**11. Write a PL/SQL Code using Cursors, Exceptions and Composite Data Types**

**CURSORS :**

**i)write an explicit cursor using simple loop.**

**Program:**

SQL> set serveroutput on;

SQL> declare

2 cursor sail\_cur is

3 select sid,sname from sailors;

4 ab sail\_cur%rowtype;

5 begin

6 open sail\_cur;

7 loop

8 fetch sail\_cur into ab;

9 exit when sail\_cur%NOTFOUND;

10 dbms\_output.put\_line(ab.sid||' '||ab.sname);

11 end loop;

12 close sail\_cur;

13 end;

14 /

**Output:**

421 leela

22 dustin

29 brutus

31 lubber

32 andy

64 horatio

71 zobra

85 art

74 ravi

95 bob

58 rusty

PL/SQL procedure successfully completed.

**ii)write an explicit cursor using while loop**

**Program:**

SQL> declare

2 cursor sail\_cur is

3 select sid,sname from sailors;

4 ab sail\_cur%rowtype;

5 begin

6 open sail\_cur;

7 fetch sail\_cur into ab;

8 while sail\_cur%FOUND

9 loop

10 dbms\_output.put\_line(ab.sid||' '||ab.sname);

11 fetch sail\_cur into ab;

12 end loop;

13 close sail\_cur;

14 end;

15 /

**Output:**

421 leela

22 dustin

29 brutus

31 lubber

32 andy

64 horatio

71 zobra

85 art

74 ravi

95 bob

58 rusty

PL/SQL procedure successfully completed.

**iii) write an explicit cursor using for loop.**

**Program:**

SQL> declare

2 cursor sail\_cur is

3 select sid,sname from sailors;

4 ab sail\_cur%rowtype;

5 begin

6 for ab in sail\_cur

7 loop

8 dbms\_output.put\_line(ab.sid||' '||ab.sname);

9 end loop;

10 end;

11 /

**Output:**

421 leela

22 dustin

29 brutus

31 lubber

32 andy

64 horatio

71 zobra

85 art

74 ravi

95 bob

58 rusty

PL/SQL procedure successfully completed.

**EXCEPTIONS:**

**i)PL/SQL program to print divide by zero exception.**

**WITHOUT EXCEPTION:**

SQL> set serveroutput on;

SQL> declare

2 id number := 12;

3 BEGIN

4 id:=12/0;

5 end;

6 /

declare

\*

ERROR at line 1:

ORA-01476: divisor is equal to zero

ORA-06512: at line 4

**WITH EXCEPTION:**

SQL> declare

2 id number:=12;

3 BEGIN

4 id:=12/0;

5 exception

6 when zero\_divide then

7 dbms\_output.put\_line('Divide by zero');

8 end;

9 /

Divide by zero

PL/SQL procedure successfully completed.

**ii) PL/SQl program to print value error exception.**

**WITHOUT EXCEPTION:**

SQL> declare

2 num number:=&num;

3 BEGIN

4 dbms\_output.put\_line('Square root of '||num||' is '||sqrt(num));

5 end;

6 /

Enter value for num: -25

old 2: num number:=&num;

new 2: num number:=-25;

declare

\*

ERROR at line 1:

ORA-06502: PL/SQL: numeric or value error

ORA-06512: at line 4

**WITH EXCEPTION:**

SQL> declare

2 num number:=&num;

3 BEGIN

4 dbms\_output.put\_line('Square root of '||num||' is'||sqrt(num));

5 exception

6 when value\_error then

7 dbms\_output.put\_line('Value error');

8 end;

9 /

Enter value for num: -25

old 2: num number:=&num;

new 2: num number:=-25;

Value error

PL/SQL procedure successfully completed.

**iii) Write a PL/SQL program to print unique constraint.**

**WITHOUT EXCEPTION:**

SQL> declare

2 roll\_no int:=502;

3 sname varchar2(20):='Alekhya';

4 marks number:=99;

5 begin

6 insert into student values(roll\_no,sname,marks);

7 end;

8 /

declare

\*

ERROR at line 1:

ORA-00001: unique constraint (CSE205B0.SYS\_C005394) violated

ORA-06512: at line 6

**WITH EXCEPTION :**

SQL> declare

2 roll\_no int:=502;

3 sname varchar2(20):='Alekhya';

4 marks number:=99;

5 begin

6 insert into student values(roll\_no,sname,marks);

7 exception

8 when dup\_val\_on\_index then

9 dbms\_output.put\_line('Unique Constraint violated');

10 end;

11 /

Unique Constraint violated

PL/SQL procedure successfully completed.

**iv) PL/SQL program to print no data found exception.**

**WITHOUT EXCEPTION:**

SQL> declare

2 id int:=&id;

3 name varchar2(20);

4 begin

5 select sname into name from student where roll\_no=id;

6 dbms\_output.put\_line(name);

7 end;

8 /

Enter value for id: 503

old 2: id int:=&id;

new 2: id int:=503;

declare

\*

ERROR at line 1:

ORA-01403: no data found

ORA-06512: at line 5

**WITH EXCEPTION:**

SQL> declare

2 id int:=&id;

3 name varchar2(20);

4 begin

5 select sname into name from student where roll\_no=id;

6 dbms\_output.put\_line(name);

7 exception

8 when no\_data\_found then

9 dbms\_output.put\_line('No data found');

10 end;

11 /

Enter value for id: 503

old 2: id int:=&id;

new 2: id int:=503;

No data found

PL/SQL procedure successfully completed.

