ASSIGNMENT

***Table 1: SalesPeople***

**Snum**is Primary key

**Sname**is Unique constraint

**Snum Sname City Comm**

1001 Peel. London .12

1002 Serres Sanjose .13

1004 Motika London .11

1007 Rifkin Barcelona .15

1003 Axelrod Newyork .10

***Table 2: Customers***

**Cnum**is Primary Key

**City**has not null constraint.

**Snum**is foreign key constraint refers Snum column of SalesPeople table.

**Cnum** **Cname City Snum**

2001  Hoffman London 1001

2002  Giovanni Rome 1003

2003  Liu Sanjose 1002

2004  Grass Berlin 1002

2006 Clemens London 1001

2008 Cisneros Sanjose 1007

2007 Pereira Rome 1004

***Table 3: Orders***

**Onum**is Primary key

**Cnum**is foreign key refers to Cnum column of Customers table. **Snum**is foreign key refers Snum column of SalesPeople table.

**Onum Amt Odate Cnum Snum**

3001 18.69 3-10-1990 2008 1007

3003 767.19 3-10-1990 2001 1001

3002 1900.10 3-10-1990 2007 1004

3005  5160.45 3-10-1990 2003 1002

3006  1098.16 3-10-1990 2008 1007

3009 1713.23 4-10-1990 2002 1003

3007  75.75 4-10-1990 2004 1002

3008  4273.00 5-10-1990 2006 1001

3010  1309.95 6-10-1990 2004 1002

3011  9891.88 6-10-1990 2006 1001

🡪First of all, we create SalesPeople, Customers and Orders tables and insert values into them.

🡪We create tables with the given relations between the tables.

-----------------------TABLE 1(SalesPeople) Creation and insertion-------------------------

CREATE TABLE SalesPeople (

Snum INT PRIMARY KEY,

Sname VARCHAR(25) UNIQUE,

City VARCHAR(25),

Comm DECIMAL(4, 2)

);

------------------ Insert data into SalesPeople table-----------------

INSERT INTO SalesPeople (Snum, Sname, City, Comm)

VALUES

(1001, 'Peel', 'London', 0.12),

(1002, 'Serres', 'Sanjose', 0.13),

(1004, 'Motika', 'London', 0.11),

(1007, 'Rifkin', 'Barcelona', 0.15),

(1003, 'Axelrod', 'Newyork', 0.10);

SalesPeople

|  |  |  |  |
| --- | --- | --- | --- |
| **Snum** | **Sname** | **City** | **Comm** |
| 1001 | Peel | London | 0.12 |
| 1002 | Serres | Sanjose | 0.13 |
| 1004 | Motika | London | 0.11 |
| 1007 | Rifkin | Barcelona | 0.15 |
| 1003 | Axelrod | Newyork | 0.1 |

-----------------------TABLE 2(Customers) Creation and insertion-------------------------

CREATE TABLE Customers (

Cnum INT PRIMARY KEY,

Cname VARCHAR(25),

City VARCHAR(25) NOT NULL,

Snum INT,

FOREIGN KEY (Snum) REFERENCES SalesPeople(Snum)

);

---------------- Insert data into Customers table-----------------

INSERT INTO Customers (Cnum, Cname, City, Snum)

VALUES

(2001, 'Hoffman', 'London', 1001),

(2002, 'Giovanni', 'Rome', 1003),

(2003, 'Liu', 'Sanjose', 1002),

(2004, 'Grass', 'Berlin', 1002),

(2006, 'Clemens', 'London', 1001),

(2008, 'Cisneros', 'Sanjose', 1007),

(2007, 'Pereira', 'Rome', 1004);

Customers

|  |  |  |  |
| --- | --- | --- | --- |
| **Cnum** | **Cname** | **City** | **Snum** |
| 2001 | Hoffman | London | 1001 |
| 2002 | Giovanni | Rome | 1003 |
| 2003 | Liu | Sanjose | 1002 |
| 2004 | Grass | Berlin | 1002 |
| 2006 | Clemens | London | 1001 |
| 2008 | Cisneros | Sanjose | 1007 |
| 2007 | Pereira | Rome | 1004 |

-----------------------TABLE 3(Orders) Creation and insertion-------------------------

CREATE TABLE Orders (

Onum INT PRIMARY KEY,

Amt DECIMAL(8, 2),

Odate DATE,

Cnum INT,

Snum INT,

FOREIGN KEY (Cnum) REFERENCES Customers(Cnum),

FOREIGN KEY (Snum) REFERENCES SalesPeople(Snum)

);

---------------- Insert data into Orders table-----------------

INSERT INTO Orders (Onum, Amt, Odate, Cnum, Snum)

