# FOOTWEAR PRODUCTS


**CIS – 5430 SPRING 2023**

**GROUP 1:**

**Navyasree Sriramoju (Team Leader)**

**Sushmitha Dandu (Database Developer)**

**Naga Sai Lohitha Karmuru (Database Designer)**

## CONTENTS

Introduction…….3

Purpose……3

Functionalities…….3

Users……4

Roles……4

GROUP PROJECT-1 …….5

Conceptual and Logic Design …….5

Business Rules…….5

Identify entity and relationship types. Fill out the relationship matrix……5

ER/EER diagram using software tools……6

Database Logical Design…….6

Establish join paths for the above relational database using the referential integrity…….8

Function analysis for each of the tables……...8

Show all the normalized tables and indicate the normalization form for each of your tables…….9

Tables in 2NF and 3NF……9

GROUP PROJECT-2……..……11

SQL DDL Script (Database creation script) ……..……11

Database structure/relational schema (DESC Table Name) …….21

Database instance (SELECT * FROM Table Name) …….....27

Insert, Update, Delete, and Create View statements in each group……...32

INSERT Values ……..……...…32

UPDATE Values……..……...33

DELETE Values……..……...34

CREATE View……..……35

SELECT statements with joins, subqueries, GROUP BY, HAVING, and function statements……...…35

PL/SQL statement block……..……...……39

Object Types usage for object columns or object tables……44

## INTRODUCTION

Footwear stores offer a variety of shoes and other items for all ages, genders, and styles, both in physical retail locations and online. A comprehensive database system is being developed for an online footwear store to manage product, customer, order, and inventory information using ORACLE SQL. The project aims to provide a scalable and optimized database system that streamlines e-store operations, increases data integrity and security, and supports reporting and analysis applications. The team has designed and built the database structure and developed SQL and PL/SQL statements to meet the store's requirements.

## PURPOSE

The project's major goal is to ensure effective and reliable data management for the e-store. Creating tables with proper constraints to contain product information, customer data, and order details, as well as assuring data quality and security. Inventory management will also be handled by the database, which will keep track of available stock and update it in real-time as orders are placed and fulfilled.

Another critical goal of the project is to improve data retrieval and reporting for the e-store. Defining views that provide meaningful and relevant information to store management, such as sales reporting, order tracking, and customer analytics, is part of this. To extract important insights from the data, the project may also require the creation of complicated SQL queries that include joins, subqueries, group by, and having clauses.

## FUNCTIONALITIES

The documentation for the Footwear products online shop database contains thorough information about the database's entity types, relationships, and properties. This data assists users in understanding how data is arranged and kept in the database.

The documentation also describes data validation rules and constraints to ensure that the data in the database is accurate and consistent. Defining data types, allowable values, and business rules to validate incoming data are all part of this.

DDL (Data Definition Language) and DML (Data Manipulation Language) statements are also provided in the documentation for performing Create, Read, Update, and Delete operations on the database. These instructions show users how to interact with the database in an effective and safe manner.

Additionally, the documentation provides support for Object-Relational Database Management System (ORDBMS) features such as PL/SQL data processing blocks, which can enhance performance and reduce data transfer. It also has Object Types for data and logic encapsulation, improving code organization, reusability, and security while interfacing with database capabilities.

## USERS

Marketing and sales personnel can use the database to evaluate customer data, track sales, and generate marketing reports for planning and strategy.

IT Administrators: These people may oversee maintaining and administering the database's technical features, such as database performance, security, backups, and troubleshooting.

Customers that visit the e-store to buy footwear may interact with the database indirectly via the front-end interface. They can create accounts, place orders, track order status, and check their purchasing history.

Customer Service Representatives: These users can access the database to help customers with enquiries, order management, returns, and refunds. They may also use the database to obtain consumer information to give customized customer support.

Suppliers: Suppliers who supply footwear to the e-commerce site may interface with the database to update product availability, manage inventory, and process purchase orders.

## ROLES

All the team members worked effectively together and contributed significantly to the project. In both Project 1 and Project 2, each team member plays a distinct role. The project's team leader did an excellent job, and the developers were quite helpful. All the team members worked well together and communicated effectively.

# GROUP PROJECT 1

## Conceptual and Logical Design

The online store database needs to keep track of orders for its inventory. When a customer places orders, the system must record that the order and order items. The system must update the available quantity on hand to reflect that the by product(s) has been sold. When an employee processes orders, the system must confirm that the ordered items are in stock. The online store needs to keep track of customers and employees, too. The system must update the available quantity on hand to reflect that the by product(s) has been sold. Each team create your store, database and sell your own products.

## Business Rules

One customer may or may not place many orders.

One order must be placed by one and only one customer.

One order must contain one or more product.

One product may or may not be in many orders.

One employee may process one or more orders.

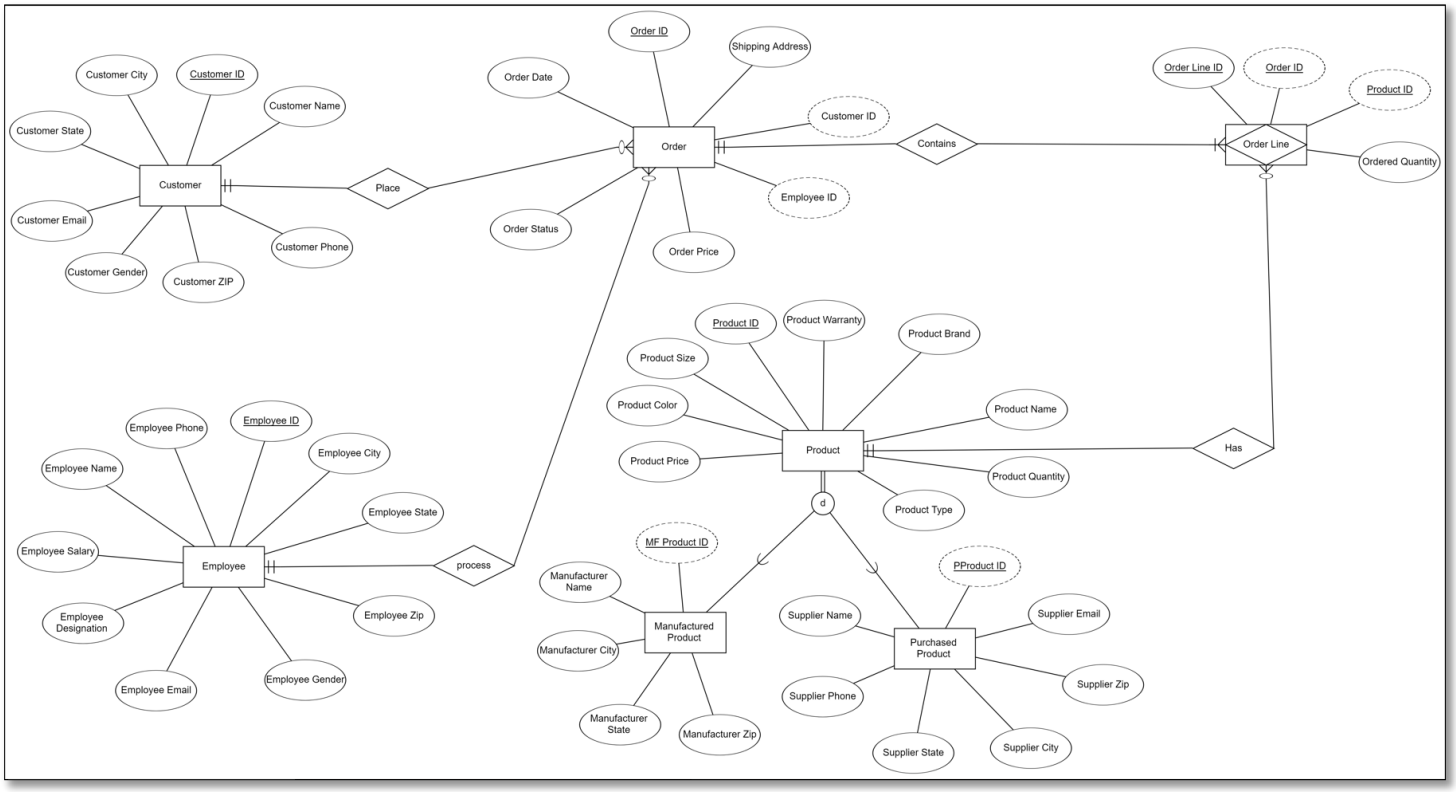
One order must be processed by one and only one employee.

One product must be either manufactured or purchased.

## Identify entity types and relationship types. Fill out the following relationship matrix.

| | Customer | Order | product | employee |
|---|---|---|---|---|
| Customer | -- | Places | -- | -- |
| Order | Is Placed | -- | Contains | Is Processed |
| Product | -- | Has | Manufactured/purchased | -- |
| Employee | -- | Processes | -- | -- |

**Draw an ER/EER diagram using software tools includes 1) entity types, 2) relationship types, 3) keys, 4) attributes, and cardinality constraints (must show participation).**




## Database Logical Design

**Map the ER diagram to a relational database schema indicating the relation name, primary key and foreign key. Add appropriate additional attributes by yourself.**

### Table Name: Customer

| Customer ID | Customer Name | Customer Phone | Customer Email | Customer Gender | Customer City | Customer State | Customer Zip |
|---|---|---|---|---|---|---|---|

### Table Name: Order

| Order ID | Order Price | Order Date | Order status | Employee ID | Customer ID | Shipping Address |
|---|---|---|---|---|---|---|

**Table Name: Order Line**

| Order Line ID | Order ID | Product ID | Ordered Quantity |
|---|---|---|---|

**Table Name: Employee**

| Employee ID | Employee Name | Employee Phone | Employee Email | Employee Designation | Employee Gender | Employee Salary | Employee City | Employee State | Employee Zip |
|---|---|---|---|---|---|---|---|---|---|

**Table Name: Product**

| Product ID | Product Name | Product Quantity | Product Type | Product Price | Product Color | Product Size | Product Warranty | Product Brand |
|---|---|---|---|---|---|---|---|---|

**Table Name: Purchased Product**

| PProduct ID | Supplier Name | Supplier Email | Supplier Phone | Supplier State | Supplier City | Supplier Zip |
|---|---|---|---|---|---|---|

**Table Name: Manufactured Product**

| MFProduct ID | Manufacturer Name | Manufacturer State | Manufacturer City | Manufacturer Zip |
|---|---|---|---|---|

**Establish join paths for the above relational database using the referential integrity by drawing arrow lines between the above tables. Indicate all the foreign keys (FK).**
**F.K. -> P.K. (Foreign Key refers to Primary Key)**