**v) Write a PL/SQL program to print case not found exception**

**WITHOUT EXCEPTION:**

SQL> declare

2 grade char(1):='&grade';

3 begin

4 case grade

5 when 'A' then dbms\_output.put\_line('Excellent');

6 when 'B' then dbms\_output.put\_line('Good');

7 when 'C' then dbms\_output.put\_line('Average');

8 when 'D' then dbms\_output.put\_line('Poor');

9 end case;

10 end;

11 /

Enter value for grade: K

old 2: grade char(1):='&grade';

new 2: grade char(1):='K';

declare

\*

ERROR at line 1:

ORA-06592: CASE not found while executing CASE statement

ORA-06512: at line 4

**WITH EXCEPTION:**

SQL> declare

2 grade char(1):='&grade';

3 begin

4 case grade

5 when 'A' then dbms\_output.put\_line('Excellent');

6 when 'B' then dbms\_output.put\_line('Good');

7 when 'C' then dbms\_output.put\_line('Average');

8 when 'D' then dbms\_output.put\_line('Poor');

9 end case;

10 exception

11 when case\_not\_found then

12 dbms\_output.put\_line('Case not found');

13 end;

14 /

Enter value for grade: J

old 2: grade char(1):='&grade';

new 2: grade char(1):='J';

Case not found

PL/SQL procedure successfully completed.

**COMPOSITE DATA TYPES:**

1. **Write a PL/SQL program to print the student name and marks using table type.**

**Program :**

SQL> declare

2 type namet is table of varchar2(20);

3 type grades is table of integer;

4 names namet;

5 marks grades;

6 total integer;

7 BEGIN

8 names:=namet('Shah','Mike','Maddi','Alex','Peter');

9 marks:=grades(92,87,98,97,78);

10 total:=names.count;

11 dbms\_output.put\_line('Total '||total||' Students');

12 for i in 1..total loop

13 dbms\_output.put\_line('Student: '||names(i)||' marks : '||marks(i));

14 end loop;

15 end;

16 /

**Output:**

Total 5 Students

Student: Shah marks : 92

Student: Mike marks : 87

Student: Maddi marks : 98

Student: Alex marks : 97

Student: Peter marks : 78

PL/SQL procedure successfully completed.

1. **Write a PL/SQL program to print student id and student name by using RECORD TYPE**

**Program:**

SQL> declare

2 type t\_name IS RECORD

3 (

4 sname student.sname%TYPE,roll\_no student.roll\_no%TYPE);

5 r\_name t\_name; --name record

6 n\_emp\_id student.roll\_no%TYPE:=502;

7 BEGIN

8 select sname,roll\_no INTO r\_name FROM student WHERE roll\_no=n\_emp\_id;

9 dbms\_output.put\_line(r\_name.sname||' , '||r\_name.roll\_no);

10 end;

11 /

**Output:**

Felix , 502

PL/SQL procedure successfully completed.

1. **Write a PL/SQL program to print student names and marks using VARRAYTYPE.**

**Program:**

SQL> declare

2 type namesarray IS VARRAY(5) OF varchar2(10);

3 type grades is VARRAY(5) OF integer;

4 names namesarray;

5 marks grades;

6 total integer;

7 BEGIN

8 names:=namesarray('Shah','Mike','Maddi','Alex','Peter');

9 marks:=grades(92,87,98,97,78);

10 total:=names.count;

11 dbms\_output.put\_line('Total'||total||'Students');

12 FOR i in 1..total LOOP

13 dbms\_output.put\_line('Student:'||names(i)||'marks:'||marks(i));

14 end loop;

15 end;

16 /

**Output:**

Total5Students

Student:Shahmarks:92

Student:Mikemarks:87

Student:Maddimarks:98

Student:Alexmarks:97

Student:Petermarks:78

PL/SQL procedure successfully completed.

**12. Write a PL/SQL Code using Procedures, Functions, Packages.**

**PROCEDURE:**

**i)Write a stored procedure to print hello message.**

SQL> create procedure greet

2 as

3 begin

4 dbms\_output.put\_line('Hello');

5 end;

6 /

Procedure created.

SQL> begin

2 greet;

3 end;

4 /

Hello

PL/SQL procedure successfully completed.

SQL> exec greet;

Hello

PL/SQL procedure successfully completed.

**ii)Write a stored procedure to find addition of two numbers.**

**Program:**

SQL> create procedure sum\_c(a in number,b in number, c out number)

2 as

3 begin

4 c:=a+b;

5 end;

6 /

Procedure created.

