1. Create a table EMPLOYEE with following schema:
   (Emp_no, E_name, E_address, E_ph_no, Dept_no, Dept_name, Job_id, Salary)
   a. Add a new column; HIREDATE to the existing relation.
   b. Change the datatype of JOB_ID from char to varchar2.
   c. Change the name of column/field Emp_no to E_no.
   d. Modify the column width of the job field of emp table.

1)

```
Create table Employee (Emp_no Int primary key,
                       E-name varchar2(50) not null,
                       E-address varchar2(225),
                       E-ph_no varchar2(15),
                       Dept_no Int,
                       Dept-name varchar2(50),
                       Job_id ~~varchar~~ char(10),
                       Salary decimal(10,2));
```

a) Add a new Column Hiredate:

```
alter table employee add hiredate date;
```

b) Change the datatype of job_id to varchar2:

```
alter table employee modify job_id varchar2(10);
```

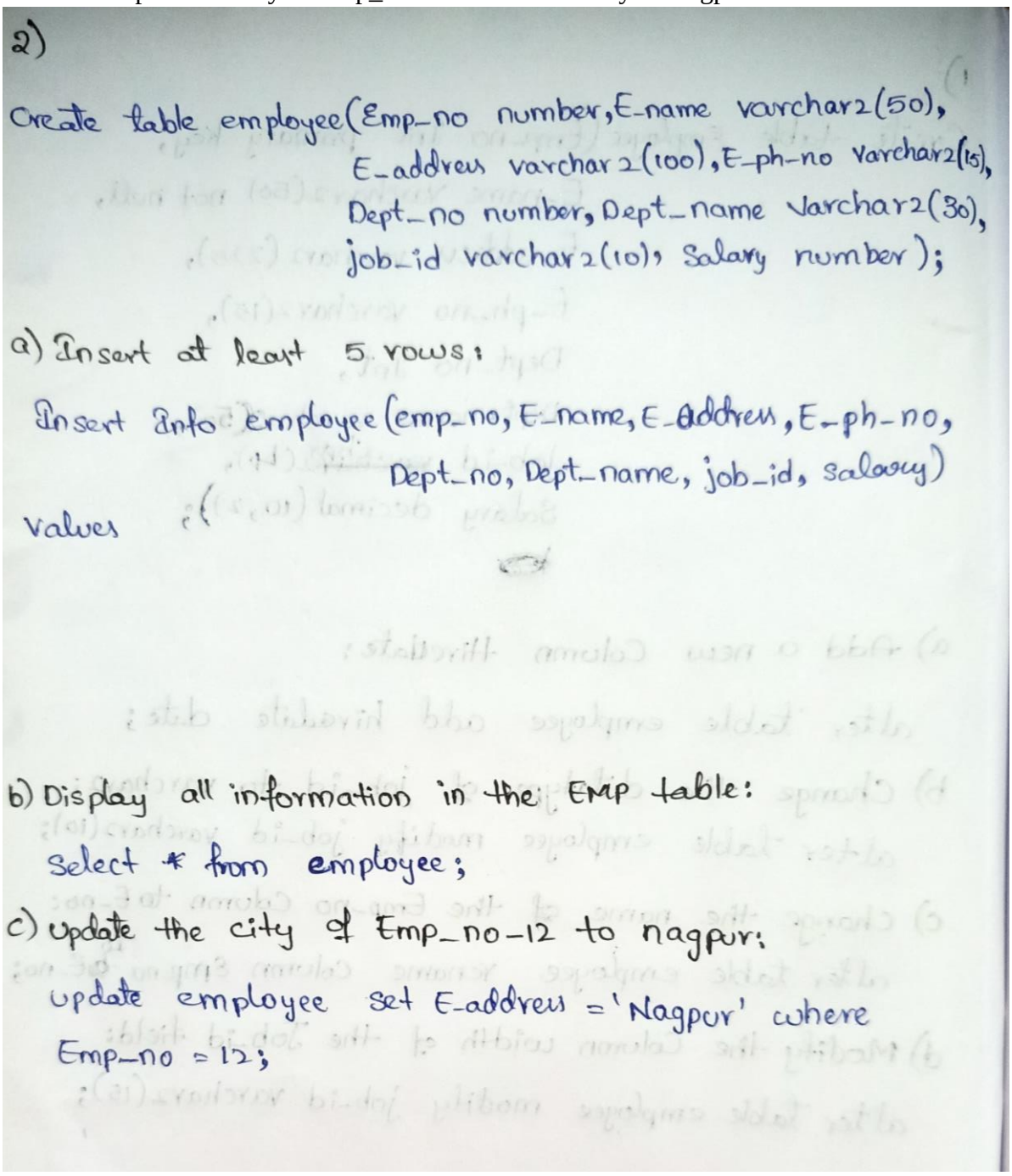
c) Change the name of the Emp_no Column to E_no:

```
alter table employee rename column Emp_no to E_no;
```

d) Modify the Column width of the Job_id field:

```
alter table employee modify job_id varchar2(15);
```

2. Create a table EMPLOYEE with following schema: (Emp_no, E_name, E_address, E_ph_no, Dept_no, Dept_name, Job_id, Salary)
   Write SQL queries for following question:
   a. Insert a least 5 rows in the table.
   b. Display all the information of EMP table.
   c. Update the city of Emp_no-12 with current city as Nagpur.



2)

```
Create table employee(Emp_no number,E_name varchar2(50),
                      E_address varchar2(100),E_ph_no varchar2(15),
                      Dept_no number, Dept_name varchar2(30),
                      job_id varchar2(10), Salary number);
```

a) Insert at least 5 rows:

```
Insert into employee(emp_no, E_name, E_Address, E_ph_no,
                     Dept_no, Dept_name, job_id, salary)
values
```

b) Display all information in the EMP table:

```
Select * from employee;
```

c) Update the city of Emp_no-12 to nagpur:

```
update employee set E_address = 'Nagpur' where
Emp_no = 12;
```

3. Create a table EMPLOYEE with following schema: (Emp_no, E_name, E_address, E_ph_no, Dept_no, Dept_name, Job_id, Salary)
   Write SQL queries for following question:
   a. Display the details of Employee who works in department MECH.
   b. Delete the email_id of employee James.
   c. Display the complete record of employees working in SALES Department.

3)

```
Create table employee (emp_no Int primary key,
                       E_name varchar2(50),
                       E_address varchar2(255),
                       E_ph_no varchar2(15),
                       Dept_no Int,
                       Dept_name varchar2(50),
                       Job_id varchar2(10),
                       Salary decimal(10,2));
```

a) Display the details of employee in the MECH department:

```
Select * from employee where Dept_name='MECH';
```

b) Delete the email_id of employee james (assuming email_id is stored in a column named "E_email"):

```
update employee set E_email = null where E_name
                                    ='James';
```

c) Display the complete record of employees working in the SALES department:

```
select * from employee where Dept_name='Sales';
```

4. Create a table EMPLOYEE with following schema:

| E_id | E_name | Age | Salary |
|---|---|---|---|
| 101 | Anu | 22 | 9000 |
| 102 | Shane | 29 | 8000 |
| 103 | Rohan | 34 | 6000 |
| 104 | Scott | 44 | 10000 |
| 105 | Tiger | 35 | 8000 |
| 106 | Alex | 27 | 7000 |
| 107 | Abhi | 29 | 8000 |

Write SQL queries for following question:

a. Count number of employee names from employee table.
b. Find the Maximum age from employee table.
c. Find the Minimum age from employee table.