F.K.Order.EmployeeID-->P.K.Employee.EmployeeID

F.K.Order.CustomerID-->P.K.Customer.CustomerID

F.K.OrderLine.OrderID-->P.K.Order.OrderID

F.K.OrderLine.ProductID-->P.K.Product.ProductID

F.K.PurchasedProduct.PProductID--> P.K.Product.ProductID

F.K.ManufacturedProduct.MFProductID--> P.K.Product.ProductID

**Do function analysis for each of your tables**
**Attribute A --> Attribute B (Determinant attribute(s) Determines Dependent Attribute(s))**

**Transitive Dependencies**

Customer Zip-->Customer City, Customer State

Employee Zip-->Employee City, Employee State

Supplier Zip-->Supplier City, Supplier State

Manufacturer Zip-->Manufacturer City, Manufacturer State

**Full Dependencies**

Customer ID-->Customer Name, Customer Phone, Customer Email, Customer Gender, Customer City, Customer State, Customer Zip

Order ID-->Order Price, Order Date, Order status, Shipping Address

Order Line-->Order Quantity

Employee ID-->Employee ID, Employee Name, Employee Phone, Employee Email, Employee Designation, Employee Gender, Employee Salary, Employee City, Employee State, Employee Zip, Employee Salary

Product ID--> Product Name, Product Quantity, Product Type, Product Price, Product Color, Product Size, Product Warranty, Product Brand

PProduct ID--> Supplier Name, Supplier Phone, Supplier Email, Supplier State, Supplier City, Supplier Zip

MFProduct ID--> Manufacturer Name, Manufacturing State, Manufacturing City, Manufacturer Zip

**Show all the normalized tables and indicate the normalization form for each of your tables.**

| Table Name | 1NF | 2NF | 3NF |
|---|---|---|---|
| Customer | ✓ | ✓ | |
| Order | ✓ | ✓ | ✓ |
| Order Line | ✓ | ✓ | ✓ |
| Employee | ✓ | ✓ | |
| Product | ✓ | ✓ | ✓ |
| Purchased Product | ✓ | ✓ | |
| Manufactured Product | ✓ | ✓ | |
| Customer Address | ✓ | ✓ | ✓ |
| Supplier Address | ✓ | ✓ | ✓ |
| Employee Address | ✓ | ✓ | ✓ |
| Manufacturer Address | ✓ | ✓ | ✓ |

**Tables in 2NF and 3NF:**

**Customer (2NF)**

| Customer ID | Customer Name | Customer Phone | Customer Gender | Customer Email | Customer Zip | Customer City | Customer State |
|---|---|---|---|---|---|---|---|

**Customer Address(3NF)**

| Customer ID | Customer Zip | Customer City | Customer State |
|---|---|---|---|

**Order (3NF)**

| Order ID | Order Price | Order Date | Order Status | Shipping Address | Employee ID | Customer ID |
|---|---|---|---|---|---|---|

**Order Line(3NF)**

| Order Line ID | Order ID | Product ID | Ordered Quantity |
|---|---|---|---|

**Employee (2NF)**

| Employee ID | Employee Name | Employee Phone | Employee Email | Employee Designation | Employee gender | Employee Salary | Employee City | Employee State | Employee Zip |
|---|---|---|---|---|---|---|---|---|---|

**Employee Address (3NF)**

| Employee ID | Employee State | Employee City | Employee Zip |
|---|---|---|---|

**Product (3NF)**

| Product ID | Product Name | Product Quantity | Product Type | Product Price | Product Color | Product Size | Product Warranty | Product Brand |
|---|---|---|---|---|---|---|---|---|

**Purchased Product (2NF)**

| PProduct ID | Supplier Name | Supplier Email | Supplier Phone | Supplier City | Supplier State | Supplier Zip |
|---|---|---|---|---|---|---|

**Supplier Address(3NF)**

| PProduct ID | Supplier City | Supplier State | Supplier Zip |
|---|---|---|---|

**Manufactured Product(2NF)**

| MFProduct ID | Manufacturer Name | Manufacturer State | Manufacturer City | Manufacturer Zip |
|---|---|---|---|---|

**Manufacturer Address(3NF)**

| MFProduct ID | Manufacturer City | Manufacturer State | Manufacturer Zip |
|---|---|---|---|

## GROUP PROJECT 2

### Database Creation Script (Tables, Constraints and Inserting data)

**Table Name: Customer (Naga Sai Lohitha Karmuru)**

```
DROP TABLE Customer CASCADE CONSTRAINTS;

CREATE TABLE Customer
(
Customer_Id VARCHAR2(20) NOT NULL,
Customer_Name VARCHAR2(25),
Customer_Phone CHAR(10),
Customer_Gender CHAR(20),
Customer_Email VARCHAR(100),
Customer_Zip VARCHAR(5),
Customer_City VARCHAR(50),
Customer_State CHAR(2),
CONSTRAINT CustomerPK PRIMARY KEY(Customer_Id),
CONSTRAINT Customer_UK_Customer_Phone UNIQUE (Customer_Phone),
CONSTRAINT Customer_NN_Customer_Name CHECK (Customer_Name IS NOT NULL)
);
```

**Inserting values into Customer Table (Naga Sai Lohitha Karmuru)**

```
INSERT INTO Customer VALUES(1,'John Smith','1234567890','M','john@gmail.com','32601','New York','NY');

INSERT INTO Customer VALUES(2,'Jane Johnson','9876543210','F','jane@gmail.com','75094','Los Angeles','CA');

INSERT INTO Customer VALUES(3,'Micheal Lee','4567890123','M','micheal@gmail.com','12209','Chicago','IL');

INSERT INTO Customer VALUES(4,'Sarah Brown','7890123456','F','sarah@gmail.com','07008','Houston','TX');

INSERT INTO Customer VALUES(5,'David Kim','3456789012','M','david@gmail.com','94206','San Francisco','CA');
```

```
INSERT INTO Customer VALUES(6,'Jessica Chen','9012345678','F','jessica@gmail.com','80514','Miami','FL');

INSERT INTO Customer VALUES(7,'Brian Johnson','6789012345','M','brian@gmail.com','97954','Seattle','WA');

INSERT INTO Customer VALUES(8,'Emily Davis','2345678901','F','emily@gmail.com','96915','Atlanta','GA');

INSERT INTO Customer VALUES(9,'Matthew Wilson','5678901234','M','matthew@gmail.com','34620','Dallas','TX');

INSERT INTO Customer VALUES(10,'Olivia Anderson','8901234567','F','olivia@gmail.com','34646','Boston','MA');

INSERT INTO Customer VALUES(11,'James Taylor','1232345644','M','james@gmail.com','07508','San Diego','CA');

INSERT INTO Customer VALUES(12,'Ava Martinez','1238799032','F','ava@gmail.com','49015','Philadelphia','PA');


INSERT INTO Customer VALUES(13,'Benjamin Lee','2512346788','M','benjamin@gmail.com','17013','Phoenix','AZ');

INSERT INTO Customer VALUES(14,'Mia Brown','4347897689','F','mia@gmail.com','96744','Denver','CO');

INSERT INTO Customer VALUES(15,'Ethan Kim','3467542345','M','ethan@gmail.com','84403','Portland','OR');
```

**Table Name: Customer Address (Sushmitha Dandu)**

```
DROP TABLE Customer_Address CASCADE CONSTRAINTS;


CREATE TABLE Customer_Address

(

Customer_Id VARCHAR2(20) NOT NULL,

Customer_City VARCHAR2(20),

Customer_State VARCHAR2(30),

Customer_Zip VARCHAR2(20),

CONSTRAINT Customer_AddressPK PRIMARY KEY (Customer_Id),

CONSTRAINT Customer_AddressFK FOREIGN KEY (Customer_Id) REFERENCES Customer(Customer_Id)

);
```

**Inserting values into Customer Address Table (Sushmitha Dandu)**

```
INSERT INTO Customer_Address VALUES('1',' New York', 'NY', '91011');
INSERT INTO Customer_Address VALUES('2','Los Angeles','CA', '75094');
INSERT INTO Customer_Address VALUES('3','Chicago', 'IL', '12209');
INSERT INTO Customer_Address VALUES('4','Houston', 'TX', '07008');
INSERT INTO Customer_Address VALUES('5','San Francisco', 'CA', '94206');
INSERT INTO Customer_Address VALUES('6','Miami', 'FL', '80514');
INSERT INTO Customer_Address VALUES('7','Seattle', 'WA', '97954');
INSERT INTO Customer_Address VALUES('8','Atlanta', 'GA', '96915');
INSERT INTO Customer_Address VALUES('9','Dallas', 'TX', '34620');
INSERT INTO Customer_Address VALUES('10','Boston','MA', '34646');
INSERT INTO Customer_Address VALUES('11','San Diego','CA', '07508');
INSERT INTO Customer_Address VALUES('12','Philadelphia','PA', '49015');
INSERT INTO Customer_Address VALUES('13','Phoenix','AZ', '91011');
INSERT INTO Customer_Address VALUES('14','Denver','CO', '96744');
INSERT INTO Customer_Address VALUES('15','Portland','OR', '84403');
```

**Table Name: Employee (Navyasree Sriramoju)**

```
DROP TABLE Employee CASCADE CONSTRAINTS;

CREATE TABLE Employee
(
Employee_ID NUMBER(5) NOT NULL,
Employee_Name VARCHAR(25),
Employee_Phone NUMBER(12) NOT NULL,
Employee_Email VARCHAR(25),
Employee_Designation VARCHAR(46),
Employee_gender VARCHAR(20),
Employee_Salary NUMBER(10) NOT NULL,
```

```
Employee_city VARCHAR(50),

Employee_state CHAR(2),

Employee_Zip VARCHAR(9),

CONSTRAINT Employee_PK PRIMARY KEY (Employee_ID),

CONSTRAINT Employee_NN_Employee_Name CHECK (Employee_Name IS NOT NULL)

);
```

**Inserting values into Employee Table (Navyasree Sriramoju)**

```
INSERT INTO Employee VALUES(101,'Joe Gellar',6557675557,'joe.g789@gmail.com','Manager','M',20000,'Pasadena','CA',91011);

INSERT INTO Employee VALUES(102,'Kat Pierce',8889990001,'Kat.p7256@gmail.com','Cashier','F',18000,'Pasadena','CA',91011);

INSERT INTO Employee VALUES(103,'Andrew Bong',6667773546,'Andrew.b6468@gmail.com','Salesman','M',15000,'Pasadena','CA',91011);

INSERT INTO Employee VALUES(104,'Neha Rao',7810002647,'Neha.r186@gmail.com','Salesman','F',12000,'Pasadena','CA',91011);

INSERT INTO Employee VALUES(201,'Harry Jones',7778537799,'Harry8647@gmail.com','Salesman','M',12000,'Pasadena','CA',91011);

INSERT INTO Employee VALUES(202,'Vera Moon',6567652577,'Vera.m1254@gmail.com','Salesman','F',14000,'Pasadena','CA',91011);

INSERT INTO Employee VALUES(203,'Yan Chang',2576547998,'Yan.C5432@gmail.com','Salesman','F',11000,'Pasadena','CA',91011);
```