SQL> declare

2 d number;

3 begin

4 sum\_c(2,7,d);

5 dbms\_output.put\_line(d);

6 end;

7 /

9

PL/SQL procedure successfully completed.

**FUNCTION:**

1. **Write a stored function to find square of a given number.**

SQL> create or replace function sq(x in number)

2 return number

3 as

4 begin

5 return(x\*x);

6 end;

7 /

Function created.

SQL> begin

2 dbms\_output.put\_line(sq(7));

3 end;

4 /

49

PL/SQL procedure successfully completed.

SQL> select sq(7) from dual;

SQ(7)

----------

49

1. **Write a stored function to find addition of two numbers.**

SQL> create or replace function add\_c(a in number,b in number)

2 return number

3 as

4 c number;

5 begin

6 c:=a+b;

7 return c;

8 end;

9 /

Function created.

SQL> declare

2 d number;

3 begin

4 d:=add\_c(10,20);

5 dbms\_output.put\_line(d);

6 end;

7 /

30

PL/SQL procedure successfully completed.

SQL> select add\_c(20,30) from dual;

ADD\_C(20,30)

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

50

**PACKAGES:**

1. **Write a package to print hello message.**

SQL> create package ss1

2 as procedure greet;

3 end;

4 /

Package created.

SQL> create package body ss1

2 as procedure greet as

3 begin

4 dbms\_output.put\_line('Hello');

5 end;

6 end;

7 /

Package body created.

SQL> exec ss1.greet;

Hello

PL/SQL procedure successfully completed.

1. **Write a package that prints student name by passing student id.**

SQL> create or replace package pk as

2 function fun1(no in number)

3 return varchar2;

4 end;

5 /

Package created.

SQL> create or replace package body pk

2 is

3 function fun1(no in number) return varchar2

4 is

5 name varchar2(20);

6 begin

7 select sname into name from student where roll\_no = no;

8 return name;

9 end;

10 end;

11 /

Package body created.

SQL> select pk.fun1(501) from dual;

PK.FUN1(501)

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

Jungkook

**15. For a Faculty Database**

**EMPLOYEE (EMPID, FName, L Name, Address, Sex, Salary, Dept No)**

**DEPARTMENT (Dept No, D Name, HOD\_EMPID)**

**PROJECT (Proj No, P Name, Dept No)**

**WORKS\_ON (EMPID, Proj No, Hours)**

**Write SQL queries to**

**a. To Show the resulting salaries if every employee working on the ‘IoT’ project is given a 10 percent raise.**

**b. Find the sum of the salaries of all employees of the ‘IT’ department, as well as the maximum salary, the minimum salary, and the average salary in this department.**

**Program:**

SQL> create table department

2 (

3 dept\_no integer,

4 dname varchar2(20),

5 hod\_empid integer,

6 primary key(dept\_no)

7 );

Table created.

SQL> create table employeee

2 (

3 empid integer,

4 fname varchar2(20),

5 lname varchar2(20),

6 address varchar2(30),

7 sex char(1),

8 dept\_no integer,

9 salary integer,

10 primary key(empid),

11 foreign key(dept\_no) references department(dept\_no)

12 );

Table created.

SQL> create table projectt

2 (

3 proj\_no integer,

4 pname varchar2(20),

5 dept\_no integer,

6 primary key(proj\_no),

7 foreign key(proj\_no) references department(dept\_no)

8 );

Table created.

SQL> create table works\_on

2 (

3 empid integer,

4 proj\_no integer,

5 hours integer,

6 foreign key(empid) references employeee(empid),

7 foreign key(proj\_no) references projectt (proj\_no)

8 );

Table created.

**INSERTION INTO DEPARTMENT**

SQL> insert into department values(1,&#39;CSE&#39;,1240);

1 row created.

SQL> insert into department values(2,&#39;IT&#39;,1245);

1 row created.

SQL> select \* from department;

DEPT\_NO DNAME HOD\_EMPID

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

1 CSE 1240

2 IT 1245

**INSERTION INTO EMPLOYEEE**

SQL> insert into employeee values(&empid,'&fname','&lname','&address','&sex',&dept\_no,&salary);

Enter value for empid: 1201

Enter value for fname: Adarsh

Enter value for lname: Kumar

Enter value for address: Kakinada

Enter value for sex: F

Enter value for dept\_no: 1

Enter value for salary: 150000

old 1: insert into employeee values(&empid,'&fname','&lname','&address','&sex',&dept\_no,&salary)

new 1: insert into employeee values(1201,'Adarsh','Kumar','Kakinada','F',1,150000)

1 row created.

SQL> /

Enter value for empid: 1240

Enter value for fname: Mahi

Enter value for lname: John

Enter value for address: Rajahmundry

Enter value for sex: F

Enter value for dept\_no: 1

Enter value for salary: 95000

old 1: insert into employeee values(&empid,'&fname','&lname','&address','&sex',&dept\_no,&salary)

new 1: insert into employeee values(1240,'Mahi','John','Rajahmundry','F',1,95000)

1 row created.

