Simple index and Composite index.

If we need to search data frequently or if we have more than 15000 records then we can create index.

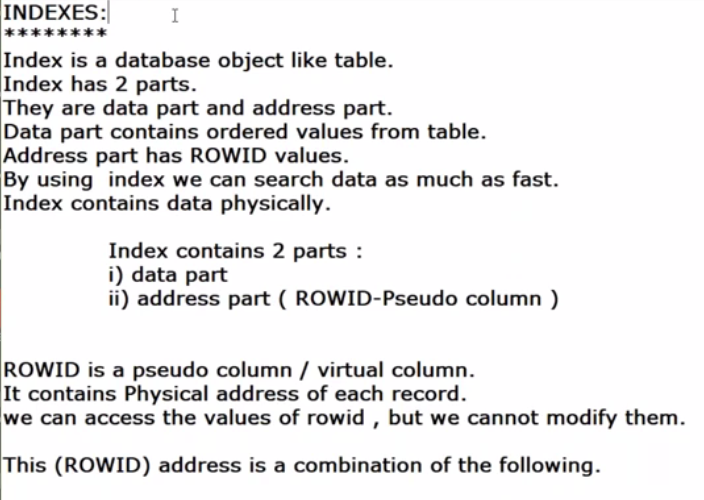
Unnecessarily don’t create any index.

Reduces search time.

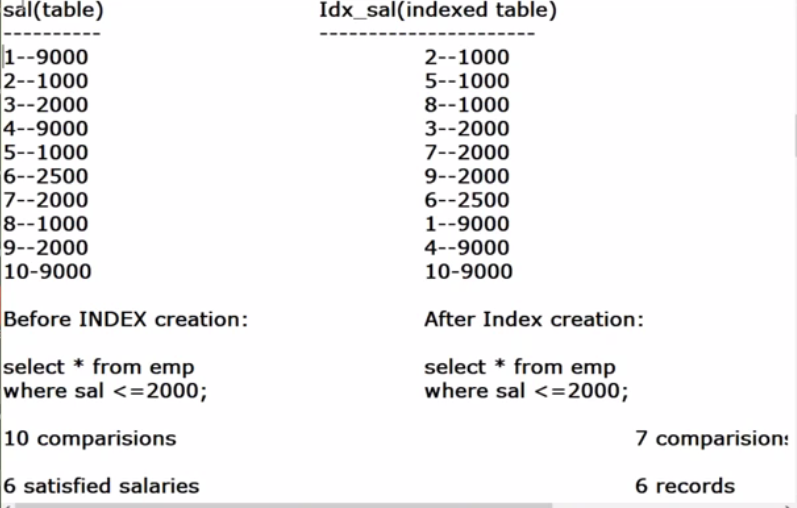
Its like index in text book to search data with in less time.

By default index maintains order.

A to z for chaar. Min to max for numerics.



If we create index on a table, then if we want to search something belong to that table.. search first goes to index.