**Table Name: Employee_Address (Sushmitha Dandu)**

```
DROP TABLE Employee_Address CASCADE CONSTRAINTS;

CREATE TABLE Employee_Address

(

Employee_ID NUMBER NOT NULL,

Employee_City VARCHAR2(50),

Employee_State CHAR(2),

Employee_Zip NUMBER(9),

CONSTRAINT Employee_AddressPK PRIMARY KEY (Employee_ID),
```

```
CONSTRAINT Employee_AddressFK FOREIGN KEY (Employee_ID) REFERENCES
Employee(Employee_ID)
);
```

**Inserting values into Employee Address Table (Sushmitha Dandu)**

```
INSERT INTO Employee_Address VALUES (101, 'Pasadena', 'CA', 91011);
INSERT INTO Employee_Address VALUES (102, 'Pasadena', 'CA', 91011);
INSERT INTO Employee_Address VALUES (103, 'Pasadena', 'CA', 91011);
INSERT INTO Employee_Address VALUES (104,'Pasadena', 'CA', 91011);
INSERT INTO Employee_Address VALUES (201,'Pasadena', 'CA', 91011);
INSERT INTO Employee_Address VALUES (202,'Pasadena', 'CA', 91011);
INSERT INTO Employee_Address VALUES (203,'Pasadena', 'CA', 91011);
INSERT INTO Employee_Address VALUES (204,'Pasadena', 'CA', 91011);
```

**Table Name: Product (Navyasree Sriramoju)**

```
DROP TABLE Product CASCADE CONSTRAINTS;
CREATE TABLE Product
(
Product_ID VARCHAR(10) NOT NULL,
Product_Name VARCHAR(30) NOT NULL,
Product_Type VARCHAR(50) NOT NULL,
Product_Price FLOAT,
Product_Color VARCHAR(15),
Product_Size Number(5),
Product_Warranty VARCHAR(30),
Product_Brand VARCHAR(30),
Product_Quantity VARCHAR(30) NOT NULL,
CONSTRAINT Product_ID_pk PRIMARY KEY (Product_ID)
);
```

**Inserting values into Product (Navyasree Sriramoju)**

```
INSERT INTO Product VALUES('M11','Sneakers','Manufactured',100,'White',7,'24months','Nike',300);
INSERT INTO Product VALUES('P22','Heels','Purchased',150,'Black',7.5,'6months','ALDO',150);
INSERT INTO Product VALUES('P33','HikingShoes','Purchased',220,'Red',6.5,'12months','Adidas',230);
INSERT INTO Product VALUES('M33','Flipflops','Manufactured',30,'Pink',6,'6months','Splash',200);
INSERT INTO Product VALUES('M55','SportShoes','Manufactured',60,'Orange',7,'18months','abc',180);
INSERT INTO Product VALUES('P55','Loafers','Purchased',70,'Brown',7.5,'12months','SteveMadden',250);
```

**Table Name: Purchased Product (Navyasree Sriramoju)**

```
DROP TABLE Purchased_Product CASCADE CONSTRAINTS;
CREATE TABLE Purchased_Product
(
PProduct_ID VARCHAR(30),
Supplier_Name VARCHAR(30) NOT NULL,
Supplier_Email VARCHAR(30),
Supplier_Phone NUMBER(10),
Supplier_City VARCHAR(20),
Supplier_State VARCHAR(2),
Supplier_Zip VARCHAR(9),
CONSTRAINT Purchased_Product_PK PRIMARY KEY (PProduct_ID),
CONSTRAINT Purchased_Product_FK FOREIGN KEY (PProduct_ID) REFERENCES Product(Product_ID)
);
```

**Inserting values into Purchased Product Table (Navyasree Sriramoju)**

```
INSERT INTO Purchased_Product VALUES('P22','New Balance','edmsupplies1242@gmail.com',6444677537,'Hartford','CT',6002);
INSERT INTO Purchased_Product VALUES('P33','NuSouce Inc','Fastenal6821@gmail.com',8566434668,'Dover','DE',19702);
INSERT INTO Purchased_Product VALUES('P55','NY Wholesale','nywhole4576@gmail.com',3797435678,'Atlanta','GA',30003);
```

**Table Name: Supplier Address (Sushmitha Dandu)**

```
DROP TABLE Supplier_Address CASCADE CONSTRAINTS;

CREATE TABLE Supplier_Address
(
PProduct_ID VARCHAR2(30) NOT NULL,
Supplier_City VARCHAR2(30),
Supplier_State VARCHAR2(20),
Supplier_Zip NUMBER(5) NOT NULL,
CONSTRAINT Supplier_AddressPK PRIMARY KEY (PProduct_ID),
CONSTRAINT Supplier_AddressFK FOREIGN KEY (PProduct_ID) REFERENCES Product(Product_ID)
);
```

**Inserting values into Supplier Address Table (Sushmitha Dandu)**

```
INSERT INTO Supplier_Address VALUES ('P22', 'Hartford', 'CT', 06002);
INSERT INTO Supplier_Address VALUES ('P33', 'Dover', 'DE', 19702);
INSERT INTO Supplier_Address VALUES ('P55', 'Atlanta', 'GA', 30003);
```

**Table Name: Manufactured Product (Navyasree Sriramoju)**

```
DROP TABLE Manufactured_Product CASCADE CONSTRAINTS;

CREATE TABLE Manufactured_Product
(
MFProduct_ID VARCHAR(30),
Manufacturer_Name VARCHAR(30) NOT NULL,
Manufacturer_State VARCHAR(2),
Manufacturer_City VARCHAR(20),
Manufacturer_Zip VARCHAR(9),
CONSTRAINT Manufactured_Product_PK PRIMARY KEY (MFProduct_ID),
CONSTRAINT Manufactured_Product_FK FOREIGN KEY (MFProduct_ID)  REFERENCES
Product(Product_ID)
);
```

**Inserting values into Manufactured Product Table (Navyasree Sriramoju)**

INSERT INTO Manufactured_Product VALUES('M11','ABCManfacturersLtd','CA','Pasadena',91011);

INSERT INTO Manufactured_Product VALUES('M33','ABCManfacturersLtd','CA','Pasadena',91011);

INSERT INTO Manufactured_Product VALUES('M55','ABCManfacturersLtd','CA','Pasadena',91011);

**Table Name: Manufacturer Address (Sushmitha Dandu)**

```
CREATE TABLE Manufacturer_Address
(
MFProduct_ID VARCHAR2(10) NOT NULL,
Manufacturer_City VARCHAR2(30),
Manufacturer_State VARCHAR2(20),
Manufacturer_Zip NUMBER(5) NOT NULL,
CONSTRAINT Manufacturer_AddressPK PRIMARY KEY (MFProduct_ID),
CONSTRAINT Manufacturer_AddressFK FOREIGN KEY (MFProduct_ID) REFERENCES
Product(Product_ID)
);
```

**Inserting values into Manufacturer Address Table (Sushmitha Dandu)**

INSERT INTO Manufacturer_Address VALUES ('M11',' Montgomery', 'AL', 35004);

INSERT INTO Manufacturer_Address VALUES ('M33', 'Phoenix', 'AZ', 85002);

INSERT INTO Manufacturer_Address VALUES ('M55','Sacramento', 'CA', 90002);

**Table Name: Orders (Naga Sai Lohitha Karmuru)**

```
DROP TABLE Orders CASCADE CONSTRAINTS;
CREATE TABLE Orders
(
Order_Id NUMBER NOT NULL,
Order_Price DECIMAL(10, 2),
Order_Date DATE,
Order_Status VARCHAR(20),
Shipping_Address VARCHAR(100),
```

Employee_ID NUMBER NOT NULL,

Customer_ID VARCHAR2(20) NOT NULL,

CONSTRAINT ORDER_PK PRIMARY KEY (Order_Id),

CONSTRAINT ORDER_FK1 FOREIGN KEY (Customer_ID) REFERENCES Customer(Customer_ID),

CONSTRAINT ORDER_FK2 FOREIGN KEY (Employee_ID) REFERENCES Employee(Employee_ID)

);

**Inserting values into Orders Table (Naga Sai Lohitha Karmuru)**

INSERT INTO ORDERS VALUES(1001, '150.99', '24 May 2022', 'Shipped', '1234 Elm St, NY', 101, 1);

INSERT INTO ORDERS VALUES(1002, '99.50', '25 May 2022', 'Delivered', '5678 Oak St, Los Angeles, CA',102, 2);

INSERT INTO ORDERS VALUES(1003, '200.00', '26 May 2022', 'Processing', '9101 Maple Ave, Chicago, IL',103, 3);

INSERT INTO ORDERS VALUES(1004, '75.25', '27 May 2022', 'Cancelled', '2468 Birch Rd, Houston, TX',104, 4);

INSERT INTO ORDERS VALUES(1005, '180.75', '3 June 2022', 'Shipped', '1357 Cedar Dr, Champaign, IL',103, 5);

INSERT INTO ORDERS VALUES(1006, '120.00', '15 June 2022', 'Delivered', '2468 Pine Ln, San Francisco, CA',102, 6);

INSERT INTO ORDERS VALUES(1007, '90.00', '27 June 2022', 'Processing', '7890 Willow Ct, Dallas, TX',202, 7);

INSERT INTO ORDERS VALUES(1008, '55.50', '9 July 2022', 'Shipped', '2345 Redwood Dr, Philadelphia, PA',102, 8);

INSERT INTO ORDERS VALUES(1009, '70.25', '20 July 2022', 'Delivered', '6789 Cedar Ct, Phoenix, AZ',201, 9);

INSERT INTO ORDERS VALUES(1010, '115.75', '15 August 2022', 'Shipped', '1234 Oakwood Ave, Peoria, IL',202, 10);

INSERT INTO ORDERS VALUES(1011, '200.50', '21 August 2022', 'Processing', '5678 Elmwood St, Denver, CO',203, 11);

INSERT INTO ORDERS VALUES(1012, '90.25', '28 August 2022', 'Cancelled', '9101 Maplewood Rd, Portland, OR',204, 12);

**Table Name: Order Line (Naga Sai Lohitha Karmuru)**

DROP TABLE OrderLine CASCADE CONSTRAINTS;

