

DBMS LABMANUAL

PROGRAM1

ExecuteDDLcommandsstudent(regnonumber,name
varchar2,dob date,marks number)

//Questions:

- a) createtheabovetablewithsuitableconstraints
- b) removetheexistingattributemarksfromthetable.
- c) changethedatatypeofregnofromnumber tovarchar2.
- d) addanewattributephnototheexisting table.
- e) insert5tuplesintothe table.
- f) displaythe tuples fromthetable.

//Query

SQL>createtablestudent(regnonumber(5),namevarchar2(10),dobdate

Tablecreated.

SQL> altertablestudentdropcolumnmarks;

Tablealtered.

SQL> altertablestudentmodifyregno varchar2(5);

Tablealtered.

SQL>altertablestudentaddphnonumber(10);

Table altered.

SQL>

SQL> insertintostudentvalues(100,'AAA','01-jan-2000',9999);

1row created.

SQL>insertintostudentvalues(101,'BBB','01-feb-2001',8888);

1row created.

SQL> insertintostudentvalues(102,'CCC','01-mar-2002',7777);

1row created.

SQL>insertintostudentvalues(103,'DDD','01-apr-2003',6666);

1row created.

SQL> insertintostudentvalues(104,'EEE','01-may-2004',5555);

1row created.

SQL> select*fromstudent;

REGNONAME	DOB	PHNO

100	AAA	01-JAN-00	9999
101	BBB	01-FEB-01	8888
102	CCC	01-MAR-02	7777
103	DDD	01-APR-03	6666
104	EEE	01-MAY-04	5555

PROGRAM2

Execute DML Commands Library(bid number,title
varchar2,authorvarchar2,publishervarchar2,year_of_pubnumber,
price number)

//Question

- a) createtheabove table.
- b) enter5tuplesintothetable.
- c) displayallthetuples.
- d) displaydiffrentpublisherfromthetable.
- e) updatepriceofall bookswith5%ofGSTamount.
- f) deletethedetailsofbookspublishedbyaspecificauthor.
- g) arrangethetuplesin thealphabeticalorder ofbooktitle.
- h) listthedetailsofallbookswhosepricerangebetween
100RS and 300RS.

//Query

```
SQL> createtablelibrary(bidnumber(3)primarykey,titlevarchar2(10),auth  
2  publishervarchar2(10),"yearofpub"number(4),pricenumber(3));
```

Tablecreated.

```
SQL>insertintolibraryvalues(100,'c-prog','shrikanth','skywords',2015,180
```

1row created.

```
SQL> insertintolibraryvalues(101,'c++-prog','shrikanth','skywords',2016
```

1row created.

```
SQL> insertintolibraryvalues(103,'java-prog','james','himalya',2017,250
```

1row created.

```
SQL> insertintolibraryvalues(104,'data-str','rajesh','skywords',2018,32
```

1row created.

```
SQL> insertintolibraryvalues(105,'dot-net','kavya','himalya',2019,280);
```

1row created.

```
SQL> insertintolibraryvalues(106,'dbms','kokhila','shiva',2020,290);
```

1row created.

```
SQL> insertintolibraryvalues(107,'python','james','skywords',2021,380);
```

1row created.

```
SQL> select*fromlibrary;
```

BID	TITLE	AUTHOR		PUBLISHER	year of pub	PRICE

100	c-prog	shrikanth skywords			2015	180
101	c++-prog	shrikanth skywords			2016	70
103	java-prog	james		himalaya	2017	250
104	data-str	rajesh		skywords	2018	320
105	dot-net	kavya		himalaya	2019	280
106	dbms	kokhila		shiva	2020	290
107	python	james		skywords	2021	380

7 rows selected.

SQL> select distinct publisher from library;

PUBLISHER

skywords

shiva

himalaya

SQL> update library set price=price+((price*5)/100);

7 rows updated.

SQL> delete from library where author='james';

2rowsdeleted.