SQL> /

Enter value for empid: 1245

Enter value for fname: Ramu

Enter value for lname: Murty

Enter value for address: Rajahmundry

Enter value for sex: M

Enter value for dept\_no: 2

Enter value for salary: 90000

old 1: insert into employeee values(&empid,'&fname','&lname','&address','&sex',&dept\_no,&salary)

new 1: insert into employeee values(1245,'Ramu','Murty','Rajahmundry','M',2,90000)

1 row created.

SQL> /

Enter value for empid: 1234

Enter value for fname: Aditya

Enter value for lname: Surya

Enter value for address: Banglore

Enter value for sex: M

Enter value for dept\_no: 1

Enter value for salary: 80000

old 1: insert into employeee values(&empid,'&fname','&lname','&address','&sex',&dept\_no,&salary)

new 1: insert into employeee values(1234,'Aditya','Surya','Banglore','M',1,80000)

1 row created.

SQL> /

Enter value for empid: 1247

Enter value for fname: Jack

Enter value for lname: Paul

Enter value for address: Banglore

Enter value for sex: M

Enter value for dept\_no: 2

Enter value for salary: 75000

old 1: insert into employeee values(&empid,'&fname','&lname','&address','&sex',&dept\_no,&salary)

new 1: insert into employeee values(1247,'Jack','Paul','Banglore','M',2,75000)

1 row created.

SQL> /

Enter value for empid: 1235

Enter value for fname: Pradeep

Enter value for lname: Chitra

Enter value for address: Rajahmundry

Enter value for sex: M

Enter value for dept\_no: 1

Enter value for salary: 78000

old 1: insert into employeee values(&empid,'&fname','&lname','&address','&sex',&dept\_no,&salary)

new 1: insert into employeee values(1235,'Pradeep ','Chitra','Rajahmundry','M',1,78000)

1 row created.

SQL> /

Enter value for empid: 1211

Enter value for fname: Srinivas

Enter value for lname: Kumar

Enter value for address: Hyderabad

Enter value for sex: M

Enter value for dept\_no: 1

Enter value for salary: 59000

old 1: insert into employeee values(&empid,'&fname','&lname','&address','&sex',&dept\_no,&salary)

new 1: insert into employeee values(1211,'Srinivas','Kumar','Hyderabad','M',1,59000)

1 row created.

SQL> /

Enter value for empid: 1492

Enter value for fname: Gopala

Enter value for lname: Rao

Enter value for address: Kakinada

Enter value for sex: M

Enter value for dept\_no: 2

Enter value for salary: 65000

old 1: insert into employeee values(&empid,'&fname','&lname','&address','&sex',&dept\_no,&salary)

new 1: insert into employeee values(1492,'Gopala','Rao','Kakinada','M',2,65000)

1 row created.

SQL> /

Enter value for empid: 1205

Enter value for fname: Eswari

Enter value for lname: Nirupama

Enter value for address: Kakinada

Enter value for sex: F

Enter value for dept\_no: 2

Enter value for salary: 65000

old 1: insert into employeee values(&empid,'&fname','&lname','&address','&sex',&dept\_no,&salary)

new 1: insert into employeee values(1205,'Eswari','Nirupama','Kakinada','F',2,65000)

1 row created.

SQL> select \* from employeee;

EMPID FNAME LNAME

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

ADDRESS S DEPT\_NO SALARY

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

1201 Adarsh Kumar

Kakinada F 1 150000

1240 Mahi John

Rajahmundry F 1 95000

1245 Ramu Murty

Rajahmundry M 2 90000

EMPID FNAME LNAME

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

ADDRESS S DEPT\_NO SALARY

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

1234 Aditya Surya

Banglore M 1 80000

1247 Jack Paul

Banglore M 2 75000

1235 Pradeep Chitra

Rajahmundry M 1 78000

EMPID FNAME LNAME

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

ADDRESS S DEPT\_NO SALARY

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

1211 Srinivas Kumar

Hyderabad M 1 59000

1492 Gopala Rao

Kakinada M 2 65000

1250 Eswari Nirupama

Kakinada F 2 65000

9 rows selected.

**INSERTION INTO PROJECTT**

SQL> insert into projectt values(&projno,'&pname',&dept\_no);

Enter value for projno: 100

Enter value for pname: IOT

Enter value for dept\_no: 1

old 1: insert into projectt values(&projno,'&pname',&dept\_no)

new 1: insert into projectt values(100,'IOT',1)

1 row created.

SQL> /

Enter value for projno: 101

Enter value for pname: CLOUD

Enter value for dept\_no: 1

old 1: insert into projectt values(&projno,'&pname',&dept\_no)

new 1: insert into projectt values(101,'CLOUD',1)

1 row created.

SQL> /

Enter value for projno: 102

Enter value for pname: BIGDATA

Enter value for dept\_no: 2

old 1: insert into projectt values(&projno,'&pname',&dept\_no)

new 1: insert into projectt values(103,'BIGDATA',2)

1 row created.

SQL> /

Enter value for projno: 103

Enter value for pname: NETWORKS

Enter value for dept\_no: 2

old 1: insert into projectt values(&projno,'&pname',&dept\_no)

new 1: insert into projectt values(103,'NETWORKS',2)

1 row created.

