</u>, SupplierProduct, StoreID)

FK StoreID→ STORE

**PRODUCT_CATEGORY** (<u>CategoryID</u>, CategoryName, Description, LastUpdatedBy, LastUpdated)

**PRODUCT** (<u>ProductID,</u> ProductName, InventoryProcessed, InventorySold, QtyInStock, Price, ExpirationDate, Description, CategoryID, LastUpdatedBy, LastUpdated)

FK CategoryID→ ProductCategory

**ORDER_DETAIL** (<u>OrderNumber, ProductID,</u> Price, Quantity, LastUpdatedBy, LastUpdated)

FK OrderNumber→ ORDER

FK ProductID → PRODUCT

**ORDER** (<u>OrderNumber</u>, OrderDate, ShipmentAddress, CustomerID, EmployeeID, LastUpdatedBy, LastUpdated)

FK CustomerID→ CUSTOMER

FK EmployeeID→ EMPLOYEE

**CUSTOMER** (<u>CustomerID</u>, CustomerFirstN, CustomerLastN, DOB, EmailAddress, CustomerStreet, CustomerCity, CustomerState, CustomerCountry, CustomerZip, PhoneNumber, LastUpdatedBy, LastUpdated)

**EMPLOYEE** (<u>EmployeeID</u>, EmployeeFirstN, EmployeeLastN, DOB, EmployeeStreet, EmployeeCity, EmployeeState, EmployeeCountry, EmployeeZip, EmailAddress, SupervisorID, LastUpdatedBy, LastUpdated)

**DEPARTMENT** (<u>DepartmentID</u>, DepartmentName, StoreID, EmployeeID, LastUpdatedBy, LastUpdated)

FK StoreID----STORE

FK EmployeeID----EMPLOYEE

Many to many relationships:

1. WAREHOUSE_SUPPLIER→ between Warehouse and Supplier
2. SUPPLIER_SHIPMENT→ between Supplier and Shipment

**WAREHOUSE_SUPPLIER** (WarehouseID, SupplierID)

FK WarehouseID → WAREHOUSE

FK SupplierID → SUPPLIER

**SUPPLIER_SHIPMENT** (SupplierID, ShipmentID)

FK SupplierID→ SUPPLIER

FK ShipmentID→ SHIPMENT

    I decided to get rid of associative entities between Warehouse-Supplier, and Supplier-Shipment while creating tables, since those associative tables will not have any specific attribute other than Primary Keys of parent tables for this scenario. Inner joins can be used if needed instead of additional redundant tables.