SQL>select*fromlibraryorderbytitleasc;

BID	TITLE	AUTHOR	PUBLISHER	yearofpub	PRICE
101	c++-prog	shrikanth	skywords	2016	74
100	c-prog	shrikanth	skywords	2015	189
104	data-str	rajesh	skywords	2018	336
106	dbms	kokhila	shiva	2020	305
105	dot-net	kavya	himalaya	2019	294

PROGRAM3

**ExecuteTCLandgroupfunctions.Considerthetable
employee(empno,empname,dept,salary,doj,branch)**

//Question

- a. Createtable.**
- b. insert5tuples.**
- c. displayallthetuples**
- d. retrieveaveragesalaryofallemployees.**
- e. retrieve number of employees.**
- f. retrievedistinctnumberofemployee indeptwise.**
- g. retrievetotalsalaryofemployee groupbyemployeenam and
count similar names**
- h. displaydetailsofemployeeswhosesalaryisgreaterthan
25000.**
- i. performcommitandrollbackoperation.**

//Query

```
SQL> create table employee(empno number(3)primary key,ename  
varchar2(10),deptvarchar2(10),salarynumber(5),dobdate,branchvarchar2(10));
```

Tablecreated.

```
SQL>insertintoemployeevalues(101,'ashok','sales',27000,'01-jan-  
2022','tumkur');
```

1row created.

```
SQL>    insertintoemployeevalues(102,'raju','accounts',25000,'01-feb-  
2022','mysore');
```

1row created.

```
SQL>    insertintoemployeevalues(103,'ramya','sales',29000,'01-mar-  
2022','tumkur');
```

1row created.

```
SQL>    insertintoemployeevalues(104,'ashok','sales',37000,'01-apr-  
2022','banglore');
```

1row created.

```
SQL>    insertintoemployeevalues(105,'radhika','marketing',23000,'01-may-  
2022','tumkur');
```

1row created.

```
SQL> insertintoemployeevalues(106,'ashok','sales',17000,'01-jun-2022','mysore');
```

1row created.

```
SQL> insertintoemployeevalues(107,'ramya','accounts',27000,'01-jul-2022','tumkur');
```

1row created.

```
SQL> insertintoemployeevalues(108,'rakshita','sales',29000,'01-aug-2022','bangalore');
```

1row created.

```
SQL> insertintoemployeevalues(109,'ashok','marketing',24000,'01-sep-2022','tumkur');
```

1row created.

```
SQL>select*fromemployee;
```

EMPNO	ENAME	DEPT	SALARY	DOB	BRANCH
101	ashok	Sales	27000	01-JAN-22	tumkur
102	raju	Accounts	25000	01-FEB-22	mysore
103	ramya	Sales	29000	01-MAR-22	tumkur
104	ashok	Sales	37000	01-APR-22	banglore

105	radhika	marketing	2300001-MAY-22	tumkur
106	ashok	sales	1700001-JUN-22	mysore
107	ramya	accounts	2700001-JUL-22	tumkur
108	rakshita	sales	2900001-AUG-22	bangalore
109	ashok	marketing	2400001-SEP-22	tumkur

9rowsselected.

SQL>selectavg(salary)fromemployee;

AVG(SALARY)

26444.4444

SQL> selectcount(*)fromemployee;

COUNT(*)

9

SQL>selectdistinctdept,count(*)fromemployeegroupbydept;

select distinct dept,count(*)from employee group by dept

*

DEPT COUNT(*)

accounts	2
sales	5
marketing	2

```
SQL>select ename,count(8),sum(salary)from employee group by ename;
```

ENAME	COUNT(8)	SUM(SALARY)

Raju	1	25000
rakshita	1	29000
radhika	1	23000
ashok	4	105000
ramya	2	56000

```
SQL>
```

```
SQL>select *from employee where salary > 25000;
```

EMPNO	ENAME	DEPT	SALARY	DOB	BRANCH

101	ashok	sales	27000	01-JAN-22	tumkur
103	ramya	sales	29000	01-MAR-22	tumkur
104	ashok	sales	37000	01-APR-22	bangalore
107	ramya	accounts	27000	01-JUL-22	tumkur
108	rakshita	sales	29000	01-AUG-22	bangalore

```
SQL>commit;
```

Commitcomplete.

```
SQL> insertintoemployeevalues(110,'kavya','marketing',25000,'01-  
oct2022','tumkur');
```

1row created.

```
SQL> rollback;
```

Rollbackcomplete.