CREATE TABLE OrderLine

```
(
OrderLine_Id VARCHAR(10) NOT NULL,
Order_Id number NOT NULL,
Product_Id VARCHAR(10) NOT NULL,
Ordered_Quantity NUMBER,
CONSTRAINT OrderLine_pk PRIMARY KEY (OrderLine_Id),
CONSTRAINT OrderLine_Orders_fk FOREIGN KEY (Order_ID) REFERENCES Orders (Order_ID),
CONSTRAINT Orders_Product_ID_fk FOREIGN KEY (Product_ID) REFERENCES Product (Product_ID)
);
```

**Inserting values into Order LineTable (Naga Sai Lohitha Karmuru)**

```
INSERT INTO ORDERLINE VALUES('OL1', '1001', 'M11', 2);
INSERT INTO ORDERLINE VALUES('OL2', '1002', 'P22', 1);
INSERT INTO ORDERLINE VALUES('OL3', '1003', 'P33', 4);
INSERT INTO ORDERLINE VALUES('OL4', '1004', 'P55', 3);
INSERT INTO ORDERLINE VALUES('OL5', '1005', 'M33', 1);
INSERT INTO ORDERLINE VALUES('OL6', '1006', 'M55', 3);
INSERT INTO ORDERLINE VALUES('OL7', '1007', 'M11', 1);
INSERT INTO ORDERLINE VALUES('OL8', '1008', 'P55', 3);
INSERT INTO ORDERLINE VALUES('OL9', '1009', 'P22', 2);
INSERT INTO ORDERLINE VALUES('OL10', '1010', 'P33', 2);
INSERT INTO ORDERLINE VALUES('OL11', '1011', 'P55', 3);
INSERT INTO ORDERLINE VALUES('OL12', '1012', 'P22', 1);
```

## Describing Tables

Desc Customer; (Naga Sai Lohitha Karmuru)

```
Desc Customer;
```

Script Output | Task completed in 0.026 seconds

```
Name            Null?    Type
--------------- -------- -------------
CUSTOMER_ID     NOT NULL VARCHAR2(20)
CUSTOMER_NAME            VARCHAR2(25)
CUSTOMER_PHONE           CHAR(10)
CUSTOMER_GENDER          CHAR(20)
CUSTOMER_EMAIL           VARCHAR2(100)
CUSTOMER_ZIP             VARCHAR2(5)
CUSTOMER_CITY            VARCHAR2(50)
CUSTOMER_STATE           CHAR(2)
```

Desc Customer_Address; (Sushmitha Dandu)

```
DESC Customer_Address;
```

Script Output | Query Result | Task completed in 0.039 seconds

```
Name            Null?    Type
--------------- -------- ------------
CUSTOMER_ID     NOT NULL VARCHAR2(20)
CUSTOMER_CITY            VARCHAR2(20)
CUSTOMER_STATE           VARCHAR2(30)
CUSTOMER_ZIP             VARCHAR2(20)
```

Desc Employee; (Navyasree Sriramoju)

```
Desc Employee;
```

Script Output

Task completed in 0.107 seconds

```
1 row inserted.


1 row inserted.


1 row inserted.

Name                   Null?    Type
---------------------- -------- ------------
EMPLOYEE_ID            NOT NULL NUMBER
EMPLOYEE_NAME                   VARCHAR2(25)
EMPLOYEE_PHONE         NOT NULL NUMBER
EMPLOYEE_EMAIL                  VARCHAR2(25)
EMPLOYEE_DESIGNATION            VARCHAR2(46)
EMPLOYEE_GENDER                 VARCHAR2(10)
EMPLOYEE_SALARY        NOT NULL NUMBER
EMPLOYEE_CITY                   VARCHAR2(50)
EMPLOYEE_STATE                  CHAR(2)
EMPLOYEE_ZIP                    VARCHAR2(9)
```

Desc Employee_Address; (Sushmitha Dandu)

```
desc Employee_Address;
```

Script Output

Task completed in 0.322 seconds

```
Name            Null?    Type
--------------- -------- ------------
EMPLOYEE_ID     NOT NULL NUMBER
EMPLOYEE_CITY            VARCHAR2(50)
EMPLOYEE_STATE           CHAR(2)
EMPLOYEE_ZIP             NUMBER(9)
```

DESC PRODUCT; (Navyasree Sriramoju)

```
DESC PRODUCT;
```

Script Output | Query Result

Task completed in 0.323 seconds

```
Name               Null?    Type
---------------- -------- ------------
PRODUCT_ID         NOT NULL VARCHAR2(10)
PRODUCT_NAME       NOT NULL VARCHAR2(30)
PRODUCT_TYPE       NOT NULL VARCHAR2(50)
PRODUCT_PRICE               FLOAT(126)
PRODUCT_COLOR               VARCHAR2(15)
PRODUCT_SIZE                NUMBER(5)
PRODUCT_WARRANTY            VARCHAR2(30)
PRODUCT_BRAND               VARCHAR2(30)
PRODUCT_QUANTITY NOT NULL VARCHAR2(30)
```

Desc Purchased_Product; (Navyasree Sriramoju)

```
Desc Purchased_Product;

Script Output   Query Result
Task completed in 0.03 seconds

Name           Null?    Type
-------------- -------- ------------
PPRODUCT_ID    NOT NULL VARCHAR2(30)
SUPPLIER_NAME  NOT NULL VARCHAR2(30)
SUPPLIER_EMAIL          VARCHAR2(30)
SUPPLIER_PHONE          NUMBER(10)
SUPPLIER_CITY           VARCHAR2(20)
SUPPLIER_STATE          VARCHAR2(2)
SUPPLIER_ZIP            VARCHAR2(9)
```

Desc Supplier_Address; (Sushmitha Dandu)

```
desc supplier_address;

Script Output
Task completed in 0.318 seconds
P33                                Dover
P55                                Atlanta

Name           Null?    Type
-------------- -------- ------------
PPRODUCT_ID    NOT NULL VARCHAR2(30)
SUPPLIER_CITY           VARCHAR2(30)
SUPPLIER_STATE          VARCHAR2(20)
SUPPLIER_ZIP   NOT NULL NUMBER(5)
```

Desc Manufactured_Product; (Navyasree Sriramoju)

```
Desc Manufactured_Product;
```

Script Output | Query Result

Task completed in 0.044 seconds

```
Name                 Null?    Type
-------------------- -------- ------------
MFPRODUCT_ID         NOT NULL VARCHAR2(30)
MANUFACTURER_NAME    NOT NULL VARCHAR2(30)
MANUFACTURER_STATE            VARCHAR2(2)
MANUFACTURER_CITY             VARCHAR2(20)
MANUFACTURER_ZIP              VARCHAR2(9)
```

Desc Manufacturer_Address; (Sushmitha Dandu)

```
desc Manufacturer_Address;
```

Script Output

Task completed in 0.329 seconds

```
M33         Phoenix                          AZ
M55         Sacramento                       CA

Name                 Null?    Type
-------------------- -------- ------------
MFPRODUCT_ID         NOT NULL VARCHAR2(10)
MANUFACTURER_CITY             VARCHAR2(30)
MANUFACTURER_STATE            VARCHAR2(20)
MANUFACTURER_ZIP     NOT NULL NUMBER(5)
```

Desc ORDERS; (Naga Sai Lohitha Karmuru)

```
desc ORDERS;
```

Script Output | Query Result

Task completed in 0.325 seconds

```
12 rows selected.

Name                 Null?    Type
---------------- -------- -------------
ORDER_ID             NOT NULL NUMBER
ORDER_PRICE                   NUMBER(10,2)
ORDER_DATE                    DATE
ORDER_STATUS                  VARCHAR2(20)
SHIPPING_ADDRESS              VARCHAR2(100)
EMPLOYEE_ID          NOT NULL NUMBER
CUSTOMER_ID          NOT NULL VARCHAR2(20)
```

DESC ORDERLINE; (Naga Sai Lohitha Karmuru)

```
DESC ORDERLINE;
```

Script Output

Task completed in 0.32 seconds

```
12 rows selected.

Name                 Null?    Type
---------------- -------- ------------
ORDERLINE_ID         NOT NULL VARCHAR2(10)
ORDER_ID             NOT NULL NUMBER
PRODUCT_ID           NOT NULL VARCHAR2(10)
ORDERED_QUANTITY              NUMBER
```

## Selecting All from Tables:

SELECT * FROM Customer; (Naga Sai Lohitha Karmuru)

```
select * from Customer;
```

Script Output × Query Result ×

SQL | All Rows Fetched: 15 in 0.032 seconds

| | CUSTOMER_ID | CUSTOMER_NAME | CUSTOMER_PHONE | CUSTOMER_GENDER | CUSTOMER_EMAIL | CUSTOMER_ZIP | CUSTOMER_CITY | CUSTOMER_STATE |
|---|---|---|---|---|---|---|---|---|
| 1 | 1 | John Smith | 1234567890 | M | john@gmail.com | 32601 | New York | NY |
| 2 | 2 | Jane Johnson | 9876543210 | F | jane@gmail.com | 75094 | Los Angeles | CA |
| 3 | 3 | Micheal Lee | 4567890123 | M | micheal@gmail.com | 12209 | Chicago | IL |
| 4 | 4 | Sarah Brown | 7890123456 | F | sarah@gmail.com | 07008 | Houston | TX |
| 5 | 5 | David Kim | 3456789012 | M | david@gmail.com | 94206 | San Francisco | CA |
| 6 | 6 | Jessica Chen | 9012345678 | F | jessica@gmail.com | 80514 | Miami | FL |
| 7 | 7 | Brian Johnson | 6789012345 | M | brian@gmail.com | 97954 | Seattle | WA |
| 8 | 8 | Emily Davis | 2345678901 | F | emily@gmail.com | 96915 | Atlanta | GA |
| 9 | 9 | Matthew Wilson | 5678901234 | M | matthew@gmail.com | 34620 | Dallas | TX |
| 10 | 10 | Olivia Anderson | 8901234567 | F | olivia@gmail.com | 34646 | Boston | MA |
| 11 | 11 | James Taylor | 1232345644 | M | james@gmail.com | 07508 | San Diego | CA |
| 12 | 12 | Ava Martinez | 1238799032 | F | ava@gmail.com | 49015 | Philadelphia | PA |
| 13 | 13 | Benjamin Lee | 2512346788 | M | benjamin@gmail.com | 17013 | Phoenix | AZ |
| 14 | 14 | Mia Brown | 4347897689 | F | mia@gmail.com | 96744 | Denver | CO |
| 15 | 15 | Ethan Kim | 3467542345 | M | ethan@gmail.com | 84403 | Portland | OR |

