LONDON
METROPOLITAN
UNIVERSITY

islington college
(इस्लिङ्टन कलेज)

Module Code & Module Title

CC5051NI

Databases

Assessment Weightage & Type

50% Individual Coursework

Year and Semester

2021-2022 Autumn

Student Name:

Group: C9

London Met ID: 20048942

College ID: NP01CP4S210237

Assignment Due Date: 28th December 2021

Assignment Submission Date:

I confirm that I understand my coursework needs to be submitted online via Google Classroom under the relevant module page before the deadline in order for my assignment to be accepted and marked. I am fully aware that late submissions will be treated as non-submission and a mark of zero will be awarded.

Table of Contents

List of Figures

List of Tables

1. Introduction

1.1 Introduction to the business

Masterpieces Limited is a new business established by Mr. Steve. Mr. Steve is a local businessman and a real estate owner. He has many other reputed business as well and he has decided to open Masterpiece Limited as an online business. The online business aims to provide arts and paintings to individuals and commercial organizations on lease or for sale. The paintings given on lease or sold are owned by Masterpieces Limited, or the artists who have drawn it.

Various paintings are collected from artists all over Nepal. The owner or artist can submit their painting for Masterpiece Limited by contacting them through mail. The paintings are checked by a group of staffs for fake paintings and after inspection, it is stored in a warehouse.

The business will have paintings on their website for sale or lease. The customers can either lease or buy the paintings. If the painting on the website is not leased by any customer for four months, the painting is given back to the painter or the owner of the painting. The customer can search for painting on the website on the basis of the theme like person, animals, landscapes, illusion, and many more.

1.2 Current Business Activities and Operations

Some of the business activities and operations of Masterpieces Limited are listed below.

→ Masterpiece Limited has a website and customers can surf on the browser to buy or take the painting on lease.

→ Masterpiece Limited only takes payment in the form of online through E-Sewa, Khalti, IME Pay, etc.

→ The paintings are delivered to the customers in three working days after the payment.

→ Masterpiece Limited runs seven days a week and twenty-four hours a day (24/7) with the staffs having different shifts to work.

→ If any paintings are not leased within four months, the painting is returned back to the artist or owner. However, they can resubmit the same painting after one month.

→ The owner or artist is paid 20% off the allocated rental price or selling price of their painting.

→ Customers can search what kind of painting they want on the basic of the theme of the painting.

1.3 Business Rules

There are some rules and regulations set by Mr. Steve for the business to run properly. The business might not gain proper profit if the rules are not followed. The business rules for Masterpieces Limited are listed below.

→ Discount are given to customers according to their category.

→ A customer can lease the same art more than once after returning.

→ 20% off the allocated rental price or selling price is given to the owner.

→ A same painting cannot be sold or rented to two or more customers.

→ The artist can put up more than one art of theirs on Masterpiece Limited.

→ An artist or owner can lease their own art.

→ A staff can take multiple orders but cannot take other staff's order.

→ Each art is only associated with one artist.

1.4 Identification of Entities and Attributes

From the given case study, the following entities and attributes are identified.

Entities – Art, Customer, Staff, Order

Attributes in Art – Art_Id (PK), ArtName, Theme, RentingPrice, SellingPrice, IssueDate, Artist_Id, ArtistName, ArtistAddress, ArtistPhNo, IssueDate, Status

Attributes in Customer – Customer_Id (PK), CustomerName, CustomerAddress, CustomerPhNo, Category, DiscountPercent, ReturnDate

Attributes in Staff – Staff_Id (PK), StaffName, StaffAddress, StaffPhNo, Salary

Attributes in Order – Order_No (PK), OrderType, OrderDate, PainterCommission, PriceAfterDiscount

2. Initial ERD

2.1 List of created objects – Entities and Attributes

The list of entities and attributes for Masterpieces Limited are given below:

Entities	Attributes
Art	Art_Id (PK), ArtName, Theme, RentingPrice, SellingPrice, IssueDate, Status, Artist_Id, ArtistName, ArtistAddress, ArtistPhNo,
Customer	Customer_Id (PK), CustomerName, CustomerAddress, CustomerPhNo, Category, DiscountPercent, ReturnDate
Staff	Staff_Id (PK), StaffName, StaffAddress, StaffPhNo, Salary
Order	Order_No(PK), OrderType, OrderDate, PainterCommission, PriceAfterDiscount

Table 1 List of Entities and Attributes

2.2 Identification and representation of Primary Keys

The primary keys identified for the entities are given below:

Primary Keys
“Art_Id” in “Art” Entity
“Customer_Id” in “Customer” Entity
“Staff_Id” in “Staff” Entity
“Order_No” in “Order” Entity

Table 2 Identification of Primary Keys

2.3 Initial Entity Relationship Diagram (ERD)

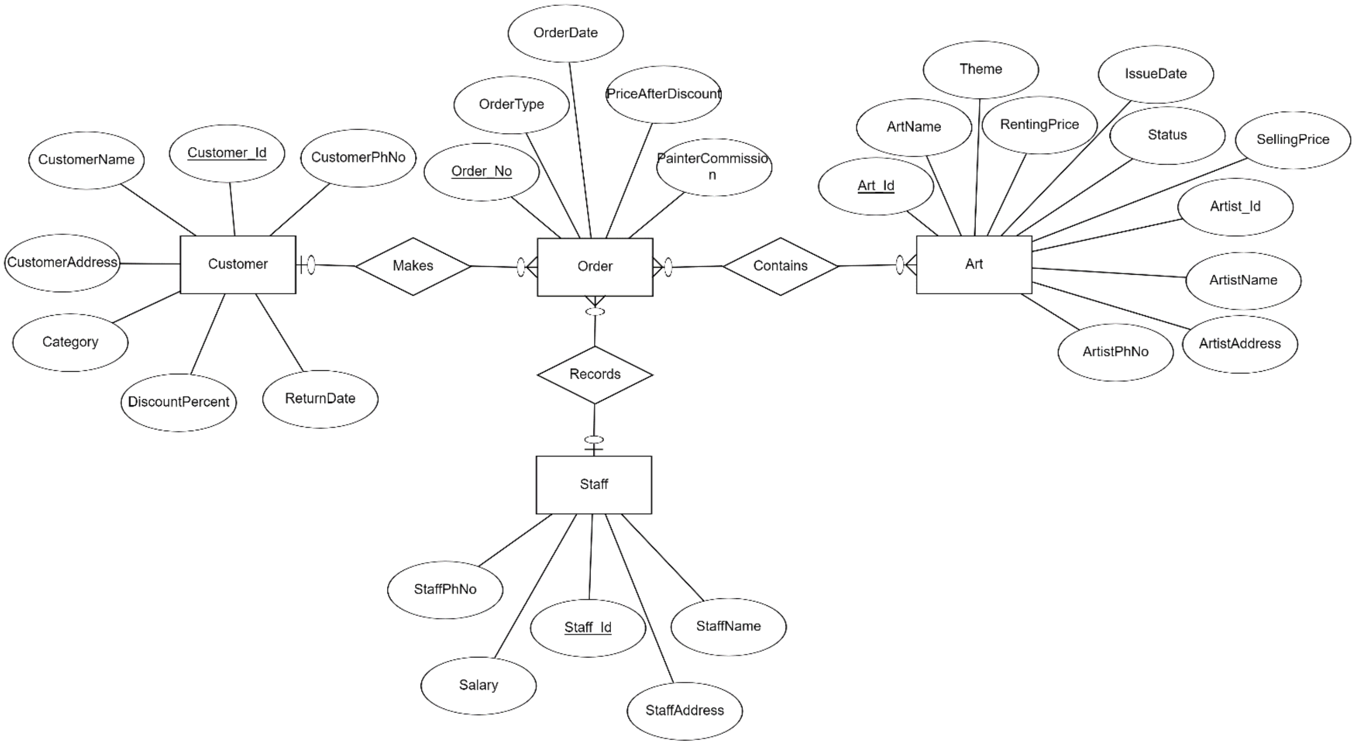


Figure 1 Initial ERD

3. Normalization

Normalization is the process of reducing redundancy from a relational table in a database. Redundancy in the data can cause insertion anomalies, update anomalies, and deletion anomalies. Normal forms are used to eliminate or reduce redundancy in database tables. Simply put, normalization divides large tables into smaller tables and connects them using relation. There are seven types of normalization in order. They are:

- 1NF (First Normal Form)
- 2NF (Second Normal Form)
- 3NF (Third Normal Form)
- BCNF (Boyce-Codd Normal Form)
- 4NF (Fourth Normal Form)
- 5NF (Fifth Normal Form)
- 6NF (Sixth Normal Form)

3.1 Assumptions

The following assumptions are made:

- One order contains only one order type.
- An artist can sell more than one painting.

3.2 Unnormalized Form (UNF)

Unnormalized form (UNF) is the basic form of the normalization process. All the data are stored in one single entity. In UNF, key elements should be identified by underlining and repeating groups should be placed inside curly brackets { }.

Customer (Customer_Id, CustomerName, CustomerAddress, CustomerPhNo, Category, DiscountPercent, ReturnDate {Order_No, OrderDate, Staff_Id, StaffName, StaffAddress, StaffPhNo, Salary {Art_Id, ArtName, Theme, RentingPrice, SellingPrice, IssueDate, Status, Artist_Id, ArtistName, ArtistAddress, ArtistPhNo, OrderType, PriceAfterDiscount, PainterCommission} })

3.3 First Normal Form (1NF)

In first normal form (1NF), the repeating groups are divided into different entities. After separating the groups into different entities, each entity should contain one key element.

Customer-1 (Customer_Id, CustomerName, CustomerAddress, CustomerPhNo, Category, DiscountPercent, ReturnDate)

Order-1 (Order_No, Customer_Id*, OrderDate, PriceAfterDiscount, PainterCommission, Staff_Id, StaffName, StaffAddress, StaffPhNo, Salary)

Art-1 (Art_Id, Order_No*,Customer_Id*, ArtName, Theme, RentingPrice, SellingPrice, IssueDate, Status, Artist_Id, ArtistName, ArtistAddress, ArtistPhNo, OrderType, PriceAfterDiscount, PainterCommission)

3.4 Second Normal Form (2NF)

In second normal form, partial dependency is checked in each entity and removed if found. Partial dependency is checked on primary keys. To change into 2NF, the data should be in 1NF.

Checking for Partial Dependencies in Customer-1

Customer-1 (Customer_Id, CustomerName, CustomerAddress, CustomerPhNo, Category, DiscountPercent, ReturnDate)

Since there is only one key in Customer-1, there is no partial dependency.

Therefore, it is already in 2NF

Checking for Partial Dependencies in Order-1

Order-1 (Order_No, Customer_Id*, OrderDate, Staff_Id, StaffName, StaffAddress, StaffPhNo, Salary)

Order_No, Customer_Id, → X

Order_No, Customer_Id → X

Order_No, → X

Customer_Id→ X

Order_No → OrderDate, Staff_Id, StaffName, StaffAddress, StaffPhNo, Salary

Checking for Partial Dependencies in Art-1

Art-1 (Art_Id, Order_No*,Customer_Id*, ArtName, Theme, RentingPrice, SellingPrice, IssueDate, Status, Artist_Id, ArtistName, ArtistAddress, ArtistPhNo, OrderType, PainterCommission, PriceAfterDiscount)

Art_Id, Order_No, Customer_Id → OrderType, PainterCommission, PriceAfterDiscount

Art_Id → ArtName, Theme, RentingPrice, SellingPrice, IssueDate, Status, Artist_Id, ArtistName, ArtistAddress, ArtistPhNo

Customer_Id → X

Order_No → X

Art_Id, Order_No → X

Art_Id, Customer_Id → X

Order_No, Customer_Id → X

Final 2NF

Customer-2 (Customer_Id, CustomerName, CustomerAddress, CustomerPhNo, Category, DiscountPercent, ReturnDate)

Order-2 (Order_No, OrderDate, Staff_Id, StaffName, StaffAddress, StaffPhNo, Salary)

Art-2 (Art_Id, ArtName, Theme, RentingPrice, SellingPrice, IssueDate, Status, Artist_Id, ArtistName, ArtistAddress, ArtistPhNo)

Order-Customer-2 (Customer_Id*, Order_No*)

Art-Order-Customer-2 (Customer_Id*, Art_Id*, Order_No*, OrderType, PainterCommission, PriceAfterDiscount)

3.5 Third Normal Form (3NF)

In third normal form, the many of the redundancies are already removed. Transitive dependencies are checked and removed if found. Similar to 2NF, the data must be in 2NF for it to be changed into 3NF.

Checking Transitive Dependencies in Customer-2:

Customer-2 (Customer_Id, CustomerName, CustomerAddress, CustomerPhNo, Category, DiscountPercent, ReturnDate)

Customer_Id → CustomerName, CustomerName → X

Customer_Id → CustomerAddress, CustomerAddress → X

Customer_Id → CustomerPhNo, CustomerPhNo → X

Customer_Id → Category, Category → DiscountPercent

Checking Transitive Dependencies in Order-2:

Order-2 (Order_No, OrderDate, Staff_Id, StaffName, StaffAddress, StaffPhNo, Salary)

Order_No → OrderDate → X

Order_No→ PainterCommission → X

Order_No→ Staff_Id → StaffName, StaffAddress, StaffPhNo, Salary)

Checking Transitive Dependencies in Art-2:

Art-2 (Art_Id, ArtName, Theme, RentingPrice, SellingPrice, IssueDate, Status, Artist_Id, ArtistName, ArtistAddress, ArtistPhNo)

Art_Id → ArtName → X

Art_Id → Theme → X

Art_Id → RentingPrice → X

Art_Id → SellingPrice → X

Art_Id → IssueDate → X

Art_Id → Status → X

Art_Id → ReturnDate → X

Art_Id → Artist_Id → ArtistName, ArtistAddress, ArtistPhNo

Final 3NF

Entity **Order-Customer-2** is not needed as it can be adequately represented within the entity **Art-Order-Customer-2.**

Art-3 (Art_Id, Artist_Id*, ArtName, Theme, RentingPrice, SellingPrice, IssueDate, Status)

Artist-3 (Artist_Id, ArtistName, ArtistAddress, ArtistPhNo)

Customer-3 (Customer_Id, Category*, CustomerName, CustomerAddress, CustomerPhNo, ReturnDate)

Customer-Category-3 (Category, DiscountPercent)

Staff-3 (Staff_Id, StaffName, StaffAddress, StaffPhNo, Salary)

Order-3 (Order_No, Staff_Id* OrderDate)

Art-Order-Customer-3 (Customer_Id*, Art_Id*, Order_No*, OrderType, PainterCommission, PriceAfterDiscount)

