to display unique values of a column

| | |
|---|---|
| select distinct job from emp; | It will display unique values from job column |
| select distinct job,mgr from emp; | It will display unique combination of job and mgr |

To arrange data in sorted order use order by clause

1. to arrange data in ascending order of salary

select * from emp order by sal

2. to arrange data in descending order of salary
   select * from emp order by sal desc

3. to arrange data in descending order of salary, if salary is same then arrange on ename
   select * from emp order by sal desc, ename

4. to arrange data in descending order of salary, if salary is same then arrange on ename in descending order
   select * from emp order by sal desc, ename desc

5. to find 2 highly paid employees
   limit will retrieve first 2 rows
   select * from emp order by sal desc
   limit 2;
6. to find 3 rd, 4th and 5 th rowhighly paid employees
   limit 2,3 will skip 1 st 2 rows and retrieve next 3 rows
   select * from emp order by sal desc
   limit 2,3

7. generate derived columns
   derived columns are the columns which are not there in the table, but can be display by using some formula or expression
   select empno,ename,sal,sal+comm
   from emp;

   groupby and having clause
   1. When you want to divide existing rows in multiple groups based on some column then use group by
   2. grouping can be done on more than one column
   3. while filtering data, if the condition is on the existing column, then use the condition in where clause, otherwise use it on having clause.
   4. in group by, in select statement, you can use only columns on which grouping is done. other columns cannot be used in select statement

   examples
   1. display sum of sal and min sal for all employees

```
select sum(sal), max(sal) from emp
```

2. list sum of sal and max sal departmentwise

```
select deptno,sum(sal),max(sal)
from emp
group by deptno;
```

3. list sum of sal and min sal count, jobwise

```
 select job,count(*), min(sal),sum(sal)
from emp
group by job
```

4. find sum of sal, avg sal for each mgrwise

```
select  mgr,sum(sal),avg(sal)
from emp
group by mgr;
```

5. find sum of sal, sum of netsal, net sal= sal+comm for each department

```
select deptno,sum(sal),sum(sal+ifnull(comm,0))
from emp
group by deptno
```

6. find sum sal, avg sal and count departmentwise, jobwise;

```
select deptno,job,sum(sal),sum(sal+ifnull(comm,0))
from emp
group by deptno,job
```

7. find how many clerks are in each department

```
select deptno,count(*),sum(sal)
from emp
where job='CLERK'
group by deptno
having count(*)>2
```

8. display department with highest number of employees

```
select deptno,count(*)
from emp
group by deptno
order by count(*) desc
limit 1;
```

9. display sum(sal),avg(sal), deptwise, mgrr wise

```
select deptno,mgr,sum(sal),avg(sal),count(*)

from emp

group by deptno,mgr

order by count(*) desc
```

If you want to display value of aggregate function, without using group by, then use partition by

partition by clause allows you to display columns which are not involved in group by.

1. to display empno,ename,deptno,sum(sal) for each department.

   Select ename,deptno,sum(sal) over (partition by deptno order by deptno)

   From emp

Find all ename which has _ in it

```
select ename
  -> from emp
  -> where ename like '%\_%';
select ename
  -> from emp
  -> where ename REGEXP '_';
```

functions used for numeric columns.

| | | |
|---|---|---|
| abs(num) | to convert -ve value into +ve | select abs(-3)<br>3 |
| pow(num,raiseto) | power of the number | select pow(3,2)<br>9 |
| floor(num) | It will remove all the digits after the decimal point, and gives the maximum lowest number | select floor(312.1567)<br>312<br>select floor(312.61567)<br>312 |
| ceil(num) | It will always give the next minimum number | select ceil(312.1567)<br>313<br>select ceil(312.61567)<br>313 |
| round(num,precision) | round will round the number upto given precision | select round(312.1567,2)<br>312.16<br>select round(312.61267,2)<br>312.61 |
| truncate(num,precesion) | truncate will truncate the number upto given precision | select truncate(312.1567,2)<br>312.15<br>select truncate(312.61267,2)<br>312.61 |
| sqrt(num) | it will display square root of given number | select sqrt(4)<br>2 |

