DBMS Assignment

Submitted By:-

- Mabbu Saivardhan Reddy
- INT240, AU 2019
- Accolite, Hyderabad

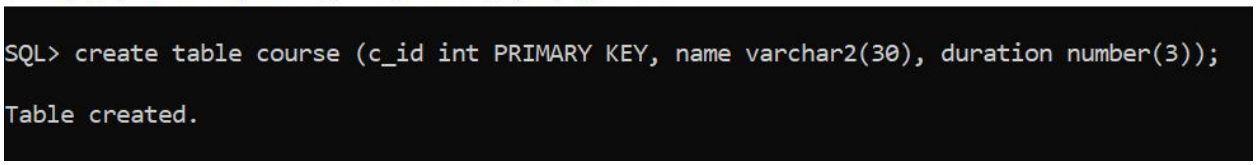
Submitted To:-

- Gouthami Mogili

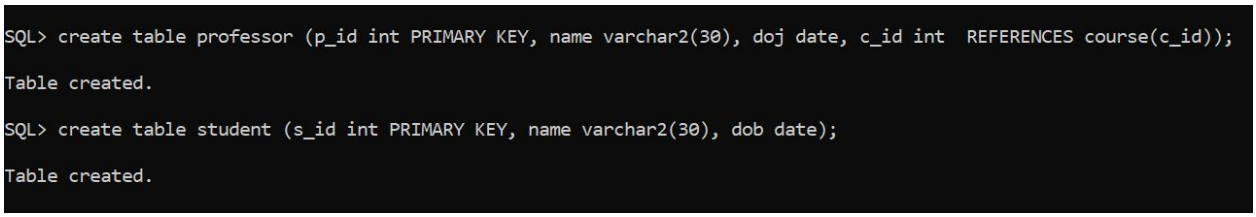
---

1) Prepare database based on your ER Model and insert some sample data

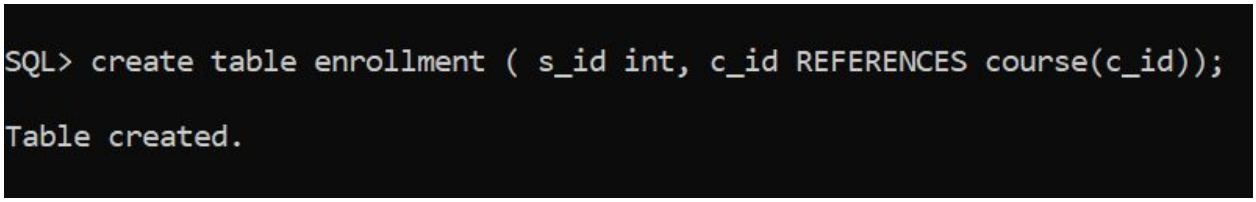
-> Courses table creation:



-> Professor, Student tables creation:



-> Enrollment table creation:



-> Inserting data into Courses table:

```
SQL> insert into course values(1, 'DBMS', 3);

1 row created.

SQL> insert into course values(2, 'JAVA', 4);

1 row created.

SQL> insert into course values(3, 'ANGULAR', 3);

1 row created.

SQL> insert into course values(4, 'REACT', 3);

1 row created.

SQL> insert into course values(5, 'GIT', 4);

1 row created.
```

-> Inserting data into Professors table:

```
SQL> insert into professor values(1, 'Goutami', TO_DATE('25/07/2016', 'dd/mm/yyyy'), 1);

1 row created.

SQL> insert into professor values(2, 'Goutami', TO_DATE('25/07/2016', 'dd/mm/yyyy'), 5);

1 row created.

SQL> insert into professor values(3, 'Ankit', TO_DATE('13/06/2014', 'dd/mm/yyyy'), 3);

1 row created.

SQL> insert into professor values(4, 'Devesh', TO_DATE('10/05/2015', 'dd/mm/yyyy'), 4);

1 row created.

SQL> insert into professor values(5, 'Chandra', TO_DATE('30/03/2012', 'dd/mm/yyyy'), 2);

1 row created.
```