4. Final Entity Relation Diagram (ERD)

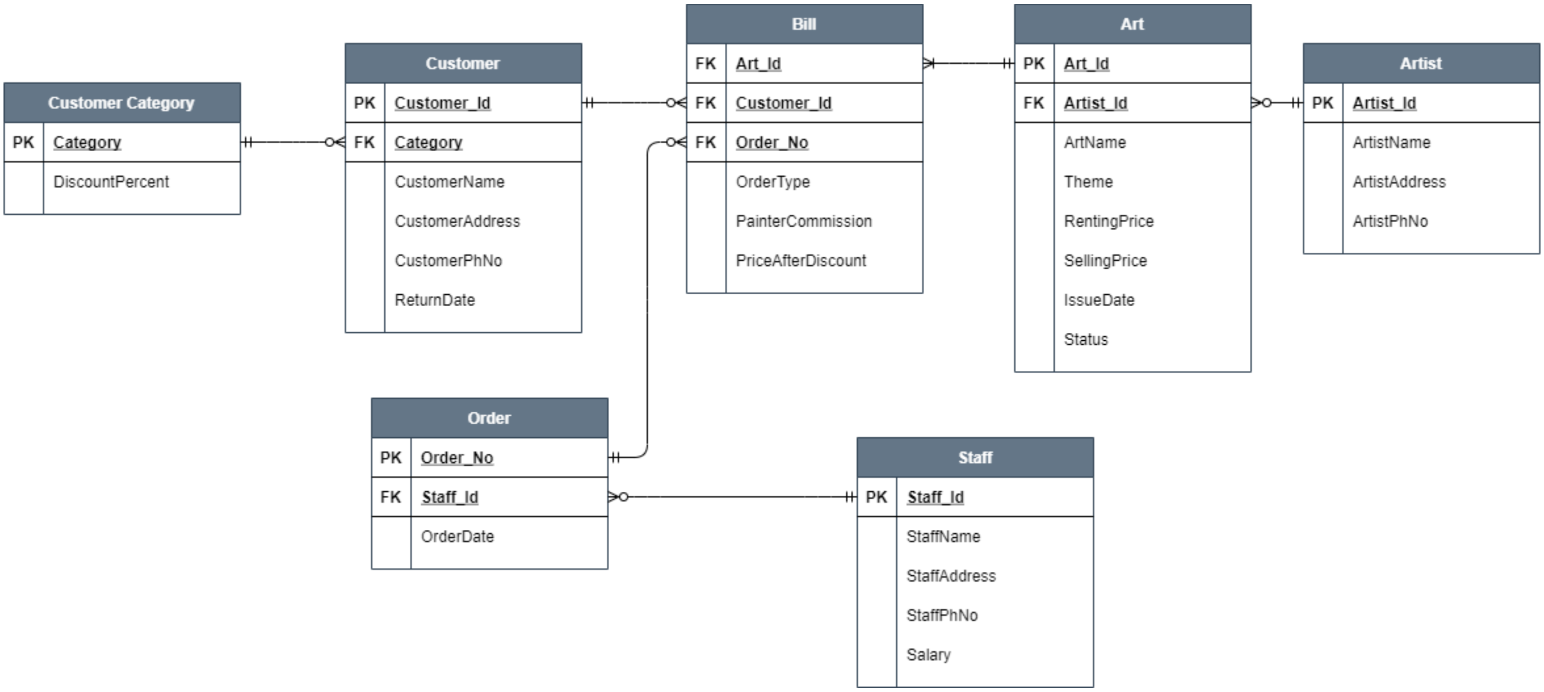
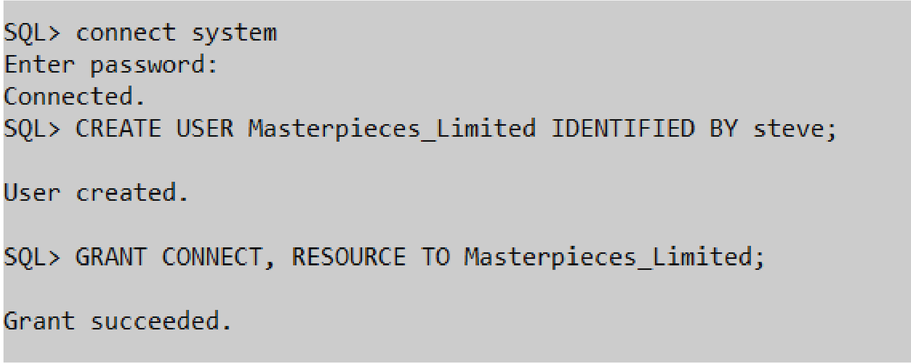


Figure 2 Final ERD

5. Implementation

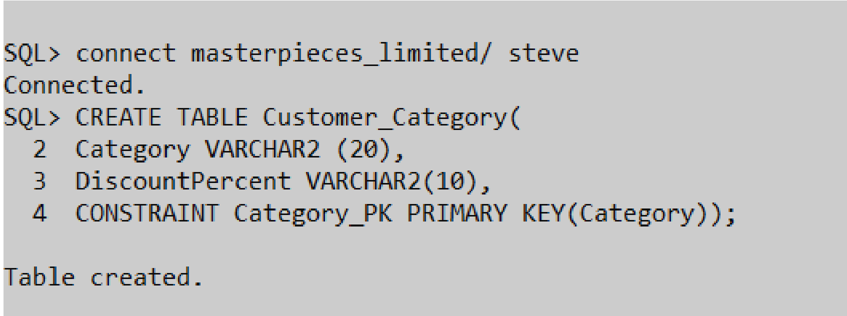
Creating and Granting Access to User



```
SQL> connect system
Enter password:
Connected.
SQL> CREATE USER Masterpieces_Limited IDENTIFIED BY steve;

User created.

SQL> GRANT CONNECT, RESOURCE TO Masterpieces_Limited;

Grant succeeded.
```

Figure 3 Creating and Granting Access to User

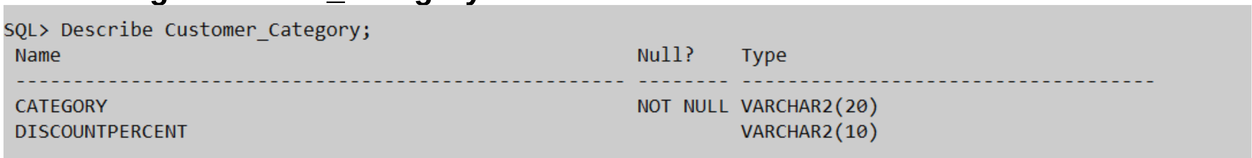
Connecting To Masterpieces Limited and Creating Customer_Category Table



```
SQL> connect masterpieces_limited/ steve
Connected.
SQL> CREATE TABLE Customer_Category(
  2  Category VARCHAR2 (20),
  3  DiscountPercent VARCHAR2(10),
  4  CONSTRAINT Category_PK PRIMARY KEY(Category));

Table created.
```

Figure 4 Connecting to User and Creating Customer_Category Table

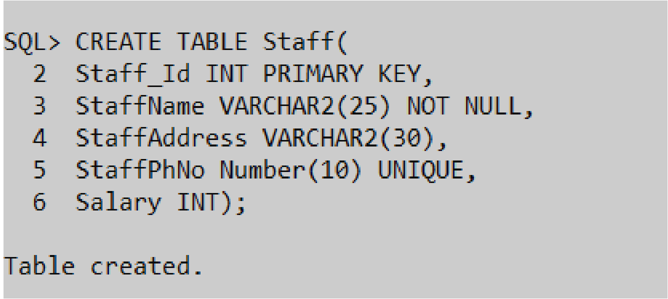
Describing Customer_Category Table



```
SQL> Describe Customer_Category;
 Name                                                  Null?    Type
 ----------------------------------------------------- -------- ------------------------------------
 CATEGORY                                              NOT NULL VARCHAR2(20)
 DISCOUNTPERCENT                                                VARCHAR2(10)
```

Figure 5 Describing Customer_Category Table

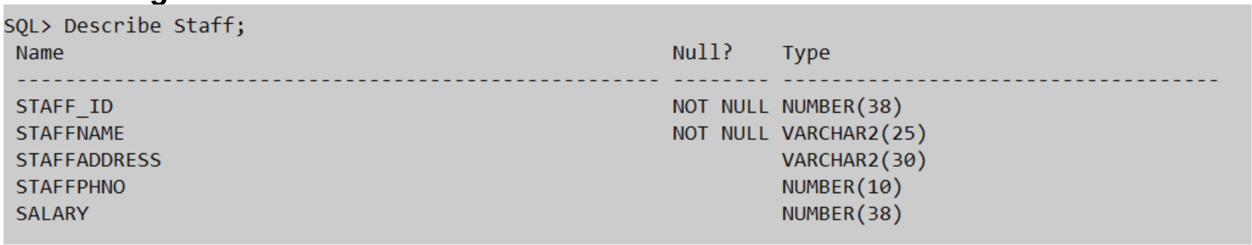
Creating Staff Table



```
SQL> CREATE TABLE Staff(
  2  Staff_Id INT PRIMARY KEY,
  3  StaffName VARCHAR2(25) NOT NULL,
  4  StaffAddress VARCHAR2(30),
  5  StaffPhNo Number(10) UNIQUE,
  6  Salary INT);

Table created.
```

Figure 6 Creating Staff Table

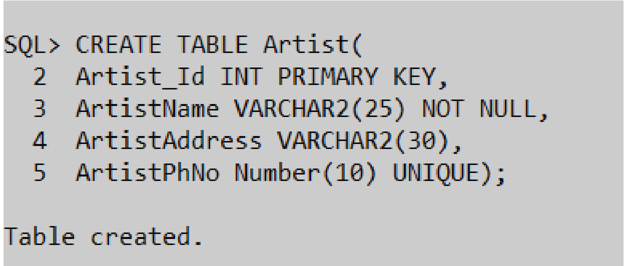
Describing Staff Table



```
SQL> Describe Staff;
 Name                                                      Null?    Type
 ----------------------------------------------------- -------- ------------------------------------
 STAFF_ID                                                  NOT NULL NUMBER(38)
 STAFFNAME                                                 NOT NULL VARCHAR2(25)
 STAFFADDRESS                                                       VARCHAR2(30)
 STAFFPHNO                                                          NUMBER(10)
 SALARY                                                             NUMBER(38)
```

Figure 7 Describing Staff Table

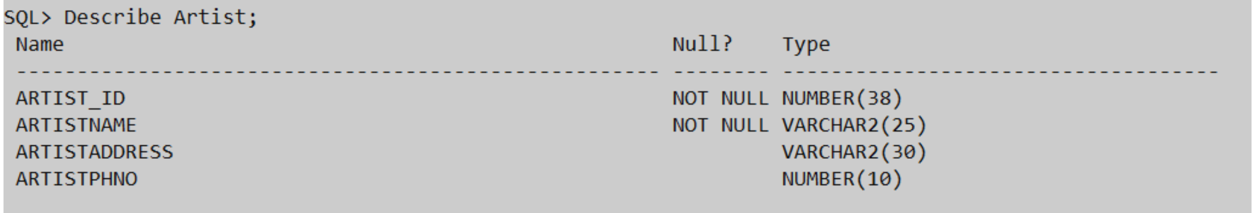
Creating Artist Table



```
SQL> CREATE TABLE Artist(
  2  Artist_Id INT PRIMARY KEY,
  3  ArtistName VARCHAR2(25) NOT NULL,
  4  ArtistAddress VARCHAR2(30),
  5  ArtistPhNo Number(10) UNIQUE);

Table created.
```

Figure 8 Creating Artist Table

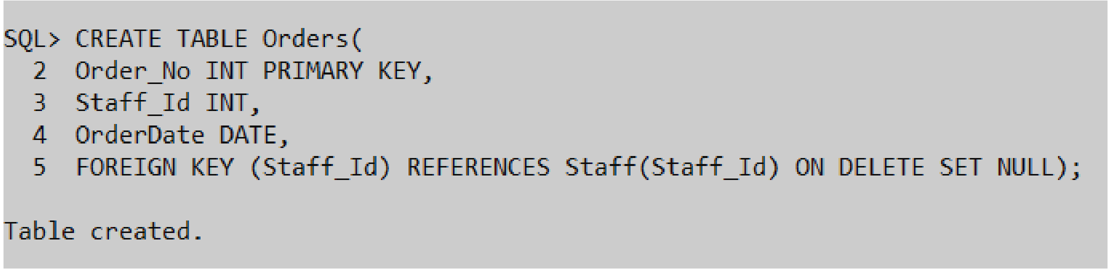
Describing Artist Table



```
SQL> Describe Artist;
 Name                                                      Null?    Type
 ----------------------------------------------------- -------- ------------------------------------
 ARTIST_ID                                                 NOT NULL NUMBER(38)
 ARTISTNAME                                                NOT NULL VARCHAR2(25)
 ARTISTADDRESS                                                      VARCHAR2(30)
 ARTISTPHNO                                                         NUMBER(10)
```

Figure 9 Describing Artist Table

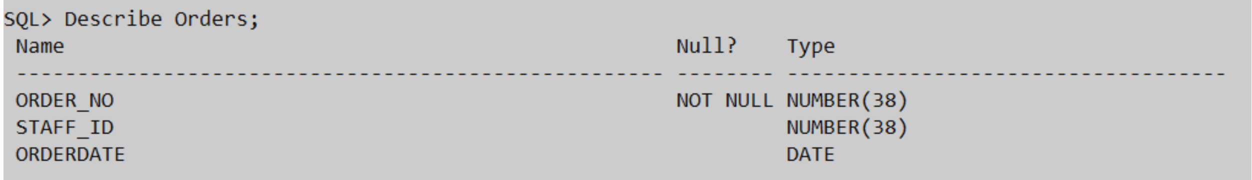
Creating Orders Table



```
SQL> CREATE TABLE Orders(
     Order_No INT PRIMARY KEY,
     Staff_Id INT,
     OrderDate DATE,
     FOREIGN KEY (Staff_Id) REFERENCES Staff(Staff_Id) ON DELETE SET NULL);

Table created.
```

Figure 10 Creating Orders Table

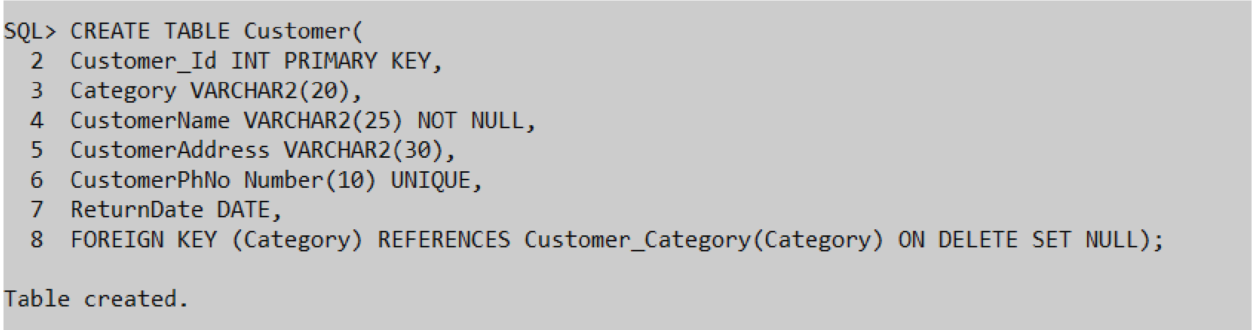
Describing Orders Table



```
SQL> Describe Orders;
 Name                                                  Null?    Type
 ----------------------------------------------------- -------- ------------------------------------
 ORDER_NO                                              NOT NULL NUMBER(38)
 STAFF_ID                                                       NUMBER(38)
 ORDERDATE                                                      DATE
```

