Problem 1:

Create Customer Table:

*Create table Customer (name varchar (20), gender char, Maritial\_Status Varchar (10),DOB date, City varchar (20), Credit\_limit Number);*

Insert the Records:

*INSERT INTO Customer VALUES ('Nancy','F','Married',null, 'New York', 8000);*

*INSERT INTO Customer VALUES ('Lewis','M','Unmarried',null, 'SFO', 10000);*

*INSERT INTO Customer VALUES ('Aiko','F','Married',null, 'Tokyo', 6000);*

*INSERT INTO Customer VALUES ('Alen','M','Unmarried',null, 'Sydney', 8000);*

*INSERT INTO Customer VALUES ('Sandey','F','Unmarried',null, 'Sydney', 8000);*

*INSERT INTO Customer VALUES ('Nick','M','Unmarried',null, 'New York', 8000);*

*INSERT INTO Customer VALUES ('Glen','F','Married',null, 'Sydney', 6500);*

*INSERT INTO Customer VALUES ('Haruto','M','Unmarried',null, 'Tokyo', 5000);*

Verify Data: SELECT \* FROM customer;

Questions:

1. Find the average credit limit of male customers residing in ‘Tokyo’ or ‘Sydney’.
2. List the customers who do not have a credit limit and were born before 1980.

Select \* from Customer where creditlimit not

1. Find the number of customers, in each city, whose marital status is ‘married’.
2. Find the Total credit limit of that city who has maximum married male customers.
3. Find the number of customers, in each city, whose credit limit is more than the average credit limit of all the customers.
4. Find the % of married customers who do not live in Sydney.

Problem 2:

Create Employee Table

Create table Employee (employee\_id int, first\_name varchar[20], last\_name varchar[20], manager\_id int, DEPTID INT, SALARY DECIMAL);

Insert the Records`

insert into employee values (4529,'Nancy','Young',4125, 1,2000);

insert into employee values (4238,'John','Simon',4329,3,3500);

insert into employee values (4329,'Martina','Candreva',4125,5,5500);

insert into employee values (4009,'Klaus','Koch',4329,2,3000);

insert into employee values (4125,'Mafalda','Ranieri',NULL,1,2200);

insert into employee values (4500,'Jakub','Hrabal',4529,1,1800);

insert into employee values (4118,'Moira','Areas',4952,5,4900);

insert into employee values (4012,'Jon','Nilssen',4952,4,2200);

insert into employee values (4952,'Sandra','Rajkovic',4529,4,4000);

insert into employee values (4444,'Seamus','Quinn',4329,4,4200);

Verify Data: SELECT \* FROM employee;

Create Department Table

Create table Dept (DeptId int, DeptName varchar (20) );

Insert the Records

insert into Dept values (1,'HR');

insert into Dept values (2,'Sales');

insert into Dept values (3,'Finance');

insert into Dept values (4,'IT');

insert into Dept values (5,' Marketing ');

Verify Data: Select \* from Dept;

Questions:

1. Show all first and last name of all employees, their managers first and last name and Department names of employees whose managers do not work in ‘Sales’ department and managers salary is more than the average salary of 'Marketing' and 'Sales'.
2. show all the employees first name, last name, their departments, salaries, and the average salary in their respective department. Order the result by department.
3. show all the employees’ first name, last name, their departments, salaries, and the highest salary in their respective department. Order the result by department.

Problem 3:

Create table Product:

CREATE TABLE Product (Pdt\_id number, suppl\_id number, pdt\_status varchar (20), list\_price number, min\_price number);

Insert the data:

Insert into Product values (1797, 102094, 'orderable',349,288);

Insert into Product values (2254, 102071, 'obsolete',453,371);

Insert into Product values (2382, 102050, 'under development',850,731);

Insert into Product values (2459, 102099, 'under development',699,568);

Insert into Product values (3127, 102087, 'orderable',498,444);

Insert into Product values (3353, 102071, 'obsolete',489,413);

Insert into Product values (3354, 102066, 'orderable',543,478);

Verify the data: SELECT \* FROM Product;

Questions:

1. Display the Product id, Product status, their minimum price and max price as $500.;
2. Display all the products whose minimum list prices are more than average list price of products having the status orderable.
3. Display the number of products whose list prices are more than the average list price.

