Practical 1

Questions for Practice: 23/07/2020

1. Create database named as college.
   Create database college

2. Inside database create table as Faculty with columns id, name, qualification, subject, age, salary. Make id as unique and not null.
   Create table Faculty (id int primary key, name char(20), qualification char(20),subject char(20),age float, salary int)

3. Insert 10 values in to the table.
   Insert into faculty values (11,"aayesha","MSC maths","Maths",25,35000)
   .
   . (9 times with different values)

4. Display the name of staff whose qualification is MCOM
   select * from Faculty where qualification ="MCOM"

5. Display the name and age of staff.
   select name as"Staff Name",age as "Staff age" from faculty

6. Display the name of staff who is expert in "JAVA"
   select name from Faculty where subject like "java"

7. Which employees earns more than 40000.
   select * from Faculty where salary > 40000

8. Which faculties have age more than 30 and less than 40 years.
   select * from Faculty where age>30 and age<40

9. Display staff details who earns highest.
   select * from faculty where salary = (select max(salary) from faculty)

10. Increase the salary of all employees by 5%.
    Update faculty set salary = salary + salary*0.05

Practical 2 **24/07/20**

By considering practical 1 table faculty solve following queries.

1. Add column percentage to faculty.
   Alter table faculty add percentage float

2. Set the percentage to 60 for all.
   Update table faculty set percentage =60

3. Delete details of faculty whose id is 15.
   delete from faculty where id =15

4. Remove column age from faculty.
   Alter table faculty drop column age

5. Delete all the data from table.

   delete from faculty

6. Delete the table.
   drop table faculty

NULL – means value is missing or not specified. DBMS treats Zero, space and null differently.

Practical 3 **13/08/20**

1. **Create a table student with attributes rollno, name, percent, grade, phone , age, weight.**
   create table student(rollno int, name char(20), percent float, grade char(3),phone int,age int,weight float)

2. **Insert 5 meaningful records in it.**

   insert into student values(101,'brijesh',67,'A',90909,45,87)

   …..

3. **Display student name whose percentage is between 40 to 60.**
   select name , percent
   from student
   where percent >=40 and percent<=60

   or

   select name , percent
   from student
   where percent between 40 and 60

4. **How many students have not given age.**

select count(*)

from student

where age is null

5. **Add a column blood group in student table.**

   alter table student add column blood_gr char(2)

6. **Set the blood group as B for students having roll no 101,104,106.**

update student

set blood_gr="B"

where rollno in (101,104,106)

7. **How many students have weight above 80 kg.**

```
select count(*)
from student
where weight >= 80
```

8. **Display the name of oldest student. (Subquery)**

```
select name
from student
where age = (select max(age) from student)
```

9. **What is the average percentage scored by students.**

```
select avg(percent) as "Average percentage"
from student
```

10. **How many students are there.**

```
select count(rollno) as "Total Students"
 from student
```

11. **Which student scored lowest percentage.**

```
select name, percent from student
where percent= (select min(percent) from student)
```

12. **Display student information as per their percentage in ascending order.**

```
select * from student order by percent
```

13. **How many students have percentage above 85?**

```
select count(*)
from student
where percent>=85
```

14. Which student is having lowest weight.

```
select name, weight from student
where weight= (select min(weight) from student)
```

15. What is the age and weight of whose name begins with ‘d’.

```
select name, age, weight
from student
where name like 'd%'
```

16. Display names of student who have grade o, a and b.

```
select name, grade
from student
where grade in ('A','O','B')
```

Learnings : order by, aggregate function, in, between, like, wild card character

Practical 4 **27/08/20**

(Multitable queries, equi join, non equi join)

Create following two tables with column and appropriate datatypes.

1. **Customer- cid primary key, cname, address, age**

   Create table customer(cid int primary key, cname char(20), address varchar(50), age int)

2. **Orders- oid,cid foreign key, prod_name, price**
   Create table orders(oid int primary key, prod_name char(20),price int, cid int references customer (cid))

3. **Insert meaningful records in table.**

Insert into customer values (101,"mohan" , "malad",56)

….