Figure 11 Describing Orders Table

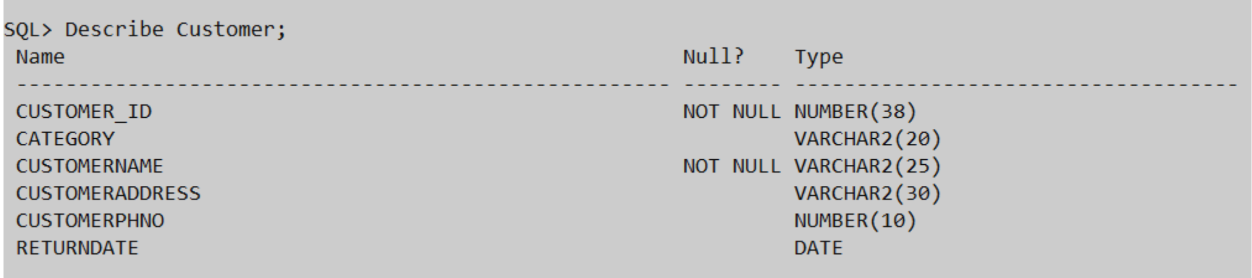
Creating Customer Table



```
SQL> CREATE TABLE Customer(
     Customer_Id INT PRIMARY KEY,
     Category VARCHAR2(20),
     CustomerName VARCHAR2(25) NOT NULL,
     CustomerAddress VARCHAR2(30),
     CustomerPhNo Number(10) UNIQUE,
     ReturnDate DATE,
     FOREIGN KEY (Category) REFERENCES Customer_Category(Category) ON DELETE SET NULL);

Table created.
```

Figure 12 Creating Customer Table

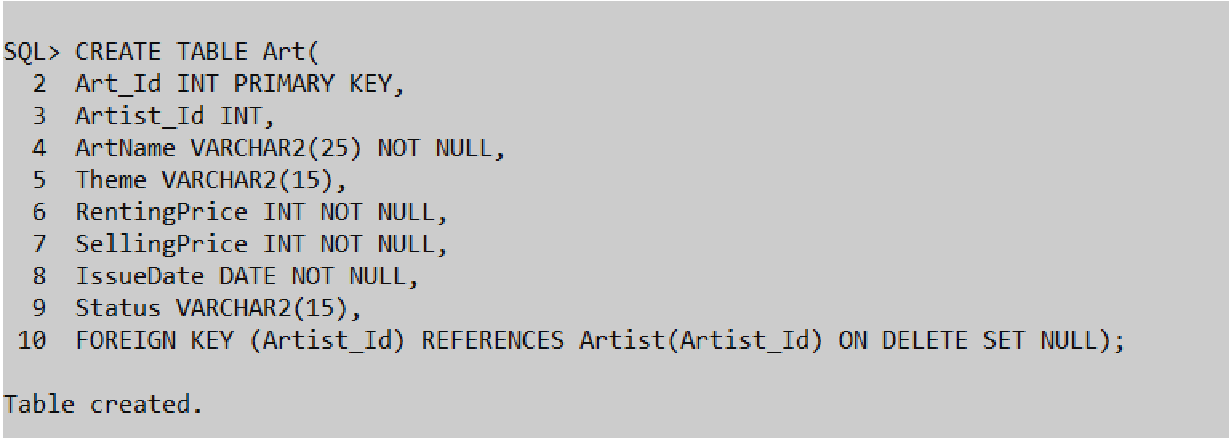
Describing Customer Table



```
SQL> Describe Customer;
 Name                                                  Null?    Type
 ----------------------------------------------------- -------- ------------------------------------
 CUSTOMER_ID                                           NOT NULL NUMBER(38)
 CATEGORY                                                       VARCHAR2(20)
 CUSTOMERNAME                                          NOT NULL VARCHAR2(25)
 CUSTOMERADDRESS                                                VARCHAR2(30)
 CUSTOMERPHNO                                                   NUMBER(10)
 RETURNDATE                                                     DATE
```

Figure 13 Describing Customer Table

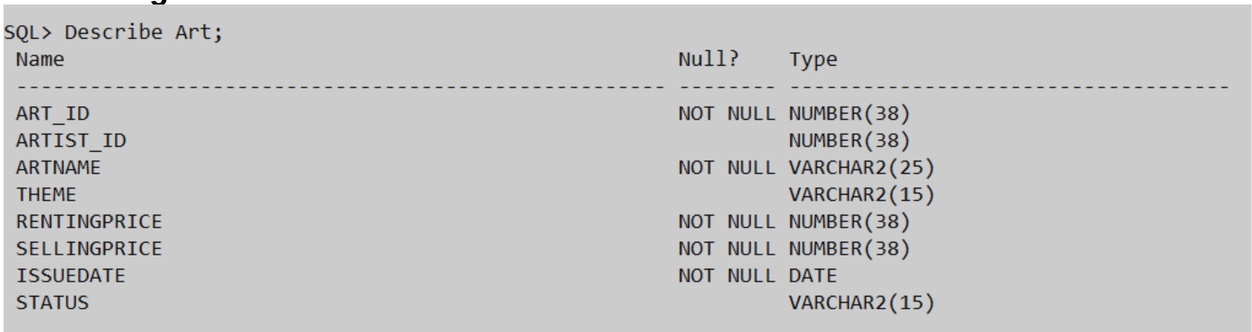
Creating Art Table



```
SQL> CREATE TABLE Art(
Art_Id INT PRIMARY KEY,
Artist_Id INT,
ArtName VARCHAR2(25) NOT NULL,
Theme VARCHAR2(15),
RentingPrice INT NOT NULL,
SellingPrice INT NOT NULL,
IssueDate DATE NOT NULL,
Status VARCHAR2(15),
FOREIGN KEY (Artist_Id) REFERENCES Artist(Artist_Id) ON DELETE SET NULL);

Table created.
```

Figure 14 Creating Art Table

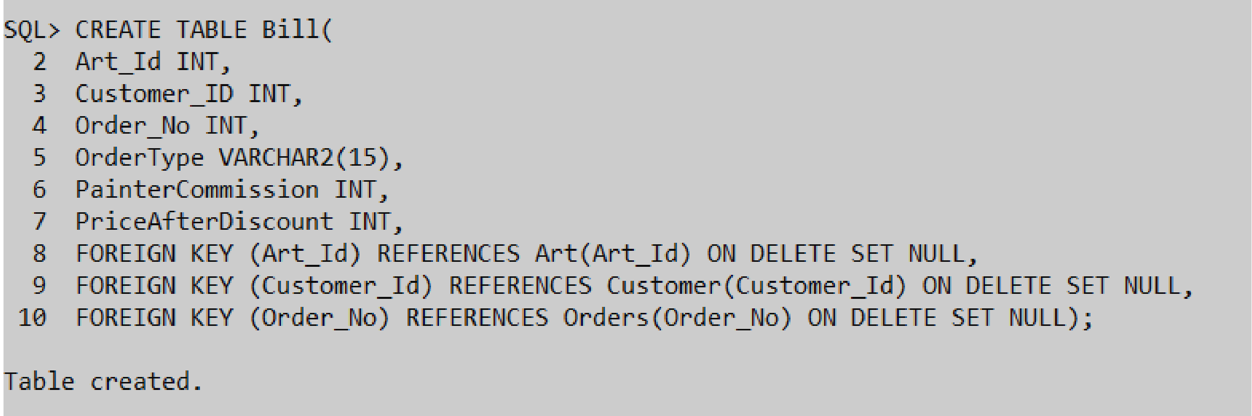
Describing Art Table



```
SQL> Describe Art;
 Name                                                  Null?    Type
 ----------------------------------------------------- -------- ------------------------------------
 ART_ID                                                NOT NULL NUMBER(38)
 ARTIST_ID                                                      NUMBER(38)
 ARTNAME                                               NOT NULL VARCHAR2(25)
 THEME                                                          VARCHAR2(15)
 RENTINGPRICE                                          NOT NULL NUMBER(38)
 SELLINGPRICE                                          NOT NULL NUMBER(38)
 ISSUEDATE                                             NOT NULL DATE
 STATUS                                                         VARCHAR2(15)
```

Figure 15 Describing Art Table

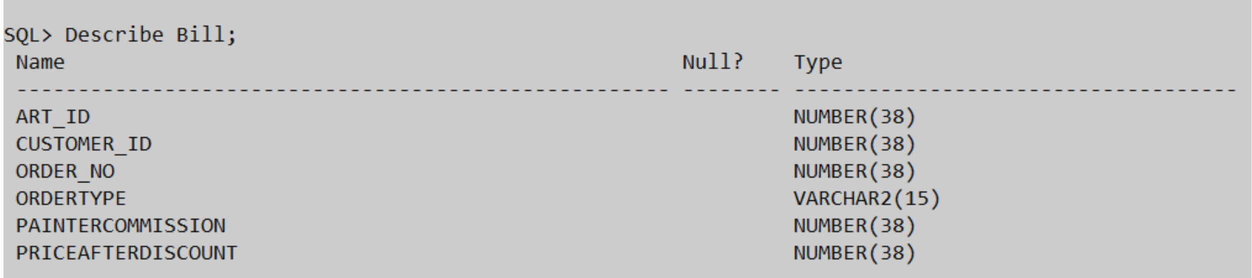
Creating Bill Table



```
SQL> CREATE TABLE Bill(
Art_Id INT,
Customer_ID INT,
Order_No INT,
OrderType VARCHAR2(15),
PainterCommission INT,
PriceAfterDiscount INT,
FOREIGN KEY (Art_Id) REFERENCES Art(Art_Id) ON DELETE SET NULL,
FOREIGN KEY (Customer_Id) REFERENCES Customer(Customer_Id) ON DELETE SET NULL,
FOREIGN KEY (Order_No) REFERENCES Orders(Order_No) ON DELETE SET NULL);

Table created.
```

Figure 16 Creating Bill Table

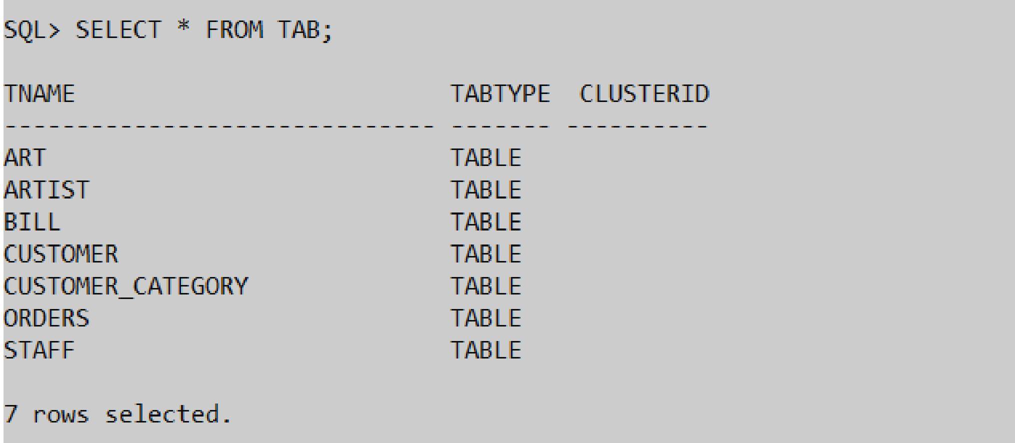
Describing Bill Table



```
SQL> Describe Bill;
 Name                                                  Null?    Type
 ----------------------------------------------------- -------- ------------------------------------
 ART_ID                                                         NUMBER(38)
 CUSTOMER_ID                                                    NUMBER(38)
 ORDER_NO                                                       NUMBER(38)
 ORDERTYPE                                                      VARCHAR2(15)
 PAINTERCOMMISSION                                              NUMBER(38)
 PRICEAFTERDISCOUNT                                             NUMBER(38)
```

Figure 17 Describing Bill Table

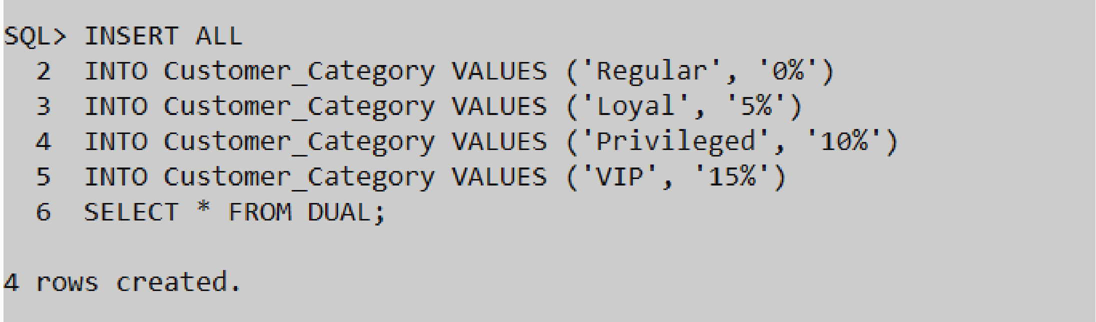
Showing all tables in Masterpieces



```
SQL> SELECT * FROM TAB;

TNAME                          TABTYPE  CLUSTERID
------------------------------ ------- ----------
ART                            TABLE
ARTIST                         TABLE
BILL                           TABLE
CUSTOMER                       TABLE
CUSTOMER_CATEGORY              TABLE
ORDERS                         TABLE
STAFF                          TABLE

7 rows selected.
```

Figure 18 Showing All Tables in User

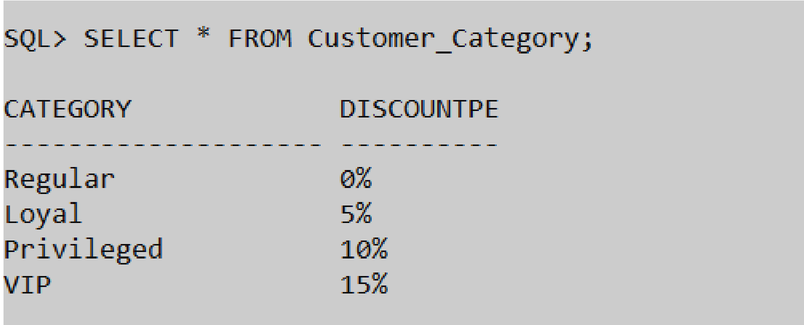
Inserting values in Customer_Category Table



```
SQL> INSERT ALL
  2  INTO Customer_Category VALUES ('Regular', '0%')
  3  INTO Customer_Category VALUES ('Loyal', '5%')
  4  INTO Customer_Category VALUES ('Privileged', '10%')
  5  INTO Customer_Category VALUES ('VIP', '15%')
  6  SELECT * FROM DUAL;

4 rows created.
```

Figure 19 Inserting Values in Customer_Category Table

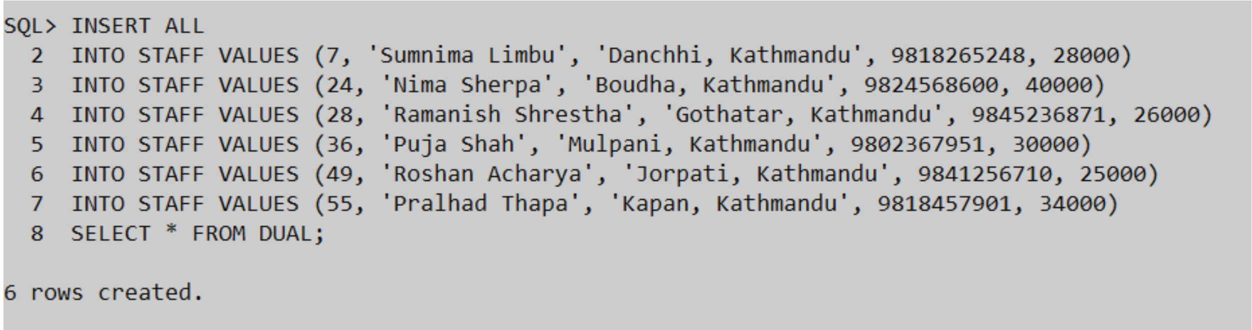
Displaying Customer_Category Table



```
SQL> SELECT * FROM Customer_Category;

CATEGORY             DISCOUNTPE
-------------------- ----------
Regular              0%
Loyal                5%
Privileged           10%
VIP                  15%
```