SQL> /

Enter value for projno: 104

Enter value for pname: IOT

Enter value for dept\_no: 2

old 1: insert into projectt values(&projno,'&pname',&dept\_no)

new 1: insert into projectt values(104,'IOT',2)

1 row created.

SQL> /

Enter value for projno: 105

Enter value for pname: NETWORKS

Enter value for dept\_no: 1

old 1: insert into projectt values(&projno,'&pname',&dept\_no)

new 1: insert into projectt values(105,'NETWORKS',1)

1 row created.

SQL> select \* from projectt;

PROJNO PNAME DEPT\_NO

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

100 IOT 1

101 CLOUD 1

102 BIGDATA 2

103 NETWORKS 2

104 IOT 2

105 NETWORKS 1

6 rows selected.

**INSERTION INTO WORKS\_ON**

SQL> insert into works\_on values(&empid,&projno,&hours);

Enter value for empid: 1245

Enter value for projno: 104

Enter value for hours: 16

old 1: insert into works\_on values(&empid,&projno,&hours)

new 1: insert into works\_on values(1245,104,16)

1 row created.

SQL> /

Enter value for empid: 1240

Enter value for projno: 101

Enter value for hours: 22

old 1: insert into works\_on values(&empid,&projno,&hours)

new 1: insert into works\_on values(1240,101,22)

1 row created.

SQL> /

Enter value for empid: 1201

Enter value for projno: 100

Enter value for hours: 31

old 1: insert into works\_on values(&empid,&projno,&hours)

new 1: insert into works\_on values(1201,100,31)

1 row created.

SQL> /

SQL> /

Enter value for empid: 1250

Enter value for projno: 102

Enter value for hours: 25

old 1: insert into works\_on values(&empid,&projno,&hours)

new 1: insert into works\_on values(1250,102,25)

1 row created.

SQL> /

Enter value for empid: 1492

Enter value for projno: 103

Enter value for hours: 25

old 1: insert into works\_on values(&empid,&projno,&hours)

new 1: insert into works\_on values(1492,103,25)

1 row created.

SQL> /

Enter value for empid: 1235

Enter value for projno: 105

Enter value for hours: 29

old 1: insert into works\_on values(&empid,&projno,&hours)

new 1: insert into works\_on values(1235,105,29)

1 row created.

SQL> select \* from works\_on;

EMPID PROJNO HOURS

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

1245 104 16

1240 101 22

1201 100 31

1492 103 25

1235 105 29

1250 102 25

6 rows selected.

1. **To Show the resulting salaries if every employee working on the ‘IoT’ project is given a 10 percent raise.**

SQL> select e.empid,(e.salary+(e.salary\*0.1)) as salraise from employeee e,projectt p,work\_on w where p.pname='IOT' and w.projno=p.projno and e.empid=w.empid;

EMPID SALRAISE

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

1245 99000

1201 165000

1. **Find the sum of the salaries of all employees of the ‘IT’ department, as well as the maximum salary, the minimum salary, and the average salary in this department.**

SQL> select sum(e.salary) from employeee e,department d where e.dept\_no=d.dept\_no and d.dname='IT';

SUM(E.SALARY)

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

295000

SQL> select max(e.salary) from employeee e ,department d where e.dept\_no=d.dept\_no and d.dname='IT';

MAX(E.SALARY)

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

90000

SQL> select min(e.salary) from employeee e ,department d where e.dept\_no=d.dept\_no and d.dname='IT';

MIN(E.SALARY)

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

65000

SQL> select avg(e.salary) from employeee e ,department d where e.dept\_no=d.dept\_no and d.dname='IT';

AVG(E.SALARY)

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

73750

**16. For a Movie Database:**

**ACTOR (Act\_id, Act\_Name, Act\_Gender)**

**DIRECTOR (Dir\_id, Dir\_Name)**

**MOVIES (Mov\_id, Mov\_Title, Mov\_Year, Dir\_id)**

**MOVIE\_CAST (Act\_id, Mov\_id, Role)**

**RATING (Mov\_id, Rev\_Stars)**

**Write SQL queries to**

**1. List the titles of all movies directed by ‘STEVEN SPIELBERG’.**

**2. Find the movie names where one or more actors acted in two or more movies.**

**3. List all actors who acted in a movie before 2015 and also in a movie after 2015 (use JOIN operation).**

**4. Find the title of movies and number of stars for each movie that has at least one rating and find the highest number of stars that movie received. Sort the result by movie title.**

**TABLES CREATION:**

SQL> create table actor

2 (

3 act\_id integer,

4 act\_name varchar2(20),

5 act\_gender char(1),

6 primary key(act\_id)

7 );

Table created.

SQL> create table director

2 (

3 did integer,

4 dname varchar2(30),

5 primary key(did)

6 );

Table created.

SQL> create table movie

2 (

3 mid integer,

4 mtitle varchar2(20),

5 myear integer,

6 did integer,

7 primary key(mid),

8 foreign key(did) references director(did)

9 );

Table created.