SELECT * FROM Customer_Address; (Sushmitha Dandu)

```
SELECT * FROM Customer_Address;
```

Script Output × Query Result ×

SQL | All Rows Fetched: 15 in 0.046 seconds

| | CUSTOMER_ID | CUSTOMER_CITY | CUSTOMER_STATE | CUSTOMER_ZIP |
|---|---|---|---|---|
| 1 | 1 | New York | NY | 91011 |
| 2 | 2 | Los Angeles | CA | 75094 |
| 3 | 3 | Chicago | IL | 12209 |
| 4 | 4 | Houston | TX | 7008 |
| 5 | 5 | San Francisco | CA | 94206 |
| 6 | 6 | Miami | FL | 80514 |
| 7 | 7 | Seattle | WA | 97954 |
| 8 | 8 | Atlanta | GA | 96915 |
| 9 | 9 | Dallas | TX | 34620 |
| 10 | 10 | Boston | MA | 34646 |
| 11 | 11 | San Diego | CA | 7508 |
| 12 | 12 | Philadelphia | PA | 49015 |
| 13 | 13 | Phoenix | AZ | 91011 |
| 14 | 14 | Denver | CO | 96744 |
| 15 | 15 | Portland | OR | 84403 |

SELECT * FROM Employee; (Navyasree Sriramoju)

```
SELECT * FROM Employee;

Script Output
Task completed in 0.023 seconds

EMPLOYEE_ID EMPLOYEE_NAME          EMPLOYEE_PHONE EMPLOYEE_EMAIL          EMPLOYEE_DESIGNATION                        EMPLOYEE_G EMPLOYEE_
----------- ---------------------- -------------- ---------------------- ------------------------------------------ ---------- --------
        102 Kat Pierce                 8889990001 Kat.p7256@gmail.com    Cashier                                    F
        103 Andrew Bong                6667773546 Andrew.b6468@gmail.com Salesman                                   M
        104 Neha Rao                   7810002647 Neha.r186@gmail.com    Salesman                                   F
        201 Harry Jones                7778537799 Harry8647@gmail.com    Salesman                                   M
        202 Vera Moon                  6567652577 Vera.m1254@gmail.com   Salesman                                   F
        203 Yan Chang                  2576547998 Yan.C5432@gmail.com    Salesman                                   F
        204 Shyam Singh                6748764676 Shyam.s6371@gmail.com  Salesman                                   M
        101 Joe Gellar                 6557675557 joe.g789@gmail.com     Manager                                    M

8 rows selected.
```

SELECT * FROM Employee_Address; (Sushmitha Dandu)

```
select * from Employee_Address;

Script Output
Task completed in 0.023 seconds

1 row inserted.

EMPLOYEE_ID EMPLOYEE_CITY                                      EM EMPLOYEE_ZIP
----------- ---------------------------------------------------- -- ------------
        101  Pasadena                                           CA        91011
        102 Pasadena                                            CA        91011
        103 Pasadena                                            CA        91011
        104 Pasadena                                            CA        91011
        201 Pasadena                                            CA        91011
        202 Pasadena                                            CA        91011
        203 Pasadena                                            CA        91011
        204 Pasadena                                            CA        91011

8 rows selected.
```

SELECT * FROM Product; (Navyasree Sriramoju)

```
select * from PRODUCT;

Script Output   Query Result
Task completed in 0.018 seconds

PRODUCT_ID PRODUCT_NAME                PRODUCT_TYPE                                   PRODUCT_PRICE PRODUCT_COLOR   PRODUCT_SIZE PRODUCT_WA
---------- ----------------------------- ------------------------------------------ ------------- -------------- ------------ ---------
M11        Sneakers                    Manufactured                                           100 White                    7 24months
P22        Heels                       Purchased                                              150 Black                    8 6months
P33        HikingShoes                 Purchased                                              220 Red                      7 12months
M33        Flipflops                   Manufactured                                            30 Pink                     6 6months
M55        SportShoes                  Manufactured                                            60 Orange                   7 18months
P55        Loafers                     Purchased                                               70 Brown                    8 12months

6 rows selected.
```

SELECT * FROM Purchased_Product; (Navyasree Sriramoju)

```
SELECT * FROM Purchased_Product;
```

Script Output | Query Result

All Rows Fetched: 3 in 0.002 seconds

| | PPRODUCT_ID | SUPPLIER_NAME | SUPPLIER_EMAIL | SUPPLIER_PHONE | SUPPLIER_CITY | SUPPLIER_STATE | SUPPLIER_ZIP |
|---|---|---|---|---|---|---|---|
| 1 | P22 | New Balance | edmsupplies1242@gmail.com | 6444677537 | Hartford | CT | 6002 |
| 2 | P33 | NuSouce Inc | Fastenal6821@gmail.com | 8566434668 | Dover | DE | 19702 |
| 3 | P55 | NY Wholesale | nywhole4576@gmail.com | 3797435678 | Atlanta | GA | 30003 |

SELECT * FROM Supplier_Address; (Sushmitha Dandu)

```
select * from supplier_address;
```

Script Output

Task completed in 0.02 seconds

```
1 row inserted.


PPRODUCT_ID                    SUPPLIER_CITY                  SUPPLIER_STATE       SUPPLIER_ZIP
------------------------------ ------------------------------ -------------------- ------------
P22                            Hartford                       CT                           6002
P33                            Dover                          DE                          19702
P55                            Atlanta                        GA                          30003
```

SELECT * FROM Manufactured_Product; (Navyasree Sriramoju)

```
SELECT * FROM Manufactured_Product;
```

Script Output | Query Result

All Rows Fetched: 3 in 0.002 seconds

| | MFPRODUCT_ID | MANUFACTURER_NAME | MANUFACTURER_STATE | MANUFACTURER_CITY | MANUFACTURER_ZIP |
|---|---|---|---|---|---|
| 1 | M11 | ABCManfacturersLtd | CA | Pasadena | 91011 |
| 2 | M33 | ABCManfacturersLtd | CA | Pasadena | 91011 |
| 3 | M55 | ABCManfacturersLtd | CA | Pasadena | 91011 |

SELECT * FROM Manufacturer_Address; (Sushmitha Dandu)

```
select * from Manufacturer_Address;

Script Output
Task completed in 0.018 seconds
1 row inserted.


MFPRODUCT_ MANUFACTURER_CITY              MANUFACTURER_STATE   MANUFACTURER_ZIP
---------- ------------------------------ -------------------- ----------------
M11         Montgomery                    AL                              35004
M33        Phoenix                        AZ                              85002
M55        Sacramento                     CA                              90002
```

SELECT * FROM Orders; (Naga Sai Lohitha Karmuru)

```
SQL Worksheet  History
0.035 seconds
Worksheet  Query Builder

select * from ORDERS;

Script Output  Query Result
Task completed in 0.035 seconds


 ORDER_ID ORDER_PRICE ORDER_DAT ORDER_STATUS         SHIPPING_ADDRESS
---------- ----------- --------- -------------------- ------------------------------------------------------------
     1001      150.99 24-MAY-22 Shipped              1234 Elm St, NY
     1002        99.5 25-MAY-22 Delivered            5678 Oak St, Los Angeles, CA
     1003         200 26-MAY-22 Processing           9101 Maple Ave, Chicago, IL
     1004       75.25 27-MAY-22 Cancelled            2468 Birch Rd, Houston, TX
     1005      180.75 03-JUN-22 Shipped              1357 Cedar Dr, Champaign, IL
     1006         120 15-JUN-22 Delivered            2468 Pine Ln, San Francisco, CA
     1007          90 27-JUN-22 Processing           7890 Willow Ct, Dallas, TX
     1008        55.5 09-JUL-22 Shipped              2345 Redwood Dr, Philadelphia, PA
     1009       70.25 20-JUL-22 Delivered            6789 Cedar Ct, Phoenix, AZ
     1010      115.75 15-AUG-22 Shipped              1234 Oakwood Ave, Peoria, IL
     1011       200.5 21-AUG-22 Processing           5678 Elmwood St, Denver, CO

 ORDER_ID ORDER_PRICE ORDER_DAT ORDER_STATUS         SHIPPING_ADDRESS
---------- ----------- --------- -------------------- ------------------------------------------------------------
     1012       90.25 28-AUG-22 Cancelled            9101 Maplewood Rd, Portland, OR
```

SELECT * FROM OrderLine; (Naga Sai Lohitha Karmuru)

```
select * from ORDERLINE;

Script Output
Task completed in 0.025 seconds

ORDERLINE_   ORDER_ID PRODUCT_ID ORDERED_QUANTITY
---------- ---------- ---------- ----------------
OL2              1002 P22                       1
OL4              1004 P55                       3
OL5              1005 P33                       1
OL6              1006 P22                       3
OL8              1008 P55                       3
OL10             1010 P33                       2
OL1              1001 M11                       2
OL3              1003 P33                       4
OL7              1007 M11                       1
OL9              1009 P22                       2
OL11             1011 P55                       3

ORDERLINE_   ORDER_ID PRODUCT_ID ORDERED_QUANTITY
---------- ---------- ---------- ----------------
OL12             1012 P22                       1
```

## Performing Insert, Update, Delete, Create Views

### INSERT values:

### Inserting records to Employee Table (Naga Sai Lohitha Karmuru)

INSERT INTO Employee
VALUES(105,'Messie',8889992345,'Messie.123@gmail.com','Cashier','M',28000,'Pasadena','CA',91011);

INSERT INTO Employee
VALUES(205,'Christina',6667777645,'Christina.46@gmail.com','Salesman','F',17000,'Pasadena','CA',91011);

```
select * from employee;

Script Output
Task completed in 0.023 seconds
1 row inserted.

EMPLOYEE_ID EMPLOYEE_NAME              EMPLOYEE_PHONE EMPLOYEE_EMAIL            EMPLOYEE_DESIGNATION                          EMPLOYEE_G EMPLOYEE_
----------- ------------------------- -------------- ------------------------- --------------------------------------------- ---------- ---------
        101 Joe Gellar                     6557675557 joe.g789@gmail.com        Manager                                       M
        102 Kat Pierce                     8889990001 Kat.p7256@gmail.com       Cashier                                       F
        103 Andrew Bong                    6667773546 Andrew.b6468@gmail.com    Salesman                                      M
        104 Neha Rao                       7810002647 Neha.r186@gmail.com       Salesman                                      F
        201 Harry Jones                    7778537799 Harry8647@gmail.com       Salesman                                      M
        202 Vera Moon                      6567652577 Vera.m1254@gmail.com      Salesman                                      F
        203 Yan Chang                      2576547998 Yan.C5432@gmail.com       Salesman                                      F
        204 Shyam Singh                    6748764676 Shyam.s6371@gmail.com     Salesman                                      M
        105 Messie                         8889992345 Messie.123@gmail.com      Cashier                                       M
        205 Christina                      6667777645 Christina.46@gmail.com    Salesman                                      F

10 rows selected.
```

