SDM College of Engineering and Technology

Dhavalagiri, Dharwad-580 002. Karnataka State. India.
Email: principal@sdmcet.ac.in, cse.sdmcet@gmail.com
Ph: 0836-2447465/ 2448327 Fax: 0836-2464638 Website: sdmcet.ac.in

Department
of
COMPUTER SCIENCE AND ENGINEERING

LABORATORY REPORT

[2UCSC501 DATABASE MANAGEMENT SYSTEMS]

Odd Semester: Aug-Dec-2024

Course Teacher: Dr. U.P.Kulkarni

2024- 2025

Submitted by
By

Ms. Ambika Chavan
2SD23CS400
5th Semester A division

W1:- CREATING EMPLOYEE TABLE

```
sql> create table employee (
        empno integer not null
        constraint EMPLOYEE_PK_VIOLATION
        primary key,
        empname char(20) not null,
        sex char(1) not null
        constraint EMPLOYEE_SEX_VIOlATION
        check(sex in ('m','f')),
        phone integer not null,
        dob date default '15-apr-68' not null );
```

****

W3:- DESCRIBING EMPLOYEE TABLE

```
sql> describe employee;
 Name                      Null?         Type
 ----------------------------------------- -------- -------------
 EMPNO                NOT NULL      NUMBER(38)
 EMPNAME              NOT NULL      CHAR(20)
 SEX                  NOT NULL      CHAR(1)
 PHONE                NOT NULL      NUMBER(38)
 DOB                  NOT NULL      DATE
```

****

W6:- INSERTING VALUES INTO EMPLOYEE TABLE

```
sql> insert into employee values(1,'manish','m',9916236624,'23-apr-05');
1 row created.
sql> insert into employee values(2,'harish','m',9916236626,'24-jul-06');
1 row created.
sql> insert into employee values(3,'ambika','f',9886507658,'16-jul-06');
1 row created.
sql> insert into employee values(4,'deepa','f',9916236620,'25-jul-08');
1 row created.
```

****

W9:- DISPLAYING VALUES OF EMPLOYEE TABLE

```
sql> select* from employee;

 EMPNO EMPNAME          S     PHONE DOB
------ -------------------- - ---------- ---------
    1 manish            m 9916236624 23-APR-05
    2 harish            m 9916236626 24-JUL-06
    3 ambika            f 9886507658 16-JUL-06
    4 deepa             f 9916236620 25-JUL-08
```

****

W2:- CREATING PROJECT TABLE

```
sql> create table project (
          projectno integer not null,
          project char(20) not null,
          chiefarchitect char(20) default 'upk' not null,
          constraint PROJECT_PK_VIOLATION
          primary key (projectno) );
```

****

W4:- DESCRIBING TABLE PROJECT

```
sql> describe project;

Name                          Null?    Type
----------------------------------------- -------- ----------------------------
 PROJECTNO                     NOT NULL NUMBER(38)
 PROJECT                       NOT NULL CHAR(20)
 CHIEFARCHITECT                NOT NULL CHAR(20)
```

****

W7:- INSERTING VALUES INTO PROJECT TABLE

```
sql> insert into project values(100,'dbms','upk');
1 row created.
sql> insert into project values(101,'web development','prattu');
1 row created.
sql> insert into project values(102,'java','deepa');
```

```
1 row created.

sql> insert into project values(103,'aiml','motu');

1 row created.
```

****

W10:- DISPLAYING VALUES OF PROJECT TABLE

```
sql> select* from project;

 PROJECTNO PROJECT            CHIEFARCHITECT
---------- -------------------- --------------------
       100 dbms                 upk
       101 web development      prattu
       102 java                 deepa
       103 aiml                 motu
```

****

W3:- CREATING ASSIGNED_TO TABLE

```
sql> create table assigned_to (
            empno integer not null,
            projectno integer not null,
            constraint ASSIGNED_TO_PK_VIOLATION
            foreign key(empno)
            references employee,
            constraint ASSIGNED_TO_FK_SD_VIOLATION
            foreign key(projectno)
            references project  );
```

****

W5:- DESCRIBING ASSIGNED_TO TABLE

```
sql> describe assigned_to;

 Name                          Null?    Type
 ----------------------------------------- -------- ----------------------
 EMPNO                         NOT NULL NUMBER(38)
 PROJECTNO                       NOT NULL NUMBER(38)
```