PROGRAM4

**ImplementNestedQueries:AnInventorydatabasehasthe
following table.ITEMS(itemcode,name,price);**

PURCHASE(itemcode,qty);

//Question

- a) Createthetableswiththeaboveattributes.**
- b) Insert5tuplesintohtables.**
- c) displayalltherecordsineachtables**
- d) Listtheitemspurchased.**
- e) Listtheitemswhichare notpurchased.**

//Query

```
SQL>createtableitem(itemcodenumber(3)primarykey,namevarchar2(10),price  
number(3));
```

Tablecreated.

```
SQL>
```

```
createtablepurchase(itemcodenumber(3)referencesitem,qtynu  
mber(3));
```

Table created.

```
SQL> insert into itemvalues(100,'rice',50);
```

1 row created.

```
SQL>
```

```
SQL> insert into itemvalues(101,'soap',35);
```

1 row created.

```
SQL>
```

```
SQL> insert into itemvalues(102,'paste',25);
```

1 row created.

```
SQL>
```

```
SQL> insert into itemvalues(103,'sugar',40);
```

1 row created.

```
SQL>
```

```
SQL> insert into itemvalues(104,'oil',20);
```

1 row created.

```
SQL>
```

```
SQL> insertintoitemvalues(105,'oil',100);
```

1row created.

```
SQL>
```

```
SQL>
```

```
SQL> insertintopurchasevalues(100,10);
```

1row created.

```
SQL>
```

```
SQL> insertintopurchasevalues(102,5);
```

1row created.

```
SQL>
```

```
SQL> insertintopurchasevalues(104,5);
```

1row created.

```
SQL>
```

```
SQL> insertintopurchasevalues(105,3);
```

1row created.

```
SQL>
```

```
SQL> select*fromitem;
```

ITEMCODE	NAME	PRICE

100	rice	50
101	soap	35
102	paste	25
103	sugar	40
104	oil	20
105	oil	100

6 rows selected.

SQL>

SQL> select * from purchase;

ITEMCODE	QTY

100	10
102	5
104	5
105	3

SQL>

SQL> select * from item where itemcode in (select itemcode from purchase);

ITEMCODE	NAME	PRICE

100	rice	50
102	paste	25
104	oil	20
105	oil	100

```
SQL>select*fromitemwhereitemcodenotin(selectitemcodefrompurchase);
```

ITEMCODENAME	PRICE

101 soap	35
103sugar	40

PROGRAM5

Implementjoinoperationin SQL

Thecompanydatabaseconsistofthefollowingtables

Department(dno,dname,mgrid,mgrjoindate);

Employee(empid,name,address,gender,salary,dno)

//Question

createtablesandinsert5tupleseachandperformthe following.

- a) Givea10%raiseinsalaryforallemployeesworkingin Research dept.**
- b) Retrievethe nameofemployeecontrolledbydeptnumber 102 using Exists operator.**
- c) Retrievethe nameofemployeesandtheirdeptname**

(using natural join).

d) Perform EQUI join operation on the given tables.

e) Perform NON-EQUI join operation on the given tables.

f) Perform OUTER join operation on the given tables.

//QUERY

**SQL> create table department1(dno number(3) primary key, dname
varchar2(10), mgr id number(3), mgr io ind
ate date);**

Table created.

**SQL> create table employee1(emp id number(3) primary key, ename
varchar2(10), address varchar2(10), gender
erv varchar2(6), salary number(5), dno number(3) references department);**

Table created.