VALUES

(3001, 18.69, '1990-10-03', 2008, 1007),

(3003, 767.19, '1990-10-03', 2001, 1001),

(3002, 1900.10, '1990-10-03', 2007, 1004),

(3005, 5160.45, '1990-10-03', 2003, 1002),

(3006, 1098.16, '1990-10-03', 2008, 1007),

(3009, 1713.23, '1990-10-04', 2002, 1003),

(3007, 75.75, '1990-10-04', 2004, 1002),

(3008, 4273.00, '1990-10-05', 2006, 1001),

(3010, 1309.95, '1990-10-06', 2004, 1002),

(3011, 9891.88, '1990-10-06', 2006, 1001);

Orders

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Onum** | **Amt** | **Odate** | **Cnum** | **Snum** |
| 3001 | 18.69 | 03-10-1990 | 2008 | 1007 |
| 3003 | 767.19 | 03-10-1990 | 2001 | 1001 |
| 3002 | 1900.1 | 03-10-1990 | 2007 | 1004 |
| 3005 | 5160.45 | 03-10-1990 | 2003 | 1002 |
| 3006 | 1098.16 | 03-10-1990 | 2008 | 1007 |
| 3009 | 1713.23 | 04-10-1990 | 2002 | 1003 |
| 3007 | 75.75 | 04-10-1990 | 2004 | 1002 |
| 3008 | 4273 | 05-10-1990 | 2006 | 1001 |
| 3010 | 1309.95 | 06-10-1990 | 2004 | 1002 |
| 3011 | 9891.88 | 06-10-1990 | 2006 | 1001 |

🡪On the basis of above tables perform given below questions

**1) Count the number of Salesperson whose name begin with ‘a’/’A’.**

***Query:***

SELECT Sname, COUNT(\*) AS count\_of\_Sales\_People\_starting\_with\_a\_or\_A

FROM SalesPeople

WHERE Sname LIKE 'a%' OR Sname LIKE 'A%'

***Output:***

|  |  |
| --- | --- |
| **Sname** | **count\_of\_Sales\_People\_starting\_with\_a\_or\_A** |
| Axelrod | 1 |

**2)**  **Display all the Salesperson whose all orders worth is more than Rs. 2000.**

This query joins the SalesPeople table with the Orders table based on the common **Snum** column. It then groups the results by **Snum** and **Sname** and applies a condition using the **HAVING** clause to filter the salespeople whose total order amount is greater than Rs. 2000.

**Query:**

SELECT S.Snum, S.Sname

FROM SalesPeople S

JOIN Orders O ON S.Snum = O.Snum

GROUP BY S.Snum, S.Sname

HAVING SUM(O.Amt) > 2000;

**Output:**

|  |  |
| --- | --- |
| **Snum** | **Sname** |
| 1001 | Peel |
| 1002 | Serres |

**3) Count the number of Salesperson belonging to *Newyork*.**

**Query:**

SELECT City, COUNT(\*) AS Sales\_people\_from\_Newyork

FROM SalesPeople

WHERE City = 'Newyork';

**Output:**

|  |  |
| --- | --- |
| **City** | **Sales\_people\_from\_Newyork** |
| Newyork | 1 |

**4) Display the number of Salespeople belonging to*London* and belonging to*Paris***.

**Query:**

SELECT City, COUNT(\*) AS Number\_Of\_Sales\_people

FROM SalesPeople

WHERE City IN ('London', 'Paris')

GROUP BY City;

**Output:**

|  |  |
| --- | --- |
| **City** | **Number\_Of\_Sales\_people** |
| London | 2 |

**5) Display the number of orders taken by each Salesperson and their date of orders.**

This query joins the **SalesPeople** and **Orders** tables on the **Snum** column. It then groups the results by the salesperson's name (Sname) and the order date (Odate). The **COUNT()** function is used to count the number of orders (Onum) for each salesperson and order date.

**Query:**

SELECT S.Sname, COUNT(O.Onum) AS Order\_Count, O.Odate

FROM SalesPeople S

JOIN Orders O ON S.Snum = O.Snum

GROUP BY S.Sname, O.Odate**;**

**Output:**

|  |  |  |
| --- | --- | --- |
| **Sname** | **Order\_Count** | **Odate** |
| Axelrod | 1 | 04-10-1990 |
| Motika | 1 | 03-10-1990 |
| Peel | 1 | 03-10-1990 |
| Peel | 1 | 05-10-1990 |
| Peel | 1 | 06-10-1990 |
| Rifkin | 2 | 03-10-1990 |
| Serres | 1 | 03-10-1990 |
| Serres | 1 | 04-10-1990 |
| Serres | 1 | 06-10-1990 |