

-- Insert into Products

```
CREATE OR REPLACE PROCEDURE insert_product(  
    p_product_id INT,  
    p_product_name VARCHAR2  
) IS  
BEGIN  
    INSERT INTO Products (Product_ID, Product_Name)  
    VALUES (p_product_id, p_product_name);  
END;  
/
```

-- Insert into Inventory

```
CREATE OR REPLACE PROCEDURE insert_inventory(  
    p_store_id INT,  
    p_product_id INT,  
    p_stock_level INT,  
    p_stock_status_date DATE,  
    p_stock_status VARCHAR2  
) IS  
BEGIN  
    INSERT INTO Inventory (Store_ID, Product_ID, Stock_Level, Stock_Status_Date,  
    Stock_Status)  
    VALUES (p_store_id, p_product_id, p_stock_level, p_stock_status_date, p_stock_status);  
END;  
/
```

-- Insert into Suppliers

```
CREATE OR REPLACE PROCEDURE insert_supplier(  
    p_supplier_id INT,  
    p_product_id INT,  
    p_lead_time INT,  
    p_delivery_history VARCHAR2  
) IS  
BEGIN  
    INSERT INTO Suppliers (Supplier_ID, Product_ID, Lead_Time, Delivery_History)  
    VALUES (p_supplier_id, p_product_id, p_lead_time, p_delivery_history);  
END;  
/
```

-- Insert into SalesTransactions

```
CREATE OR REPLACE PROCEDURE insert_sales_transaction(  
    p_transaction_id INT,
```

```

    p_product_id INT,
    p_store_id INT,
    p_quantity_sold INT,
    p_sale_date DATE
) IS
BEGIN
    INSERT INTO SalesTransactions (Transaction_ID, Product_ID, Store_ID, Quantity_Sold,
Sale_Date)
    VALUES (p_transaction_id, p_product_id, p_store_id, p_quantity_sold, p_sale_date);
END;
/

```

```

-- Read Products
CREATE OR REPLACE FUNCTION get_product(
    p_product_id INT
) RETURN VARCHAR2 IS
    v_product_name VARCHAR2(50);
BEGIN
    SELECT Product_Name INTO v_product_name
    FROM Products WHERE Product_ID = p_product_id;
    RETURN v_product_name;
EXCEPTION
    WHEN NO_DATA_FOUND THEN
        RETURN 'No product found';
END;
/

```

```

-- Read Inventory
CREATE OR REPLACE FUNCTION get_inventory(
    p_store_id INT,
    p_product_id INT
) RETURN VARCHAR2 IS
    v_stock_status VARCHAR2(20);
BEGIN
    SELECT Stock_Status INTO v_stock_status
    FROM Inventory WHERE Store_ID = p_store_id AND Product_ID = p_product_id;
    RETURN v_stock_status;
EXCEPTION
    WHEN NO_DATA_FOUND THEN
        RETURN 'No inventory found';
END;
/

```

```

-- Read Suppliers
CREATE OR REPLACE FUNCTION get_supplier(
    p_supplier_id INT
) RETURN VARCHAR2 IS
    v_delivery_history VARCHAR2(125);
BEGIN
    SELECT Delivery_History INTO v_delivery_history
    FROM Suppliers WHERE Supplier_ID = p_supplier_id;
    RETURN v_delivery_history;
EXCEPTION
    WHEN NO_DATA_FOUND THEN
        RETURN 'No supplier found';
END;
/

```

```

-- Read SalesTransactions
CREATE OR REPLACE FUNCTION get_sales_transaction(
    p_transaction_id INT
) RETURN VARCHAR2 IS
    v_sale_info VARCHAR2(200);
BEGIN
    SELECT 'Product ID: ' || Product_ID || ', Store ID: ' || Store_ID || ', Quantity Sold: ' ||
    Quantity_Sold || ', Sale Date: ' || Sale_Date
    INTO v_sale_info
    FROM SalesTransactions WHERE Transaction_ID = p_transaction_id;
    RETURN v_sale_info;
EXCEPTION
    WHEN NO_DATA_FOUND THEN
        RETURN 'No transaction found';
END;
/

```

```

-- Update Products
CREATE OR REPLACE PROCEDURE update_product(
    p_product_id INT,
    p_product_name VARCHAR2
) IS
BEGIN
    UPDATE Products
    SET Product_Name = p_product_name
    WHERE Product_ID = p_product_id;

```

END;

/

-- Update Inventory

CREATE OR REPLACE PROCEDURE update_inventory(

 p_store_id INT,

 p_product_id INT,

 p_stock_level INT,

 p_stock_status VARCHAR2

) IS

BEGIN

 UPDATE Inventory

 SET Stock_Level = p_stock_level, Stock_Status = p_stock_status

 WHERE Store_ID = p_store_id AND Product_ID = p_product_id;

END;

/

-- Update Suppliers

CREATE OR REPLACE PROCEDURE update_supplier(

 p_supplier_id INT,

 p_lead_time INT,

 p_delivery_history VARCHAR2

) IS

BEGIN

 UPDATE Suppliers

 SET Lead_Time = p_lead_time, Delivery_History = p_delivery_history

 WHERE Supplier_ID = p_supplier_id;

END;

/

-- Update SalesTransactions

CREATE OR REPLACE PROCEDURE update_sales_transaction(

 p_transaction_id INT,

 p_quantity_sold INT,

 p_sale_date DATE

) IS

BEGIN

 UPDATE SalesTransactions

 SET Quantity_Sold = p_quantity_sold, Sale_Date = p_sale_date

 WHERE Transaction_ID = p_transaction_id;

END;

/

-- Delete from Products

```
CREATE OR REPLACE PROCEDURE delete_product(  
    p_product_id INT  
) IS  
BEGIN  
    DELETE FROM Products WHERE Product_ID = p_product_id;  
END;  
/
```

-- Delete from Inventory

```
CREATE OR REPLACE PROCEDURE delete_inventory(  
    p_store_id INT,  
    p_product_id INT  
) IS  
BEGIN  
    DELETE FROM Inventory WHERE Store_ID = p_store_id AND Product_ID = p_product_id;  
END;  
/
```

-- Delete from Suppliers

```
CREATE OR REPLACE PROCEDURE delete_supplier(  
    p_supplier_id INT  
) IS  
BEGIN  
    DELETE FROM Suppliers WHERE Supplier_ID = p_supplier_id;  
END;  
/
```

-- Delete from SalesTransactions

```
CREATE OR REPLACE PROCEDURE delete_sales_transaction(  
    p_transaction_id INT  
) IS  
BEGIN  
    DELETE FROM SalesTransactions WHERE Transaction_ID = p_transaction_id;  
END;  
/
```