SQL> insert into department1 values(100,'research',500,'01-jan-2022');

1 row created.

SQL>

SQL> insert into department1 values(101,'technical',501,'01-feb-2022');

1 row created.

SQL>


```
SQL> insertintodepartment1values(102,'testing',502,'01-mar-2022');
```

1rowcreated.

```
SQL>
```

```
SQL> insertintodepartment1values(103,'coding',500,'01-apr-2022');
```

1rowcreated.

```
SQL> insert into employee1  
values(200,'abhishek','tumkur','male',25000,100);
```

1rowcreated.

```
SQL>
```

```
SQL> insert into employee1  
values(201,'manasa','bangalore','female',23000,101);
```

1rowcreated.

```
SQL>
```

```
SQL> insert into employee1  
values(202,'chandana','tumkur','female',28000,102);
```

1rowcreated.

```
SQL>
```

```
SQL> insert into employee1  
values(203,'mohan','bangalore','male',35000,103);
```

1rowcreated.

SQL>

**SQL> insert into employee1
values(204,'sanvi','tumkur','female',29000,101);**

1rowcreated.

SQL>

**SQL> insert into employee1
values(205,'rajeshwari','tumkur','female',30000,100);**

1rowcreated.

SQL>

**SQL> insert into employee1
values(206,'vidya','bangalore','female',20000,102);**

1rowcreated.

**SQL> update employee1 set salary=salary+(salary/10) where dno in (select dno
from department1 where
dname='coding');**

1rowupdated.

**SQL> select ename from employee1 where exists (select d.dno from department1
d where e.dno=102 and
d.dno=102);**

ENAME

chandana

vidya

SQL>selecte.ename,d.dnamefromemployee1enaturaljoindepartmentd;

ENAME DNAME

abhishek research

manasa technical

chandana testing

mohan coding

sanvi technical

rajeshwariresearch

vidya testing

7rowsselected.

SQL>select*fromemployee1e,departmentdwheree.dno=d.dno;

EMPID ENAME ADDRESS GENDER SALARY DNO
DNO

DNAME MGRID MGRIOINDA

200abhishek tumkur male 25000 100 100
research 50001-JAN-22

201manasa bangalorefemale 23000 101 101
technical 50101-FEB-22

202chandana tumkur female 28000 102 102
testing 50201-MAR-22

EMPID ENAME ADDRESS GENDER SALARY DNO
DNO

DNAME MGRID MGRIIONDA

203mohan bangalorefemale 38500 103 103
coding 50001-APR-22

204sanvi tumkur female 29000 101 101
technical 50101-FEB-22

205rajeshwaritumkur female 30000 100 100
research 50001-JAN-22

EMPID ENAME ADDRESS GENDER SALARY DNO
DNO

DNAME MGRID MGRIIONDA

206vidya bangalorefemale 20000 102 102
testing 50201-MAR-22

7rowsselected.

SQL>select*fromemployee1e,departmentdwheree.dno!=d.dno;