4)

```
Create table employee( E_id int primary key,
                       E_name varchar (255),
                       Age Int , Salary int );

Insert into employee (ID, E_name ,age, salary ) values
    (101, 'Anu' , 22 , 9000),
    (102, 'Shane', 29, 8000),
    (103, 'Rohan', 34, 6000),
    (104, 'Scott', 35, 10000),
    (105, 'Tiger', 27, 8000),
    (106, 'Alex' , 29 , 7000),
    (107, 'Abhi', 28, 8000);
```

a) Count the number of employee names:

```
Select Count (E_name) as number_of_employee from
Employee;
```

b) find the maximum age:

```
Select max(age) as maximum_age from employee;
```

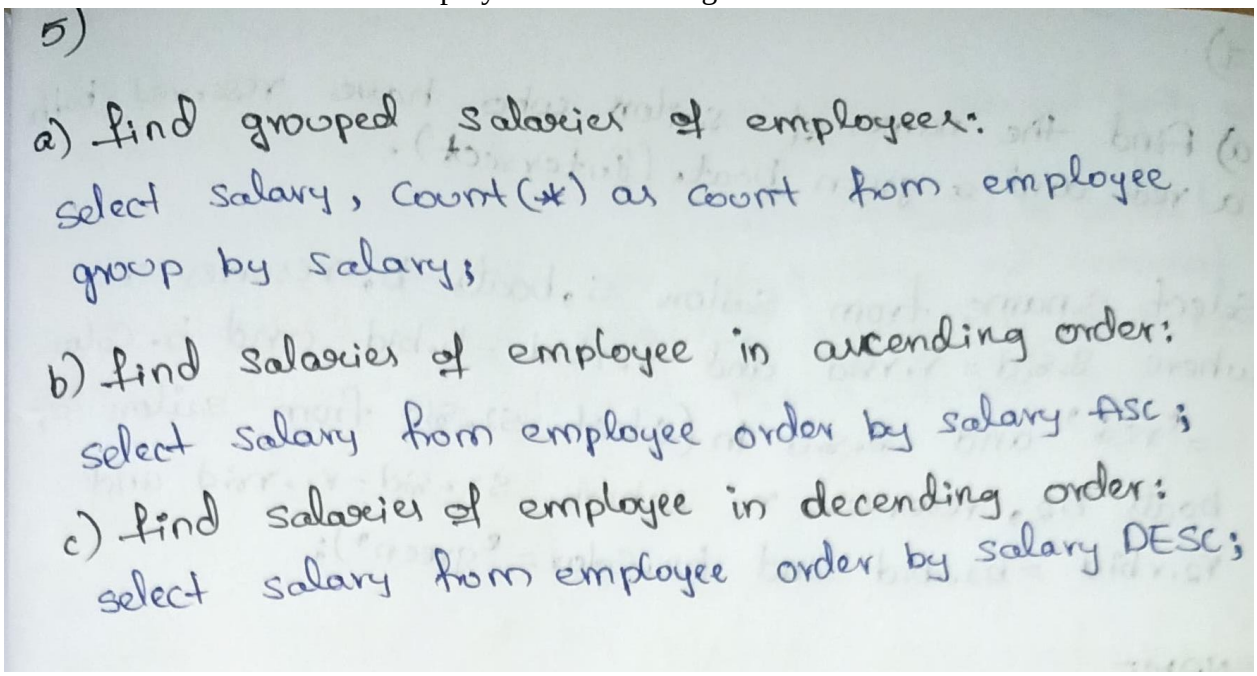
c) find the minimum age:

```
Select min(age) as minimum_age from employee;
```

5. Create a table EMPLOYEE with following schema:

| E_id | E_name | Age | Salary |
|---|---|---|---|
| 101 | Anu | 22 | 9000 |
| 102 | Shane | 29 | 8000 |
| 103 | Rohan | 34 | 6000 |
| 104 | Scott | 44 | 10000 |
| 105 | Tiger | 35 | 8000 |
| 106 | Alex | 27 | 7000 |
| 107 | Abhi | 29 | 8000 |

   a. Find grouped salaries of employees. (group by clause)
   b. Find salaries of employee in Ascending Order. (order by clause)
   c. Find salaries of employee in Descending Order.



6. Create a table EMPLOYEE with following schema:

| EMPNO | ENAME | JOB | MANAGER_NO | SAL | COMMISSION |
|---|---|---|---|---|---|
| 101 | abhi | manager | 1234 | 1100 | 70 |
| 102 | rohith | analyst | 2345 | 9000 | 65 |
| 103 | david | trainee | 3456 | 9000 | 65 |
| 104 | rahul | clerk | 4567 | 7000 | 55 |

a. Insert the any three records in the employee table and use rollback. Check the result.
b. Add primary key constraint and not null constraint to the employee table.
c. Insert null values to the employee table and verify the result.

6)

```
Create table employee( Empno number,
                       Ename varchar2(20),
                       Job varchar2(20),
                       manager_no number,
                       sal number,
                       Commision number );
```

a) Insert three records and rollback:

Insert the given values in the given table

```
rollback;
```

b) Add Constraints:

```
alter table employee add constraint pk_empno
Primary key(empno), modify ename varchar2(20) not
null;
```

c) insert null values:

```
insert into employee values (104, NULL, 'clerk', 4567,
                             7000, 55);
```