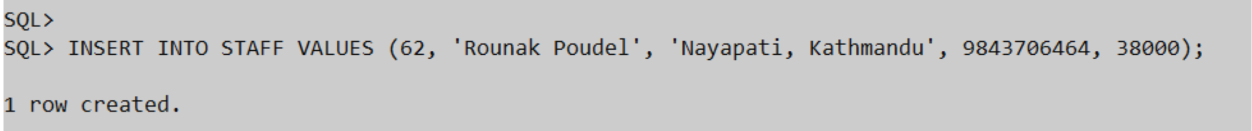
Figure 20 Displaying Customer_Category Table

Inserting values in Staff Table



```
SQL> INSERT ALL
     INTO STAFF VALUES (7, 'Sumnima Limbu', 'Danchhi, Kathmandu', 9818265248, 28000)
     INTO STAFF VALUES (24, 'Nima Sherpa', 'Boudha, Kathmandu', 9824568600, 40000)
     INTO STAFF VALUES (28, 'Ramanish Shrestha', 'Gothatar, Kathmandu', 9845236871, 26000)
     INTO STAFF VALUES (36, 'Puja Shah', 'Mulpani, Kathmandu', 9802367951, 30000)
     INTO STAFF VALUES (49, 'Roshan Acharya', 'Jorpati, Kathmandu', 9841256710, 25000)
     INTO STAFF VALUES (55, 'Pralhad Thapa', 'Kapan, Kathmandu', 9818457901, 34000)
     SELECT * FROM DUAL;

6 rows created.
```

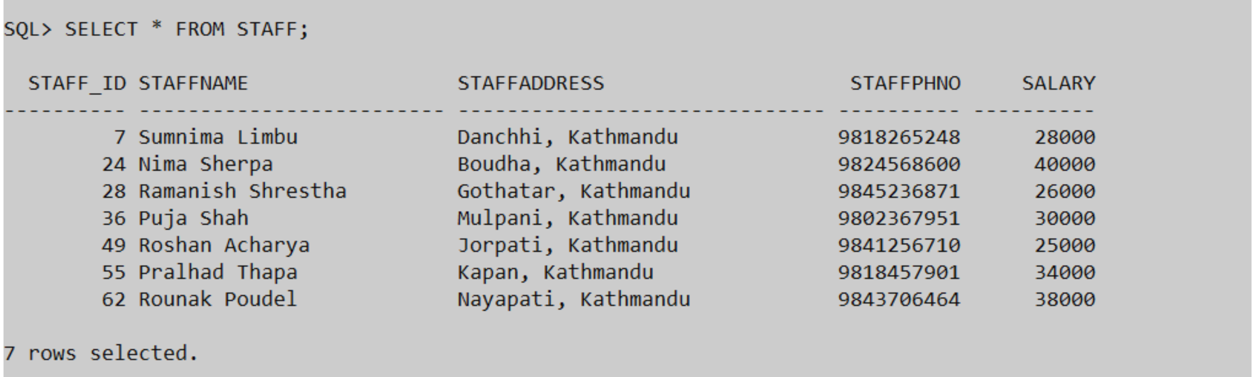
Figure 21 Inserting Values in Staff Table



```
SQL>
SQL> INSERT INTO STAFF VALUES (62, 'Rounak Poudel', 'Nayapati, Kathmandu', 9843706464, 38000);

1 row created.
```

Figure 22 Inserting Values in Staff Table (2)

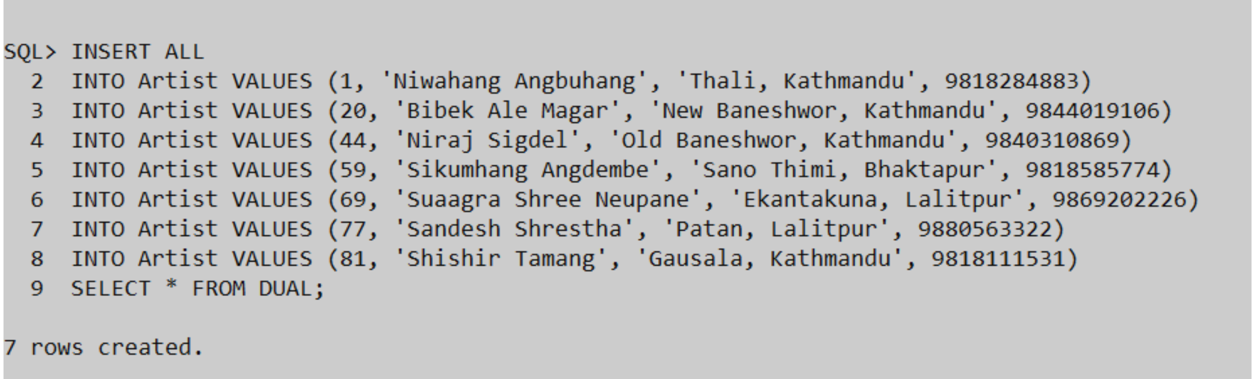
Displaying Staff Table



```
SQL> SELECT * FROM STAFF;

  STAFF_ID STAFFNAME                 STAFFADDRESS                    STAFFPHNO     SALARY
---------- ------------------------- ------------------------------ ---------- ----------
         7 Sumnima Limbu             Danchhi, Kathmandu              9818265248      28000
        24 Nima Sherpa               Boudha, Kathmandu               9824568600      40000
        28 Ramanish Shrestha         Gothatar, Kathmandu             9845236871      26000
        36 Puja Shah                 Mulpani, Kathmandu              9802367951      30000
        49 Roshan Acharya            Jorpati, Kathmandu              9841256710      25000
        55 Pralhad Thapa             Kapan, Kathmandu                9818457901      34000
        62 Rounak Poudel             Nayapati, Kathmandu             9843706464      38000

7 rows selected.
```

Figure 23 Displaying Staff Table

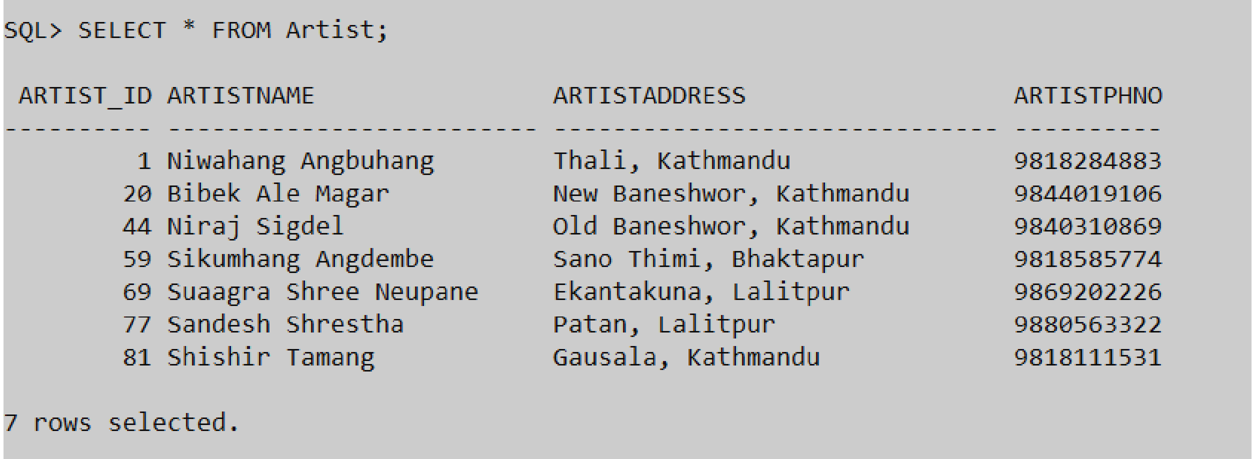
Inserting values in Artist Table



```
SQL> INSERT ALL
     INTO Artist VALUES (1, 'Niwahang Angbuhang', 'Thali, Kathmandu', 9818284883)
     INTO Artist VALUES (20, 'Bibek Ale Magar', 'New Baneshwor, Kathmandu', 9844019106)
     INTO Artist VALUES (44, 'Niraj Sigdel', 'Old Baneshwor, Kathmandu', 9840310869)
     INTO Artist VALUES (59, 'Sikumhang Angdembe', 'Sano Thimi, Bhaktapur', 9818585774)
     INTO Artist VALUES (69, 'Suaagra Shree Neupane', 'Ekantakuna, Lalitpur', 9869202226)
     INTO Artist VALUES (77, 'Sandesh Shrestha', 'Patan, Lalitpur', 9880563322)
     INTO Artist VALUES (81, 'Shishir Tamang', 'Gausala, Kathmandu', 9818111531)
     SELECT * FROM DUAL;

7 rows created.
```

Figure 24 Inserting Values in Artist Table

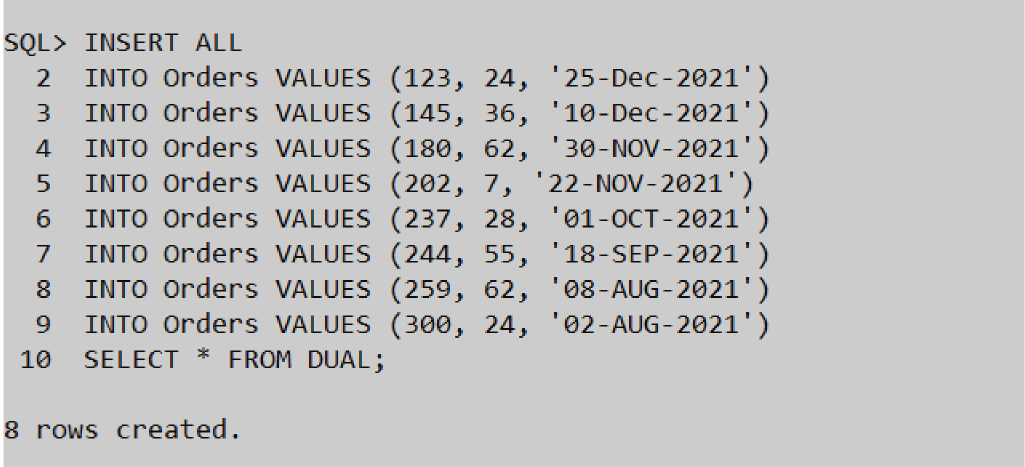
Displaying Artist Table



```
SQL> SELECT * FROM Artist;

 ARTIST_ID ARTISTNAME                ARTISTADDRESS                  ARTISTPHNO
---------- ------------------------- ------------------------------ ----------
         1 Niwahang Angbuhang        Thali, Kathmandu               9818284883
        20 Bibek Ale Magar           New Baneshwor, Kathmandu       9844019106
        44 Niraj Sigdel              Old Baneshwor, Kathmandu       9840310869
        59 Sikumhang Angdembe        Sano Thimi, Bhaktapur          9818585774
        69 Suaagra Shree Neupane     Ekantakuna, Lalitpur           9869202226
        77 Sandesh Shrestha          Patan, Lalitpur                9880563322
        81 Shishir Tamang            Gausala, Kathmandu             9818111531

7 rows selected.
```

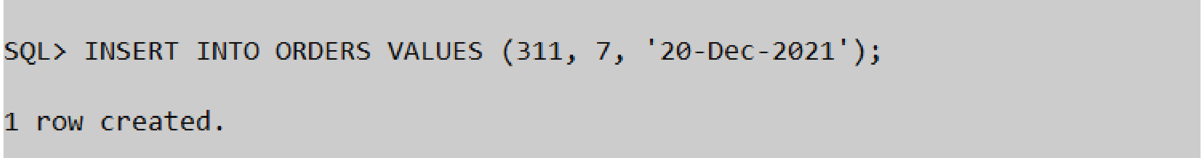
Figure 25 Displaying Artist Table

Inserting values in Orders Table



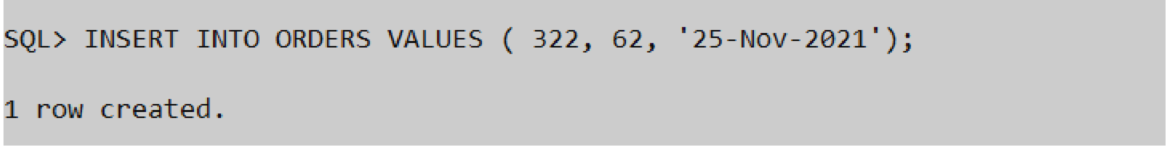
```
SQL> INSERT ALL
     INTO Orders VALUES (123, 24, '25-Dec-2021')
     INTO Orders VALUES (145, 36, '10-Dec-2021')
     INTO Orders VALUES (180, 62, '30-NOV-2021')
     INTO Orders VALUES (202, 7, '22-NOV-2021')
     INTO Orders VALUES (237, 28, '01-OCT-2021')
     INTO Orders VALUES (244, 55, '18-SEP-2021')
     INTO Orders VALUES (259, 62, '08-AUG-2021')
     INTO Orders VALUES (300, 24, '02-AUG-2021')
     SELECT * FROM DUAL;

8 rows created.
```

Figure 26 Inserting Values in Orders Table



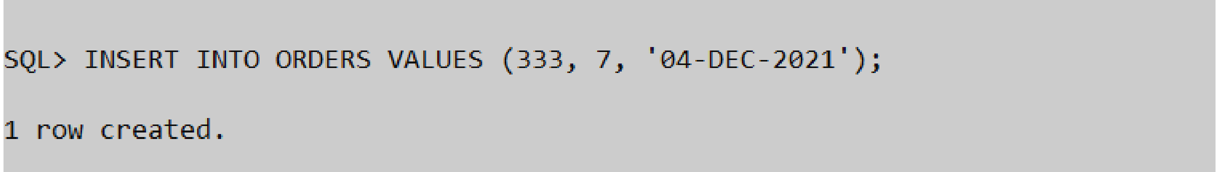
```
SQL> INSERT INTO ORDERS VALUES (311, 7, '20-Dec-2021');

1 row created.
```

Figure 27 Inserting Values in Orders Table (2)



```
SQL> INSERT INTO ORDERS VALUES ( 322, 62, '25-Nov-2021');

1 row created.
```

