**NAME:ROHAN NYATI**

**SAP ID:500075940**

**ROLL NO. : R177219148**

**BATCH-5(AI&ML)**

**EXPERIMENT-7**

**Title: 7.** To understand the concepts of Index.

**Objective:** Students will be able to implement the concept of index.

**1.** Execute the following index related queries:

1) Create an index of name employee\_idx on EMPLOYEES with column Last\_Name, Department\_id 2) Find the ROWID for the above table and create a unique index on employee\_id column of the EMPLOYEES.

3) Create a reverse index on employee\_id column of the EMPLOYEES.

4) Create a unique and composite index on employee\_id and check whether there is duplicity of tuples or not.

5) Create Function-based indexes defined on the SQL functions UPPER(column\_name) or LOWER(column\_name) to facilitate case-insensitive searches(on column Last\_Name).

6) Drop the function based index on column Last\_Name.

**TABLE:**

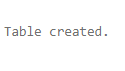
create table employees( employee\_id varchar(10) ,

first\_name varchar(30) not null ,

last\_name varchar(30) not null ,

salary integer not null ,

department\_id number(5));



**DATA:**

insert into employees(employee\_id ,first\_name , last\_name , salary , department\_id)values('e101' , 'tony' , 'jackson' , 80000 , 111) ;

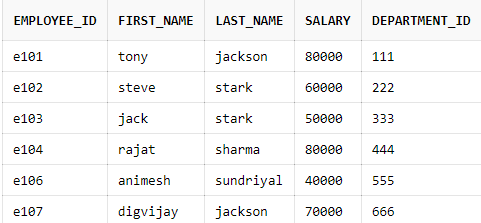
insert into employees(employee\_id ,first\_name , last\_name , salary , department\_id)values('e102' , 'steve' , 'stark' , 60000 , 222) ;

insert into employees(employee\_id ,first\_name , last\_name , salary , department\_id)values('e103' , 'jack' , 'stark' , 50000 , 333) ;

insert into employees(employee\_id ,first\_name , last\_name , salary , department\_id)values('e104' , 'rajat' , 'sharma' , 80000 , 444) ;

insert into employees(employee\_id ,first\_name , last\_name , salary , department\_id)values('e106' , 'animesh' , 'sundriyal' , 40000 , 555) ;

insert into employees(employee\_id ,first\_name , last\_name , salary , department\_id)values('e107' , 'digvijay' , 'jackson' , 70000 , 666) ;



**ANSWERS:**

create index employee\_idx on employees(last\_name , department\_id) ;

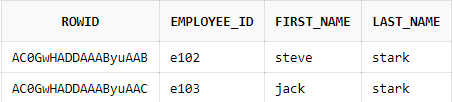


select ROWID , employee\_id , first\_name , last\_name

from employees

where last\_name = 'stark' ;

create unique index unique\_idx on employees(employee\_id) ;



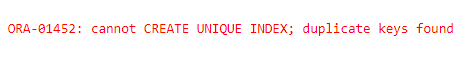


create index reverse\_idx on employees(employee\_id) REVERSE ;



insert into employees(employee\_id ,first\_name , last\_name , salary , department\_id)values('e106' , 'animesh' , 'sundriyal' , 40000 , 555) ;

create unique index uni\_idx on employees(employee\_id);

Create index emp\_idx ON employees (UPPER(last\_name));

drop index emp\_idx ;