Problem 4:

Create table Orders:

Create table Orders (Order\_No Number ,order\_date Date, order\_amount Number, customer\_id Number);

Insert the data:

Insert into Orders values (10,'10/1/2021',42.12,3);

Insert into Orders values (20,'08/5/2022',14.10,4);

Insert into Orders values (30,'01/1/2023',42.12,2);

Insert into Orders values (40,'02/14/2023',42.15,2);

Insert into Orders values (50,'04/18/2023',42.44,4);

Insert into Orders values (60,'04/24/2023',88.44,1);

Verify data:                   SELECT \* FROM Orders;

Create table Customer:

Create table CUST (custid number, custname varchar (10));

Insert the data:

Insert into cust values (1,'James');

Insert into cust values (3,'Smith');

Insert into cust values (3,'Glen');

Insert into cust values (4,'Mike');

Insert into cust values (5,'Alen');

Verify data:                     SELECT \* FROM CUST;

Questions:

1. Write a query to show the average number of orders per customer.
2. How many days in a row each customer places an order.

=====================================================================================

Swetha answers:

Problem 1:

Create Customer Table:

*Create table Customer (name varchar (20), gender char, Maritial\_Status Varchar (10),DOB date, City varchar (20), Credit\_limit Number);*

Insert the Records:

*INSERT INTO Customer VALUES ('Nancy','F','Married',null, 'New York', 8000);*

*INSERT INTO Customer VALUES ('Lewis','M','Unmarried',null, 'SFO', 10000);*

*INSERT INTO Customer VALUES ('Aiko','F','Married',null, 'Tokyo', 6000);*

*INSERT INTO Customer VALUES ('Alen','M','Unmarried',null, 'Sydney', 8000);*

*INSERT INTO Customer VALUES ('Sandey','F','Unmarried',null, 'Sydney', 8000);*

*INSERT INTO Customer VALUES ('Nick','M','Unmarried',null, 'New York', 8000);*

*INSERT INTO Customer VALUES ('Glen','F','Married',null, 'Sydney', 6500);*

*INSERT INTO Customer VALUES ('Haruto','M','Unmarried',null, 'Tokyo', 5000);*

Verify Data: SELECT \* FROM customer;

Questions:

1.Find the average credit limit of male customers residing in ‘Tokyo’ or ‘Sydney’.

select avg(credit\_limit) from customer where gender="M" and city in ('Tokyo','Sydney');

2.List the customers who do not have a credit limit and were born before 1980.

select \* from customer where credit\_limit is null and date(DOB) < "01-01-1980";

3.Find the number of customers, in each city, whose marital status is ‘married’.

select City, count(\*) from Customer

where Maritial\_Status="Married"

group by 1 order by 2 desc

4.Find the Total credit limit of that city who has maximum married male customers.

select sum(credit\_limit) from Customer

where city in (select city from customer where

               Maritial\_Status="Married" and gender="M"

               group by 1 order by count(\*) desc limit 1)

5.Find the number of customers, in each city, whose credit limit is more than the average credit limit of all the customers.

select sum(credit\_limit) from Customer

where city in (select city from customer where

               Maritial\_Status="Married" and gender="M"

               group by 1 order by count(\*) desc limit 1)

6.Find the % of married customers who do not live in Sydney.

select count(name) from customer where maritial\_status="Married" and city!="Sydney"

select e1.first\_name,e1.last\_name, e2.first\_name as mgr\_first,e2.last\_name as mgr\_last, d.deptname from employee e1

join employee e2 on e2.manager\_id=e1.employee\_id

join Dept d

on e2.deptid=d.DeptId

where d.deptname not in ('Sales') and e2.salary >

(select avg(e.salary) from Employee e join dept d on e.deptid=d.deptid where d.deptname in ('Marketing','Sales') );

create table dept\_avg as

select deptid,avg(salary) as avg\_sal from Employee e

group by deptid order by 2 desc;

group by 1 order by 2 desc;

select e.first\_name,e.last\_name,e.deptid,e.salary,d2.avg\_sal

from employee e join dept\_avg d2

on e.deptid=d2.deptid;

             create table dept\_max as

select deptid,max(salary) as max\_sal from Employee e

group by deptid order by 2 desc;

select e.first\_name,e.last\_name,e.deptid,e.salary,d2.max\_sal

from employee e join dept\_max d2

on e.deptid=d2.deptid

select pdt\_id,pdt\_status,min\_price, 500 as max\_price from Product;

select pdt\_id,pdt\_status,min\_price, 500 as max\_price from Product;

select \* from Product where list\_price> (select avg(list\_price) from Product where pdt\_status="orderable");

 select count(\*) from Product where list\_price> (select avg(list\_price) from Product)

select c.custname, count(o.order\_no)

from cust c join orders O

on c.custid=o.customer\_id

group by 1 order by 2 desc;