SQL> create table movie\_cast

2 (

3 act\_id integer,

4 mid integer,

5 role varchar2(20),

6 primary key(act\_id,mid),

7 foreign key(act\_id) references actor(act\_id),

8 foreign key(mid) references movie(mid)

9 );

Table created.

SQL> create table rating

2 (

3 mid integer,

4 rev\_stars integer,

5 foreign key(mid) references movie(mid)

6 );

Table created.

**INSERTION INTO ACTOR**

SQL> insert into actor values(&act\_id,'&act\_name','&act\_gender');

Enter value for act\_id: 101

Enter value for act\_name: DICAPRIO

Enter value for act\_gender: M

old 1: insert into actor values(&act\_id,'&act\_name','&act\_gender')

new 1: insert into actor values(101,'DICAPRIO','M')

1 row created.

SQL> /

Enter value for act\_id: 102

Enter value for act\_name: KATE WINSLET

Enter value for act\_gender: F

old 1: insert into actor values(&act\_id,'&act\_name','&act\_gender')

new 1: insert into actor values(102,'KATE WINSLET ','F')

1 row created.

SQL> /

Enter value for act\_id: 103

Enter value for act\_name: SAM WORTHINGTON

Enter value for act\_gender: M

old 1: insert into actor values(&act\_id,'&act\_name','&act\_gender')

new 1: insert into actor values(103,'SAM WORTHINGTON','M')

1 row created.

SQL> /

Enter value for act\_id: 104

Enter value for act\_name: SAM NEIL

Enter value for act\_gender: M

old 1: insert into actor values(&act\_id,'&act\_name','&act\_gender')

new 1: insert into actor values(104,'SAM NEIL ','M')

1 row created.

SQL> /

Enter value for act\_id: 105

Enter value for act\_name: CATE BLANCHETT

Enter value for act\_gender: F

old 1: insert into actor values(&act\_id,'&act\_name','&act\_gender')

new 1: insert into actor values(105,'CATE BLANCHETT','F')

1 row created.

SQL> /

Enter value for act\_id: 106

Enter value for act\_name: CHRIS PRATT

Enter value for act\_gender: M

old 1: insert into actor values(&act\_id,'&act\_name','&act\_gender')

new 1: insert into actor values(106,'CHRIS PRATT','M')

1 row created.

SQL> /

Enter value for act\_id: 107

Enter value for act\_name: BRYCE DALLAS

Enter value for act\_gender: F

old 1: insert into actor values(&act\_id,'&act\_name','&act\_gender')

new 1: insert into actor values(107,'BRYCE DALLAS ','F')

1 row created.

SQL> /

Enter value for act\_id: 108

Enter value for act\_name: LAURA DERN

Enter value for act\_gender: F

old 1: insert into actor values(&act\_id,'&act\_name','&act\_gender')

new 1: insert into actor values(108,'LAURA DERN','F')

1 row created.

SQL> /

Enter value for act\_id: 109

Enter value for act\_name: DANIEL YORK

Enter value for act\_gender: F

old 1: insert into actor values(&act\_id,'&act\_name','&act\_gender')

new 1: insert into actor values(109,'DANIEL YORK','F')

1 row created.

SQL> select \* from actor;

ACT\_ID ACT\_NAME A

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

101 DICAPRIO M

102 KATE WINSLET F

103 SAM WORTHINGTON M

104 SAM NEIL M

105 CATE BLANCHETT F

106 CHRIS PRATT M

107 BRYCE DALLAS F

108 LAURA DERN F

109 DANIEL YORK F

9 rows selected.

**INSERTION INTO DIRECTOR**

SQL> insert into director values(&did,'&dname');

Enter value for did: 10

Enter value for dname: STEVEN SPIELBERG

old 1: insert into director values(&did,'&dname')

new 1: insert into director values(10,'STEVEN SPIELBERG')

1 row created.

SQL> /

Enter value for did: 11

Enter value for dname: JAMES CAMERON

old 1: insert into director values(&did,'&dname')

new 1: insert into director values(11,'JAMES CAMERON')

1 row created.

SQL> /

Enter value for did: 12

Enter value for dname: MARTIN SCORSESE

old 1: insert into director values(&did,'&dname')

new 1: insert into director values(12,'MARTIN SCORSESE')

1 row created.

SQL> /

Enter value for did: 13

Enter value for dname: BAZ LUHRMANN

old 1: insert into director values(&did,'&dname')

new 1: insert into director values(13,'BAZ LUHRMANN')

1 row created.

SQL> /

Enter value for did: 14

Enter value for dname: CHRISTOPHER NOLAN

old 1: insert into director values(&did,'&dname')

new 1: insert into director values(14,'CHRISTOPHER NOLAN')

1 row created.

SQL> /

Enter value for did: 15

Enter value for dname: COLIN TREVORROW

old 1: insert into director values(&did,'&dname')

new 1: insert into director values(15,'COLIN TREVORROW')

1 row created.

SQL> /

Enter value for did: 16

Enter value for dname: RIDLEY SCOTT

old 1: insert into director values(&did,'&dname')

new 1: insert into director values(16,'RIDLEY SCOTT')

1 row created.