EMPID ENAME ADDRESS GENDER SALARY DNO
DNO

DNAME MGRID MGRIOINDA

201manasa bangalorefemale 23000 101 100
research 50001-JAN-22

202chandana tumkur female 28000 102 100
research 50001-JAN-22

203mohan bangaloremale 38500 103 100
research 50001-JAN-22

EMPID ENAME ADDRESS GENDER SALARY DNO
DNO

DNAME MGRID MGRIOINDA

204sanvi tumkur female 29000 101 100

research 50001-JAN-22

 206vidya bangalorefemale 20000 102 100
research 50001-JAN-22

 200abhishek tumkur male 25000 100 101
technical 50101-FEB-22

 EMPID ENAME ADDRESS GENDER SALARY DNO
DNO

DNAME MGRID MGRIOINDA

 202chandana tumkur female 28000 102 101
technical 50101-FEB-22

 203mohan bangaloremale 38500 103 101
technical 50101-FEB-22

 205rajeshwaritumkur female 30000 100 101
technical 50101-FEB-22

 EMPID ENAME ADDRESS GENDER SALARY DNO
DNO

DNAME MGRID MGRIOINDA

206vidya bangalorefemale 20000 102 101
technical 50101-FEB-22

200abhishek tumkur male 25000 100 102
testing 50201-MAR-22

201manasa bangalorefemale 23000 101 102
testing 50201-MAR-22

EMPID ENAME ADDRESS GENDER SALARY DNO
DNO

DNAME MGRID MGRIOINDA

203mohan bangaloremale 38500 103 102
testing 50201-MAR-22

204sanvi tumkur female 29000 101 102
testing 50201-MAR-22

205rajeshwaritumkur female 30000 100 102
testing 50201-MAR-22

EMPID ENAME ADDRESS GENDER SALARY DNO
DNO

DNAME MGRID MGRIOINDA

200abhishek tumkur male 25000 100 103
coding 50001-APR-22

201manasa bangalorefemale 23000 101 103
coding 50001-APR-22

202chandana tumkur female 28000 102 103
coding 50001-APR-22

EMPID ENAME ADDRESS GENDER SALARY DNO
DNO

DNAME MGRID MGRIOINDA

204sanvi tumkur female 29000 101 103
coding 50001-APR-22

205rajeshwaritumkur female 30000 100 103
coding 50001-APR-22

206vidya bangalorefemale 20000 102 103
coding 50001-APR-22

21rowsselected.

```
SQL>select*fromemployee1efullouterjoindepartment1done.dno=d.dno;
```

EMPID ENAME ADDRESS GENDER SALARY DNO
DNO

DNAME MGRID MGRIOINDA

205rajeshwaritumkur female 30000 100 100
research 50001-JAN-22

200abhishek tumkur male 25000 100 100
research 50001-JAN-22

204sanvi tumkur female 29000 101 101
technical 50101-FEB-22

EMPID ENAME ADDRESS GENDER SALARY DNO
DNO

DNAME MGRID MGRIOINDA

201manasa bangalorefemale 23000 101 101
technical 50101-FEB-22

206vidya bangalorefemale 20000 102 102

testing 50201-MAR-22

202chandana tumkur female 28000 102 102
testing 50201-MAR-22

EMPID ENAME ADDRESS GENDER SALARY DNO
DNO

DNAME MGRID MGRIOINDA

203mohan bangaloremale 38500 103 103
coding 50001-APR-22

7rowsselected.

PROGRAM 6

Create views for a particular table the railway reservation system
databases consist of the following table

TRAIN(tno,tname,splace,destination)

AVAILABILITY(tno,class splace,destination ,seats)

//Question

- a. Create view SLEEPER to display Train No , Start-Place , Destination which have Sleeper class and perform the following
 - 1. Insert new record

2. Update destination='Mangalore' where trainno = 'rjd16'
3. Delete a record which have train no ='kke55'
- b. Create view DETAILS to display train no , train name , class
- c. Create view TOTAL-SEATS to display train number , start-place , use count function to number of seats , group by start-place and perform the following
 1. Insert new record
 2. Update start-place = 'Hubli' where train-no = 'jn58'
 3. Delete last row of the view
- d. Rename view sleeper to sleeper class
- e. Delete view DETAILS

//QUERY

```
SQL> create table trainn(tno varchar2(5),name char(10),splace varchar2(10),dest  
varchar2(15));
```

Table created.

```
SQL> create table availablityy(tno varchar2(10),class char(10),splace varchar2(10),dest  
varchar2(10)  
,seats number(10));
```

Table created.

```
SQL> create view sleeperr as select tno,splace,dest from trainn;
```

View created.

```
SQL> insert into sleeperr values('&tno','&name','&dest');
```

Enter value for tno: rjd16

Enter value for name: banglore

Enter value for dest: hubli

old 1: insert into sleeperr values('&tno','&name','&dest')

new 1: insert into sleeperr values('rjd16','banglore','hubli')