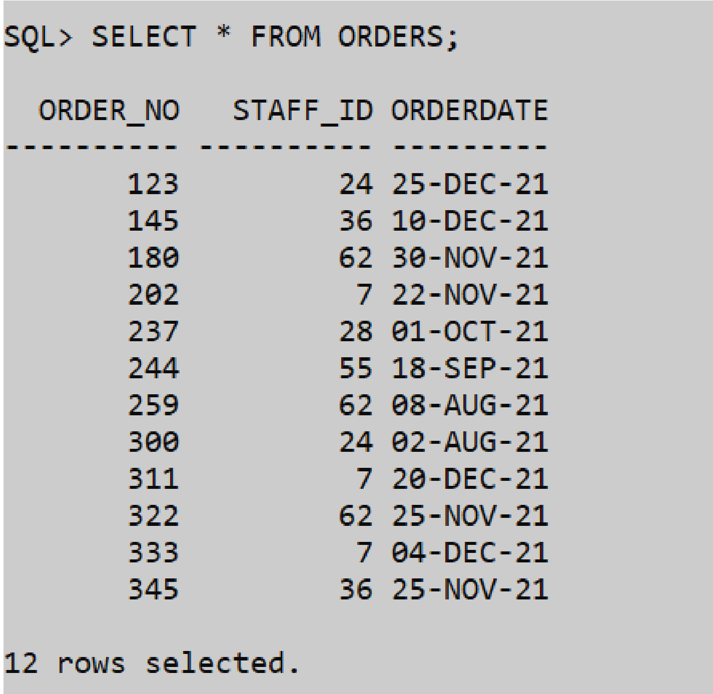
Figure 28 Inserting Values in Orders Table (3)



```
SQL> INSERT INTO ORDERS VALUES (333, 7, '04-DEC-2021');

1 row created.
```

Figure 29 Inserting Values in Orders Table (4)

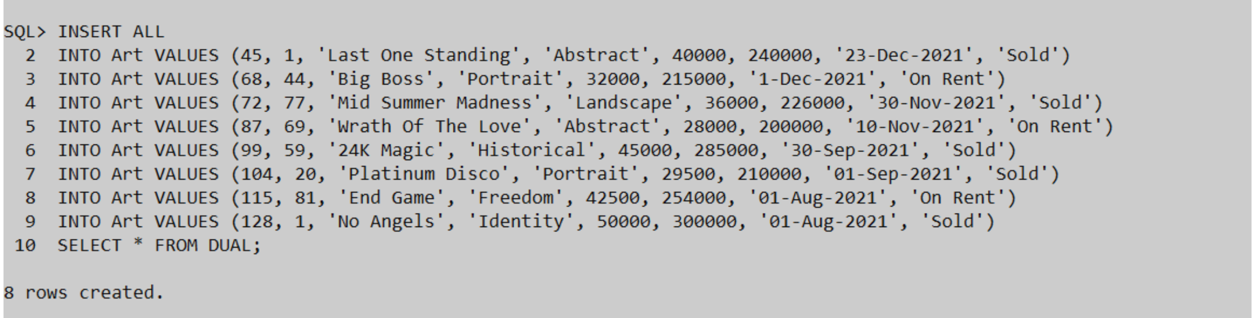
Displaying Orders Table



```
SQL> SELECT * FROM ORDERS;

  ORDER_NO   STAFF_ID ORDERDATE
---------- ---------- ---------
       123         24 25-DEC-21
       145         36 10-DEC-21
       180         62 30-NOV-21
       202          7 22-NOV-21
       237         28 01-OCT-21
       244         55 18-SEP-21
       259         62 08-AUG-21
       300         24 02-AUG-21
       311          7 20-DEC-21
       322         62 25-NOV-21
       333          7 04-DEC-21
       345         36 25-NOV-21

12 rows selected.
```

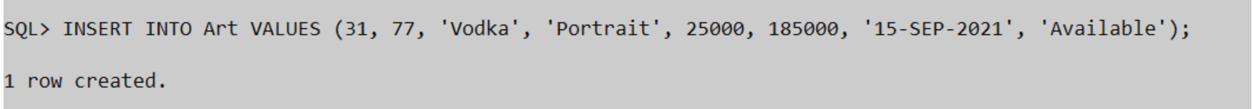
Figure 30 Displaying Orders Table

Inserting values in Art Table



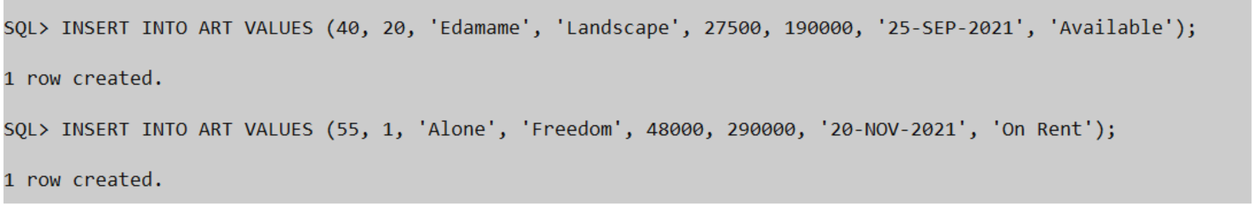
```
SQL> INSERT ALL
     INTO Art VALUES (45, 1, 'Last One Standing', 'Abstract', 40000, 240000, '23-Dec-2021', 'Sold')
     INTO Art VALUES (68, 44, 'Big Boss', 'Portrait', 32000, 215000, '1-Dec-2021', 'On Rent')
     INTO Art VALUES (72, 77, 'Mid Summer Madness', 'Landscape', 36000, 226000, '30-Nov-2021', 'Sold')
     INTO Art VALUES (87, 69, 'Wrath Of The Love', 'Abstract', 28000, 200000, '10-Nov-2021', 'On Rent')
     INTO Art VALUES (99, 59, '24K Magic', 'Historical', 45000, 285000, '30-Sep-2021', 'Sold')
     INTO Art VALUES (104, 20, 'Platinum Disco', 'Portrait', 29500, 210000, '01-Sep-2021', 'Sold')
     INTO Art VALUES (115, 81, 'End Game', 'Freedom', 42500, 254000, '01-Aug-2021', 'On Rent')
     INTO Art VALUES (128, 1, 'No Angels', 'Identity', 50000, 300000, '01-Aug-2021', 'Sold')
     SELECT * FROM DUAL;

8 rows created.
```

Figure 31 Inserting Values in Art Table



```
SQL> INSERT INTO Art VALUES (31, 77, 'Vodka', 'Portrait', 25000, 185000, '15-SEP-2021', 'Available');

1 row created.
```

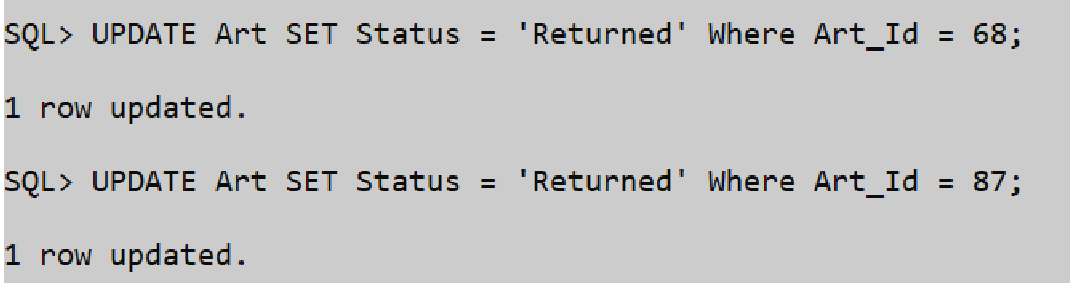
Figure 32 Inserting Values in Art Table (2)



```
SQL> INSERT INTO ART VALUES (40, 20, 'Edamame', 'Landscape', 27500, 190000, '25-SEP-2021', 'Available');

1 row created.

SQL> INSERT INTO ART VALUES (55, 1, 'Alone', 'Freedom', 48000, 290000, '20-NOV-2021', 'On Rent');

1 row created.
```

Figure 33 Inserting Values in Art Table (3)

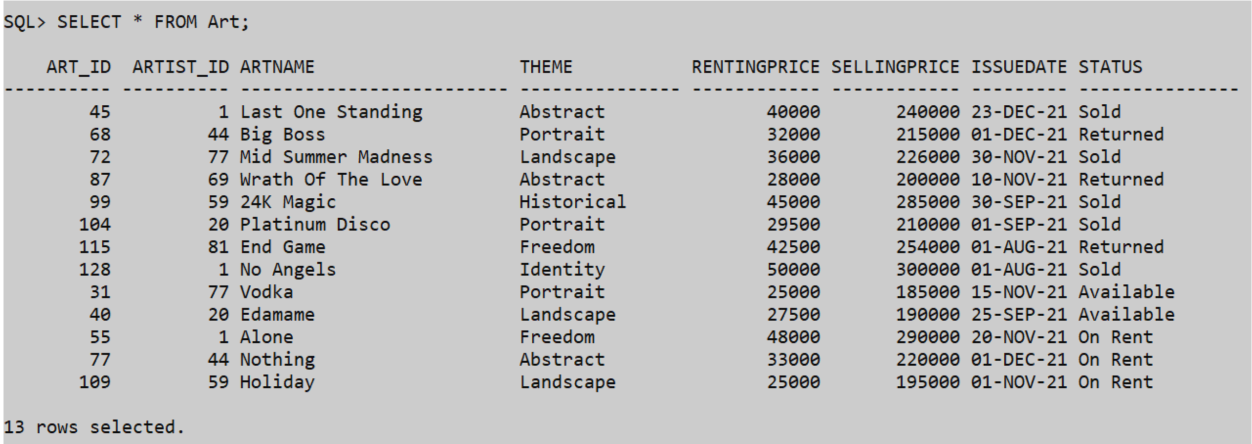
Updating Status in Art Table



```
SQL> UPDATE Art SET Status = 'Returned' Where Art_Id = 68;

1 row updated.

SQL> UPDATE Art SET Status = 'Returned' Where Art_Id = 87;

1 row updated.
```

Figure 34 Updating Status in Art Table

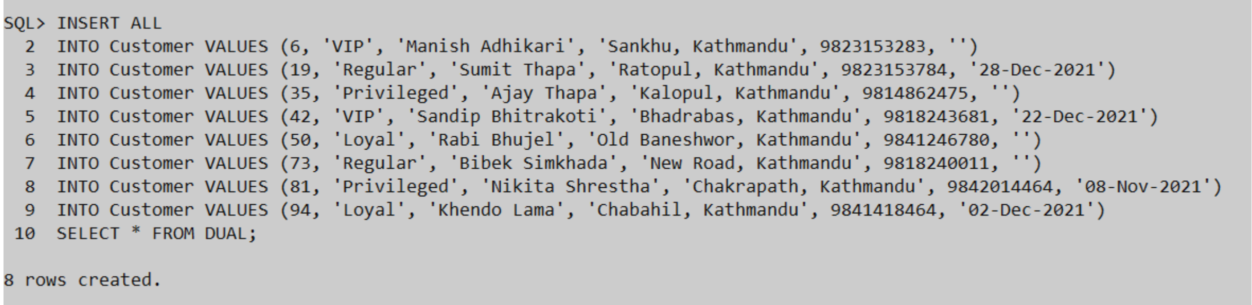
Displaying Art Table



```
SQL> SELECT * FROM Art;

    ART_ID  ARTIST_ID ARTNAME                   THEME           RENTINGPRICE SELLINGPRICE ISSUEDATE STATUS
---------- ---------- ------------------------- --------------- ------------ ------------ --------- ---------------
        45          1 Last One Standing         Abstract               40000       240000 23-DEC-21 Sold
        68         44 Big Boss                  Portrait               32000       215000 01-DEC-21 Returned
        72         77 Mid Summer Madness        Landscape              36000       226000 30-NOV-21 Sold
        87         69 Wrath Of The Love         Abstract               28000       200000 10-NOV-21 Returned
        99         59 24K Magic                 Historical             45000       285000 30-SEP-21 Sold
       104         20 Platinum Disco            Portrait               29500       210000 01-SEP-21 Sold
       115         81 End Game                  Freedom                42500       254000 01-AUG-21 Returned
       128          1 No Angels                 Identity               50000       300000 01-AUG-21 Sold
        31         77 Vodka                     Portrait               25000       185000 15-NOV-21 Available
        40         20 Edamame                   Landscape              27500       190000 25-SEP-21 Available
        55          1 Alone                     Freedom                48000       290000 20-NOV-21 On Rent
        77         44 Nothing                   Abstract               33000       220000 01-DEC-21 On Rent
       109         59 Holiday                   Landscape              25000       195000 01-NOV-21 On Rent

13 rows selected.
```

Figure 35 Displaying Art Table

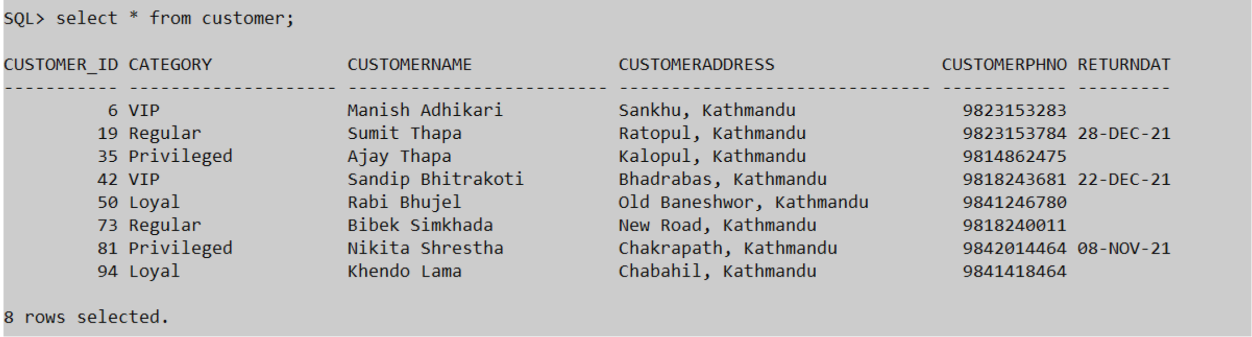
Inserting values in Customer Table



```
SQL> INSERT ALL
  2  INTO Customer VALUES (6, 'VIP', 'Manish Adhikari', 'Sankhu, Kathmandu', 9823153283, '')
  3  INTO Customer VALUES (19, 'Regular', 'Sumit Thapa', 'Ratopul, Kathmandu', 9823153784, '28-Dec-2021')
  4  INTO Customer VALUES (35, 'Privileged', 'Ajay Thapa', 'Kalopul, Kathmandu', 9814862475, '')
  5  INTO Customer VALUES (42, 'VIP', 'Sandip Bhitrakoti', 'Bhadrabas, Kathmandu', 9818243681, '22-Dec-2021')
  6  INTO Customer VALUES (50, 'Loyal', 'Rabi Bhujel', 'Old Baneshwor, Kathmandu', 9841246780, '')
  7  INTO Customer VALUES (73, 'Regular', 'Bibek Simkhada', 'New Road, Kathmandu', 9818240011, '')
  8  INTO Customer VALUES (81, 'Privileged', 'Nikita Shrestha', 'Chakrapath, Kathmandu', 9842014464, '08-Nov-2021')
  9  INTO Customer VALUES (94, 'Loyal', 'Khendo Lama', 'Chabahil, Kathmandu', 9841418464, '02-Dec-2021')
 10  SELECT * FROM DUAL;

8 rows created.
```