Insert into orders values(1001, "maggi", 200, 102)

…..

cid	cname	address	age
101	mohan	malad	56
102	neeraj	goregaon	34
103	suraj	malad	24
104	prakash	mahim	30

oid	prod_name	price	cid
1001	Maggi	200	102
1002	pasta	300	104
1003	poha	100	102
1004	oats	500	103

4. **Which customers have given orders?**
   select distinct cname
   from customer , orders
   where customer.cid = orders.cid

inner join –

```
select distinct cname
from customer inner join orders
on customer.cid = orders.cid
```

Note : distinct – removes duplicate entry.

5. **MAGGI is purchased by which customer?**

```
select cname
from customer c, orders o
where prod_name="Maggi" and c.cid = o.cid
```

inner join-

```
select cname
from customer c inner join orders o
on prod_name="Maggi" and c.cid = o.cid
```

Note:
C and o are table alias.
Table alias are the short name of table.

6. **What is the age of customer who have purchase product PASTA?**

```
select age
from customer c , orders o
where prod_name="pasta" and c.cid = o.cid
```

inner join -

```
select age
from customer c inner join orders o
on prod_name="pasta" and c.cid = o.cid
```

7. Customers of MALAD have purchased which product?

```
select prod_name
from orders o,customer c
where address = "malad" and c.cid=o.cid
```

8. Display product name as per their price.

```
select prod_name, price
from orders
order by price
```

9. Which customers have not purchased any product?

```
select cid,cname
from customer
where cid not in (select cid from orders)
```

10. Which product is purchased by customer 101?

```
Select prod_name
From orders o, customer c
Where o.cid = c.cid and o.cid=101
```

11. Which customer has purchase product of price above 300?

```
Select *
From orders o, customer c
Where price >300 and c.cid=o.cid
```

12. Which customer has purchased cheapest product?

```
select cname, prod_name, price
from customer c, orders o
where c.cid=o.cid and prod_name like (
select prod_name from orders where price =
(select min(price) from orders))
```

13.Display the customer name having same age.

```
select c1.cname,c1.age
from customer c1,customer c2
where c1.age=c2.age and c1.cid<>c2.cid
```

Practical No. 5 **4/09/2020**

(Group by and Having clause)

Consider the tables in practical no. 4 and answer following queries:

cid	cname	address	age
101	mohan	malad	56
102	neeraj	goregaon	34
103	suraj	malad	24
104	prakash	mahim	30
105	priya	mahim	56
106	brijesh	malad	34

1. **Display the locations.**

   select distinct address from customer
   or
   select address from customer group by address

2. **display the number of people location wise.**

   select address, count(cid) from customer group by address

3. **display the location where more than 2 people stays.**

   select address, count(cid) from customer group by address having count(cid)>2

4. **How many people have age more than 30.**

   select count(age) from customer where age>30

5. **Display number of people from each age group.**

   select count(cid), age from customer group by age

Practical No. 6 **11/09/2020**

(views and joins)

1. **Create table books with atleast 10 columns and 10 rows.**
   (bid primary key, bname, pid foreign key , auid foreign key, pages, price, types, language, year, stars)

2. **Create table author**
   (auid primary key, auname, age, phone, qualification, experience)

3. **Create table publisher** (pid primary key, pname, year, ISOno, stars)

4. **Create view called as study_book which will contain only study types of books.**

5. **Create view called as threestar which will contain 3 star book with details such as bid, bname, pages, price, auid, auname, experience.**

6. **Create view allbook which will contain bid, bname, auname and pname.**

7. **Create view author_pub which will contain aid,auname,pid,pname of the books which are published in years 2019,2016, 2012.**

8. **Display the publisher name of the book published in year 2012.**

9. **Display the bname, auname and pname of all the books.**

10. **Display the author details of the books which have got more than 4 stars.**

11. **How many books are there of each type.**

12. **Which publisher have received maximum stars.**

13.Display the name of book which is costliest.

14. Which book is written by the oldest author.

15. Display the number books written in different languages.