SQL> select \* from director;

DID DNAME

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

10 STEVEN SPIELBERG

11 JAMES CAMERON

12 MARTIN SCORSESE

13 BAZ LUHRMANN

14 CHRISTOPHER NOLAN

15 COLIN TREVORROW

16 RIDLEY SCOTT

7 rows selected.

**INSERTION INTO MOVIE**

SQL> insert into movie values(&mid,'&mtitle',&myear,&did);

Enter value for mid: 1001

Enter value for mtitle: JURASSIC PARK

Enter value for myear: 1993

Enter value for did: 10

old 1: insert into movie values(&mid,'&mtitle',&myear,&did)

new 1: insert into movie values(1001,'JURASSIC PARK',1993,10)

1 row created.

SQL> /

Enter value for mid: 1002

Enter value for mtitle: TITANIC

Enter value for myear: 1997

Enter value for did: 11

old 1: insert into movie values(&mid,'&mtitle',&myear,&did)

new 1: insert into movie values(1002,'TITANIC',1997,11)

1 row created.

SQL> /

Enter value for mid: 1003

Enter value for mtitle: THE AVIATOR

Enter value for myear: 2004

Enter value for did: 12

old 1: insert into movie values(&mid,'&mtitle',&myear,&did)

new 1: insert into movie values(1003,'THE AVIATOR',2004,12)

1 row created.

SQL> /

Enter value for mid: 1004

Enter value for mtitle: BODY OF LIES

Enter value for myear: 2008

Enter value for did: 16

old 1: insert into movie values(&mid,'&mtitle',&myear,&did)

new 1: insert into movie values(1004,'BODY OF LIES ',2008,16)

1 row created.

SQL> /

Enter value for mid: 1005

Enter value for mtitle: AVATAR

Enter value for myear: 2009

Enter value for did: 11

old 1: insert into movie values(&mid,'&mtitle',&myear,&did)

new 1: insert into movie values(1005,'AVATAR',2009,11)

1 row created.

SQL> /

Enter value for mid: 1006

Enter value for mtitle: INCEPTION

Enter value for myear: 2010

Enter value for did: 14

old 1: insert into movie values(&mid,'&mtitle',&myear,&did)

new 1: insert into movie values(1006,'INCEPTION ',2010,14)

1 row created.

SQL> /

Enter value for mid: 1007

Enter value for mtitle: THE GREAT GATSBY

Enter value for myear: 2013

Enter value for did: 13

old 1: insert into movie values(&mid,'&mtitle',&myear,&did)

new 1: insert into movie values(1007,'THE GREAT GATSBY',2013,13)

1 row created.

SQL>

SQL> /

Enter value for mid: 1008

Enter value for mtitle: JURASSIC WORLD

Enter value for myear: 2015

Enter value for did: 15

old 1: insert into movie values(&mid,'&mtitle',&myear,&did)

new 1: insert into movie values(1008,'JURASSIC WORLD ',2015,15)

1 row created.

SQL> /

Enter value for mid: 1009

Enter value for mtitle: THE BFG

Enter value for myear: 2016

Enter value for did: 10

old 1: insert into movie values(&mid,'&mtitle',&myear,&did)

new 1: insert into movie values(1009,'THE BFG',2016,10)

1 row created.

SQL> /

Enter value for mid: 1010

Enter value for mtitle: THE POST

Enter value for myear: 2017

Enter value for did: 10

old 1: insert into movie values(&mid,'&mtitle',&myear,&did)

new 1: insert into movie values(2010,'THE POST',2017,10)

1 row created.

SQL> select \* from movie;

MID MTITLE MYEAR DID

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

1001 JURASSIC PARK 1993 10

1002 TITANIC 1997 11

1003 THE AVIATOR 2004 12

1004 BODY OF LIES 2008 16

1005 AVATAR 2009 11

1006 INCEPTION 2010 14

1007 THE GREAT GATSBY 2013 13

1008 JURASSIC WORLD 2015 15

1009 THE BFG 2016 10

1010 THE POST 2017 10

10 rows selected.

**INSERTION INTO MOVIE\_CAST**

SQL> insert into movie\_cast values(&act\_id,&mid,'&role');

Enter value for act\_id: 104

Enter value for mid: 1001

Enter value for role: Hero

old 1: insert into movie\_cast values(&act\_id,&mid,'&role')

new 1: insert into movie\_cast values(104,1001,'Hero')

1 row created.

SQL> /

Enter value for act\_id: 108

Enter value for mid: 1001

Enter value for role: Heroine

old 1: insert into movie\_cast values(&act\_id,&mid,'&role')

new 1: insert into movie\_cast values(108,1001,'Heroine')

1 row created.

SQL> /

Enter value for act\_id: 101

Enter value for mid: 1002

Enter value for role: Hero

old 1: insert into movie\_cast values(&act\_id,&mid,'&role')

new 1: insert into movie\_cast values(101,1002,'Hero')

1 row created.