Figure 36 Inserting Values in Customer Table

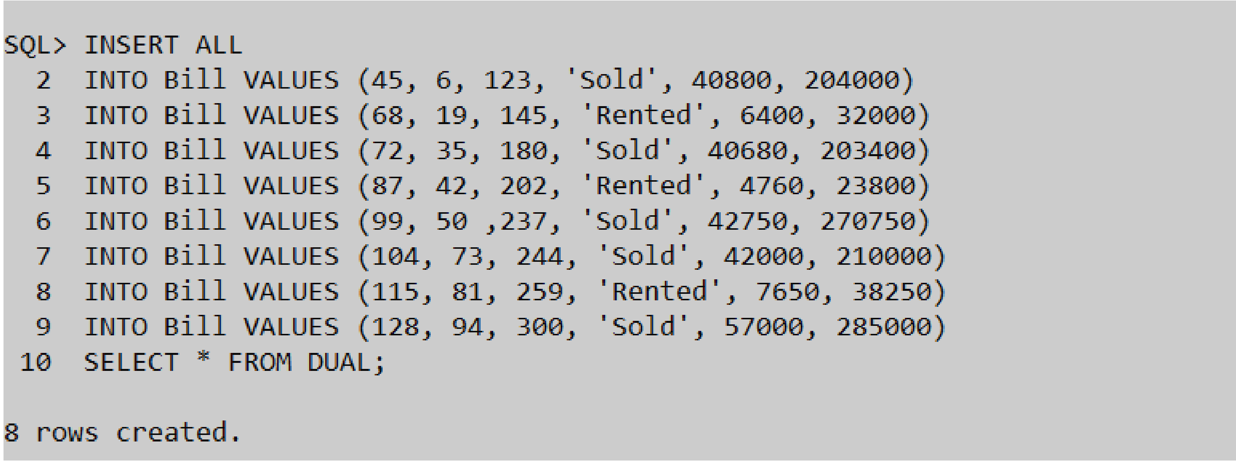
Displaying Customer Table



```
SQL> select * from customer;

CUSTOMER_ID CATEGORY             CUSTOMERNAME              CUSTOMERADDRESS                CUSTOMERPHNO RETURNDAT
----------- -------------------- ------------------------- ------------------------------ ------------ ---------
          6 VIP                  Manish Adhikari           Sankhu, Kathmandu                9823153283
         19 Regular              Sumit Thapa               Ratopul, Kathmandu               9823153784 28-DEC-21
         35 Privileged           Ajay Thapa                Kalopul, Kathmandu               9814862475
         42 VIP                  Sandip Bhitrakoti         Bhadrabas, Kathmandu             9818243681 22-DEC-21
         50 Loyal                Rabi Bhujel               Old Baneshwor, Kathmandu         9841246780
         73 Regular              Bibek Simkhada            New Road, Kathmandu              9818240011
         81 Privileged           Nikita Shrestha           Chakrapath, Kathmandu            9842014464 08-NOV-21
         94 Loyal                Khendo Lama               Chabahil, Kathmandu              9841418464

8 rows selected.
```

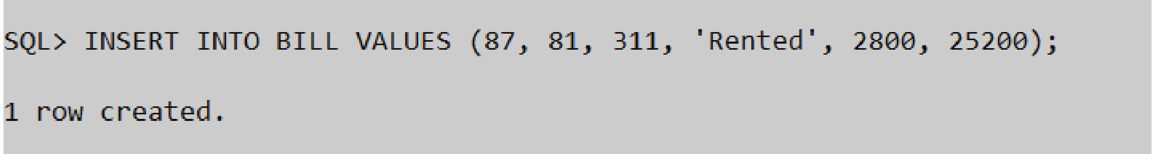
Figure 37 Displaying Customer Table

Inserting values in Bill Table



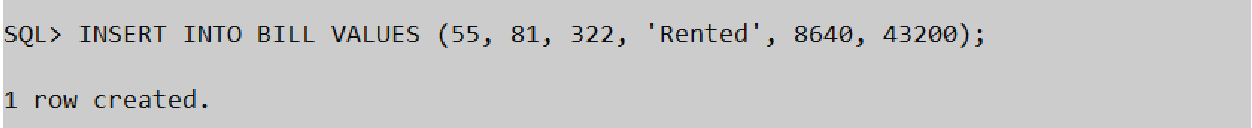
```
SQL> INSERT ALL
     INTO Bill VALUES (45, 6, 123, 'Sold', 40800, 204000)
     INTO Bill VALUES (68, 19, 145, 'Rented', 6400, 32000)
     INTO Bill VALUES (72, 35, 180, 'Sold', 40680, 203400)
     INTO Bill VALUES (87, 42, 202, 'Rented', 4760, 23800)
     INTO Bill VALUES (99, 50 ,237, 'Sold', 42750, 270750)
     INTO Bill VALUES (104, 73, 244, 'Sold', 42000, 210000)
     INTO Bill VALUES (115, 81, 259, 'Rented', 7650, 38250)
     INTO Bill VALUES (128, 94, 300, 'Sold', 57000, 285000)
     SELECT * FROM DUAL;

8 rows created.
```

Figure 38 Inserting Values in Bill Table



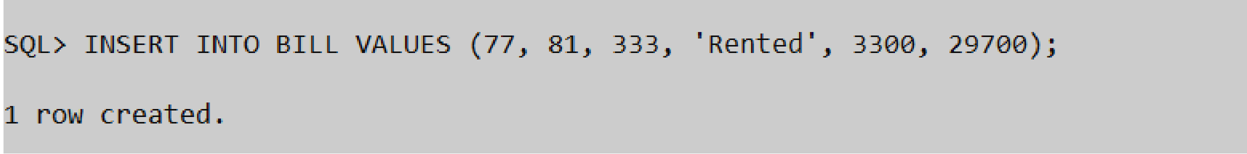
```
SQL> INSERT INTO BILL VALUES (87, 81, 311, 'Rented', 2800, 25200);

1 row created.
```

Figure 39 Inserting Values in Bill Table (2)



```
SQL> INSERT INTO BILL VALUES (55, 81, 322, 'Rented', 8640, 43200);

1 row created.
```

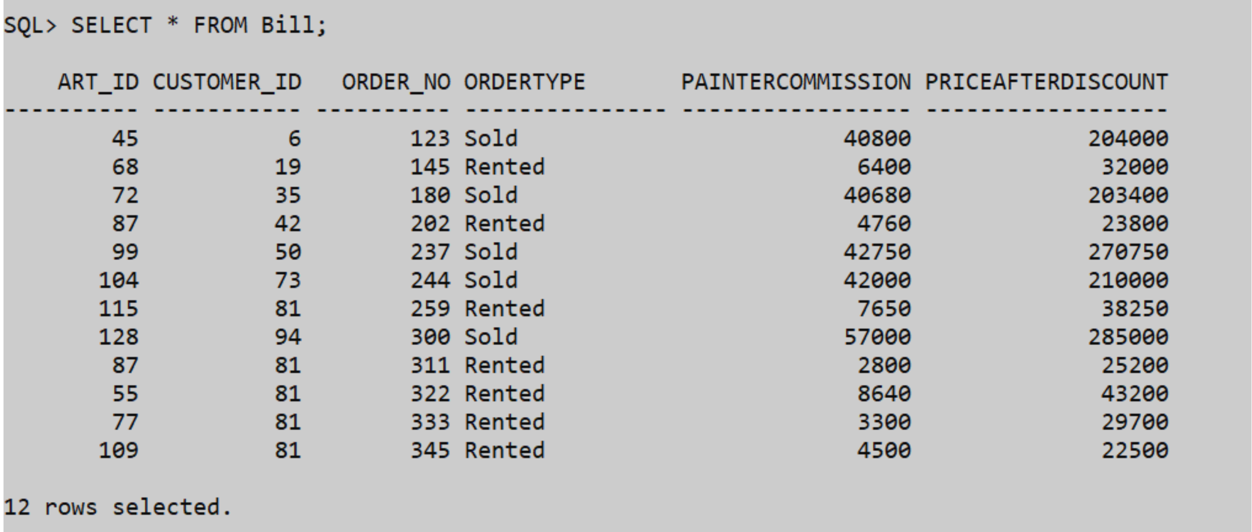
Figure 40 Inserting Values in Bill Table (3)



```
SQL> INSERT INTO BILL VALUES (77, 81, 333, 'Rented', 3300, 29700);

1 row created.
```

Figure 41 Inserting Values in Bill Table (4)

Displaying Bill Table



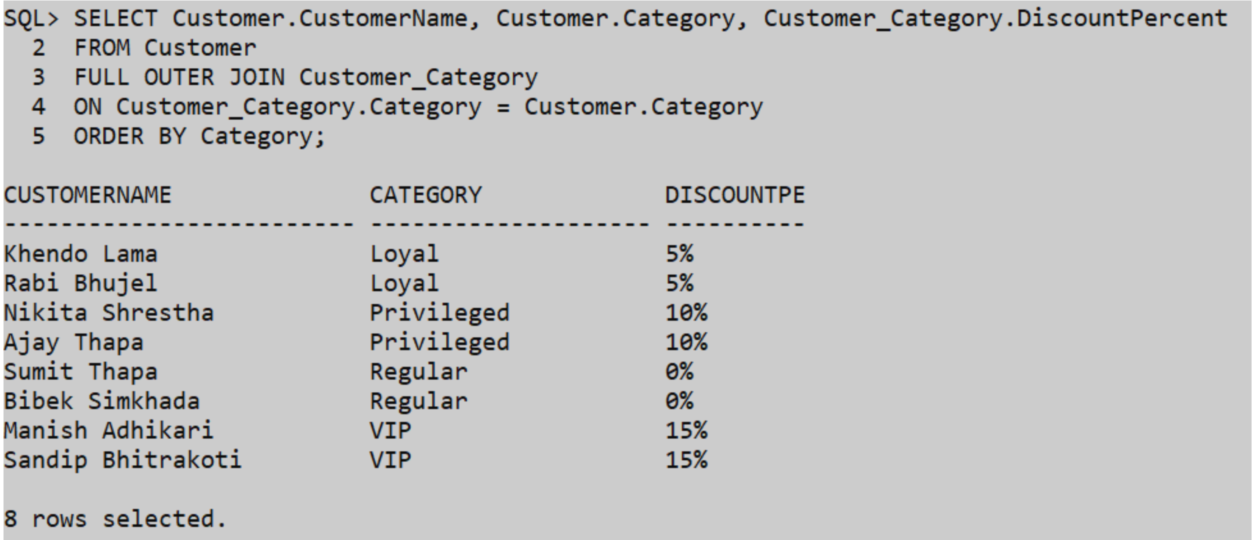
```
SQL> SELECT * FROM Bill;

    ART_ID CUSTOMER_ID   ORDER_NO ORDERTYPE       PAINTERCOMMISSION PRICEAFTERDISCOUNT
---------- ----------- ---------- --------------- ----------------- ------------------
        45           6        123 Sold                        40800             204000
        68          19        145 Rented                       6400              32000
        72          35        180 Sold                        40680             203400
        87          42        202 Rented                       4760              23800
        99          50        237 Sold                        42750             270750
       104          73        244 Sold                        42000             210000
       115          81        259 Rented                       7650              38250
       128          94        300 Sold                        57000             285000
        87          81        311 Rented                       2800              25200
        55          81        322 Rented                       8640              43200
        77          81        333 Rented                       3300              29700
       109          81        345 Rented                       4500              22500

12 rows selected.
```

Figure 42 Displaying Bill Table

6. Database Querying

Information Query 1

List all customers according to category.

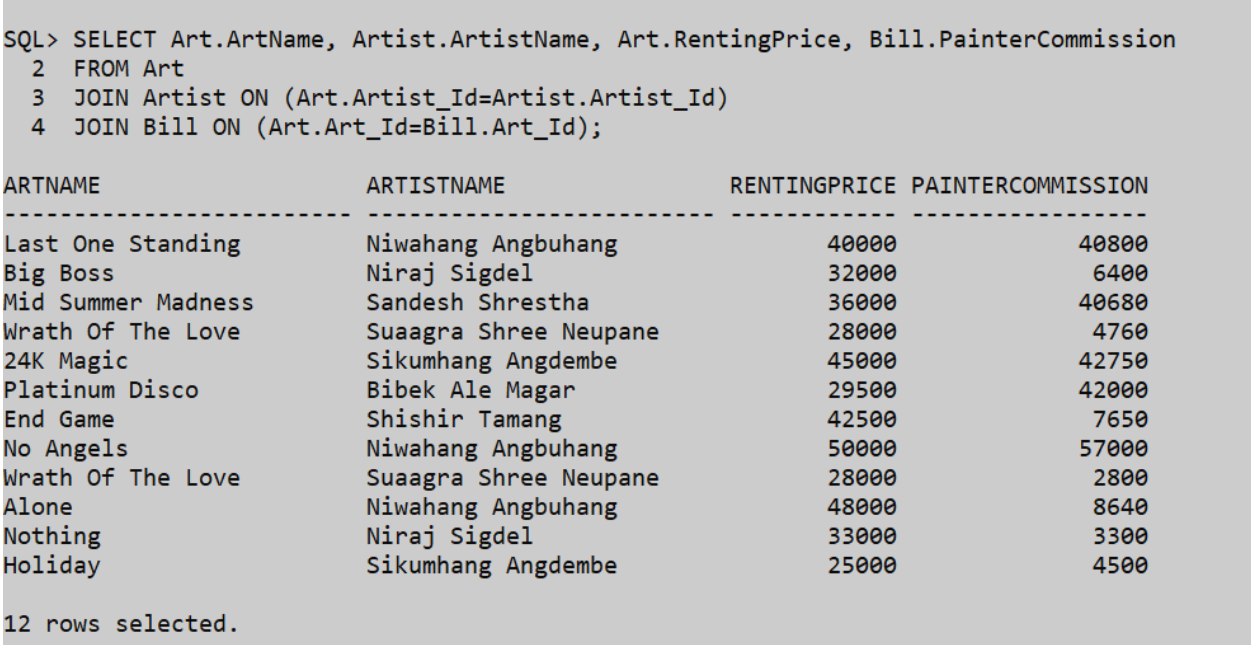


```
SQL> SELECT Customer.CustomerName, Customer.Category, Customer_Category.DiscountPercent
  2  FROM Customer
  3  FULL OUTER JOIN Customer_Category
  4  ON Customer_Category.Category = Customer.Category
  5  ORDER BY Category;

CUSTOMERNAME              CATEGORY             DISCOUNTPE
------------------------- -------------------- ----------
Khendo Lama               Loyal                5%
Rabi Bhujel               Loyal                5%
Nikita Shrestha           Privileged           10%
Ajay Thapa                Privileged           10%
Sumit Thapa               Regular              0%
Bibek Simkhada            Regular              0%
Manish Adhikari           VIP                  15%
Sandip Bhitrakoti         VIP                  15%

8 rows selected.
```

Figure 43 Information Query 1

Information Query 2

List paintings and their artist with monthly rental price and paid price.