****

W8:- INSERTING VALUES INTO ASSIGNED_TO TABLE

```
sql> insert into assigned_to values(1,101);

1 row created.

sql> insert into assigned_to values(2,100);

1 row created.

sql> insert into assigned_to values(4,103);

1 row created.

sql> insert into assigned_to values(3,102);

1 row created.

sql>  insert into assigned_to values(3,101);

sql>  insert into assigned_to values(3,103);

sql>  insert into assigned_to values(3,100);
```

****

W10:- DISPLAYING VALUES OF ASSIGNED_TO TABLE

```
sql> select* from assigned_to;

 EMPNO  PROJECTNO

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

    1       101

    2       100

    4       103

    3       102
```

****

TERMWORK- 1:-a) entity integrity

```
sql> insert into employee values(2,'harish','m',9916236626,'24-jul-06');

ERROR at line 1:

ORA-00001: unique constraint (23CS400.EMPLOYEE_PK_VIOLATION) violated
```

---

b) row integrity

```
sql> insert into employee values(2,'harish','a',9916236626,'24-jul-06');

ERROR at line 1:

ORA-02290: check constraint (23CS400.EMPLOYEE_SEX_VIOLATION) violated
```

---

c) referential integrity

```
sql> insert into assigned_to values(4,109);

ERROR at line 1:

ORA-02291: integrity constraint (23CS400.ASSIGNED_TO_PK_VIOLATION) violated -

parent key not found
```

****

TERMWORK- 2:- a) write sql statement to obtain the 'employeeno#' of employees working on project '101'

```
sql> select empno from assigned_to where projectno='101';

EMPNO

---------

     1
```

****

termwork-2 b) :- write sql statement to display details of all employees working on project = 101.

```
sql> select e.*from employee e,assigned_to at where e.empno=at.empno and projectno='101';

output:-

            EMPNO EMPNAME     S     PHONE       DOB

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

             1 manish        m     9916236624  23-APR-05
```

****

TERMWORK-2 c):- write a sql statement to display details of all employess working on "dbms" project.

```
sql>  select e.*from employee e, assigned_to at, project pr where e.empno=at.empno and pr.projectno=at.projectno
and pr.project='dbms';

output:-

                  EMPNO   EMPNAME       S      PHONE       DOB

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

                      2        harish            m   9916236626     24-JUL-06
```

****

TERMWORK-2 d):- write a sql statement to display details of all employees working on project 1 and 2.

sql> select e.*from employee e, assigned_to at, project pr where e.empno=at.empno and pr.projectno=at.projectno and pr.projectno IN('101','102');

output :-

EMPNO	EMPNAME	S	PHONE	DOB
1	manish	m	9916236624	23-APR-05
3	ambika	f	9886507658	16-JUL-06

****

TERMWORK- 2 e) :- write a sql statement to display details of all employees working on project 101 or 102.

SQL> select e.*from Employee e, Assigned_to at where e.empno=at.empno AND(projectno=1 OR projectno=2);

Output:-

EMPNO	EMPNAME	S	PHONE	DOB	SALARY
101	John	m	9923456745	23-APR-04	
103	Pinky	f	9354456745	15-JUN-20	70000

****

TERMWORK -3) modify the schema to store information about fine to be paid by employees(multivalued attributes)

```
sql> create table fine (
      empno integer not null,
      constraint fine_pk_violation
      primary key(empno),
      constraint fine_fk_violation
      foreign key(empno)
      references employee,
      ammount integer not null,
      dob date default '15-apr-05' not null  );
```

output:- describe fine;

Name	Null?	Type
EMPNO	NOT NULL	NUMBER(38)

AMMOUNT NOT NULL NUMBER(38)

DOB NOT NULL DATE

sql> insert into fine values(1, 35000,'17-apr-05');

sql>insert into fine values(2, '30000', '12-may-87');

sql>insert into fine values(3, '15000', '17-aug-60');

sql> select * from fine;

EMPNO AMMOUNT DOB

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

2 30000 12-MAY-87

3 15000 17-AUG-60

1 35000 17-APR-05

****

TERMWORK -4) modify the schema to store the details of dependents (week entity) for all employees if exist.

sql> create table dependent (

dname varchar(20) not null,

constraint dependent_pk_violation

primary key(dname),

sex char(1) not null

constraint dependent_sex_violation

check(sex in ('m','f')),

relationship varchar(20) not null);

output: - describe dependent;

Name Null? Type

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

DNAME NOT NUL VARCHAR2(20)

SEX NOT NULL CHAR(1)

RELATIONSHIP NOT NULL VARCHAR2(20)