### Inserting records to Customer Table (Sushmitha Dandu)

INSERT INTO Customer VALUES(16,'Mark','5762348900','M','Mark@gmail.com','32691','Dallas','TX');
INSERT INTO Customer
VALUES(17,'Marie','7768148900','F','Marie222@gmail.com','34311','Sacramento','CA');

Script Output | Query Result
All Rows Fetched: 17 in 0.044 seconds

| | CUSTOMER_ID | CUSTOMER_NAME | CUSTOMER_PHONE | CUSTOMER_GENDER | CUSTOMER_EMAIL | CUSTOMER_ZIP | CUSTOMER_CITY | CUSTOMER_STATE |
|---|---|---|---|---|---|---|---|---|
| 1 | 1 | John Smith | 1234567890 | M | john@gmail.com | 32601 | New York | NY |
| 2 | 2 | Jane Johnson | 9876543210 | F | jane@gmail.com | 75094 | Los Angeles | CA |
| 3 | 3 | Micheal Lee | 4567890123 | M | micheal@gmail.com | 12209 | Chicago | IL |
| 4 | 4 | Sarah Brown | 7890123456 | F | sarah@gmail.com | 07008 | Houston | TX |
| 5 | 5 | David Kim | 3456789012 | M | david@gmail.com | 94206 | San Francisco | CA |
| 6 | 6 | Jessica Chen | 9012345678 | F | jessica@gmail.com | 80514 | Miami | FL |
| 7 | 7 | Brian Johnson | 6789012345 | M | brian@gmail.com | 97954 | Seattle | WA |
| 8 | 8 | Emily Davis | 2345678901 | F | emily@gmail.com | 96915 | Atlanta | GA |
| 9 | 9 | Matthew Wilson | 5678901234 | M | matthew@gmail.com | 34620 | Dallas | TX |
| 10 | 10 | Olivia Anderson | 8901234567 | F | olivia@gmail.com | 34646 | Boston | MA |
| 11 | 11 | James Taylor | 1232345644 | M | james@gmail.com | 07508 | San Diego | CA |
| 12 | 12 | Ava Martinez | 1238799032 | F | ava@gmail.com | 49015 | Philadelphia | PA |
| 13 | 13 | Benjamin Lee | 2512346788 | M | benjamin@gmail.com | 17013 | Phoenix | AZ |
| 14 | 14 | Mia Brown | 4347897689 | F | mia@gmail.com | 96744 | Denver | CO |
| 15 | 15 | Ethan Kim | 3467542345 | M | ethan@gmail.com | 84403 | Portland | OR |
| 16 | 16 | Mark | 5762348900 | M | Mark@gmail.com | 32691 | Dallas | TX |
| 17 | 17 | Marie | 7768148900 | F | Marie222@gmail.com | 34311 | Sacramento | CA |

**Inserting records to Product Table (Navyasree Sriramoju)**

INSERT INTO Product VALUES('P11','Jordans','Purchased',120,'Green',7,'24 months','Nike',100);
INSERT INTO Product VALUES('M22','Office Shoes','Manufactured',80,'Cream',7.5,'12 months','xyz',80);

```
select * from product;
```

Script Output | Task completed in 0.019 seconds

```
1 row inserted.

PRODUCT_ID PRODUCT_NAME                   PRODUCT_TYPE                                       PRODUCT_PRICE PRODUCT_COLOR   PRODUCT_SIZE PRODUCT_WA
---------- ------------------------------ -------------------------------------------------- ------------- --------------- ------------ ----------
M11        Sneakers                       Manufactured                                                 100 White                        7 24months
P22        Heels                          Purchased                                                    150 Black                        8 6months
P33        HikingShoes                    Purchased                                                    220 Red                          7 12months
M33        Flipflops                      Manufactured                                                  30 Pink                         6 6months
M55        SportShoes                     Manufactured                                                  60 Orange                       7 18months
P55        Loafers                        Purchased                                                     70 Brown                        8 12months
P11        Jordans                        Purchased                                                    120 Green                        7 24 months
M22        Office Shoes                   Manufactured                                                  80 Cream                        8 12 months

8 rows selected.
```

**UPDATE values: (Navyasree Sriramoju)**

Updating Product_Name from Flipflops to Wedges in Product Table.

Change the Product FlipFlops to Wedges

UPDATE product

SET Product_Name = 'Wedges'

WHERE Product_Name = 'Flipflops';

```
Change the Product FlipFlops to Wedges
UPDATE product
SET Product_Name = 'Wedges'
WHERE Product_Name = 'Flipflops';

select * from product;
```

Script Output | Task completed in 0.02 seconds

```
1 row updated.

PRODUCT_ID PRODUCT_NAME                   PRODUCT_TYPE                                       PRODUCT_PRICE PRODUCT_COLOR   PRODUCT_SIZE PRODUCT_WA
---------- ------------------------------ -------------------------------------------------- ------------- --------------- ------------ ----------
M11        Sneakers                       Manufactured                                                 100 White                        7 24months
P22        Heels                          Purchased                                                    150 Black                        8 6months
P33        HikingShoes                    Purchased                                                    220 Red                          7 12months
M33        Wedges                         Manufactured                                                  30 Pink                         6 6months
M55        SportShoes                     Manufactured                                                  60 Orange                       7 18months
P55        Loafers                        Purchased                                                     70 Brown                        8 12months
P11        Jordans                        Purchased                                                    120 Green                        7 24 months
M22        Office Shoes                   Manufactured                                                  80 Cream                        8 12 months

8 rows selected.
```

**DELETE values: (Naga Sai Lohitha Karmuru)**

Deleting OL2 values from OrderLine Table.

DELETE from OrderLine where OrderLine_ID = 'OL2';

```
DELETE from OrderLine where OrderLine_ID = 'OL2';

select * from OrderLine;
```

Script Output

Task completed in 0.025 seconds

```
1 row deleted.


ORDERLINE_   ORDER_ID PRODUCT_ID ORDERED_QUANTITY
---------- ---------- ---------- ----------------
OL4              1004 P55                       3
OL5              1005 P33                       1
OL6              1006 P22                       3
OL8              1008 P55                       3
OL10             1010 P33                       2
OL1              1001 M11                       2
OL3              1003 P33                       4
OL7              1007 M11                       1
OL9              1009 P22                       2
OL11             1011 P55                       3
OL12             1012 P22                       1
```

## CREATE VIEW: (Sushmitha Dandu)

**Creating a view for Customer in states California and Texas:**

CREATE OR REPLACE VIEW CA_TX_Vu AS SELECT customer_name, customer_state FROM customer WHERE customer_state IN ('CA', 'TX');

SELECT * FROM CA_TX_Vu;

```
CREATE OR REPLACE VIEW CA_TX_Vu AS SELECT customer_name, customer_state FROM customer WHERE customer_state IN ('CA', 'TX') ;

SELECT * FROM CA_TX_Vu;
```

Script Output

Task completed in 0.024 seconds

```
CUSTOMER_NAME              CU
-------------------------- --
Jane Johnson               CA
Sarah Brown                TX
David Kim                  CA
Matthew Wilson             TX
James Taylor               CA
Mark                       TX
Marie                      CA

7 rows selected.
```

Testing Database with (Select, join, where, group by, having) Queries

**What are the product IDs, names, and total counts of products manufactured by Nike, based on the Manufactured_Product and Product tables? (Navyasree Sriramoju)**

SELECT

Product_Id,

Product_Name,

COUNT(*) as Product_count

FROM

Manufactured_Product

INNER JOIN

Product

ON

Manufactured_Product.MFProduct_ID = Product.Product_ID

GROUP BY

Product_Id,

Product_Name

HAVING

MAX(Product_Brand) = 'Nike';

```
SELECT
    Product_Id,
    Product_Name,
    COUNT(*) as Product_count
FROM
    Manufactured_Product
INNER JOIN
    Product
ON
    Manufactured_Product.MFProduct_ID = Product.Product_ID
GROUP BY
    Product_Id,
    Product_Name
HAVING
    MAX(Product_Brand) = 'Nike';
```

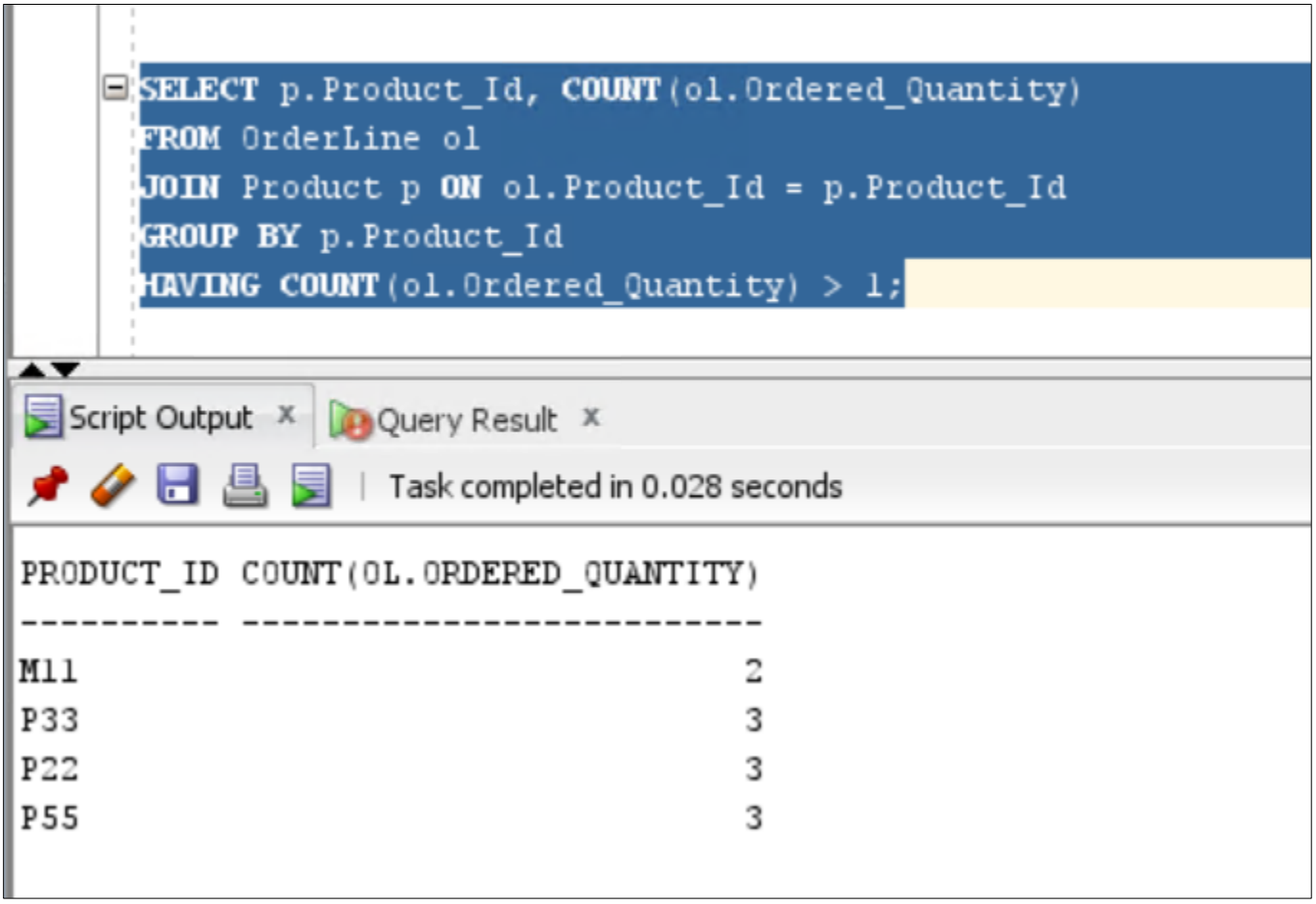
Script Output x  Query Result x

Task completed in 0.016 seconds

```
PRODUCT_ID PRODUCT_NAME                    PRODUCT_COUNT
---------- ------------------------------ -------------
M11        Sneakers                                   1
```