-> Inserting data into Students table:

```
SQL> insert into student values(1, 'Kasturi', TO_DATE('10/08/1998', 'dd/mm/yyyy'));

1 row created.

SQL> insert into student values(2, 'vardhan', TO_DATE('25/07/1998', 'dd/mm/yyyy'));

1 row created.

SQL> insert into student values(3, 'Shravya', TO_DATE('10/04/1998', 'dd/mm/yyyy'));

1 row created.

SQL> insert into student values(4, 'Manohar', TO_DATE('02/06/1997', 'dd/mm/yyyy'));

1 row created.

SQL> insert into student values(5, 'Kavya', TO_DATE('28/01/1999', 'dd/mm/yyyy'));

1 row created.

SQL> insert into student values(6, 'Anjali', TO_DATE('25/08/1998', 'dd/mm/yyyy'));

1 row created.

SQL> insert into student values(7, 'Nitish', TO_DATE('19/02/1998', 'dd/mm/yyyy'));

1 row created.
```

-> Inserting data into Enrollment table:

```
SQL> insert into enrollment values(1,1);

1 row created.

SQL> insert into enrollment values(1,3);

1 row created.

SQL> insert into enrollment values(2,2);

1 row created.

SQL> insert into enrollment values(2,1);

1 row created.

SQL> insert into enrollment values(2,4);

1 row created.

SQL> insert into enrollment values(3,5);

1 row created.

SQL> insert into enrollment values(3,4);

1 row created.

SQL> insert into enrollment values(3,1);

1 row created.

SQL> insert into enrollment values(4,2);

1 row created.

SQL> insert into enrollment values(4,5);

1 row created.
```

2a) SQL Query to provide Student details with which course he/she is enrolled for

select s.s_id, s.name, s.dob, c.name "Course Name" from student s, course c, enrollment e where e.s_id=2 and e.s_id=s.s_id and e.c_id=c.c_id;

```
SQL> select  s.s_id, s.name, s.dob, c.name "Course Name" from student s, course c, enrollment e where e.s_id=1 and e.s_id=s.s_id and e.c_id=c.c_id;

      S_ID NAME                           DOB       Course Name
---------- ------------------------------ --------- ------------------------------
         1 Kasturi                        10-AUG-98 DBMS
         1 Kasturi                        10-AUG-98 ANGULAR

SQL> select  s.s_id, s.name, s.dob, c.name "Course Name" from student s, course c, enrollment e where e.s_id=2 and e.s_id=s.s_id and e.c_id=c.c_id;

      S_ID NAME                           DOB       Course Name
---------- ------------------------------ --------- ------------------------------
         2 vardhan                        25-JUL-98 JAVA
         2 vardhan                        25-JUL-98 DBMS
         2 vardhan                        25-JUL-98 REACT
```

2b) SQL Query to get the course which maximum students enrolled for and which professor is taking up that course

select pr.name "Professor", co.name "Course" from professor pr, course co where pr.c_id=co.c_id and co.c_id=(select * from (select e.c_id from enrollment e group by e.c_id order by count(*) desc) where rownum=1);

```
SQL> select pr.name "Professor", co.name "Course" from professor pr, course co where pr.c_id=co.c_id and co.c_id=(select * from (select e.c_id from enrollment e  group by e
.c_id order by count(*) desc ) where rownum=1);

Professor                 Course
------------------------- -------------------------
Goutami                   DBMS
```

3) Write a procedure which accepts Student id as input and output would be the professor details for the courses he enrolled for.
Cases to be handled : if student_id doesn't exists in Student table, then delete the entry.

```
CREATE OR REPLACE PROCEDURE profNames(student_id IN int)
IS

c_pname varchar2(30);
c_sname varchar2(30);
c_cname varchar2(30);

CURSOR c_profs IS select c.name, p.name from course c, professor p, enrollment e where e.s_id=student_id and e.c_id=p.c_id and p.c_id=c.c_id ;

BEGIN
select name into c_sname from student where s_id=student_id ;
dbms_output.put_line('Student Name: '||c_sname);

OPEN c_profs;
LOOP
FETCH c_profs into c_cname, c_pname;
EXIT WHEN c_profs%notfound;
dbms_output.put_line('Course: '||c_cname||' Professor: '||c_pname);
END LOOP;
CLOSE c_profs;

END;
```