Lab 2 : **Dipesh Singh - 190905520**

Question 1 :

Create Employee table with following constraints:

•Make EmpNo as Primary key.

•Do not allow EmpName, Gender, Salary and Address to have null values.

•Allow Genderto have one of the two values: 'M', 'F'.

```
create table Employee (EmpNo char(9) primary key, EmpName
varchar(20) not null, gender char(1) check(gender in ('F', 'M')) not
null, salary number(8,0) not null, address varchar(50) not null, Dno
number(10));
```

Question 2 :

Create Department table with following:

•Make DeptNo as Primary key

•Make DeptName as candidate key

```
create table Department(DeptNo number(10) primary key, DeptName
varchar(20) unique, location varchar(50));
```

Question 3 :

Make DNo of Employee as foreign key which refers to DeptNo of Department.

```
alter table Employee modify foreign key(Dno) references
Department(DeptNo);
```

Question 4 :

Insert few tuples into Employee and Department which satisfies the above constraints.

```
insert all into Department values(5522, 'Computer Sc', 'NLH') into
Department values(5520,'Chemical', 'AB-2') into Department
values(5524, 'IT', 'AB-5') select * from dual;
```

```
insert all into Employee values('190905520', 'Dipesh', 'M', 300000,
'Delhi', 5522) into Employee values('190905522', 'Ayush', 'M',
200000, 'Kolkata', 5522) into Employee values('190905524',
'Swarnim', 'F', 400000, 'Banaras', 5522) into Employee
values('190909008', 'Khushal', 'M', 100000, 'Mumbai', 5524) select *
from dual;
```

Question 5 :

Try to insert few tuples intoEmployee and Department which violates some of the above constraints.

```
insert into Employee values('190905520', 'Pritima', 'F', 300000,
'Delhi', 5522);

insert into Employee values('190905569', 'Pritima', 'X', 3000,
'Delhi', 5522);

insert into Employee values('190905569',, 'M', 3000, 'Delhi', 5522);
```

Question 6 :

Try to modify/delete a tuple which violates a constraint.

```
delete from Department where DeptNo=5522;
```

Question 7 :

Modify the foreign key constraint of Employee table such that whenever a department tuple is deleted, the employees belonging to that department will also be deleted.

```
select constraint_name, table_name, constraint_type from
user_constraints;

alter table Employee drop constraint SYS_C007022;

alter table Employee add constraint FK foreign key(Dno) references
Department(DeptNo) on delete cascade;
```

Question 8 :

Create a named constraint to set the default salary to 10000 and test the constraint by inserting a new record.

```
alter table Employee modify(salary default 10000);
```

Question 9 :

List all Students with names and their department names.

```
select name, dept_name from student;
```

Question 10 :

List all instructors in CSE department.

```
select * from instructor where dept_name='Comp. Sci.';
```

Question 11 :

Find the names of courses in CSE department which have 3 credits.

```
select title from course where dept_name='Comp. Sci.' and credits=3;
```

Question 12 :

For the student with ID 12345 (or any other value), show all course-id and title of all courses registered for by the student.

```
select takes.course_id, course.title from takes, course where
ID=12345 and takes.course_id=course.course_id;
```

Question 13 :

List all the instructors whose salary is in between 40000 and 90000.

```
select * from instructor where salary>=40000 and salary<=90000;
```

Question 14 :

Display the IDs of all instructors who have never taught a course.

```
select * from instructor natural left outer join teaches where
sec_id is null;
```

Question 15 :

Find the student names, course names, and the year, for all students those who have attended classes in room-number 303.

```
select name, title, takes.year from student, section, course, takes
where room_number=514 and course.course_id=section.course_id and
course.course_id=takes.course_id and takes.ID=student.ID and
takes.year=section.year and takes.sec_id=section.sec_id and
section.semester=takes.semester;
```

Question 16 :

For all students who have opted courses in 2015, find their names and course id's with the attribute course title replaced by c-name.

```
select name, course_id as c_name from student natural join takes
where takes.year=2015;
```

Question 17 :

Find the names of all instructors whose salary is greater than the salary of at least one instructor of CSE department and salary replaced by inst-salary.

```
select distinct a.name, a.salary as inst_salary from instructor a,
instructor b where b.dept_name='Comp. Sci.' and a.salary>b.salary;
```

Question 18 :

Find the names of all instructors whose department name includes the substring 'ch'.

```
select name from instructor where dept_name like '%ch%';
```

Question 19 :

List the student names along with the length of the student names.

```
select name, length(name) from student;
```

Question 20 :

List the department names and 3 characters from 3rdposition of each department name.

```
select dept_name, substr(dept_name, 3, 3) from Department;
```

Question 21 :

List the instructor names in upper case.

```
select upper(name) from instructor;
```

Question 22 :

Replace NULL with value1(say 0) for a column in any of the table

```
select nvl(grade,'F') from takes;
```

Question 23 :

Display the salary and salary/3 rounded to nearest hundred from Instructor.

```
select salary, round(salary/3, -2) as rounded_salary from
instructor;
```