**Display Product IDs with more than 1 order.** **(Sushmitha Dandu)**

```
SELECT p.Product_Id, COUNT(ol.Ordered_Quantity)
FROM OrderLine ol
JOIN Product p ON ol.Product_Id = p.Product_Id
GROUP BY p.Product_Id
HAVING COUNT(ol.Ordered_Quantity) > 1;
```



**What is the list of male employees and their contact information along with their corresponding addresses?**
**(Naga Sai Lohitha Karmuru)**

```
SELECT
   E.EMPLOYEE_NAME,
   E.EMPLOYEE_PHONE,
   E.EMPLOYEE_DESIGNATION,
   E.EMPLOYEE_STATE,
   E.EMPLOYEE_CITY,
   E.EMPLOYEE_ZIP
FROM
   EMPLOYEE E
RIGHT JOIN
   EMPLOYEE_ADDRESS EA ON E.EMPLOYEE_ID = EA.EMPLOYEE_ID
WHERE
   E.EMPLOYEE_GENDER = 'M'
GROUP BY
   E.EMPLOYEE_NAME,
   E.EMPLOYEE_PHONE,
   E.EMPLOYEE_DESIGNATION,
   E.EMPLOYEE_STATE,
   E.EMPLOYEE_CITY,
   E.EMPLOYEE_ZIP
HAVING
   COUNT(*) > 0;
```

```
RIGHT JOIN
    EMPLOYEE_ADDRESS EA ON E.EMPLOYEE_ID = EA.EMPLOYEE_ID
WHERE
    E.EMPLOYEE_GENDER = 'M'
GROUP BY
    E.EMPLOYEE_NAME,
    E.EMPLOYEE_PHONE,
    E.EMPLOYEE_DESIGNATION,
    E.EMPLOYEE_STATE,
    E.EMPLOYEE_CITY,
    E.EMPLOYEE_ZIP
HAVING
    COUNT(*) > 0;
```

Script Output | Query Result

Task completed in 0.025 seconds

| EMPLOYEE_NAME | EMPLOYEE_PHONE | EMPLOYEE_DESIGNATION | EM | EMPLOYEE_CITY | EMPL |
|---|---|---|---|---|---|
| Shyam Singh | 6748764676 | Salesman | CA | Pasadena | 9101 |
| Joe Gellar | 6557675557 | Manager | CA | Pasadena | 9101 |
| Andrew Bong | 6667773546 | Salesman | CA | Pasadena | 9101 |
| Harry Jones | 7778537799 | Salesman | CA | Pasadena | 9101 |

## PL/SQL - Procedural Language extension to Structured Query Language

**Raise Employee Salary by 5%** **(Navyasree Sriramoju)**

```
CREATE OR REPLACE PROCEDURE raise_Employee_Salary
  (v_id in Employee.Employee_ID%type)
  IS
  BEGIN
  UPDATE Employee
  SET Employee_Salary = Employee_Salary*1.05
  WHERE Employee_ID = v_id;
  END raise_Employee_Salary;
```

```
CREATE OR REPLACE PROCEDURE raise_Employee_Salary
   (v_id in Employee.Employee_ID%type)
   IS
   BEGIN
   UPDATE Employee
   SET Employee_Salary = Employee_Salary*1.05
   WHERE Employee_ID = v_id;
   END raise_Employee_Salary;

EXECUTE raise_Employee_Salary(105)

select * from Employee
WHERE employee_ID = 105;
```

Script Output | Query Result

Task completed in 0.019 seconds

```
Procedure RAISE_EMPLOYEE_SALARY compiled
```

EXECUTE raise_Employee_Salary(105)

```
EXECUTE raise_Employee_Salary(105)
```

Script Output | Query Result

Task completed in 0.029 seconds

```
PL/SQL procedure successfully completed.
```

select * from Employee

WHERE employee_ID = 105;

```
select * from Employee
WHERE employee_ID = 105;
```

Script Output | Query Result

Task completed in 0.025 seconds

```
        EMPLOYEE_DESIGNATION                          EMPLOYEE_G EMPLOYEE_SALARY EMPLOYEE_CITY                                      EM EMPLOYEE_
-------- ---------------------------------------------- ---------- --------------- ------------------------------------------------ -- ---------
com     Cashier                                       M                    29400 Pasadena                                         CA 91011
```

**Create a pl/sql block to find the total number of customers in New York (Naga Sai Lohitha Karmuru)**

```
CREATE OR REPLACE PROCEDURE customer_count
AS
  v_count NUMBER;
BEGIN
SELECT COUNT(*) INTO v_count
FROM Customer c JOIN Customer_Address ca ON c.Customer_Id = ca.Customer_Id
WHERE ca.Customer_City = 'Chicago';
DBMS_OUTPUT.PUT_LINE('The number of customers in Chicago is: ' || v_count);
END;
/
```

```
CREATE OR REPLACE PROCEDURE customer_count
AS
  v_count NUMBER;
BEGIN
SELECT COUNT(*) INTO v_count
FROM Customer c JOIN Customer_Address ca ON c.Customer_Id = ca.Customer_Id
WHERE ca.Customer_City = 'Chicago';
DBMS_OUTPUT.PUT_LINE('The number of customers in Chicago is: ' || v_count);
END;
/
```

Script Output | Query Result

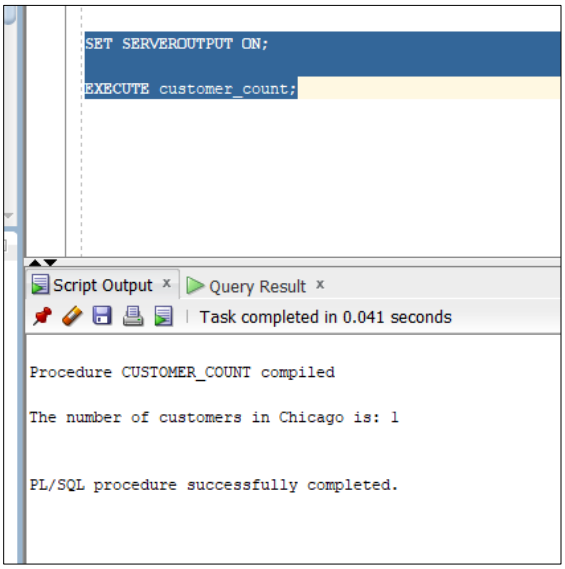
Task completed in 0.041 seconds

```
Procedure CUSTOMER_COUNT compiled
```

**Calling the procedure**

SET SERVEROUTPUT ON;

EXECUTE customer_count;




**Display the total revenue generated by a specific order (Sushmitha Dandu)**

```
CREATE OR REPLACE PROCEDURE order_revenue(v_order_id NUMBER)
AS
  v_revenue NUMBER;
BEGIN
  SELECT SUM(Product_Price * Ordered_Quantity) INTO v_revenue FROM OrderLine ol JOIN Product p ON ol.Product_Id = p.Product_Id WHERE ol.Order_Id = v_order_id;
  IF v_revenue IS NULL THEN
    DBMS_OUTPUT.PUT_LINE('No revenue found for order ID ' || v_order_id);
  ELSE
    DBMS_OUTPUT.PUT_LINE('The total revenue generated by Order ID ' || v_order_id || ' is: ' || v_revenue);
  END IF;
END;
/
```

```
CREATE OR REPLACE PROCEDURE order_revenue(v_order_id NUMBER)
AS
  v_revenue NUMBER;
BEGIN
  SELECT SUM(Product_Price * Ordered_Quantity) INTO v_revenue FROM OrderLine ol JOIN Product p ON ol.Product_Id = p.Product_Id WHERE ol.Order_Id = v_order_id;
  IF v_revenue IS NULL THEN
    DBMS_OUTPUT.PUT_LINE('No revenue found for order ID ' || v_order_id);
  ELSE
    DBMS_OUTPUT.PUT_LINE('The total revenue generated by Order ID ' || v_order_id || ' is: ' || v_revenue);
  END IF;
END;
/
```

Script Output | Query Result

Task completed in 0.04 seconds

```
Procedure ORDER_REVENUE compiled
```

SET SERVEROUTPUT ON;

EXECUTE order_revenue(1001);

```
-- Display the total revenue generated by a specific order
CREATE OR REPLACE PROCEDURE order_revenue(v_order_id NUMBER)
AS
  v_revenue NUMBER;
BEGIN
  SELECT SUM(Product_Price * Ordered_Quantity) INTO v_revenue FROM OrderLine ol JOIN Product p ON ol.Product_Id = p.Product_Id WHERE ol.Order_Id = v_order_id;
  IF v_revenue IS NULL THEN
    DBMS_OUTPUT.PUT_LINE('No revenue found for order ID ' || v_order_id);
  ELSE
    DBMS_OUTPUT.PUT_LINE('The total revenue generated by Order ID ' || v_order_id || ' is: ' || v_revenue);
  END IF;
END;
/


SET SERVEROUTPUT ON;
EXECUTE order_revenue(1001);
```