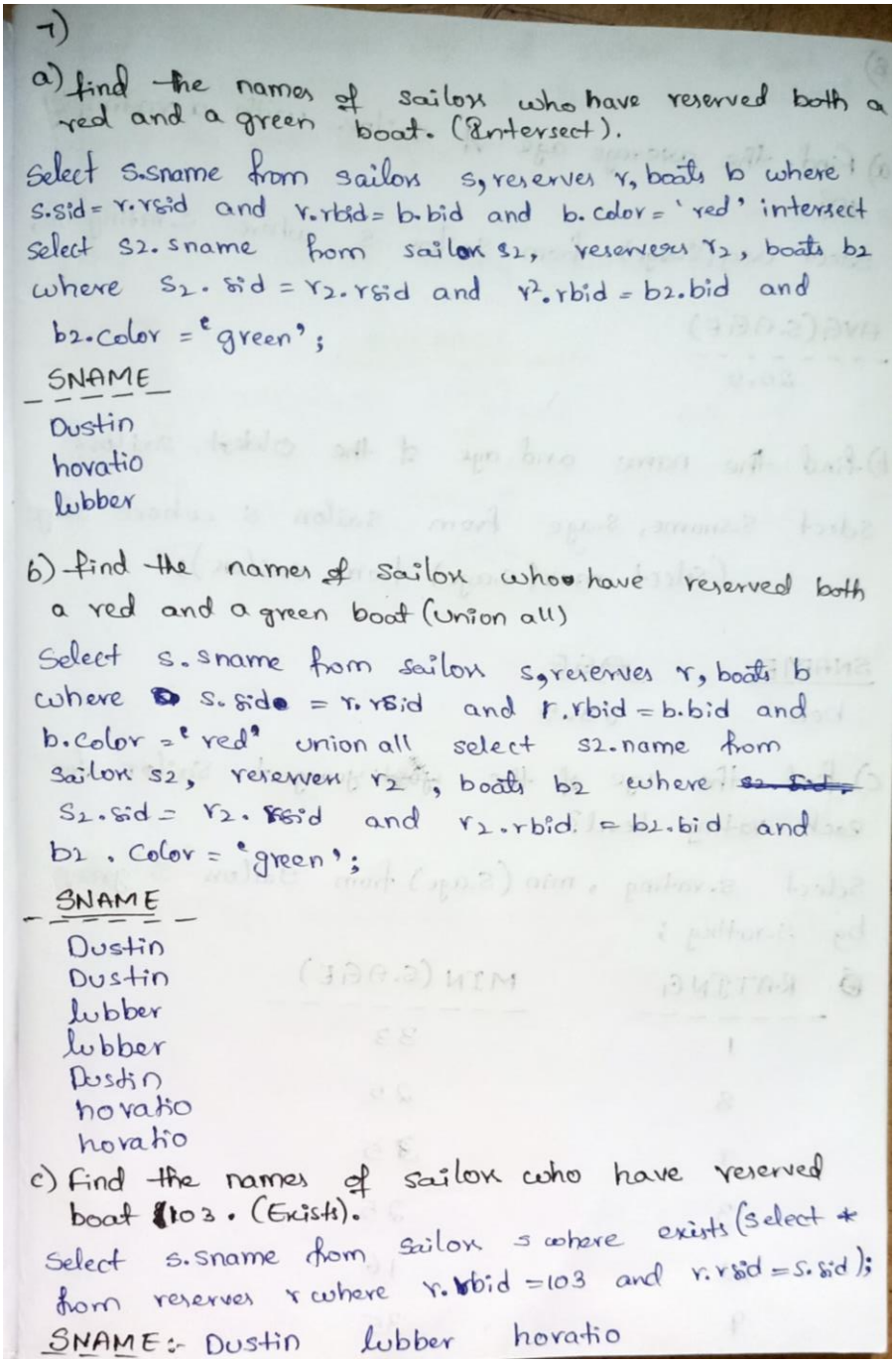
7. Create a table sailor, reserves, boats:

| *sid* | *sname* | *rating* | *age* |
|---|---|---|---|
| 22 | Dustin | 7 | 45.0 |
| 29 | Brutus | 1 | 33.0 |
| 31 | Lubber | 8 | 55.5 |
| 32 | Andy | 8 | 25.5 |
| 58 | Rusty | 10 | 35.0 |
| 64 | Horatio | 7 | 35.0 |
| 71 | Zorba | 10 | 16.0 |
| 74 | Horatio | 9 | 35.0 |
| 85 | Art | 3 | 25.5 |
| 95 | Bob | 3 | 63.5 |

| *sid* | *bid* | *day* |
|---|---|---|
| 22 | 101 | 10/10/98 |
| 22 | 102 | 10/10/98 |
| 22 | 103 | 10/8/98 |
| 22 | 104 | 10/7/98 |
| 31 | 102 | 11/10/98 |
| 31 | 103 | 11/6/98 |
| 31 | 104 | 11/12/98 |
| 64 | 101 | 9/5/98 |
| 64 | 102 | 9/8/98 |
| 74 | 103 | 9/8/98 |

| *bid* | *bname* | *color* |
|---|---|---|
| 101 | Interlake | blue |
| 102 | Interlake | red |
| 103 | Clipper | green |
| 104 | Marine | red |

a. Find the names of sailors who have reserved both a red and a green boat. (Intersect)
b. Find the names of sailors who have reserved both a red and a green boat. (union all)
c. Find the names of sailors who have reserved boat 103. (Exists)



7)

a) find the names of sailors who have reserved both a red and a green boat. (Intersect).

```
Select S.sname from sailors s, reserves r, boats b where
s.sid = r.rsid and r.rbid = b.bid and b.color = 'red' intersect
select s2.sname from sailor s2, reserves r2, boats b2
where s2.sid = r2.rsid and r2.rbid = b2.bid and
b2.color = 'green';
```

SNAME
Dustin
horatio
lubber

b) find the names of sailor who have reserved both a red and a green boat (union all)

```
Select s.sname from sailor s, reserves r, boats b
where s.sid = r.rsid and r.rbid = b.bid and
b.color = 'red' union all select s2.name from
sailor s2, reserves r2, boats b2 where
s2.sid = r2.rsid and r2.rbid = b2.bid and
b2.Color = 'green';
```

SNAME
Dustin
Dustin
lubber
lubber
Dustin
horatio
horatio

c) find the names of sailor who have reserved boat 103. (Exists).

```
Select s.sname from sailor s where exists (select *
from reserves r where r.rbid = 103 and r.rsid = s.sid);
```

SNAME :- Dustin lubber horatio

8. Create a table sailor, reserves, boats:

| *sid* | *sname* | *rating* | *age* |
|---|---|---|---|
| 22 | Dustin | 7 | 45.0 |
| 29 | Brutus | 1 | 33.0 |
| 31 | Lubber | 8 | 55.5 |
| 32 | Andy | 8 | 25.5 |
| 58 | Rusty | 10 | 35.0 |
| 64 | Horatio | 7 | 35.0 |
| 71 | Zorba | 10 | 16.0 |
| 74 | Horatio | 9 | 35.0 |
| 85 | Art | 3 | 25.5 |
| 95 | Bob | 3 | 63.5 |

| *sid* | *bid* | *day* |
|---|---|---|
| 22 | 101 | 10/10/98 |
| 22 | 102 | 10/10/98 |
| 22 | 103 | 10/8/98 |
| 22 | 104 | 10/7/98 |
| 31 | 102 | 11/10/98 |
| 31 | 103 | 11/6/98 |
| 31 | 104 | 11/12/98 |
| 64 | 101 | 9/5/98 |
| 64 | 102 | 9/8/98 |
| 74 | 103 | 9/8/98 |

| *bid* | *bname* | *color* |
|---|---|---|
| 101 | Interlake | blue |
| 102 | Interlake | red |
| 103 | Clipper | green |
| 104 | Marine | red |

a. Find the average age of sailors with a rating of 10?
b. Find the name and age of the oldest sailor?
c. Find the age of the youngest sailor for each rating level?
d. Find the average age of sailors for each rating level that has at least two sailors? (group by and Having)