Functions used with strings

| | | |
|---|---|---|
| upper(ename) | convert all characters in uppercase | select upper('asdf')<br>ASDF |
| lower(ename) | convert all the characters in small case | select lower('ASDF')<br>asdf |
| concat(s1,s2,s3) | concatenate the strings | concate('xxx','.','yyy')<br>xxx.yyy |
| concat_ws(":",ename,job,sal) | concatenate all the values with separator | concat_ws(":",'xxx','yyyy',1234)<br>xxx:yyy:1234 |
| format(sal,precision) | It will display number in formatted manner, it will display 1000 seperator in the number, and the number of digits after decimal point | format(312,2)<br>312.00<br>format(314356452,2)<br>31,43,56,452.00 |
| substr(string,startpos,number of character) | it will display number of characters from the starting position, counting starts with 1 | substr('testing',3,4)<br>stin |
| left(s,num of characters) | it will display number of characters from left side | left('welcome',3)<br>wel |
| right(s,num of characters) | it will display number of characters from right side | right('welcome',3)<br>ome |
| length(str) | it will display number of characters in the string | length('xxxx')<br>4 |
| lpad(str,length,character) | it will add characters on the left side of the string so that the total characters will be = length | lpad('welcome',12,'-')<br>-----welcome |
| rpad(str,length,character) | it will add characters on the right side of the string so that the total characters will be = length | rpad('welcome',12,'-')<br>welcome----- |
| rtrim(str) | it will remove trailing spaces, i.e the space on the right side | select rtrim('  hello  ')<br>  hello |
| trim(str) | it will remove leading and trailing spaces | select trim('  hello  ')<br>hello |
| ltrim(str) | it will remove leading spaces, i.e the spaces on the left side | select ltrim('  hello  ')<br>hello |
| instr(str,s1) | it will find the position of the first occurrence in the given string | instr('welcome','el')<br>2 |
| reverse(str) | reverse the string | reverse("hello")<br>olleh |
| replace(str,searchstr,newstr) | it replace all occurrence of the searchstr with newstr | replace('testing string','ing','aaaa')<br>testaaa straaa |

| | | |
|---|---|---|
| insert(str,pos,length,newstr) | it overnights from the pos character character onward length characters by newstr | select insert("welcome",2,3,"test") wtestome<br>select insert("welcome",2,0,"test") wtestelcome |
| repeate(string,count) | it will print the string count times | repeate(“aaa”,5) aaaaaaaaaaaaaaa |

1. find email of the employee by concatenating first 3 letters of ename, followed by . and last 3 characters of job followed by “@mycompany.com”

   select empno,ename,concat(left(ename,3),’.’,right(job,3)) email
   from emp;  -----better approach

   select empno,ename,concat(substr(ename,1,3),’.’,substr(job,length(job)-3,3),”@mycompany.com”)

2. display 3,4,5,6 character of job as a jobcode
   select empno,ename,job,substr(job,3,4) jobcode
   from emp;

3. display ename, every ename length should be 12, add extra required number of * on the right side
   select empno,ename,rpad(ename,12,’*’)
   from emp;

date related functions

| | |
|---|---|
| now() | it display today’s date and time |
| curdate() | it display today’s date |
| datediff(date1,date2) | it displays difference between 2 dates in terms of days |
| date_format(date1,format) | It will display the date in user required format<br>Y --- will display 4 digit year<br>y----will display 2 digit year<br>M- month name in character<br>m-month in number<br>d- date in number<br>D- display th or st after date<br>b--- display months in 3 letter (jan, feb,....)<br>r ---- to print time in 12 hrs (hh:mm:ss AM/PM)<br>h--- to display hour<br>i----to display minutes |

| | |
|---|---|
| | p—to display AM/PM |
| date_add(date, interval n unit) | it will find the date after given interval<br>to find the date after 2 months<br>date_add(curdate(),interval 2 month) |
| date_sub(date, interval n unit) | it will find the date before given interval |
| day(date) | to find day from the given date |
| month(date) | to find month from the given date |
| year(date) | to find year from the given date |
| quarter(date) | to find the quarter |
| week(date) | to find the week of the date |
| extract(day from date) | to retrieve portion of the date, this is available in oracle also<br>select extract(day from curdate());<br>select extract(month from curdate());<br>---------- to retrieve month<br>select extract(year from curdate()); |
| monthname(curdate) | will display month name in characters |
| dayname(curdate) | it prints days, like Monday, Tuesday,.... |
| last_day(curdate()) | to find last day of the give month |
| Str_to_date('22/11/1982','%d/%m/%y')<br>#####1982-11-22 | convert given date in mysql format |