1 row created.

```
SQL> /
```

Enter value for tno: rjd17

Enter value for name: banglore

Enter value for dest: dharwad

old 1: insert into sleeperr values('&tno','&name','&dest')

```
new 1: insert into sleeperr values('rjd17','banglore','dharwad')
```

1 row created.

```
SQL> rjd18
```

```
SP2-0042: unknown command "rjd18" - rest of line ignored.
```

```
SQL> /
```

```
Enter value for tno: rjd18
```

```
Enter value for name: banglore
```

```
Enter value for dest: mysore
```

```
old 1: insert into sleeperr values('&tno','&name','&dest')
```

```
new 1: insert into sleeperr values('rjd18','banglore','mysore')
```

1 row created.

```
SQL> /
```

```
Enter value for tno: kke55
```

```
Enter value for name: banglore
```

```
Enter value for dest: udupi
```

```
old 1: insert into sleeperr values('&tno','&name','&dest')
```

```
new 1: insert into sleeperr values('kke55','banglore','udupi')
```

1 row created.

```
SQL> /
```

```
Enter value for tno: kke56
```

```
Enter value for name: banglore
```

```
Enter value for dest: belgam
```

```
old 1: insert into sleeperr values('&tno','&name','&dest')
```

```
new 1: insert into sleeperr values('kke56','banglore','belgam')
```

1 row created.

```
SQL> /
```

```
Enter value for tno: kke57
```

```
Enter value for name: banglore
```

```
Enter value for dest: tumkur
```

```
old 1: insert into sleeperr values('&tno','&name','&dest')
```

```
new 1: insert into sleeperr values('kke57','banglore','tumkur')
```

1 row created.

```
SQL> select* from sleeperr;
```

TNO SPLACE DEST

rjd16 banglore hubli
rjd17 banglore dharwad
rjd18 banglore mysore
kke55 banglore udupi
kke56 banglore belgam
kke57 banglore tumkur

6 rows selected.

SQL> update sleeperr set dest='mangalore' where tno='rjd16';

1 row updated.

SQL> delete from sleeperr where tno='kke55';

1 row deleted.

SQL> select* from sleeperr;

TNO SPLACE DEST

rjd16 banglore mangalore
rjd17 banglore dharwad
rjd18 banglore mysore
kke56 banglore belgam
kke57 banglore tumkur

SQL> create view details as select t.tno,t.name,a.class from trainn t,availablityy a;

View created.

SQL> create view totallseatss as select tno,splace from availablityy;

View created.

SQL> insert into totallseatss values(' &tno','&splace');
Enter value for tno: rjd17
Enter value for splace: banglore
old 1: insert into totallseatss values(' &tno','&splace')
new 1: insert into totallseatss values('rjd17','banglore')

1 row created.

```
SQL> /
Enter value for tno: rjd18
Enter value for splace: banglore
old 1: insert into totallseatss values('&tno','&splace')
new 1: insert into totallseatss values('rjd18','banglore')

1 row created.
```

```
SQL> /
Enter value for tno: rjd19
Enter value for splace: banglore
old 1: insert into totallseatss values('&tno','&splace')
new 1: insert into totallseatss values('rjd19','banglore')

1 row created.
```

```
SQL> /
Enter value for tno: jn55
Enter value for splace: banglore
old 1: insert into totallseatss values('&tno','&splace')
new 1: insert into totallseatss values('jn55','banglore')

1 row created.
```

```
SQL> /
Enter value for tno: rjd20
Enter value for splace: banglore
old 1: insert into totallseatss values('&tno','&splace')
new 1: insert into totallseatss values('rjd20','banglore')

1 row created.
```

```
SQL> select * from totallseatss;
```

TNO	SPLACE
-----	-----
rjd17	banglore
rjd18	banglore
rjd19	banglore
jn55	banglore
rjd20	banglore

```
SQL> update totallseatss set splace='hubli' where tno='jn55';
```

1 row updated.