Script Output | Query Result

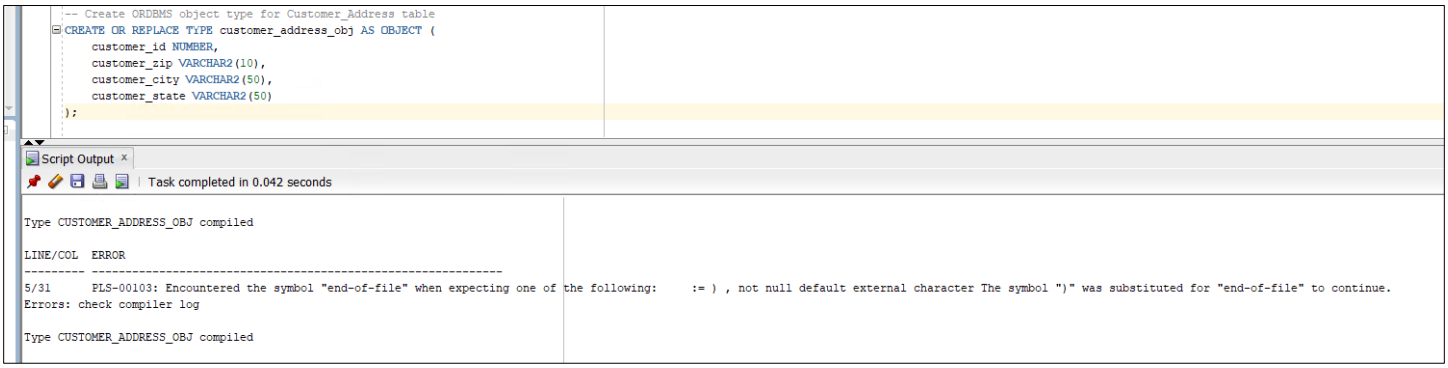
Task completed in 0.043 seconds

```
Procedure ORDER_REVENUE compiled

The total revenue generated by Order ID 1001 is: 200


PL/SQL procedure successfully completed.
```

## ORDBMS - Object–Relational Database Management System

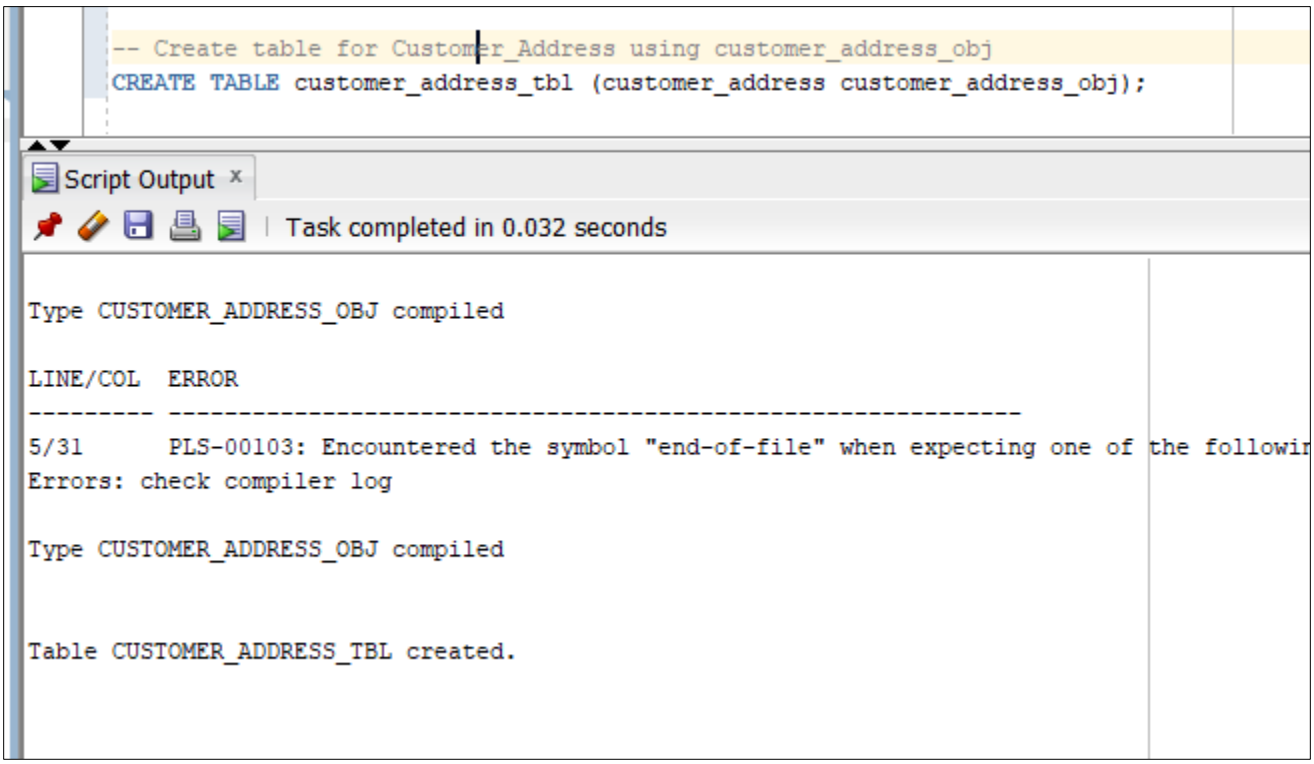
**Creating ORDBMS object type for Customer_Address table** (Navyasree Sriramoju)

```
CREATE OR REPLACE TYPE customer_address_obj AS OBJECT (
    customer_id NUMBER,
    customer_zip VARCHAR2(10),
    customer_city VARCHAR2(50),
    customer_state VARCHAR2(50)
);
```




**Creating table for Customer_Address using customer_address_obj** (Navyasree Sriramoju)

```
CREATE TABLE customer_address_tbl (customer_address customer_address_obj);
```



**Describing customer_address_tbl** (Naga Sai Lohitha Karmuru)

DESCRIBE customer_address_tbl;

```
-- Describe customer_address_tbl
DESCRIBE customer_address_tbl;
```

Script Output

Task completed in 0.327 seconds

```
Type CUSTOMER_ADDRESS_OBJ compiled

LINE/COL  ERROR
--------- -------------------------------------------------------------
5/31      PLS-00103: Encountered the symbol "end-of-file" when expecting c
Errors: check compiler log

Type CUSTOMER_ADDRESS_OBJ compiled


Table CUSTOMER_ADDRESS_TBL created.

Name             Null? Type
---------------- ----- --------------------
CUSTOMER_ADDRESS       CUSTOMER_ADDRESS_OBJ
```

**Inserting the values** (Naga Sai Lohitha Karmuru)

```
-- Describe customer_address_tbl
DESCRIBE customer_address_tbl;

INSERT INTO customer_address_tbl VALUES (customer_address_obj(1, '12345', 'Los Angeles', 'CA'));

SELECT * from customer_address_tbl;
```

Script Output

Task completed in 0.029 seconds

```
Type CUSTOMER_ADDRESS_OBJ compiled

LINE/COL  ERROR
--------- -------------------------------------------------------------
5/31      PLS-00103: Encountered the symbol "end-of-file" when expecting one of the following:     := ) , not
Errors: check compiler log

Type CUSTOMER_ADDRESS_OBJ compiled


Table CUSTOMER_ADDRESS_TBL created.

Name             Null? Type
---------------- ----- --------------------
CUSTOMER_ADDRESS       CUSTOMER_ADDRESS_OBJ
no rows selected

1 row inserted.
```

**Displaying the values** (Navyasree Sriramoju)

SELECT * from customer_address_tbl;

```
SELECT * from customer_address_tbl;
```

Script Output

Task completed in 0.032 seconds

```
CUSTOMER_ADDRESS(CUSTOMER_ID, CUSTOMER_ZIP, CUSTOMER_CITY, CUSTOMER_STATE)
--------------------------------------------------------------------------------
CUSTOMER_ADDRESS_OBJ(1, '12345', 'Los Angeles', 'CA')
```

**Creating ORDBMS object type for Product table** (Sushmitha Dandu)

```
CREATE OR REPLACE TYPE product_obj AS OBJECT (
   product_id NUMBER,
   product_name VARCHAR2(50),
   product_quantity NUMBER,
   product_type VARCHAR2(50),
   product_price NUMBER,
   product_color VARCHAR2(50),
   product_size VARCHAR2(20),
   product_warranty VARCHAR2(50),
   product_brand VARCHAR2(50)
);
```

```
CREATE OR REPLACE TYPE product_obj AS OBJECT (
    product_id NUMBER,
    product_name VARCHAR2(50),
    product_quantity NUMBER,
    product_type VARCHAR2(50),
    product_price NUMBER,
    product_color VARCHAR2(50),
    product_size VARCHAR2(20),
    product_warranty VARCHAR2(50),
    product_brand VARCHAR2(50)
);
```

```
Script Output
Task completed in 0.049 seconds

Type PRODUCT_OBJ compiled
```

**Creating table for Product_t using product_obj** (Sushmitha Dandu)

CREATE TABLE product_t(product product_obj);

```
CREATE TABLE product_t(product product_obj);
```

```
Script Output
Task completed in 0.027 seconds

Type PRODUCT_OBJ compiled


Table PRODUCT_T created.
```

**Describing product_t** (Naga Sai Lohitha Karmuru)

```
CREATE TABLE product_t(product product_obj);

DESCRIBE product_t;
```

Script Output

Task completed in 0.327 seconds

```
Type PRODUCT_OBJ compiled


Table PRODUCT_T created.

Name    Null? Type
------- ----- -----------
PRODUCT       PRODUCT_OBJ
```

**Inserting Values into the table** (Naga Sai Lohitha Karmuru)

INSERT INTO product_t VALUES (product_obj(1, 'Widget', 10, 'Gadget', 9.99, 'Blue', 'Small', '1 year', 'ABC Corp'));

```
INSERT INTO product_t VALUES (product_obj(1, 'Widget', 10, 'Gadget', 9.99, 'Blue', 'Small', '1 year', 'ABC Corp'));
```

Script Output

Task completed in 0.038 seconds

```
1 row inserted.
```

**Displaying the Values** (Sushmitha Dandu)

select * from product_t;

```
select * from product_t;
```

Script Output

Task completed in 0.037 seconds

```
1 row inserted.


PRODUCT(PRODUCT_ID, PRODUCT_NAME, PRODUCT_QUANTITY, PRODUCT_TYPE, PRODUCT_PRICE, PRODUCT_COLOR, PRODUCT_SIZE, PRODUCT_WARRANTY, PRODUCT_BRAND)
-----------------------------------------------------------------------------------------------------------------------------------------
PRODUCT_OBJ(1, 'Widget', 10, 'Gadget', 9.99, 'Blue', 'Small', '1 year', 'ABC Corp')
```