Day 9:

Functions to handle null values

1)NVL function

Function use to handle null values

something + null --> null

ex 100+null ---> null

100+0 --> 100

>select salary,comm,salary+comm as gross\_salary from employ;

NVL is a buit in function , which takes 2 argument.

syntax --> nvl(arg1,arg2)

if arg1 is null then returns arg2

if arg1 is not null then nvl returns arg1 only

-dual is a inbuilt table in oracle,having 1 dummy column with value'X'

& 1 row & having varchar2(1) data type

(It is often used in sql queries for testing and evaluating

expression)(use for testing sql statements)

>select nvl(null,1) from dual;

1

>select nvl(100,1) from dual;

100

>select 100+null from dual;

null

>select ename,salary,comm,salary+nvl(comm,0) as total\_salary from employ;

2)NVL2 -->enhanced version of nvl.

it is a built in function, which takes 3 argument.

syntax-->

nvl2(arg1,arg2,arg3)

if arg1 is null then it returns arg3

if arg1 is not null then it returns arg2

>select nvl2(null,50,100) from dual;

100

>select nvl2(90,50,100) from dual;

50

3) coalesce --> it helps you to deal with null values

coalesce can take multiple arguments

syntax--> coalesce(arg1,arg2,.....argN)

It returns the first not null argument from the list of

arguments.

>select coalesce(null,3,5,8,4) from dual;

3

>select coalesce(null,null,5,8,4) from dual;

5

>select coalesce(3,5,null,8,4) from dual;

3

>select ename,salary,comm,salary+coalesce(comm,0) as total\_salary from employ;

####Aggregate functions

Aggregate function is function where values of multiple

rows/records are joined together to get a single value output

Here rows are grouped into single row

Common aggregate functions are

-max

-min

-sum

-avg

-count

Fetch maximum salary from table

>select max(salary) from employ

Fetch minimum salary from table

>select min(salary) from employ

Fetch sum of salary from the table

>select sum(salary) from employ;

Fetch average salary from the table

>select avg(salary) from employ;

Fetch number/count of records from the table

>select count(\*) from employees;

>select count(\*) as num\_of\_records from employees;

-Fetch number of employees have salary more than 9000

>select count(\*) from employees where salary>9000;

####group by

used to group or categorize the data

deptno

20

30

20

20

40

50

>select deptno from emp

group by deptno;

20

30

40

50

Input

deptno emp\_name

20 kk

30 aa

20 bb

20 cc

40 dd

50 ee

>Fetch number of employees in each department

>select deptno,count(\*) as total\_num\_of\_employees from avdemp

group by deptno;

deptno total\_num\_of\_employees

20 3

30 1

40 1

50 1

Fetch department that contains more than 2 employees

>select deptno,count(\*) as total\_number\_of\_employees from avdemp

group by deptno

having count(\*)>2;

where--> used to filter table data

u cannot use aggregate function in where clause

ex

>select deptno,count(\*) as total\_number\_of\_employees from avdemp

group by deptno

where count(\*)>2

-->error

having-->use to filter group by data

u can use aggregate function in having clause

having clause only use after group by

>select \* from (

select deptno,count(\*) as total\_num\_of\_employees from avdemp

group by deptno)

where total\_num\_of\_employees>2;

Fetch department and maximum salary in that department

>select deptno,max(sal) from emp

group by deptno;

Fetch department and maximum salary in that department

and maximum salary greater than 70000

>select deptno,max(sal) from emp

group by deptno

having max(sal)>70000;

ex--->Fetch duplicate country name

Input

country

India

pakistan

india

srilanka

output

India

>select country,count(\*) from country1

group by country

having count(\*)>1

>select country\_name,state,count(\*) from states

group by country\_name,state

having count(\*)>1;