

* Calculation by using logical operators

In oracle we can perform logical conditions by using three logical operators that are the AND operator :-

The oracle engine will process all rows in a table & display the result only when all of the condⁿ's specified using the 'and' operator are satisfied.

eg:-

find the products whose selling price is greater than 2000 & less than or equal to 5000.

```
select prod-name from product-master
where    cell-price > 2000 and cell-price <=
5000;
```

2) OR operator :-

The oracle engine will process all rows in a table & display the result only when any of the conditions specified using the 'or' operator are satisfied.

eg:- find the products whose selling-price is greater than 2000 or less than or equal to 5000

select Prod-name from Product-master
where cell-price ≥ 20

~~Not (cell-price ≥ 2000~~

cell-price > 2000 or cell-price ≤ 5000 ;

3) Not operator :-

The Oracle engine will process all rows in a table & display the result only when none of the condition specified using 'not' operator are specified.

eg:- find the list of all clients who are not in a city Bombay or than Dehli.

select * from Client-master
where not (city = 'Bombay' or
city = 'Dehli');

* Range ~~searching~~ searching :-

In order to select data i.e. within a range of values, the 'bet' operator is used. This operator

allows the selection of rows that contain values within a specified lower & upper limit. The range given after the word 'betⁿ' is inclusive.

The lower value must be coded first. The two values in betⁿ the range must be linked with the keyword 'and'. A between operator can be used with both character & numeric data types.

eg: ① select name from client-master
where salary between 5000 and 10000

The above select will retrieve all the records from client-master table where the salary is in betⁿ 5000 & 10000 (both values inclusive).

② Retrieve prod-no, description & profit-percent from the prod-master table where the values contained in the field profit-percent are not betⁿ 10 & 15 both inclusive.

select prod-no, description, profit-percent
from prod-master
where profit-percent NOT Between
10 AND 15;

The above select will retrieve all the records from the prod-master table except where the profit-percent is in betⁿ 10 & 15.

17 Retrieve all the information about suppliers whose name begins with the letters Ra from supplier-master

→ select * from supplier-master
where name like 'Ra%'; ↵

21 Retrieve the supplier_name, address₁, address₂, city from the table supplier-master where the supplier name is three characters long & at two characters are as

→ select supplier_name, address₁, address₂, city from supplier-master
where supplier_name like 'as-'; ↵

31 Find the names of all clients having A as the second letter in their names

→ select name from client-master
where name like '-a%'; ↵

41 Find out the clients whose state is a city whose second letter is

A.

→ select * from client-master
where city like '-a%'; ←

5] Retrieve the supplier name, city from the table supplier-master where the supplier name is either Rama, krushna or shyam.

→ select supplier name, city from
supplier-master
where supplier name In ('Rama', 'krushna',
'shyam'); ←

6] Retrieve the supplier name, city from the table supplier-master where the supplier name is not either Rama, krushna, or shyam.

→ select supplier name, city from
supplier-master
where supplier name NOT In ('Rama',
'krushna', 'shyam');

* Pattern matching :-

The use of the like predicate :-

The comparison operators discussed used to compare one value exactly to one other value. Such a precision may not always be desired or necessary. For this purpose Oracle provides a predicate 'like'.

The 'like' predicate permits to you for comparison of one string value with another string value, which is not identical. This is achieved by using wildcard characters. These characters are

- 1) % :- The percent sign matches any string
- 2) _ :- The underscore matches any character. single character.

eg:- @Retrieve all infoⁿ about suppliers whose name begin with the letters 'ja' from supplier-master

```
select * from supplier_master  
where supplier_name like 'ja%';
```


② Retrieve all information about supplier where the second character of name are either 'r' or 'h'

Select * from supplier-master
where supplier name like 'r%'
OR supplier name like 'h%';

The IN and NOT IN predicates :-

The arithmetic operator (=) compares a single value to another single value. In case values need to be compared to a list of values then 'IN' predicate is used. One can check a single value against multiple values by using the IN predicate.

eg:- Retrieve the name, add1, add2, city from the table supplier-master where the sup name is either Mayura or Supriya or Sangita or Amarata.

Select name, add1, add2, city p
from supplier-master
where name IN ('Mayura', 'Supriya',
'Sangita', 'Amarata');

29/01/05

The NOT IN predicate is the opposite of the IN predicate. This will select all the rows where values do not match all of the values in the list

eg:-

```
select name, add1, add2, city
from supplier-master
where name NOT IN ('Mayura', 'Sypriya',
                  'Sangita', 'Amarata');
```

The above statement retrieve all the rows from the supplier-master table where the supplier name is not equal to the ones supplied.