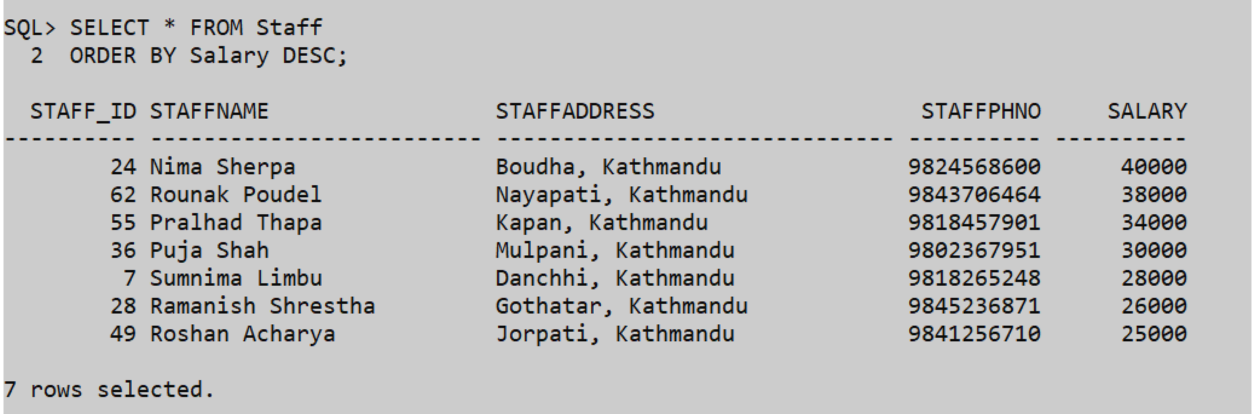


```
SQL> SELECT Art.ArtName, Artist.ArtistName, Art.RentingPrice, Bill.PainterCommission
     FROM Art
     JOIN Artist ON (Art.Artist_Id=Artist.Artist_Id)
     JOIN Bill ON (Art.Art_Id=Bill.Art_Id);

ARTNAME                   ARTISTNAME                RENTINGPRICE PAINTERCOMMISSION
------------------------- ------------------------- ------------ -----------------
Last One Standing         Niwahang Angbuhang               40000             40800
Big Boss                  Niraj Sigdel                     32000              6400
Mid Summer Madness        Sandesh Shrestha                 36000             40680
Wrath Of The Love         Suaagra Shree Neupane            28000              4760
24K Magic                 Sikumhang Angdembe               45000             42750
Platinum Disco            Bibek Ale Magar                  29500             42000
End Game                  Shishir Tamang                   42500              7650
No Angels                 Niwahang Angbuhang               50000             57000
Wrath Of The Love         Suaagra Shree Neupane            28000              2800
Alone                     Niwahang Angbuhang               48000              8640
Nothing                   Niraj Sigdel                     33000              3300
Holiday                   Sikumhang Angdembe               25000              4500

12 rows selected.
```

Figure 44 Information Query 2

Information Query 3

Show total staff in Masterpieces Limited sorted by higher salary.

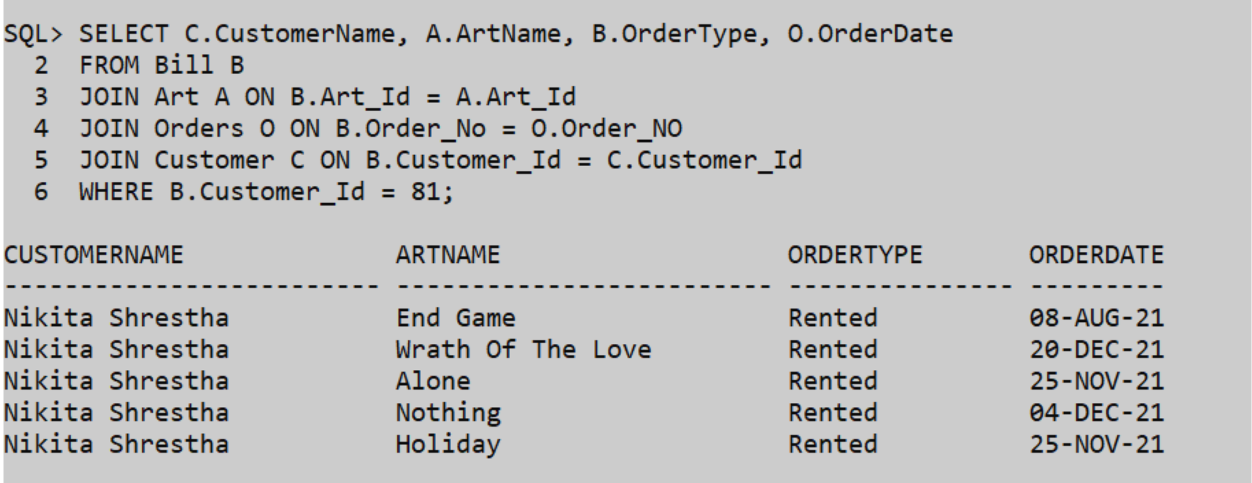


```
SQL> SELECT * FROM Staff
     ORDER BY Salary DESC;

  STAFF_ID STAFFNAME                 STAFFADDRESS                    STAFFPHNO     SALARY
---------- ------------------------- ------------------------------ ---------- ----------
        24 Nima Sherpa               Boudha, Kathmandu              9824568600      40000
        62 Rounak Poudel             Nayapati, Kathmandu            9843706464      38000
        55 Pralhad Thapa             Kapan, Kathmandu               9818457901      34000
        36 Puja Shah                 Mulpani, Kathmandu             9802367951      30000
         7 Sumnima Limbu             Danchhi, Kathmandu             9818265248      28000
        28 Ramanish Shrestha         Gothatar, Kathmandu            9845236871      26000
        49 Roshan Acharya            Jorpati, Kathmandu             9841256710      25000

7 rows selected.
```

Figure 45 Information Query 3

Information Query 4

Show paintings leased before and currently by any one customer.

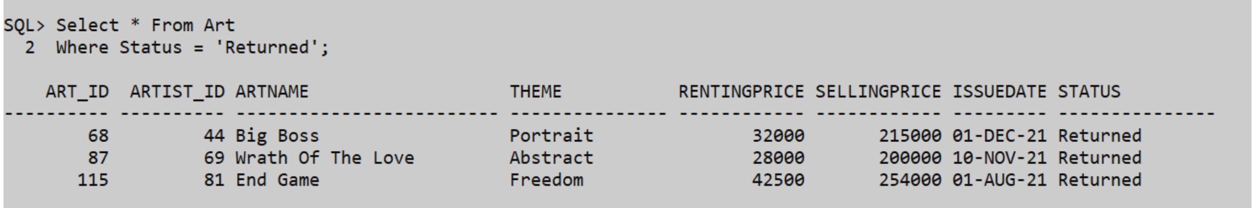


```
SQL> SELECT C.CustomerName, A.ArtName, B.OrderType, O.OrderDate
     FROM Bill B
     JOIN Art A ON B.Art_Id = A.Art_Id
     JOIN Orders O ON B.Order_No = O.Order_NO
     JOIN Customer C ON B.Customer_Id = C.Customer_Id
     WHERE B.Customer_Id = 81;

CUSTOMERNAME              ARTNAME                   ORDERTYPE       ORDERDATE
------------------------- ------------------------- --------------- ---------
Nikita Shrestha           End Game                  Rented          08-AUG-21
Nikita Shrestha           Wrath Of The Love         Rented          20-DEC-21
Nikita Shrestha           Alone                     Rented          25-NOV-21
Nikita Shrestha           Nothing                   Rented          04-DEC-21
Nikita Shrestha           Holiday                   Rented          25-NOV-21
```

Figure 46 Information Query 4

Information Query 5

List all paintings that have been returned to the owner.

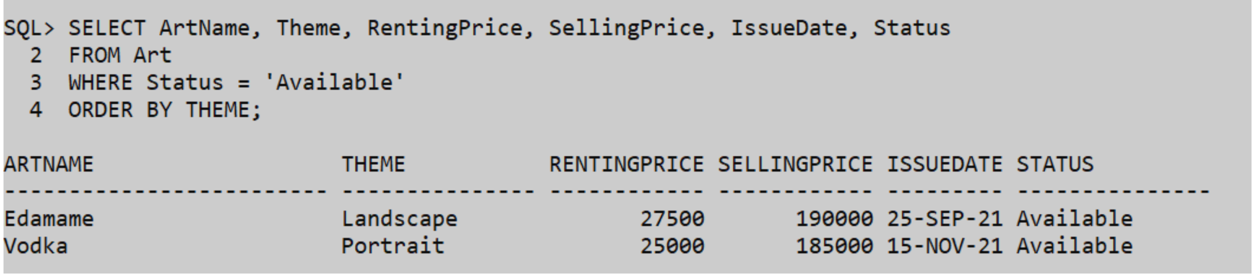


```
SQL> Select * From Art
     Where Status = 'Returned';

    ART_ID  ARTIST_ID ARTNAME                   THEME          RENTINGPRICE SELLINGPRICE ISSUEDATE STATUS
---------- ---------- ------------------------- -------------- ------------ ------------ --------- ---------------
        68         44 Big Boss                  Portrait              32000       215000 01-DEC-21 Returned
        87         69 Wrath Of The Love         Abstract              28000       200000 10-NOV-21 Returned
       115         81 End Game                  Freedom               42500       254000 01-AUG-21 Returned
```

Figure 47 Information Query 5

Transaction Query 1

List the number of paintings available for rent according to category.

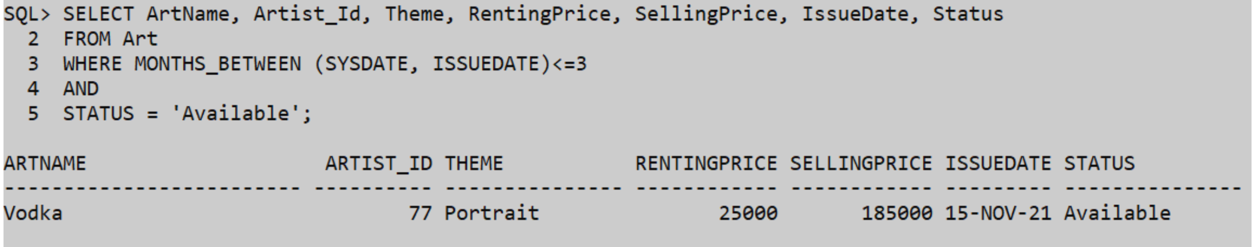


```
SQL> SELECT ArtName, Theme, RentingPrice, SellingPrice, IssueDate, Status
     FROM Art
     WHERE Status = 'Available'
     ORDER BY THEME;

ARTNAME                   THEME          RENTINGPRICE SELLINGPRICE ISSUEDATE STATUS
------------------------- -------------- ------------ ------------ --------- ---------------
Edamame                   Landscape             27500       190000 25-SEP-21 Available
Vodka                     Portrait              25000       185000 15-NOV-21 Available
```

Figure 48 Transaction Query 1

Transaction Query 2

List the details of paintings that have not been leased within three months.

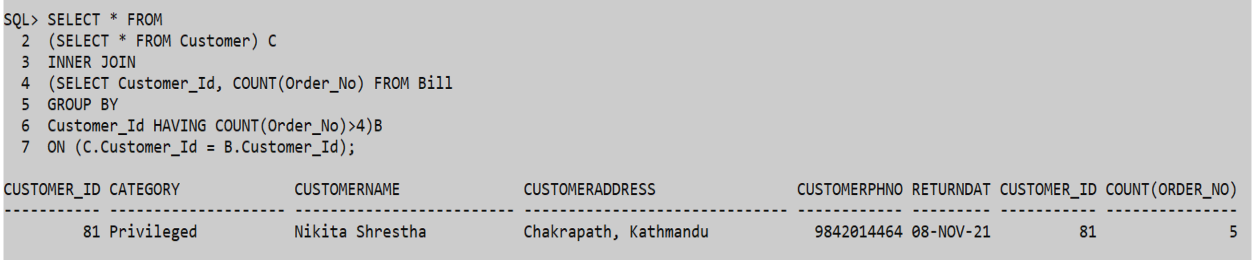


```
SQL> SELECT ArtName, Artist_Id, Theme, RentingPrice, SellingPrice, IssueDate, Status
     FROM Art
     WHERE MONTHS_BETWEEN (SYSDATE, ISSUEDATE)<=3
     AND
     STATUS = 'Available';

ARTNAME                    ARTIST_ID THEME           RENTINGPRICE SELLINGPRICE ISSUEDATE STATUS
------------------------- ---------- --------------- ------------ ------------ --------- ---------------
Vodka                             77 Portrait               25000       185000 15-NOV-21 Available
```

Figure 49 Transaction Query 2

Transaction Query 3

List the details of customers who have leased the painting more than four times.

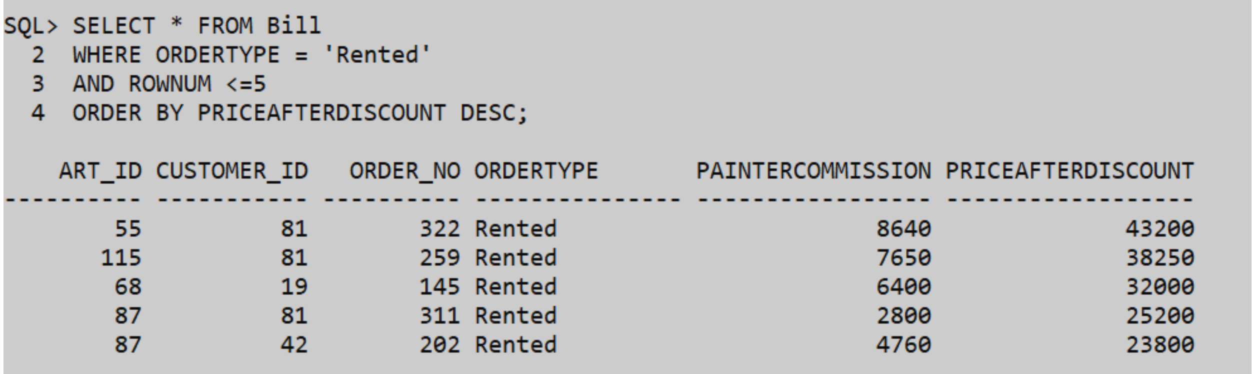


```
SQL> SELECT * FROM
     (SELECT * FROM Customer) C
     INNER JOIN
     (SELECT Customer_Id, COUNT(Order_No) FROM Bill
     GROUP BY
     Customer_Id HAVING COUNT(Order_No)>4)B
     ON (C.Customer_Id = B.Customer_Id);

CUSTOMER_ID CATEGORY             CUSTOMERNAME              CUSTOMERADDRESS                CUSTOMERPHNO RETURNDAT CUSTOMER_ID COUNT(ORDER_NO)
----------- -------------------- ------------------------- ------------------------------ ------------ --------- ----------- ---------------
         81 Privileged           Nikita Shrestha           Chakrapath, Kathmandu            9842014464 08-NOV-21          81               5
```

Figure 50 Transaction Query 3

Transaction Query 4

List top 5 paintings based on total collected rental amount.

