LAB1

create table employee(emp\_no INT, emp\_name VARCHAR(20), emp\_address VARCHAR(30));

insert into employee values(1, 'ri', 'delhi');

insert into employee values(2, 'keyur', 'gujju');

insert into employee values(3, 'mri', 'kolk');

insert into employee values(4, 'aditi', 'hyd');

select \* from employee;

select emp\_name from employee;

select \* from employee where emp\_address='hyd';

alter table employee add(salary numeric(10,2));

update employee set salary = 5000;

select \* from employee;

desc employee;

delete from employee where emp\_address='kolk';

rename employee to employee1;

desc employee1;

drop table employee1;

LAB2

select title from course where credits=3;

select \* from takes;

select takes.course\_id, course.title from takes, course

where id=12345 and takes.course\_id=course.course\_id;

create table emp(

empno int,

gender varchar(1) not null,

salary numeric(5,2) not null,

address varchar(30) not null,

primary key(empno),

check (gender in('M', 'F'))

);

create table dept

(

deptno int,

deptname varchar(20) unique,

location varchar(20),

primary key(deptno)

);

alter table emp modify (dno integer);

alter table emp add constraint fkdept foreign key(dno) references dept(deptno);

insert into dept values(1, 'a', 'b');

insert into dept values(1, 'n', 'c');

insert into emp values(1, 'M', 200, 'del', 1);

insert into emp values(2, 'M', 200, 'del', 1);

select \* from emp

alter table emp drop constraint fkdept;

alter table emp add constraint fk foreign key (dno) references dept(deptno) on delete cascade;

alter table emp modify salary default 100;

insert into emp(empno, gender, address, dno) values(3, 'F', 'kol', 1)

select id from instructor where salary between 40000 and 90000;

select id from instructor where instructor.id not in(select id from teaches);

select \* from section

select student.name, course.title, takes.year from student

join takes on student.id=takes.id

join course on takes.course\_id=course.course\_id

join section on section.course\_id=takes.course\_id where section.room\_number=303

select student.name, takes.course\_id as "c-name"

from student join takes on student.id=takes.id

where takes.year = 2010

select distinct t.name, t.salary as "inst-salary"

from instructor t, instructor s

where t.salary<s.salary;

select name from instructor

where dept\_name like '%ch%'

select name, length(name) from student

select dept\_name, substr(dept\_name, 3, 3) from department

select upper(name) from instructor

select nvl(name, 0) from instructor

select salary, round((salary/3), .01) from instructor

alter table emp add dob date

insert into emp values(8, 'M', 100, 'del', 1, to\_date('10112005', 'dd-mm-yy'))

select name, to char

select course\_id from course join section on course\_id where

LAB 3

select course\_id from section where semester='Fall' and year=2009

minus

select course\_id from section where semester='Spring' and year=2010

select title

from course

where course.course\_id not in (select course\_id from takes)

select course\_id

from section

where semester='Fall' and year ='2009'

and course\_id

in (select course\_id

from section

where semester='Summer' and year ='2010')

SELECT count(takes.ID)

from teaches join instructor on teaches.id=instructor.id

join takes on takes.course\_id=teaches.course\_id

where instructor.id=10101;

select count(distinct takes.id) from teaches join instructor on teaches.id=instructor.id

join takes on takes.course\_id=teaches.course\_id where instructor.id=10101;

select name from student where name in (select name from instructor

select name from instructor where salary > some(select salary from instructor)

select name from instructor where name not in

(select name from instructor where salary < some(select salary from instructor))

select dept\_name, avg(salary) as avr

from instructor

group by dept\_name

order by avr desc

select dept\_name from department

where budget < (select avg(salary) from instructor)

select course\_id from course where dept\_name='Biology'

intersect

select course\_id from takes

select course\_id from

(

select course\_id, year, count(course\_id) as c from section

where year=2009

group by course\_id, year

) where c =1

select \* from department

select id from(

select takes.id, count(takes.course\_id) as c from takes join course on takes.course\_id=course.course\_id

where dept\_name = 'Comp. Sci.'

group by id

) where c = 2

select i.dept\_name, avg(i.salary)from

instructor i join department d on i.dept\_name=d.dept\_name

group by i.dept\_name

having avg(i.salary) > 42000

create view all\_courses as

select s.course\_id, s.sec\_id, s.building, s.room\_number

from section s join department d on s.building=d.building

where dept\_name = 'Physics' and semester='Fall' and year = 2009

select \* from all\_courses

create view department\_total\_salary as

select dept\_name

LAB 4

SELECT course\_id, count(id)

from takes group by course\_id

select d.dept\_name, count(t.id)

from takes t join course c on t.course\_id=c.course\_id

join department d on d.dept\_name = c.dept\_name

group by d.dept\_name having count(t.id)>10

select d.dept\_name, count(c.course\_id)

from department d join ciewourse c on c.dept\_name=d.dept\_name

group by d.dept\_name

select dep, avr

from (

select d.dept\_name as dep, avg(i.salary) as avr

from department d join instructor i on d.dept\_name=i.dept\_name

group by d.dept\_name

) where avr >42000

select s.sec\_id, count(t.id)

from section s join takes t on s.course\_id = t.course\_id

where s.semester='Spring' and s.year=2009

group by s.sec\_id;

select \* from prereq order by course\_id

select max(s) from(

select d.dept\_name, sum(i.salary) as s

from instructor i join department d on d.dept\_name=i.dept\_name

group by d.dept\_name)

select max(cnt) from(

select s.sec\_id, count(t.id) as cnt

from section s join takes t on s.course\_id = t.course\_id

where s.semester='Spring' and s.year=2010

group by s.sec\_id);

select d.dept\_name, count(i.id), avg(i.salary)

from department d join instructor i on d.dept\_name = i.dept\_name

group by d.dept\_namev

having count(i.id) > 5 and avg(i.salary) > 50000

update student

set dept\_name = case when dept\_name='Comp. Sci.' then 'Biology'

else dept\_name

end

select \* from student

update instructor

set salary = case

when salary>100000 then salary\*1.03

else salary\*1.05

end;

select \* from instructor