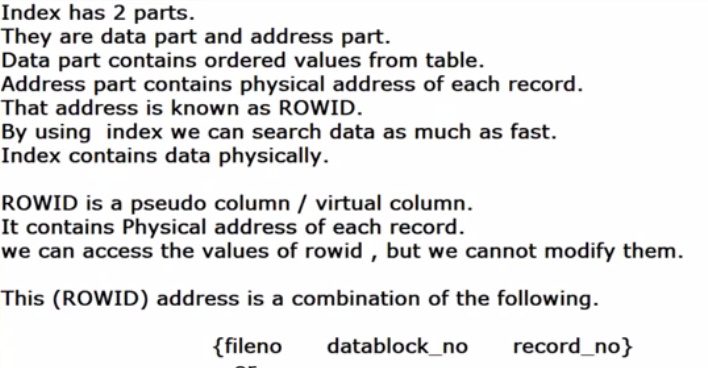


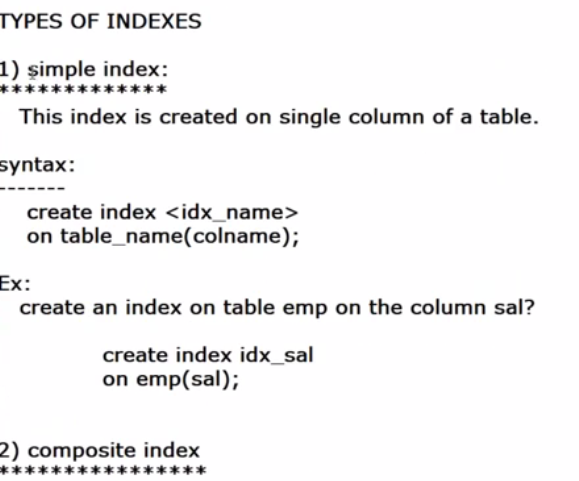
Search happens upto 2500 only,

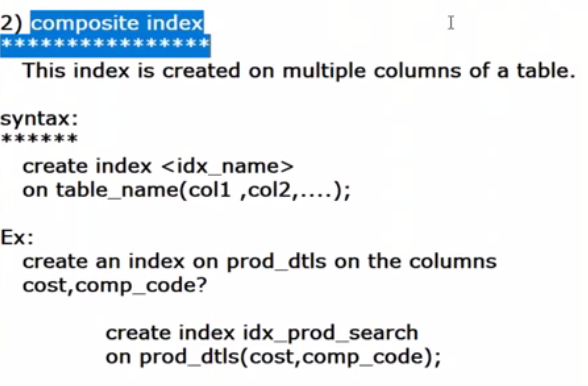
At 2500 condition failed, then it wont verify further as the column is in ascending order.

Only 7 comparisons done in index.

Index contains data physically but views contains data logically.







Searching records based on conditions applied to 2 or more columns is called composite index.

SQL> --before index creation

SQL> select sal from emp;

SAL

----------

5000

2850

2450

2975

3000

3000

800

1600

1250

1250

1500

SAL

----------

1100

950

1300

14 rows selected.

SQL> create index idx\_sal on emp(sal);

Index created.

SQL> --after index created.

SQL> select sal from emp;

SAL

----------

5000

2850

2450

2975

3000

3000

800

1600

1250

1250

1500

SAL

----------

1100

950

1300

14 rows selected.

SQL>

SQL> --before index created.

SQL> select sal from emp where sal<=2000;

SAL

----------

800

1600

1250

1250

1500

1100

950

1300

8 rows selected.

SQL> create index idx\_sal on emp(sal);

Index created.

SQL> --after index created.

SQL> select sal from emp where sal<=2000;

SAL

----------

800

950

1100

1250

1250

1300

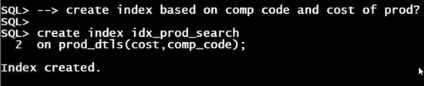
1500

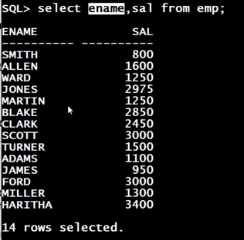
1600

8 rows selected.

In the above example, the salaries displayed after index creatiojn are ordered salaries.

Bcz when we create index data arranges in sorting order.





SQL> select initcap(ename), sal, 0.2\*sal from emp where sal<=2000;

INITCAP(EN SAL 0.2\*SAL

---------- ---------- ----------

Smith 800 160

Allen 1600 320

Ward 1250 250

Martin 1250 250

Turner 1500 300

Adams 1100 220

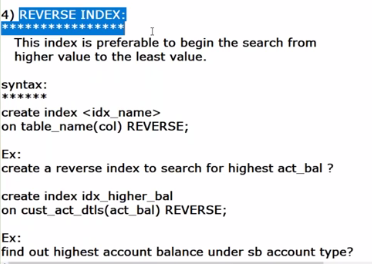
James 950 190

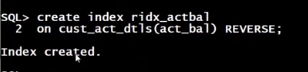
Miller 1300 260

8 rows selected.

SQL> create index idx\_emp

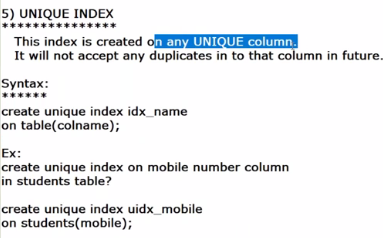
2 on emp(initcap(ename), sal, (0.2\*sal));



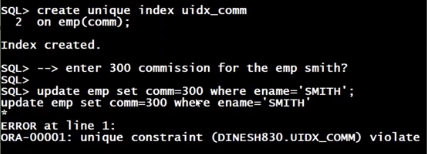


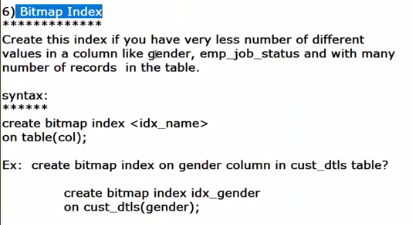
Data will be searched from higher to lower.

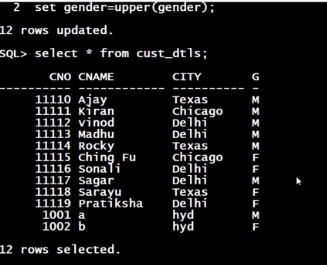
Search in descending order.



It also acts as unique constraint, it wont accept duplicate values after index created.









Primary keys also considered as indexes ?