8)

a) Find the average age of sailors with a rating of 10?

```
Select avg(S.age) from Sailors S where S.rating=10;
```

| AVG(S.AGE) |
|---|
| 25.5 |

b) find the name and age of the oldest sailor?

```
Select S.sname, S.age from Sailors S where S.age
= (Select max(S.age) from Sailors);
```

| SNAME | AGE |
|---|---|
| bob | 63.5 |

c) find the age of the ~~youn~~ youngest sailor for each rating level?

```
Select S.rating, min(S.age) from Sailors S group
by S.rating;
```

| RATING | MIN(S.AGE) |
|---|---|
| 1 | 33 |
| 8 | 25 |
| 7 | 35 |
| 3 | 25.5 |
| 10 | 16 |
| 9 | 35 |

d) find the average age of Sailors for each rating level that has at least two sailors? (group by and having).

```
Select S.rating, avg(S.age) as average from Sailors S
group by S.rating having Count(*)>1;
```

| RATING | AVERAGE |
|---|---|
| 8 | 40.5 |
| 7 | 40 |
| 3 | 44.5 |
| 10 | 25.5 |

9. Create a table customer and order table:

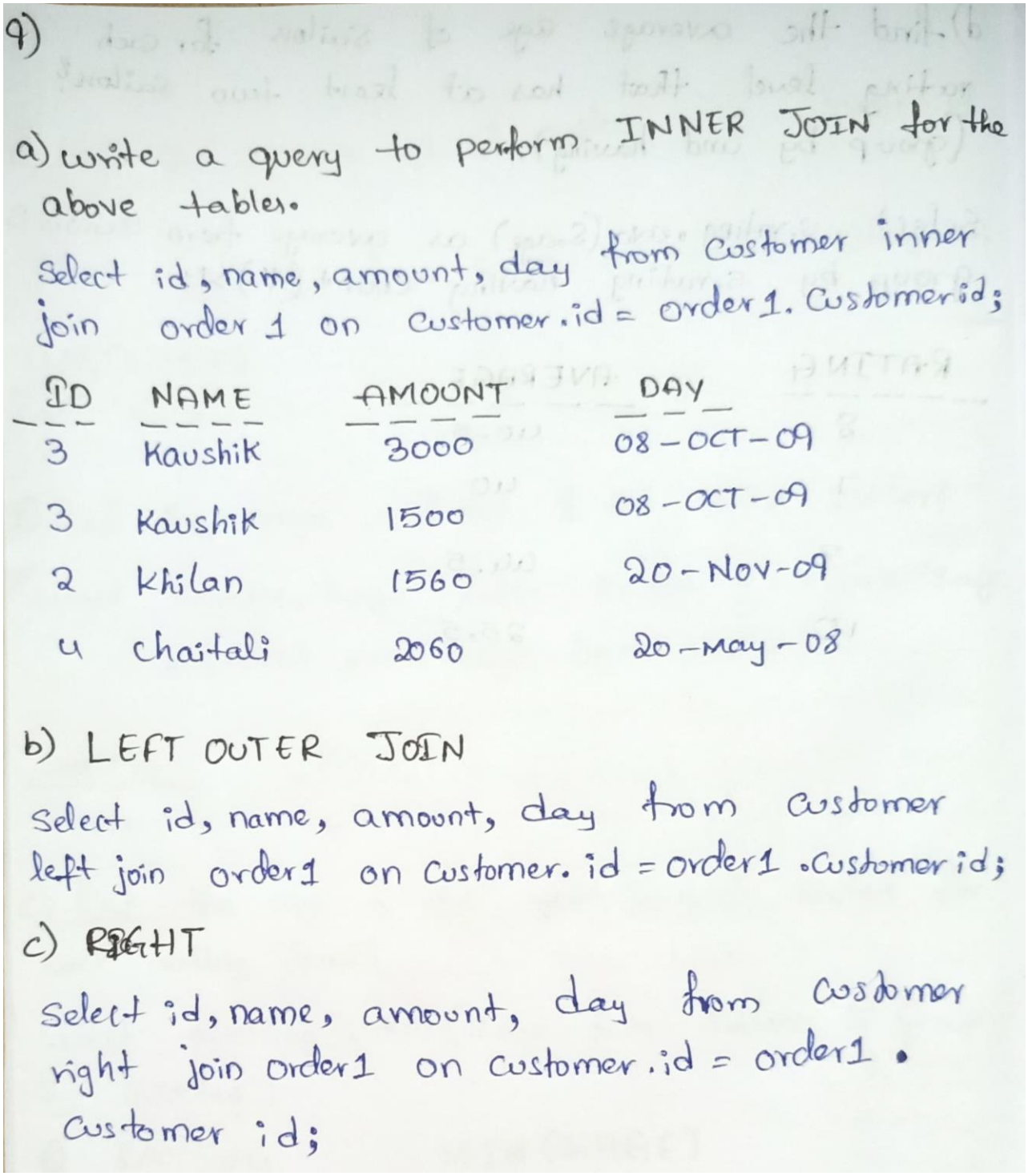
CUSTOMER TABLE

| ID | NAME | AGE | ADDRESS | SALARY |
|---|---|---|---|---|
| 1 | Ramesh | 32 | Ahmedabad | 2000.00 |
| 2 | Khilan | 25 | Delhi | 1500.00 |
| 3 | Kaushik | 23 | Kota | 2000.00 |
| 4 | Chaitali | 25 | Mumbai | 6500.00 |
| 5 | Hardik | 27 | Bhopal | 8500.00 |
| 6 | Komal | 22 | MP | 4500.00 |
| 7 | Muffy | 24 | Indore | 10000.00 |

ORDER TABLE

| OID | DAY | CUSTOMER_ID | AMOUNT |
|---|---|---|---|
| 102 | 2009-10-08 | 3 | 3000 |
| 100 | 2009-10-08 | 3 | 1500 |
| 101 | 2009-11-20 | 2 | 1560 |
| 103 | 2008-05-20 | 4 | 2060 |

a. Write a query to perform INNER JOIN for the above tables.
b. Write a query to perform LEFT OUTER JOIN for the above tables.
c. Write a query to perform RIGHT OUTER JOIN for the above tables.



9)

a) write a query to perform INNER JOIN for the above tables.

```
Select id, name, amount, day from customer inner
join order1 on customer.id = order1.customerid;
```

| ID | NAME | AMOUNT | DAY |
|---|---|---|---|
| 3 | Kaushik | 3000 | 08-OCT-09 |
| 3 | Kaushik | 1500 | 08-OCT-09 |
| 2 | Khilan | 1560 | 20-NOV-09 |
| 4 | Chaitali | 2060 | 20-May-08 |

b) LEFT OUTER JOIN

```
Select id, name, amount, day from customer
left join order1 on customer.id = order1.customerid;
```

c) RIGHT

```
Select id, name, amount, day from customer
right join order1 on customer.id = order1.
customer id;
```