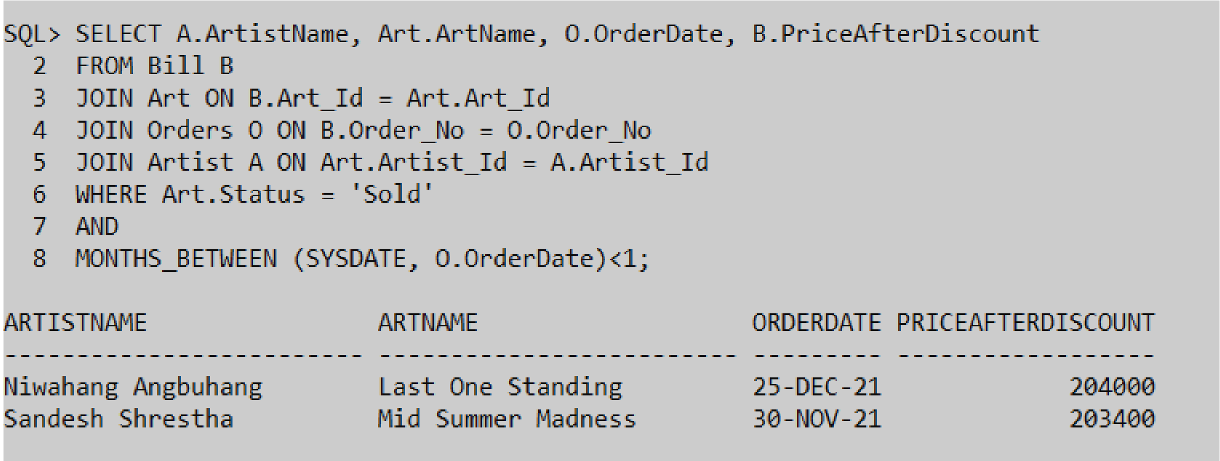


```
SQL> SELECT * FROM Bill
     WHERE ORDERTYPE = 'Rented'
     AND ROWNUM <=5
     ORDER BY PRICEAFTERDISCOUNT DESC;

    ART_ID CUSTOMER_ID   ORDER_NO ORDERTYPE       PAINTERCOMMISSION PRICEAFTERDISCOUNT
---------- ----------- ---------- --------------- ----------------- ------------------
        55          81        322 Rented                       8640              43200
       115          81        259 Rented                       7650              38250
        68          19        145 Rented                       6400              32000
        87          81        311 Rented                       2800              25200
        87          42        202 Rented                       4760              23800
```

Figure 51 Transaction Query 4

Transaction Query 5

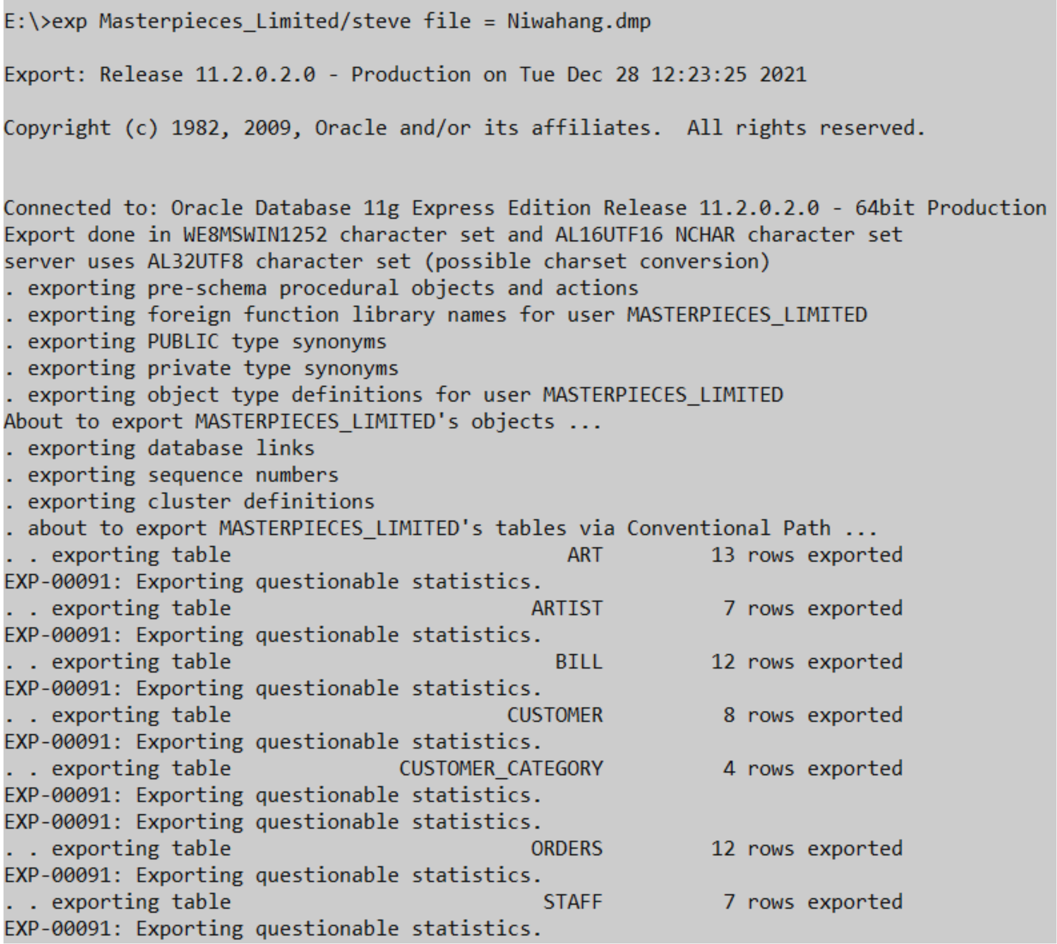
Show the name of the painter and their paintings sold value (in total) for the current month.



```
SQL> SELECT A.ArtistName, Art.ArtName, O.OrderDate, B.PriceAfterDiscount
     FROM Bill B
     JOIN Art ON B.Art_Id = Art.Art_Id
     JOIN Orders O ON B.Order_No = O.Order_No
     JOIN Artist A ON Art.Artist_Id = A.Artist_Id
     WHERE Art.Status = 'Sold'
     AND
     MONTHS_BETWEEN (SYSDATE, O.OrderDate)<1;

ARTISTNAME                ARTNAME                   ORDERDATE PRICEAFTERDISCOUNT
------------------------- ------------------------- --------- ------------------
Niwahang Angbuhang        Last One Standing         25-DEC-21             204000
Sandesh Shrestha          Mid Summer Madness        30-NOV-21             203400
```

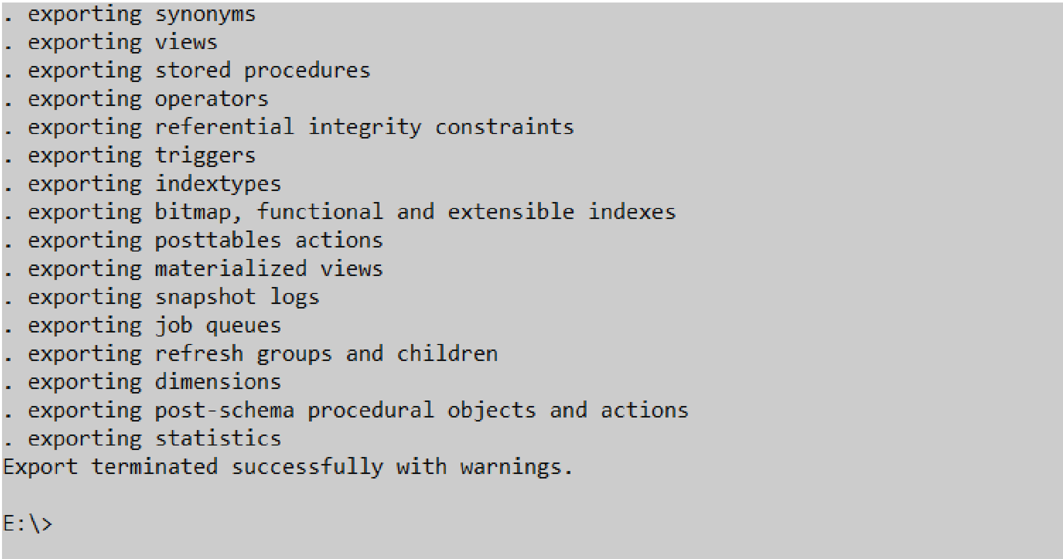
Figure 52 Transaction Query 5

Creating A Dump File



```
E:\>exp Masterpieces_Limited/steve file = Niwahang.dmp

Export: Release 11.2.0.2.0 - Production on Tue Dec 28 12:23:25 2021

Copyright (c) 1982, 2009, Oracle and/or its affiliates.  All rights reserved.


Connected to: Oracle Database 11g Express Edition Release 11.2.0.2.0 - 64bit Production
Export done in WE8MSWIN1252 character set and AL16UTF16 NCHAR character set
server uses AL32UTF8 character set (possible charset conversion)
. exporting pre-schema procedural objects and actions
. exporting foreign function library names for user MASTERPIECES_LIMITED
. exporting PUBLIC type synonyms
. exporting private type synonyms
. exporting object type definitions for user MASTERPIECES_LIMITED
About to export MASTERPIECES_LIMITED's objects ...
. exporting database links
. exporting sequence numbers
. exporting cluster definitions
. about to export MASTERPIECES_LIMITED's tables via Conventional Path ...
. . exporting table                            ART         13 rows exported
EXP-00091: Exporting questionable statistics.
. . exporting table                         ARTIST          7 rows exported
EXP-00091: Exporting questionable statistics.
. . exporting table                           BILL         12 rows exported
EXP-00091: Exporting questionable statistics.
. . exporting table                       CUSTOMER          8 rows exported
EXP-00091: Exporting questionable statistics.
. . exporting table              CUSTOMER_CATEGORY          4 rows exported
EXP-00091: Exporting questionable statistics.
EXP-00091: Exporting questionable statistics.
. . exporting table                         ORDERS         12 rows exported
EXP-00091: Exporting questionable statistics.
. . exporting table                          STAFF          7 rows exported
EXP-00091: Exporting questionable statistics.
```

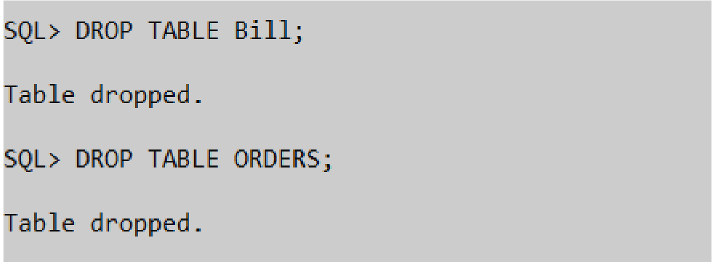
Figure 53 Creating A Dump File



```
. exporting synonyms
. exporting views
. exporting stored procedures
. exporting operators
. exporting referential integrity constraints
. exporting triggers
. exporting indextypes
. exporting bitmap, functional and extensible indexes
. exporting posttables actions
. exporting materialized views
. exporting snapshot logs
. exporting job queues
. exporting refresh groups and children
. exporting dimensions
. exporting post-schema procedural objects and actions
. exporting statistics
Export terminated successfully with warnings.

E:\>
```

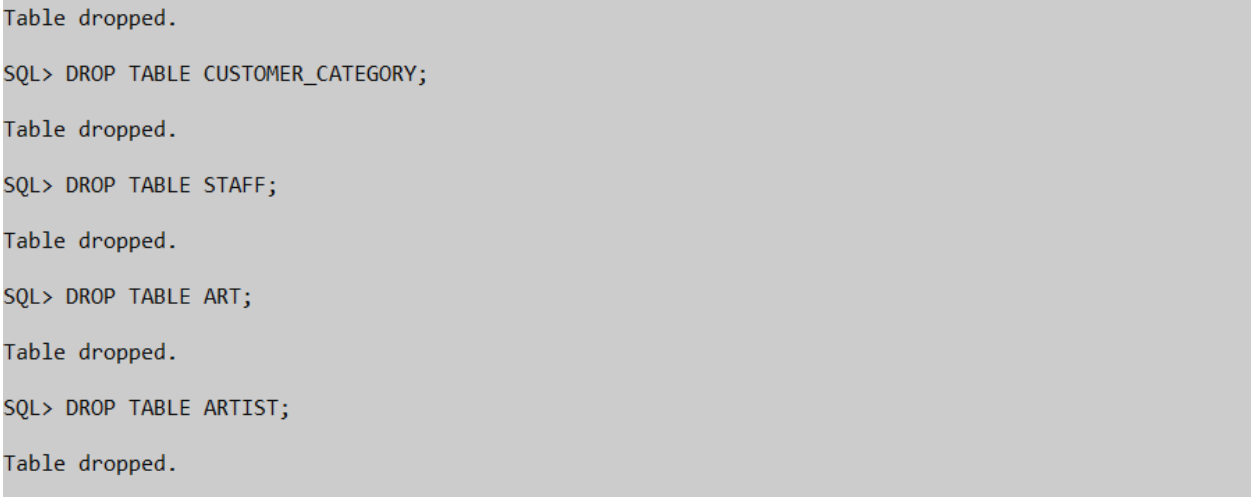
Figure 54 Creating A Dump File 2

Dropping Tables



```
SQL> DROP TABLE Bill;

Table dropped.

SQL> DROP TABLE ORDERS;

Table dropped.
```

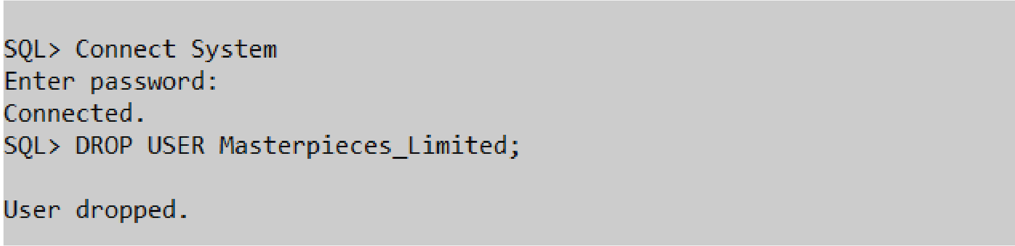
Figure 55 Dropping Tables



```
Table dropped.

SQL> DROP TABLE CUSTOMER_CATEGORY;

Table dropped.

SQL> DROP TABLE STAFF;

Table dropped.

SQL> DROP TABLE ART;

Table dropped.

SQL> DROP TABLE ARTIST;

Table dropped.
```

Figure 56 Dropping Tables (2)

Dropping User



```
SQL> Connect System
Enter password:
Connected.
SQL> DROP USER Masterpieces_Limited;

User dropped.
```

Figure 57 Dropping User

7. Critical Evaluation

For the given coursework, a database was implemented for an online business name Masterpieces Limited using Oracle database system. Database is implemented in many other modules like Programming, Software Engineering. Database is really important for a business or an application software. As database handles large amounts of data, it would be hard to handle all those data in a journal or a spreadsheet. Also, it is easy to update the data if needed. I found database to be really important for my career as there are many scope in the IT sector. This module has helped me learn about database system management, database designs, and its capability. I got to learn different SQL queries on creating and manipulating tables and data.

After completing the coursework, my knowledge in databases has improved a lot. This coursework has helped me develop my skills about database design. We were taught about database on the first semester of our course in Information Systems as well but now it was more detailed. Due to learning about the database before, it was a little easier. Although normalization was really hard and confusing when it was first taught. I was confused on some relationship for Entity Relationship Diagram. It became clear after the doing some research. In the case of normalization, Mr. Biwas Adhikari sir really helped me a lot in explaining and solving the normalization process. Even when the deadline of this coursework was approaching, he helped me solve my problems. The coursework was completed after much struggle but I have gained much more knowledge on databases.