SQL> select * from totallseatss;

TNO	SPLACE
-----	-----
rjd17	banglore
rjd18	banglore
rjd19	banglore
jn55	hubli
rjd20	banglore

SQL> delete from totallseatss where tno='rjd20';

1 row deleted.

SQL> select * from totallseatss;

TNO	SPLACE
-----	-----
rjd17	banglore
rjd18	banglore
rjd19	banglore
jn55	hubli

SQL> rename sleeperr to classs;

Table renamed.

SQL> select * from classs;

TNO	SPLACE	DEST
-----	-----	-----
rjd16	banglore	mangalore
rjd17	banglore	dharwad
rjd18	banglore	mysore
kke56	banglore	belgam
kke57	banglore	tumkur

SQL>

SQL> drop view details;

View dropped.

PROGRAM7

Write PL/SQL procedure to compute factorial of a number using recursion

```

declare
    num number;
    factorial number;
    function caluclatefact(x number) return
    number
    is
        f number;
begin
    if x=0 then
        f:=1;
    else
        f:=x*caluclatefact(x-1);
    end if;
    return f;
end;

begin
    num:=6;
    factorial:=caluclatefact(num);
    dbms_output.put_line('factorial' || num || 'is' || factorial);
end;

/

```

PL/SQL procedure successfully completed.

OUTPUT

SQL>setserveroutputon

SQL> /

Factorial6is720

PL/SQLproceduresuccessfullycompleted.

PROGRAM8

Given the table EMPLOYEE(EmpNo, Name, Salary, Designation, DeptID)writeacursorinPL/SQLtoselectthefivehighestpaid employees from the table.

SQL>createtableemployee(empnnumber(10),namevarchar2(10),salarynumber(5)); Table Created

SQL>descemployee

Name	Null?	Type

EMPNO		NUMBER(5)
NAME		VARCHAR2(10)
SALARY		NUMBER(5)

SQL>insertintoemployeevalues(&empno,&'name',&salary);

Enter value for empno: 100

Enter value for name: kohli

Entervalueforsalary:99999

old 1:insertintoemployevalues(&empno,&'name',&salary)

new 1: insert into employe values(100,'kohli',99999)

1row created.

SQL> /

Enter value for empno: 101

Enter value for name: rahul

Enter value for salary: 52000

old 1: insert into employee values(&empno, '&name', &salary)

new 1: insert into employee values(101, 'rahul', 52000)

1 row created

SQL> /

Enter value for empno: 103

Enter value for name: padikal

Enter value for salary: 67000

old 1: insert into employee values(&empno, '&name', &salary)

new 1: insert into employee values(103, 'padikal', 67000)

1 row created.

SQL> /

Enter value for empno: 104

Enter value for name: hardik

Enter value for salary: 88888

old 1: insert into employee values(&empno, '&name', &salary)

new 1: insert into employee values(104, 'hardik', 88888)

1 row created.

SQL> /

Enter value for empno: 105

Enter value for name: dhoni

Enter value for salary: 90000

old1: insert into employee values(&empno, '&name', &salary)

new1: insert into employee values(105, 'dhoni', 90000)

1 row created.

SQL> declare

2 cursor emp is select name, salary from employee order by salary desc;

3 begin

4 for i in emp

5 loop

6 if emp%rowcount <= 5 then

7 dbms_output.put_line(i.name || ' ' || i.salary);

8 endif;

9 endloop;

10 end;

11 /

PL/SQL procedure successfully completed.