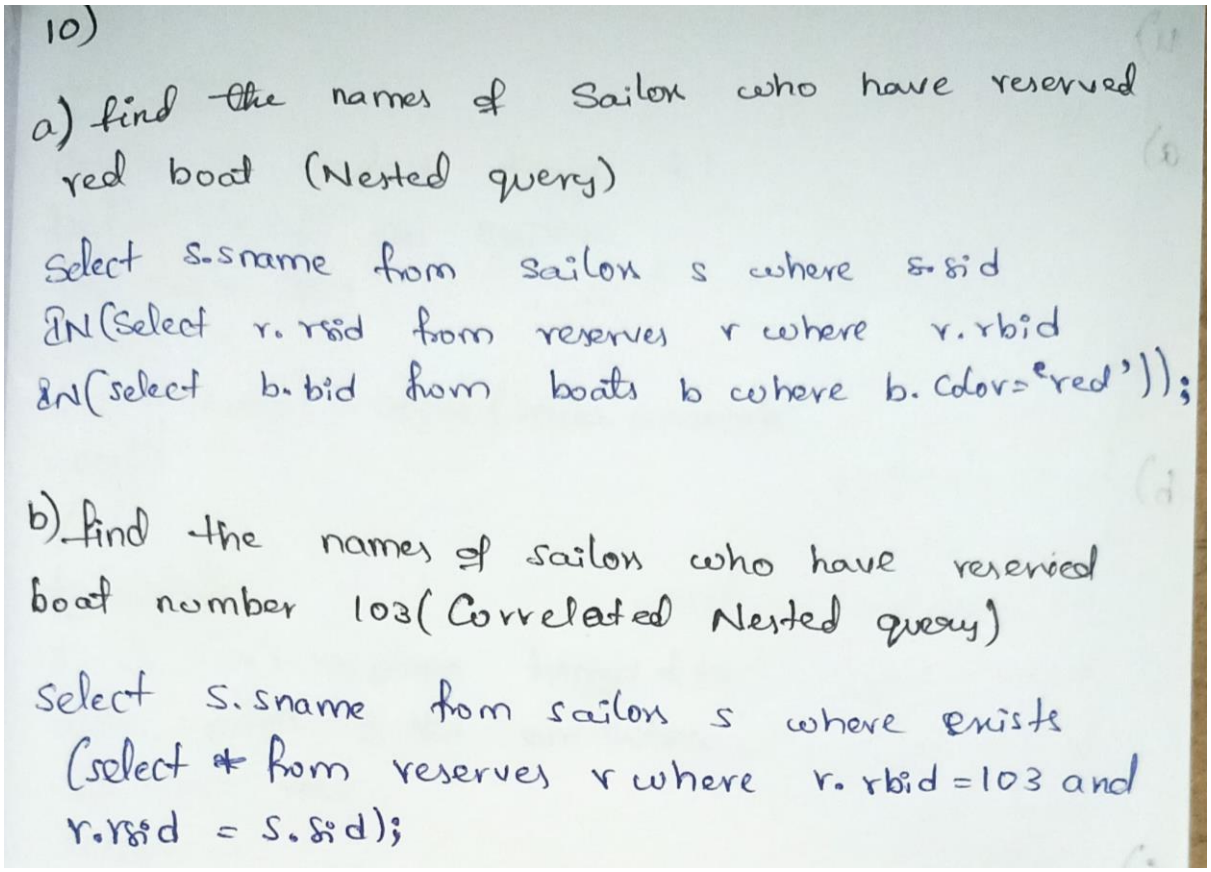
10. Create a table sailor, reserves, boats:

| *sid* | *sname* | *rating* | *age* |
|---|---|---|---|
| 22 | Dustin | 7 | 45.0 |
| 29 | Brutus | 1 | 33.0 |
| 31 | Lubber | 8 | 55.5 |
| 32 | Andy | 8 | 25.5 |
| 58 | Rusty | 10 | 35.0 |
| 64 | Horatio | 7 | 35.0 |
| 71 | Zorba | 10 | 16.0 |
| 74 | Horatio | 9 | 35.0 |
| 85 | Art | 3 | 25.5 |
| 95 | Bob | 3 | 63.5 |

| *sid* | *bid* | *day* |
|---|---|---|
| 22 | 101 | 10/10/98 |
| 22 | 102 | 10/10/98 |
| 22 | 103 | 10/8/98 |
| 22 | 104 | 10/7/98 |
| 31 | 102 | 11/10/98 |
| 31 | 103 | 11/6/98 |
| 31 | 104 | 11/12/98 |
| 64 | 101 | 9/5/98 |
| 64 | 102 | 9/8/98 |
| 74 | 103 | 9/8/98 |

| *bid* | *bname* | *color* |
|---|---|---|
| 101 | Interlake | blue |
| 102 | Interlake | red |
| 103 | Clipper | green |
| 104 | Marine | red |

a. Find the names of sailors who have reserved red boat. (Nested Query)
b. Find the names of sailors who have reserved boat number 103. (correlated Nested Query)



10)

a) find the names of Sailor who have reserved red boat (Nested query)

```
Select S.sname from sailor s where s.sid
IN(Select r.rsid from reserves r where r.rbid
IN(select b.bid from boats b where b.color='red'));
```

b) find the names of sailor who have reserved boat number 103(Correlated Nested query)

```
Select S.sname from sailor s where exists
(select * from reserves r where r.rbid=103 and
r.rsid = s.sid);
```

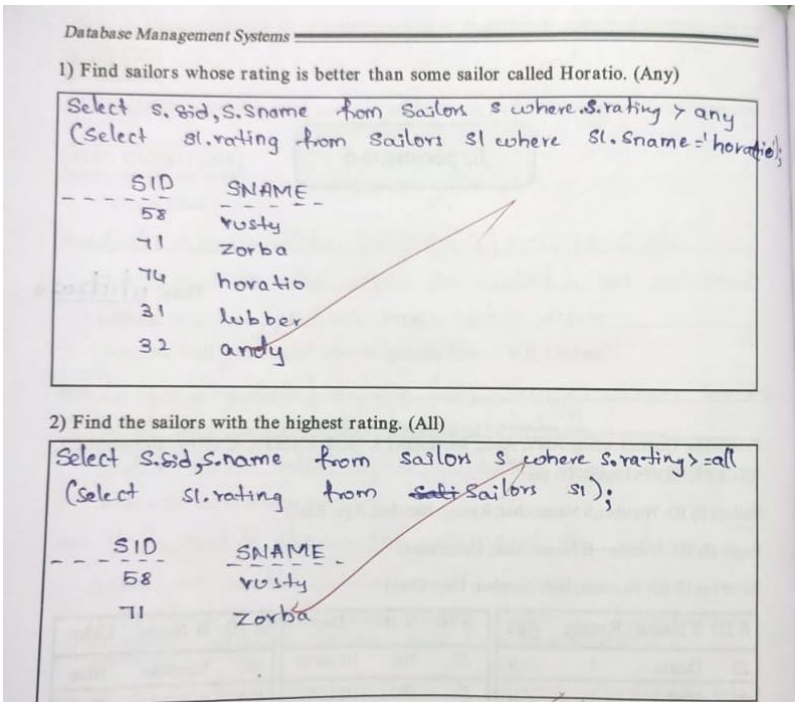
11. Create a table sailor, reserves, boats:

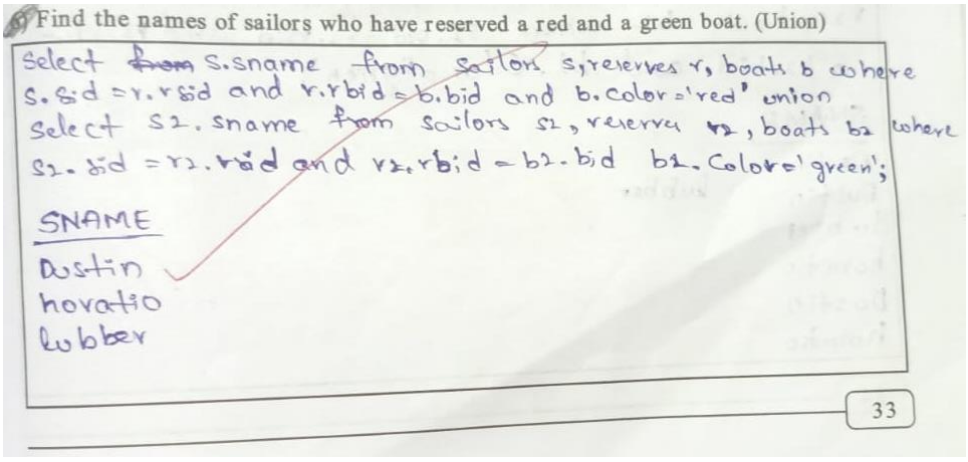
| *sid* | *sname* | *rating* | *age* |
|---|---|---|---|
| 22 | Dustin | 7 | 45.0 |
| 29 | Brutus | 1 | 33.0 |
| 31 | Lubber | 8 | 55.5 |
| 32 | Andy | 8 | 25.5 |
| 58 | Rusty | 10 | 35.0 |
| 64 | Horatio | 7 | 35.0 |
| 71 | Zorba | 10 | 16.0 |
| 74 | Horatio | 9 | 35.0 |
| 85 | Art | 3 | 25.5 |
| 95 | Bob | 3 | 63.5 |