1. to find portion of the date
   select
   year(curdate()),month(curdate()),day(curdate()),quarter(curdate()),week(curdate()));

if we have medicine table
(medid,mname,mfgdate,expdate)

1. to find all medicines which are manufactured 3 months before.

   select medid,mname,mfgdate
   from medicine
   where datediff(curdate(),mfgdate)>=90

   or

   select medid,mname,mfgdate

   from medicine
   where date_sub(curdate(),interval 3 month)>=mfgdate -------better aproach

2. display expiry date of medicines if it is after 6 months 7 days from mfg date
   select medid,mname,mfgdate,date_add(date_add(mfgdate,interval 6 month),interval 7 day)

from medicine

3. find all employees joined in dec 1981

```
select empno,ename,hiredate,extract(month from hiredate)
  -> from emp
  -> where extract(month from hiredate)=12 and extract(year from
hiredate)=1981;
```

4. find date which is after 2 years, 7 months 10 days

```
select date_add(date_add(date_add(curdate(),interval 2 year),interval 7
month),interval 10 day)
```

5. find all medicines which will expire after 3 months.

```
select medid,mname,mfgdate,expdate
from medicine
where datediff(expdate,curdate())>=90
```

6. to find all employees with experience >41 years

```
select *, floor(datediff(curdate,hierdate)/365) experience
from emp
where floor(datediff(curdate,hierdate)/365)>=41
```

7. Write a query to get the distinct Thursday from hiredate in emp table.

```
select distinct hiredate,dayname(hiredate)
  -> from emp
  -> where dayname(hiredate)='Thursday';
```

8. find a particular string occurs how many times in the given string
   “saaavaaadaaa” find aaa appears how many times in the given string
   saaavaaadaaa 12
   svd----------3

```
select floor((length("saaataaayaaa")-
length(replace("saaataaayaaa","aaa","")))/length("aaa"));
```

case statement
When you want to display some data, based on condition, then we use case statement
in case statement all the values that you are displaying should be of same type

```
case when condition then o/p
when condition then o/p
else o/p end alias_name

case columnname when val1 then o/p
when val1 then o/p
when val1 then o/p
else o/p end alias_name
```

example
1. if deptno=10 then display accounts, if it 20 then display sales otherwise display purchase

```
select empno,ename,deptno, case deptno when 10 then "accounts"
  -> when 20 then "sales"
  -> else "purchase" end dname
  -> from emp;
```

2. if comm is null or 0 then display “poor performance”
if comm>=300 and <500 then display “ok performance”
if comm>=500 and <1000 then display good performance
otherwise display “excellent performance”

```
select empno,ename,sal,comm,case when comm is null or comm=0 then "poor performance"
  -> when comm>=300 and comm<450 then "ok performance"
  -> when comm>=450 and comm<1000 then "good performance"
  -> else "excellent performance" end comment
  -> from emp;
```

Table: Boxoffice

| Movie_id | Rating | Domestic_sales | International_sales |
|---|---|---|---|
| 5 | 8.2 | 380843261 | 555900000 |
| 14 | 7.4 | 268492764 | 475066843 |
| 8 | 8 | 206445654 | 417277164 |
| 12 | 6.4 | 191452396 | 368400000 |
| 3 | 7.9 | 245852179 | 239163000 |
| 6 | 8 | 261441092 | 370001000 |

if rating < 3 then ‘bad’
if rating >=3 and <5 the ok
if rating>=5 and <7 then good
otherwise excellent

```
select movieid,rating, case when rating<3 then ‘bad’
when rating<5 then ‘ok’
when rating<7 then ‘good’
else ‘excellent’ end status
from boxoffice
```