OUTPUT

SQL> set serveroutput on

SQL> /

kohli 99999

dhoni 90000

hardik 88000

padikal67000
rahu152000

PROGRAM9

Given the table MOVIE(MID,MTitle,Language,Director,Year) write a function in PL/SQL to find the total number of movies in the table.

```
SQL>create table movie(mid number(10),title varchar2(10),language varchar2(10), director
      varchar2(10), year number(3));
```

Table created.

```
SQL>desc movie
```

Name	Null?	Type

MID		NUMBER(10)
TITLE		VARCHAR2(10)
LANGUAGE		VARCHAR2(10)
DIRECTOR		VARCHAR2(10)
YEAR		NUMBER(3)

```
SQL>insert into movie values(&mid,'&title','&language','&director',&year);
```

Enter value for mid: 101
Enter value for title: aaa
Enter value for language: kannada
Enter value for director: pppEnter
value for year: 01

```
old 1:insertintomovievalues(&mid,'&title','&language','&director','&year)
```

```
new 1: insert into movie values(100,'aaa','kannada','ppp',01)
```

1 rowcreated.

```
SQL> /
```

Entervalueformid:102

Entervaluefortitle:bbb

Entervalueforlanguage:hindi

Enter value for director: xyz

Enter value for year: 02

```
old 1:insertintomovievalues(&mid,'&title','&language','&director','&year)
```

```
new 1: insert into movie values(101,'bbb','hindi','xyz',02)
```

1 rowcreated.

```
SQL> /
```

Entervalueformid:103

Enter value for title:ccc

Entervalueforlanguage:telugu

Enter value for director: abc

Enter value for year: 03

```
old 1:insertintomovievalues(&mid,'&title','&language','&director','&year)
```

```
new 1: insert into movie values(103,'ccc','telugu','abc',03)
```

1 rowcreated.

```
SQL> /
```

Entervalueformid:104

Entervaluefortitle:ddd

Enter value for language: tamil

Enter value for director: prs

Enter value for year: 04

old 1: insert into movie values(&mid, '&title', '&language', '&director', &year)

new 1: insert into movie values(104, 'ddd', 'tamil', 'prs', 04)

1 row created.

SQL> CREATE OR REPLACE FUNCTION moviesf

2 RETURN NUMBER IS

3 total NUMBER := 0;

4 BEGIN

5 SELECT COUNT(*) INTO total FROM movies;

6 RETURN total;

7 END;

8 /

Function created.

SQL> DECLARE

2 n NUMBER;

3 BEGIN

4 n := moviesf();

5 DBMS_OUTPUT.PUT_LINE('number of movies: ' || n);

6 end;

7 /

PL/SQL procedures successfully completed.

OUTPUT:

SQL>set serveroutput on
Totalnumberofmovies:4

PROGRAM10

Given the table CUSTOMERS (CID,Cname,Address)write a PL/SQL programwhichasksforcustomerID,iftheuserentriesinvalid ID then the exception invalid_id is raised.

SQL>createtablecustomer(cidnumber(3),namevarchar2(10),addressvarchar2(10));

Tablecreated.

SQL>desccustomer

Name	Null?	Type

CID		NUMBER(3)
NAME		VARCHAR2(10)
ADDRESS		VARCHAR2(10)

Insertintocustomervvalues(100,'abc','tumkuru');
Insert into customer values(101,'xyz','mysore');
Insert into customer values(102,'pqr','ooty');
Insert into customer values(103,'abd','goa');
Insert into customer values(104,'vk','banglore');
Insert into customer values(105,'abb','panaji');

SQL>declare


```

2  c_idcustomer.cid%type:=0;
3  c_namecustomer.name%type;
4  c_addresscustomer.address%type;
5  begin
6  c_id:=&c_id;
7  selectname,addressintoc_name,c_addressfromcustomerwherecid=c_id;
8  dbms_output.put_line('name='||c_name);
9  dbms_output.put_line('address='||c_address);
10 Exception
11 whenNO_DATA_FOUNDthen
12 dbms_output.put_line('Invalidcustomer-ID');
13 end;
14 /

```

OUTPUT

Entervalueforc_id:101

old 6: c_id:=&c_id;

new 6:c_id:=101;

name=xyz

address=mysore

PL/SQLproceduresuccessfullycompleted.

SQL> /

Entervalueforc_id:105

old 6: c_id:=&c_id;

new 6:c_id:=105;

Invalid cust-ID