| *sid* | *bid* | *day* |
|---|---|---|
| 22 | 101 | 10/10/98 |
| 22 | 102 | 10/10/98 |
| 22 | 103 | 10/8/98 |
| 22 | 104 | 10/7/98 |
| 31 | 102 | 11/10/98 |
| 31 | 103 | 11/6/98 |
| 31 | 104 | 11/12/98 |
| 64 | 101 | 9/5/98 |
| 64 | 102 | 9/8/98 |
| 74 | 103 | 9/8/98 |

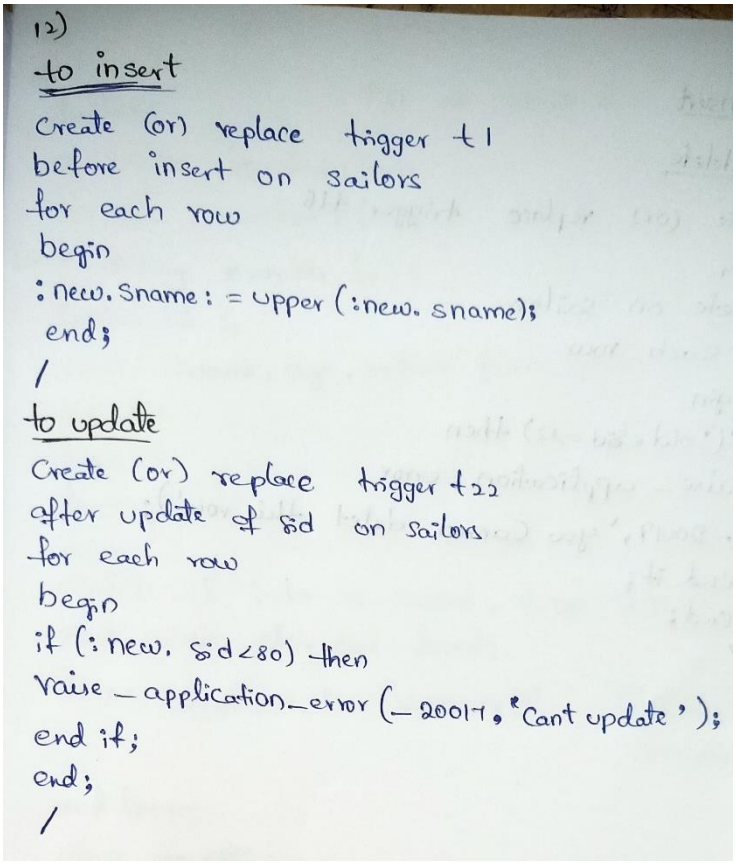
| *bid* | *bname* | *color* |
|---|---|---|
| 101 | Interlake | blue |
| 102 | Interlake | red |
| 103 | Clipper | green |
| 104 | Marine | red |

a. Find sailors whose rating is better than some sailor called Horatio. (Any)
b. Find the sailors with the highest rating. (All)
c. Find the names of sailors who have reserved a red and a green boat. (Union)



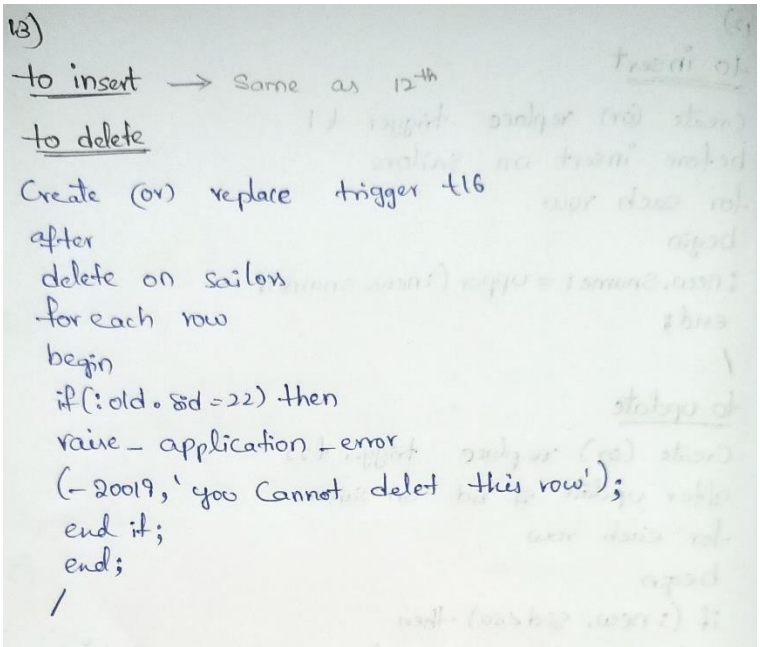



12. Write a PL/SQL code for creation of Trigger to insert and to update data into a table



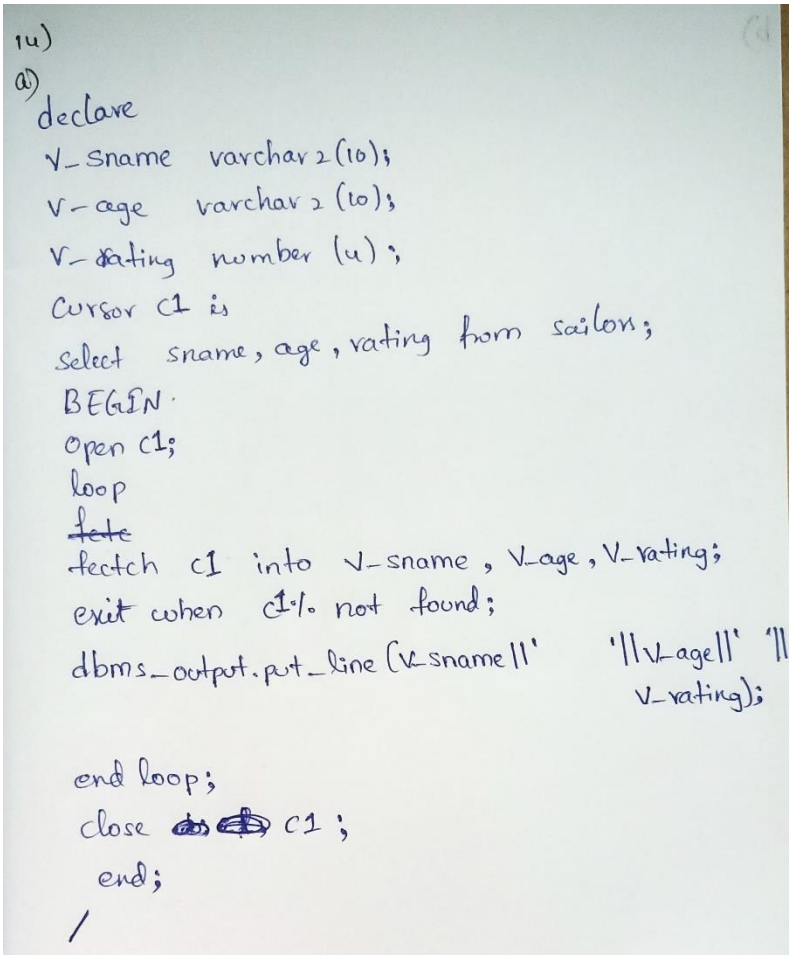
```
12)
to insert
Create (or) replace trigger t1
before insert on sailors
for each row
begin
:new.sname := upper (:new. sname);
end;
/
to update
Create (or) replace trigger t22
after update of sid on sailors
for each row
begin
if (:new. sid<80) then
raise_application_error (-20017, 'Cant update');
end if;
end;
/
```