Sql>insert into dependent values('neha' , 'f', 'sister');

Sql>insert into dependent values('rithesh' , 'm', 'brother');

Output:- select * from dependent;

DNAME S RELATIONSHIP

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

neha f sister

rithesh m brother

****

TERMWORK -5) a) deleting rows

sql> insert into employee values(8, 'riya', 'f', 123456789,'19-sep-09');

select * from employee;

EMPNO	EMPNAME	S	PHONE	DOB
1	manish	m	9916236624	23-APR-05
2	harish	m	9916236626	24-JUL-06
3	ambika	f	9886507658	16-JUL-06
4	deepa	f	9916236620	25-JUL-08
8	riya	f	123456789	19-SEP-09

sql> delete from employee where empname='riya';

output:-

EMPNO	EMPNAME	S	PHONE	DOB
1	manish	m	9916236624	23-APR-05
2	harish	m	9916236626	24-JUL-06
3	ambika	f	9886507658	16-JUL-06
4	deepa	f	9916236620	25-JUL-08

****

b) droping table

```
sql> create table temp (
        tempno integer not null,
        tname char(20) not null,
        primary key (tempno)  );
```

describe temp;

Name	Null?	Type

TEMPNO NOT NULL NUMBER(38)

TNAME NOT NULL CHAR(20)

sql> drop table temp;

output:- table dropped.

Sql> describe temp;

ERROR:

ORA-04043: object temp does not exist

****

c) updating rows

sql> select * from employee;

EMPNO EMPNAME S PHONE DOB

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

1 manish m 9916236624 23-APR-05

2 harish m 9916236626 24-JUL-06

3 ambika f 9886507658 16-JUL-06

4 deepa f 9916236620 25-JUL-08

sql> update employee SET dob = '27-oct-04' where empno='3';

output:-

EMPNO EMPNAME S PHONE DOB

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

1 manish m 9916236624 23-APR-05

2 harish m 9916236626 24-JUL-06

3 ambika f 9886507658 27-OCT-04

4 deepa f 9916236620 25-JUL-08

****

TERMWORK -6)) study the impact of deleting rows, and droping table on foreign key.

SQL> delete from employee where empno='102';

delete from employee where empno='102'

*

ERROR at line 1:

ORA-02292: integrity constraint (SYSTEM.SYS_C008341) violated - child record found

---

droping table for foreign key.

```
SQL> drop table Employee;

drop table Employee
         *
ERROR at line 1:
ORA-02449: unique/primary keys in table referenced by foreign keys
```

****

TERMWORK -7) display name and dob of all employees who are on bench(not working).

```
SQL>  SELECT E.empname, E.dob
        FROM Employee E
        LEFT JOIN Assigned_to A ON E.empno = A.empno
        WHERE A.projectno IS NULL;
```

Output: - EMPNAME DOB

Raj 01-JAN-12

****

TERMWORK -8)) display names of all employees working on all projects

```
SQL>
   SELECT E.empname
   FROM Employee E
   JOIN Assigned_to A ON E.empno = A.empno
   GROUP BY E.empno, E.empname
   HAVING COUNT(DISTINCT A.projectno) = (SELECT COUNT(*) FROM Project);
```

Output:- EMPNAME

Rani

****

TERMWORK -9)) display the names of all employees working on atleast all of the project that employee 101 is working.

SQL>

```
SELECT E.empname
FROM Employee E
WHERE NOT EXISTS (
   SELECT A.projectno
   FROM Assigned_to A
   WHERE A.empno = 101
   AND A.projectno NOT IN (
      SELECT A2.projectno
      FROM Assigned_to A2
      WHERE A2.empno = E.empno )
);
```

Output:- EMPNAME

John

Rani

****

TERMWORK -10)) display details of top 3 senior employees

```
SQL>
   SELECT *
   FROM Employee
   ORDER BY dob ASC
   FETCH FIRST 3 ROWS ONLY;
```

Output:-

EMPNO	EMPNAME	S	PHONE	DOB
101	John	m	9923456745	23-APR-04
102	Smith	m	9923456756	25-APR-05
104	Rani	f	9923386745	13-JAN-09

****

TERMWORK -11)) find for each employee the penalty incurred

```
SQL>    SELECT E.empno, E.empname, F.ammount, F.f_date
        FROM Employee E
```

```
        LEFT JOIN Fine F ON E.empno = F.empno;
```

Output:

EMPNO	EMPNAME	AMMOUNT	F_DATE
102	Smith	2000	24-FEB-19
104	Rani	6420	21-SEP-16
101	John	7500	09-DEC-03
105	Raj		
103	Pinky		

****

TERMWORK -12)) display the number of employees working under each project having count > 3

```
SQL>
  SELECT P.projectno, P.pname, COUNT(A.empno) AS EmployeeCount
  FROM Project P
  JOIN Assigned_to A ON P.projectno = A.projectno
  GROUP BY P.projectno, P.pname
  HAVING COUNT(A.empno) > 3;
```

Output:-

PROJECTNO	PNAME	EMPLOYEECOUNT
2	python	4

****

TERMWORK -13) a) study of ALTER statement

```
SQL>  ALTER table Employee ADD address varchar(40);
```

Output:-

EMPNO	EMPNAME	S	PHONE	DOB	ADDRESS
1	manish	m	9916236624	23-APR-05	
2	harish	m	9916236626	24-JUL-06	
3	ambika	f	9886507658	27-OCT-04	
4	deepa	f	9916236620	25-JUL-08	
8	riya	f	123456789	19-SEP-09	

---

b) study of ORDER BY CLAUSE

```
SQL>
        select  e.*
        from Employee e
        order by e.dob
        fetch first 3 rows only;
```

****

TERMWORK -14) study of ststistical function

a))

```
sql> select max(ammount) from fine;

      MAX(AMMOUNT)
       ----------
          35000
```

```
sql> select min(ammount) from fine;

       MIN(AMMOUNT)
        ----------
           15000
```

```
b))  sql>  select sum(ammount) from fine;

        SUM(AMMOUNT)
         ----------
             80000
```

```
sql>  select avg(ammount) from fine;

          AVG(AMMOUNT)
           ----------
           26666.6667
```

****

TERMWORK -15) study of

a) between

```
sql>  select  empno from fine  where ammount  between 10000 and 400000;

    EMPNO
   ----------
       2
       3
       1
```

---

b) like

```
sql>  select empno from Employee where empname LIKE 'd%';

            EMPNO
            ----------
                4
```

---

c) all

```
sql>    select count(*) from Employee;

       COUNT(*)
        ----------
           5
```

```
sql>  select DISTINCT(ammount) from fine;

  AMMOUNT
  -----------
   30000
   15000
   35000
```

```
sql> select empname from Employee where empno IN(1,2);

EMPNAME
---------
manish
harish
```

```
sql> select VARIANCE(ammount) from fine;

VARIANCE(AMMOUNT)

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

      108333333

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

sql> select  STDDEV(ammount) from fine;

STDDEV(AMMOUNT)

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

     10408.33
```

****

TERMWORK - 16)) Study at DATE related functions

```
sql>  select * from Employee where Employee.dob <= all(select dob from Employee);


   EMPNO EMPNAME        S     PHONE           DOB       ADDRESS

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

       3       ambika            f     9886507658  27-OCT-04
```

```
sql> select e.empname,to_char(dob,'yyyy')

       from Employee e;

EMPNAME        TO_C

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

manish             2005

harish             2006

ambika             2004

deepa              2008

riya               2009
```

```
sql> select * from Employee where Employee.dob >= all(select dob from Employee);

   EMPNO EMPNAME      S     PHONE      DOB       ADDRESS

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

      8        riya              f    123456789   19-SEP-09
```

```
sql> select * from Employee e
       where e.dob < any (select dob
                          from Employee);
    EMPNO EMPNAME        S     PHONE        DOB   ADDRESS
       ---------- ---------- - ---------- -----------
       3       ambika          f     9886507658   27-OCT-04
       1       manish          m     9916236624   23-APR-05
       2       harish          m     9916236626    24-JUL-06
       4       deepa           f     9916236620   25-JUL-08
sql>  select to_char(dob, 'SYEAR')
        FROM Employee;
TO_CHAR(DOB,'SYEAR')
 TWO THOUSAND FIVE
 TWO THOUSAND SIX
 TWO THOUSAND FOUR
 TWO THOUSAND EIGHT
 TWO THOUSAND NINE
sql>   select to_char(dob, 'yyyy mm  dd hh  mi ss')
         from Employee;
TO_CHAR(DOB,'YYYYMMDD
2005   04  23   12   00  00
2006   07  24   12   00  00
2004   10  27   12   00  00
2008   07  25   12   00  00
2009   09  19   12   00  00
sql>  select TO_CHAR (dob,'MONTH')
       from Employee;
```

```
TO_CHAR(D
---------
APRIL
JULY
OCTOBER
JULY
SEPTEMBER
```

```
SQL>  select TO_CHAR (dob,'W')
         from  Employee;
T
-
4
4
4
4
3
```

```
sql>  select TO_CHAR (dob,'WW')
         from  Employee;
TO
--
17
30
43
30
38
```

```
SQL> SELECT TO_CHAR (dob, 'DAY') from Employee;
TO_CHAR(D
---------
SATURDAY
```