SQL> /

Enter value for act\_id: 102

Enter value for mid: 1002

Enter value for role: Heroine

old 1: insert into movie\_cast values(&act\_id,&mid,'&role')

new 1: insert into movie\_cast values(102,1002,'Heroine')

1 row created.

SQL> /

Enter value for act\_id: 101

Enter value for mid: 1003

Enter value for role: Hero

old 1: insert into movie\_cast values(&act\_id,&mid,'&role')

new 1: insert into movie\_cast values(101,1003,'Hero')

1 row created.

SQL> /

Enter value for act\_id: 109

Enter value for mid: 1003

Enter value for role: Heroine

old 1: insert into movie\_cast values(&act\_id,&mid,'&role')

new 1: insert into movie\_cast values(109,1003,'Heroine')

1 row created.

SQL> /

Enter value for act\_id: 101

Enter value for mid: 1004

Enter value for role: Hero

old 1: insert into movie\_cast values(&act\_id,&mid,'&role')

new 1: insert into movie\_cast values(101,1004,'Hero')

1 row created.

SQL> /

Enter value for act\_id: 103

Enter value for mid: 1005

Enter value for role: Hero

old 1: insert into movie\_cast values(&act\_id,&mid,'&role')

new 1: insert into movie\_cast values(103,1005,'Hero')

1 row created.

SQL> /

Enter value for act\_id: 101

Enter value for mid: 1006

Enter value for role: Hero

old 1: insert into movie\_cast values(&act\_id,&mid,'&role')

new 1: insert into movie\_cast values(101,1006,'Hero')

1 row created.

SQL> /

Enter value for act\_id: 101

Enter value for mid: 1007

Enter value for role: Hero

old 1: insert into movie\_cast values(&act\_id,&mid,'&role')

new 1: insert into movie\_cast values(101,1007,'Hero')

1 row created.

SQL> /

Enter value for act\_id: 106

Enter value for mid: 1008

Enter value for role: Hero

old 1: insert into movie\_cast values(&act\_id,&mid,'&role')

new 1: insert into movie\_cast values(106,1008,'Hero')

1 row created.

SQL> /

Enter value for act\_id: 107

Enter value for mid: 1008

Enter value for role: Heroine

old 1: insert into movie\_cast values(&act\_id,&mid,'&role')

new 1: insert into movie\_cast values(107,1008,'Heroine')

1 row created.

SQL> select \* from movie\_cast;

ACT\_ID MID ROLE

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

104 1001 Hero

108 1001 Heroine

101 1002 Hero

102 1002 Heroine

101 1003 Hero

109 1003 Heroine

101 1004 Hero

103 1005 Hero

101 1006 Hero

101 1007 Hero

106 1008 Hero

ACT\_ID MID ROLE

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

107 1008 Heroine

12 rows selected.

**INSERTION INTO RATING**

SQL> insert into rating values(&mid,&rev\_stars);

Enter value for mid: 1001

Enter value for rev\_stars: 5

old 1: insert into rating values(&mid,&rev\_stars)

new 1: insert into rating values(1001,5)

1 row created.

SQL> /

Enter value for mid: 1002

Enter value for rev\_stars: 6

old 1: insert into rating values(&mid,&rev\_stars)

new 1: insert into rating values(1002,6)

1 row created.

SQL> /

Enter value for mid: 1003

Enter value for rev\_stars: 3

old 1: insert into rating values(&mid,&rev\_stars)

new 1: insert into rating values(1003,3)

1 row created.

SQL> /

Enter value for mid: 1004

Enter value for rev\_stars: 4

old 1: insert into rating values(&mid,&rev\_stars)

new 1: insert into rating values(1004,4)

1 row created.

SQL> /

Enter value for mid: 1005

Enter value for rev\_stars: 4

old 1: insert into rating values(&mid,&rev\_stars)

new 1: insert into rating values(1005,4)

1 row created.

SQL> /

Enter value for mid: 1006

Enter value for rev\_stars: 2

old 1: insert into rating values(&mid,&rev\_stars)

new 1: insert into rating values(1006,2)

1 row created.

SQL> /

Enter value for mid: 1007

Enter value for rev\_stars: 2

old 1: insert into rating values(&mid,&rev\_stars)

new 1: insert into rating values(1007,2)

1 row created.

SQL> /

Enter value for mid: 1008

Enter value for rev\_stars: 6

old 1: insert into rating values(&mid,&rev\_stars)

new 1: insert into rating values(1008,6)

1 row created.

SQL> /

Enter value for mid: 1009

Enter value for rev\_stars: 4

old 1: insert into rating values(&mid,&rev\_stars)

new 1: insert into rating values(1009,4)

1 row created.

SQL> /

Enter value for mid: 1010

Enter value for rev\_stars: 2

old 1: insert into rating values(&mid,&rev\_stars)

new 1: insert into rating values(1010,2)

1 row created.

SQL> select \* from rating;

MID REV\_STARS

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

1001 5

1002 6

1003 3

1004 4

1005 4

1006 2

1007 2

1008 6

1009 4

1010 2

10 rows selected.