13. Write a PL/SQL code for creation of Trigger to insert and to delete data into a table



```
13)
to insert → Same as 12th
to delete
Create (or) replace trigger t16
after
delete on sailors
for each row
begin
if (:old.sid=22) then
raise_application_error
(-20019,'you Cannot delet this row');
end if;
end;
/
```

14. a) Write a PL/SQL program that uses cursor operation on any data base.



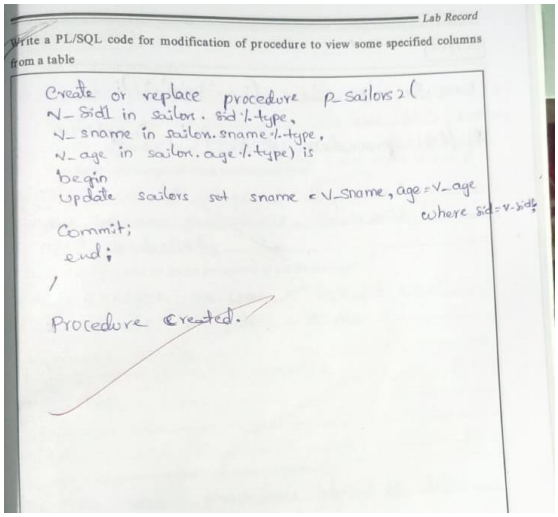
```
14)
a)
declare
v_sname  varchar2(10);
v_age  varchar2(10);
v_rating  number(4);
Cursor c1 is
Select  sname, age, rating from sailors;
BEGIN
Open c1;
loop
fetch c1 into v_sname, v_age, v_rating;
exit when c1%. not found;
dbms_output.put_line(v_sname||'  '||v_age||'  '||
                     v_rating);

end loop;
close c1;
 end;
/
```

b) Write a PL/SQL program for displaying multiplication of any number

```
DECLARE
  num NUMBER := 7;  -- Replace with the desired number
  range NUMBER := 12;  -- Replace with the desired range
BEGIN
  FOR i IN 1..range LOOP
    dbms_output.put_line(num || ' x ' || i || ' = ' || num * i);
  END LOOP;
END;
/
```

15. a) Write a PL/SQL code for modification of procedure to view some specified columns from a table.



Lab Record

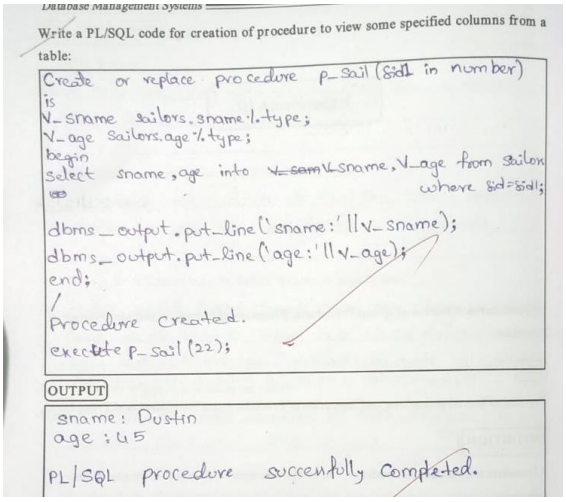
Write a PL/SQL code for modification of procedure to view some specified columns from a table

```
Create or replace procedure p_sailors2(
v_sid1 in sailor.sid%type,
v_sname in sailor.sname%type,
v_age in sailor.age%type) is
begin
update sailors set sname = v_sname, age = v_age
                                    where sid = v_sid;
Commit;
end;
/
Procedure Created.
```

b) Write a PL/SQL program for displaying multiplication of any number

```
DECLARE
num NUMBER := 7;  -- Replace with the desired number
range NUMBER := 12;  -- Replace with the desired range
BEGIN
FOR i IN 1..range LOOP
dbms_output.put_line(num || ' x ' || i || ' = ' || num * i);
END LOOP;
END;
/
```

16. a) Write a PL/SQL code for creation of procedure to view some specified columns from a table.



Database Management Systems

Write a PL/SQL code for creation of procedure to view some specified columns from a table:

```
Create or replace procedure p_sail (sid1 in number)
is
v_sname sailors.sname%type;
v_age sailors.age%type;
begin
select sname, age into v_sname, v_age from sailor
                                    where sid=sid1;

dbms_output.put_line('sname:'||v_sname);
dbms_output.put_line('age:'||v_age);
end;
/
Procedure created.
execute p_sail(22);
```

OUTPUT

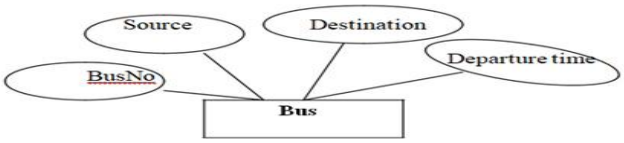
```
sname: Dustin
age : 45

PL/SQL procedure succesfully completed.
```

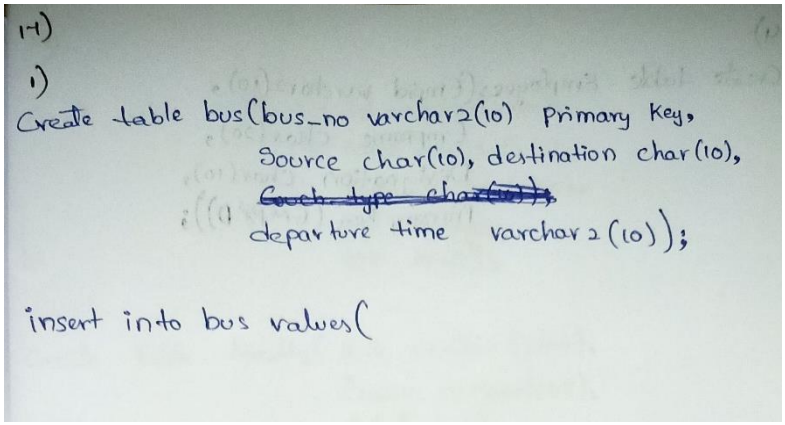
b) Write a PL/SQL program for displaying factorial of any number.