MONDAY

WEDNESDAY

FRIDAY

SATURDAY

---

sql> select empname, TO_CHAR(dob,'DAY MONTH W') FROM Employee;

EMPNAME TO_CHAR(DOB,'DAY MONTH

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

manish SATURDAY APRIL 4

harish MONDAY JULY 4

ambika WEDNESDAY OCTOBER 4

deepa FRIDAY JULY 4

riya SATURDAY SEPTEMBER 3

---

sql> select TO_CHAR (dob,'YY')

from Employee;

TO

--

05

06

04

08

09

SQL> select TO_CHAR (dob,'YYY')

from Employee;

TO_

005

006

004

008

009

SQL> select TO_CHAR (dob,'Y,YYY')

from Employee;

TO_CH

2,005

2,006

2,004

2,008

2,009

---

sql> select TO_CHAR (dob,'SYYYY')

from Employee;

TO_CH

2005

2006

2004

2008

2009

****

TERMWORK -17)) Study of views(with and without check option).

SQL>

```
create view Employee_1 as
select empname,dob
from Employee
where dob>='1-jan-68';
```

output:-

EMPNAME DOB

```
    John        23-APR-04
    Smith       25-APR-05
    Pinky       15-JUN-20
    Rani        13-JAN-09
    Raj         13-AUG-70
**********************************************************************************************
SQL>
  create view Employee_3 as
  select empname,dob,sex,salary
  from Employee
  where salary>='35000'
  with check option;
output:-
View created.
SQL> select * from Employee_3;
EMPNAME      DOB       S    SALARY
---------------------------------------
  Pinky       15-JUN-20    f    50000
**********************************************************************************************
SQL>
   update Employee_3
   set salary =70000
   where empname='Pinky';
output:-
1 row updated.
SQL> select * from Employee_3;
EMPNAME      DOB       S    SALARY
---------------------------------------
  Pinky       15-JUN-20   f    70000
**********************************************************************************************
SQL>
   DROP view Employee_2;
```

output:- View dropped.

SQL> select * from Employee_2;

select * from Employee_2

*

ERROR at line 1:

ORA-00942: table or view does not exist

****

TERMWORK -18:-a)) Study of coping the tables.

SQL>

CREATE TABLE Employee_Backup AS select * from Employee;

output:-

Table created.

SQL> select * from Employee_backup;

Output:- EMPNO EMPNAME S PHONE DOB SALARY

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

101 John m 9923456745 23-APR-04

102 Smith m 9923456756 25-APR-05

103 Pinky f 9354456745 15-JUN-20 70000

104 Rani f 9923386745 13-JAN-09

105 Raj m 9923388745 13-AUG-70

****

TERMWORK - 19) Study of PL/SQL features.

SQL>

```
declare
  projectcount integer;
begin
insert into Project values(6,'software_engg');
select count(projectno) into projectcount from Project;
dbms_output.put_line('NO OF PROJECTS=' || projectcount);
end;
```

```
output:- NO OF PROJECTS=6

PL/SQL procedure successfully completed.

**********************************************************************************************

SQL>

declare

   projectcount integer;

   status char(28);

begin

select count(projectno) into projectcount from Project;

if projectcount >2 then

  status:='very few projects';

else

  status:='safficient projects';

END if;

dbms_output.put_line('NO OF PROJECTS=' || projectcount || 'STATUS=' || status);

end;

ouput:-    NO OF PROJECTS=6   STATUS=very few projects

PL/SQL procedure successfully completed.

**********************************************************************************************

SQL>

declare

   projectcount integer;

   PName char(20);

   rowno integer;

begin

rowno:=1;

select count(projectno) into projectcount from Project;

  while rowno<=4 loop

          select pname into PName from Project where projectno=rowno;

  dbms_output.put_line('project name=' || PName);

  rowno:= rowno+1;

  END loop;
```

```
end;
```

output:- project name=hakerrank

project name=python

project name=dbms

project name=research

PL/SQL procedure successfully completed.

****