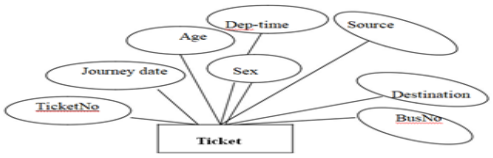
```
DECLARE
  num NUMBER := 6;  -- Replace with the desired number
  fact NUMBER := 1;
BEGIN
  FOR i IN 1..num LOOP
    fact := fact * i;
  END LOOP;
  dbms_output.put_line('Factorial of ' || num || ' is ' || fact);
END;
/
```

17. Converting ER Model to Relational Model (Represent entities and relationships in Tabular form, represent attributes as columns, identifying keys)

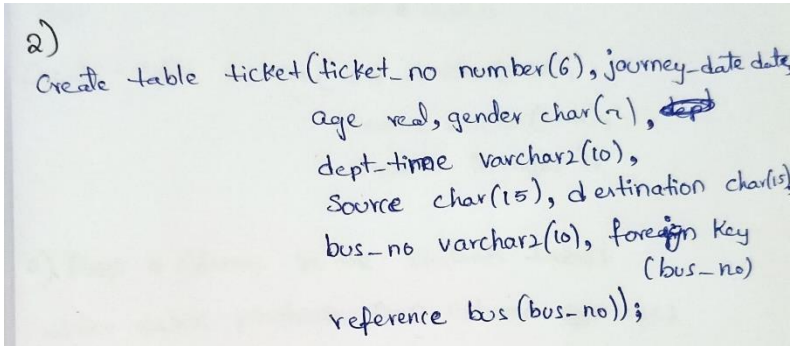


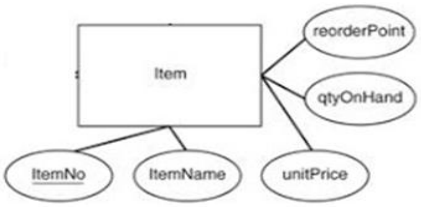

1.



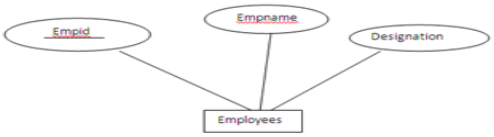

2.



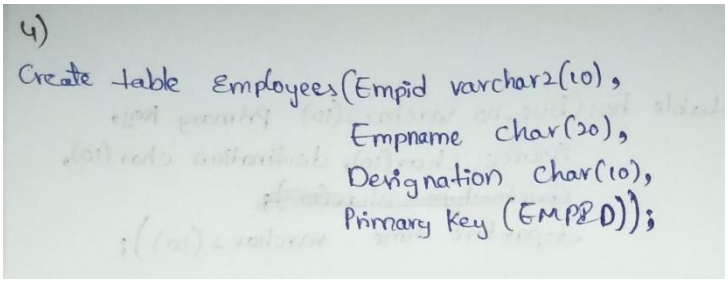



3.

```
CREATE TABLE Item (
  itemno INT PRIMARY KEY,
  itemname VARCHAR(255) NOT NULL,
  unitprice DECIMAL(10,2) NOT NULL,
  reorderpoint INT,
  qtyonhand INT
);
```




4.



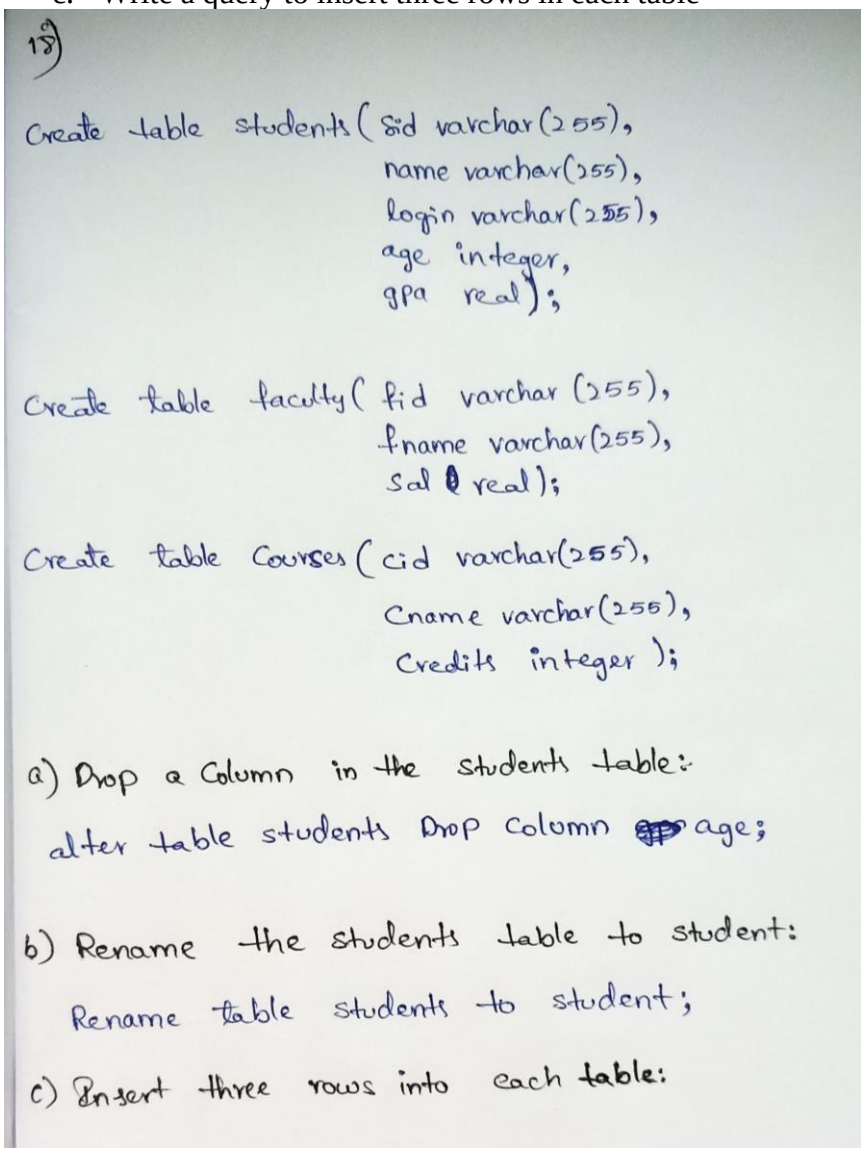
18. Create tables for following schemas

Students (*sid: string*, *name:* string, *login:* string, *age:* integer, *gpa:* real)

Faculty (*fid:* string, *fname:* string, *sal:* real)

Courses (*cid:* string, *cname:* string, *credits:* integer)

a. write a sql query to drop a column in students table.
b. Write a query to rename table students to STUDENT
c. Write a query to insert three rows in each table



18)

```
Create table students ( sid varchar(255),
                        name varchar(255),
                        login varchar(255),
                        age integer,
                        gpa real);

Create table faculty ( fid varchar (255),
                       fname varchar(255),
                       sal real);

Create table Courses ( cid varchar(255),
                       Cname varchar(255),
                       Credits integer );
```

a) Drop a Column in the students table:

```
alter table students Drop column age;
```

b) Rename the students table to student:

```
Rename table students to student;
```

c) Insert three rows into each table: