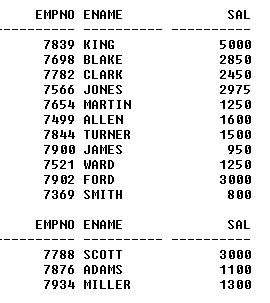
**WORKSHEET 1**

**1.LIST EMPNO,EMPNAME AND SALARY.**

**A.**  **SQL>** select empno,ename,sal from emp;

**OUTPUT:**



**2.LIST THE NAMES OF ALL MANAGERS.**

**A. SQL>** select ename from emp where job='MANAGER';

**OUTPUT:**



**3.LIST ALL CLERKS IN DEPTNO. 30.**

**A. SQL>** select ename from emp where job='CLERK' and deptno=30;

**OUTPUT:**



**4.LIST THE EMPLOYEES TO WHO MANAGER IS 7698.**

**A. SQL>** select ename from emp where mgr=7698;

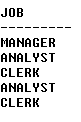
**OUTPUT:**



**5.LIST JOBS IN DEPTNO. 20.**

**A.SQL>** select job from emp where deptno=20;

**OUTPUT:**



**6.LIST EMPLOYEE NAMES WHOSE SALARY IS BETWEEN 2000 AND 3000.**

**A**. **SQL>** select ename from emp where sal between 2000 and 3000;

**OUTPUT:**



**7.LIST EMPLOYEES IN DEPARTMENTS 10,20.**

**A.SQL>** select ename from emp where deptno=10 or deptno=20;

**OUTPUT:**



**8.LIST EMPLOYEE NAMES WHICH BEGINS WITH S.**

**A.SQL>** select ename from emp where ename like 'S%';

**OUTPUT:**



**9.LIST EMPLOYEE NAMES HAVING A IN THEIR NAMES.**

**A.SQL>** select ename from emp where ename like '%A%';

**OUTPUT:**



**10.LIST** **EMPLOYEES WHO HAVE JOINED IN JAN.**

**A.** **SQL>** select ename from emp where to\_char(hiredate,'MON')='JAN';

**OUTPUT:**



**11.LIST EMPLOYEES WHO HAVE JOINED IN THE YEAR 81.**

**A. SQL>** select ename from emp where to\_char(hiredate,'YY')='81'**;**

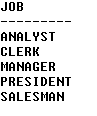
**OUTPUT:**



**12.LIST ALL DISTINCT JOBS.**

**A.** **SQL>** select distinct(job) from emp;

**OUTPUT:**



**13.LIST EMPLOYEE NAMES IN ALPHABETICAL ORDER.**

**A. SQL>** select ename from emp order by ename;

**OUTPUT:**



**14.LIST EMPLOYEE NAMES ALPHABETICALLY DEPARTMENT WISE.**

**A.** **SQL>** select ename from emp order by deptno,ename;

**OUTPUT:**



**15.LIST EMPLOYEE NAMES ALPHABETICALLY JOB WISE.**

**A.** **SQL>** select ename from emp order by job,ename;

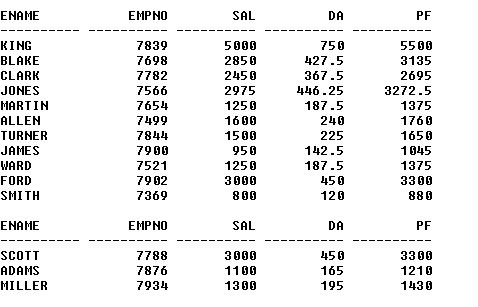
**OUTPUT:**



**16.LIST EMPLOYEE NUMBERS,NAME SAL,DA(15% OF SAL) AND PF(10% OF SAL).**

**A.** **SQL>** select ename,empno,sal,sal\*0.15 as DA,sal\*1.10 as PF from emp;

**OUTPUT:**



**17.LIST EMPLOYEE NAMES HAVING AN EXPERIENCE MORE THAN 15 YEARS.**

**A.** **SQL>** select ename from emp where round((months\_between(sysdate,hiredate))/12)>'15';

**OUTPUT:**



**18.LIST EMPLOYEE NAMES WHOSE COMMISSION IS NULL.**

**A. SQL>** select ename from emp where comm is NULL;

**OUTPUT:**



**19.LIST EMPLOYEES WHO DO NOT REPORT TO ANYBODY.**

**A. SQL>** select ename from emp where mgr is NULL**;**

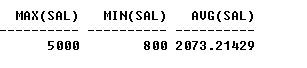
**OUTPUT:**



**20.LIST MAXIMUM SAL,MINIMUM SAL,AVERAGE SAL.**

**A.** **SQL>** select max(sal),min(sal),avg(sal) from emp;

**OUTPUT:**



**21.LIST THE NUMBERS OF JOBS.**

**A. SQL>** select count(distinct(job)) from emp;

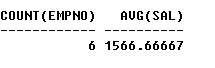
**OUTPUT:**



**22.LIST THE NUMBERS OF PEOPLE AND AVERAGE SALARY IN DEPTNO 30.**

**A**. **SQL>** select count(empno),avg(sal) from emp where deptno=30;

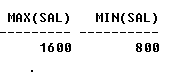
**OUTPUT:**



**23.LIST MAXIMUM SAL AND MINIMUM SAL IN THE DESIGNATIONS SALESMAN AND CLERK.**

**A**. **SQL>** select max(sal),min(sal) from emp where job='SALESMAN' or job='CLERK';

**OUTPUT:**



**24.LIST THE NUMBERS OF PEOPLE AND AVERAGE SALARY OF EMPLOYEES JOINED IN 81,82 AND 83.**

**A.** **SQL>** select count(empno),avg(sal) from emp where to\_char(hiredate,'YY')='81' or to\_char(hiredate,'YY'

)='82' or to\_char(hiredate,'YY')='83';

**OUTPUT:**



**25.LIST JOBS THAT ARE UNIQUE TO DEPTNO 20 SET OPERATIONS(ADD MORE PROBLEMS).**

**A**. **SQL>** select distinct(job) from emp where deptno=20;

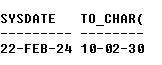
**OUTPUT:**



**26.DISPLAY TODAY’S DATE AND PRESENT TIME.**

**A.** **SQL>** select sysdate,to\_char(sysdate,'HH-MM-SS')from dual;

**OUTPUT:**

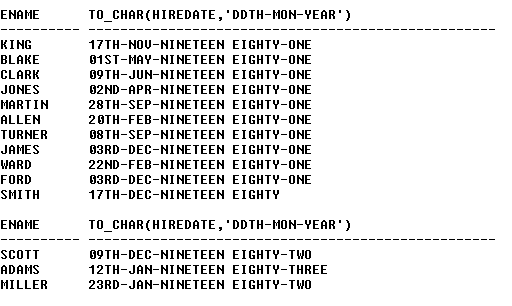


**27.LIST EMPLOYEE NAMES AND JOINING DATE IN THE FOLLOWING FORMATS.**

**A.SMITH 19TH DEC NINETEEN EIGHTY.**

**A.** **SQL>** select ename,to\_char(hiredate,'DDth-MON-YEAR') from emp;

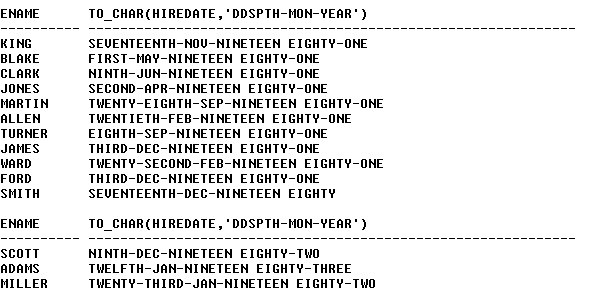
**OUTPUT:**



**B.SMITH SEVENTEENTH DEC NINETEEN EIGHTY.**

**A.SQL>** select ename,to\_char(hiredate,'DDSPTH-MON-YEAR') from emp;

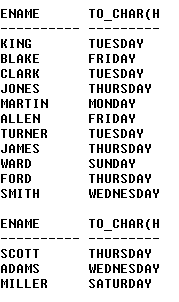
**OUTPUT:**



**C.SMITH WEEK DAY OF JOINING.**

**A.** **SQL>** select ename,to\_char(hiredate,'DAY') from emp;

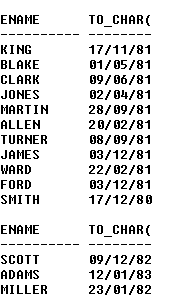
**OUTPUT:**



**D.SMITH 17/12/80.**

**A.SQL>**select ename,to\_char(hiredate,'DD/MM/YY') from emp;

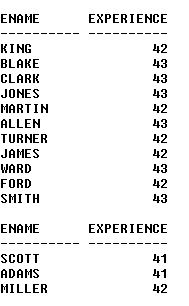
**OUTPUT:**



**28.LIST EMPLOYEE NAMES AND THEIR EXPERIENCE IN YEARS.**

**A.SQL>** select ename,round((months\_between(sysdate,hiredate))/12) as experience from emp;

**OUTPUT:**



**29.LIST EMPLOYEE NAMES WHO JOINED INDEC AND ON MONDAY OR FRIDAY.**

**A. SQL>** select ename from emp where to\_char(hiredate,'MON')='DEC' and to\_char(hiredate,'DY') in ('MON','FRI');

**OUTPUT:**



**30.DISPLAY A GIVEN DATE AS A STRING IN DIFFERENT FORMATS.**

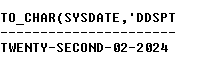
**A.** **SQL>** select to\_char(sysdate,'DD-MM-YY') from dual;

**OUTPUT:**



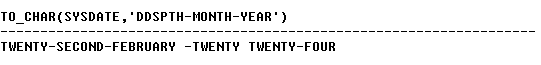
**SQL>**select to\_char(sysdate,'DDSPTH-MM-YYYY') from dual;

**OUTPUT:**



**SQL>**select to\_char(sysdate,'DDSPTH-MONTH-YEAR') from dual;

**OUTPUT:**



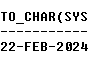
**SQL>** select to\_char(sysdate,'DD-MON-YEAR') from dual;

**OUTPUT:**



**SQL>**select to\_char(sysdate,'DD-MON-YYYY') from dual;

**OUTPUT:**



**SQL>** select to\_char(sysdate,'DAY') from dual;

**OUTPUT:**



**SQL>** select to\_char(sysdate,'WW') from dual;

**OUTPUT:**

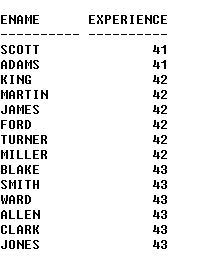


**WORKSHEET 2**

**1.LIST EMPLOYEE NAMES AND THEIR HIRE DATES SORTED IN THE ORDER OF THEIR EXPERIENCE.**

**A. SQL>** select ename,round((months\_between(sysdate,hiredate))/12) as experience from emp order by experience;

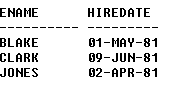
**OUTPUT:**



**2.LIST MANAGERS NAMES AND THEIR JOINING DATES COMPLETELY SPELLED IN ALPHABETICAL ORDER OF NAMES.**

**A.** **SQL>** select ename,hiredate from emp where job='MANAGER' order by ename;

**OUTPUT:**

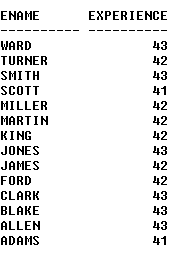


**3.LIST EMPLOYEE NAMES AND THEIR EXPERIENCE IN YEARS WITH NAMES ARRANGED IN DESCENDING ORDER.**

**A.** **SQL>** select ename,round((months\_between(sysdate,hiredate))/12) as experience from emp order by

desc;

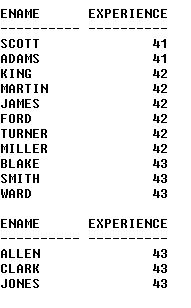
**OUTPUT:**



**4.LIST EMPLOYEE NAMES HAVING A MINIMUM OF 2 YEARS EXPERIENCE SORTED ON EXPERIENCE.**

**A.** **SQL>** select ename,round((months\_between(sysdate,hiredate))/12) as experience from emp where round((months\_between(sysdate,hiredate))/12)>=2 order by round((months\_between(sysdate,hiredate))/12);

**OUTPUT:**



**5.LIST EMPLOYEE NAMES WITH ALL CAPITAL LETTERS, WITH ALL SMALL LETTERS AND WITH FIRST LETTER ONLY AS CAPITAL.**

**A. SQL>** select upper(ename),lower(ename),initcap(ename) from emp;

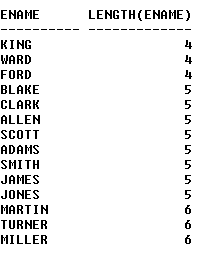
**OUTPUT:**



**6.LIST EMPLOYEE NAMES WITH LENGTH OF THE NAME SORTEDON LENGTH.**

**A.** **SQL>** select ename,length(ename) from emp order by length(ename);

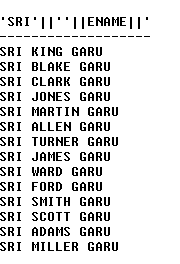
**OUTPUT:**



**7.LIST EMPLOYEE NAMES APPENDING SRI TO THE BEGINNING AND GARU TO THE END.**

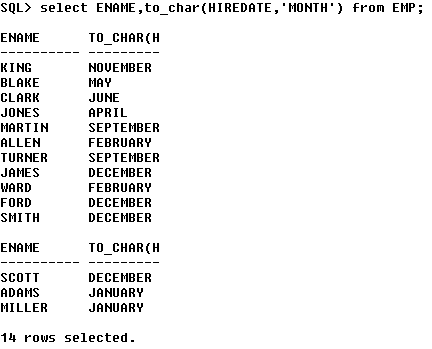
**A.** **SQL>** select 'SRI'||' '||ename||' '||'GARU' from emp;

**OUTPUT:**



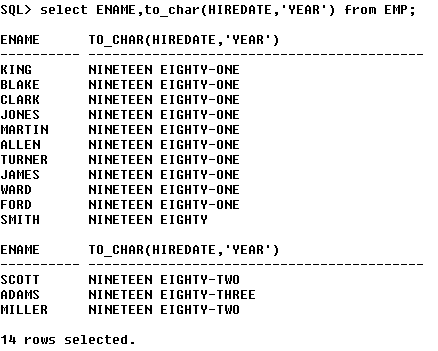
**8.LIST THE EMPLOYEE NAMES AND MONTH NAMES OF JOINING.**

**A.**



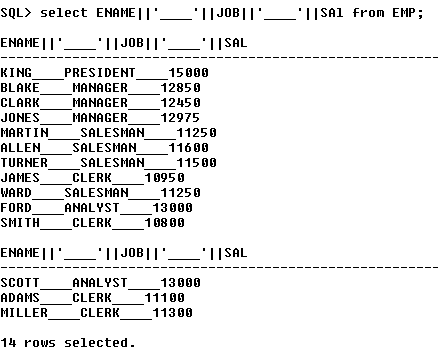
**9.LIST EMPLOYEE NAMES AND YEAR OF JOINING IN WORDS.**

**A.**

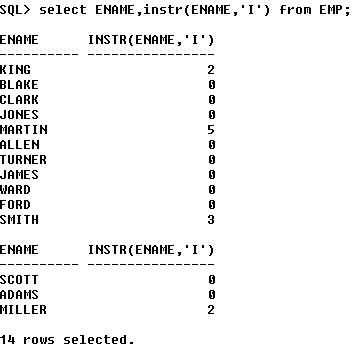


**10.LIST EMPLOYEE NAMES,JOB AND SALARY WITH 5 HYPHENS IN BETWEEN.**

**A.**

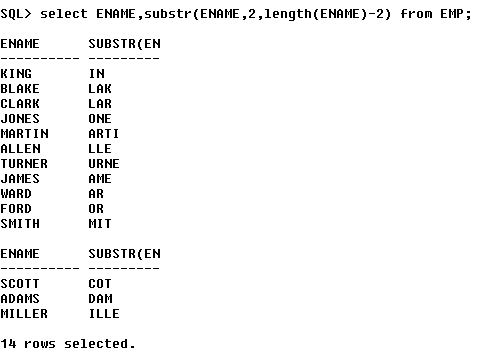


**11.LIST EMPLOYEE NAMES AND POSITION OF FIRST OCCURRENCE OF I IN THEIR NAMES.**



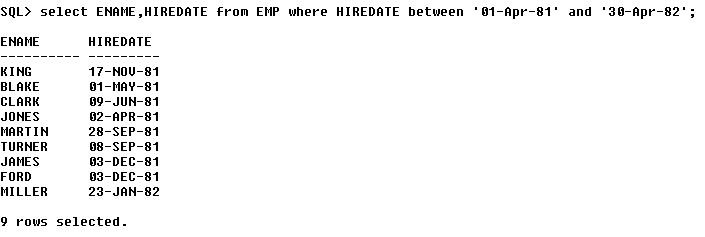
**12.LIST EMPLOYEE NAMES AND THE STRING WITHOUT FIRST CHARACTER AND LAST CHARACTER IN THEIR NAME.**

**A.**



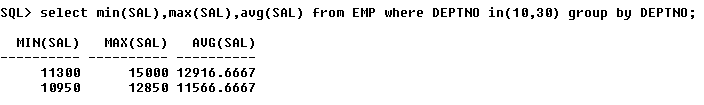
**13.LIST EMPLOYEES WHO JOINED BETWEEN APR 81 AND APR 82.**

**A.**



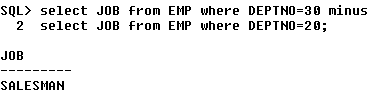
**14.LIST MAX SAL,MIN SAL,AVERAGE SAL OF DEPTS. 10,30.**

**A.**

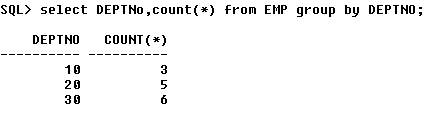


**15.LIST THE DESIGNATION IN DEPT 30 BUT NOT IN 20.**

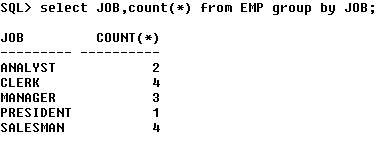
**A.**

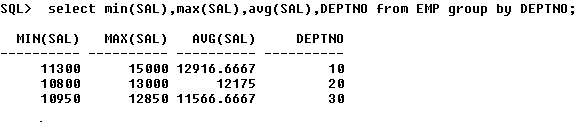


**16.LIST THE NUMBER OF EMPLOYEES IN EACH DEPARTMENT ALONG WITH DEPT NUMBERS.**

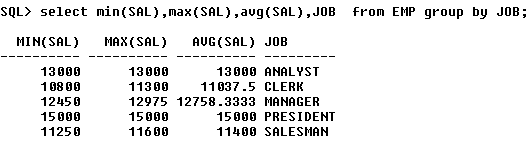


**17.LIST THE NUMBER OF EMPLOYEES JOINED YEAR WISE.**

 **19.LIST MAX SAL,MIN SAL,AVERAGE SAL DEPT WISE.**



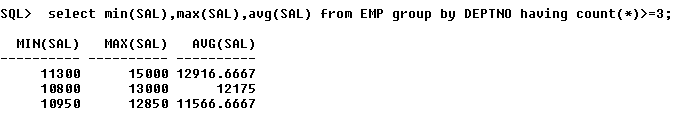
**20. LIST MAX SAL,MIN SAL,AVERAGE SAL JOB WISE.**



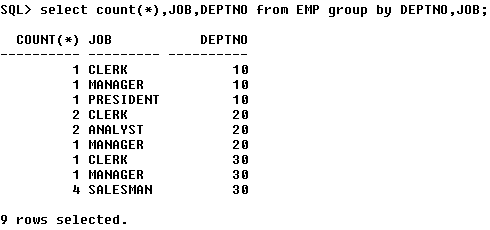
**21.LIST MAX SAL,MIN SAL FOR THE JOBS MANAGER AND CLERK.**



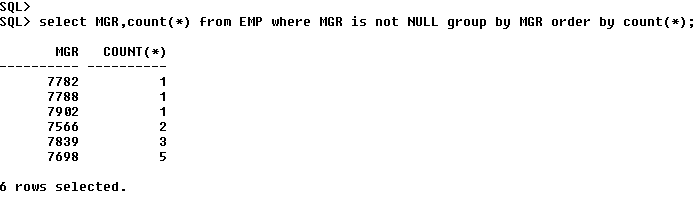
**22.LIST MAX SAL,MIN SAL,AVERAGE SAL OF THE DEPTS. HAVING A MINIMUM 3 EMPLOYEES.**



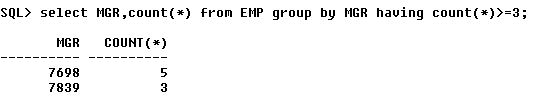
**23.LIST THE NUMBER OF EMPLOYEES IN EACH JOB AND EACH DEPARTMENT.**



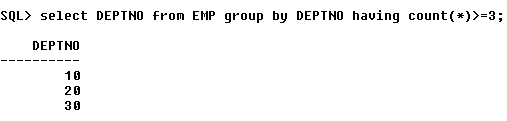
**24.LIST MGR AND THE NUMBER OF EMPLOYEES REPORT TO THEM IN THE SORTED ORDER.**



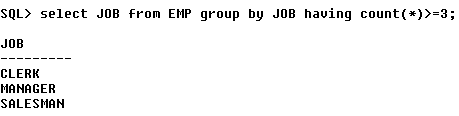
**25.LIST EMP NUMBER OF EMPLOYEES TO WHOM A MINIMUM OF 3 PEOPLE REPORT.**



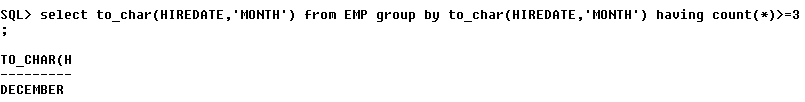
**26.LIST DEPT NUMBERS HAVING A MNIMUM OF 3 PERSONS.**



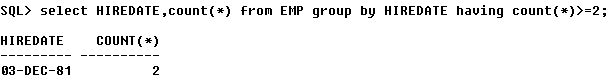
**27.LIST NUMBER OF JOBS HAVING A MINIMUM OF 3 PERSONS IN THAT JOB.**



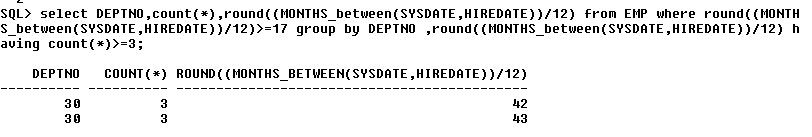
**28.LIST NAMES OF MONTHS IN WHICH A MINIMUM OF 3 PERSONS JOINED.**



**29.LIST HIREDATES OF EMPLOYEES HAVING 2 OR MORE EMPLOYEES HAVING THE SAME HIREDATE.**



**30.LIST DEPARTMENTS HAVING MINIMUM OF 3 PEOPLE HAVING A MINIMUM OF 13 YEARS OF EXPERIENCE.**

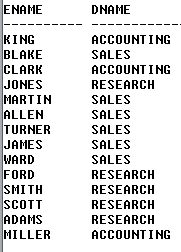


**WORKSHEET 3**

**1)List the employees names and dept names with which hey are associated.**

**SQL>** select ename,dname from emp,dept where emp.deptno=dept.deptno;

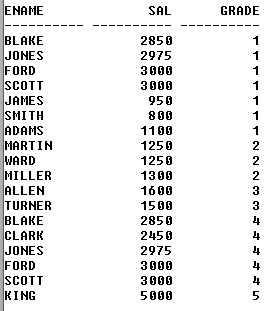
**Output:**



**2)List employee names, salary and their grade.**

**SQL>**: select ename,sal,grade from emp,salgrade where sal between losal and hisal;

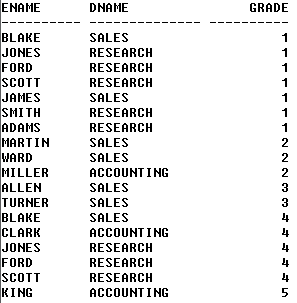
**Output:**



**3)** **List employee name, dept name along with grade.**

**SQL>:** select ename,dname,grade from emp,dept,salgrade where emp.deptno=dept.deptno and sal between losal and hisal;

**Output:**



**4)** **List employee names and their manager names.**

**SQL>**: select e1.ename,e2.ename from emp e1,emp e2 where e1.mgr=e2.empno;

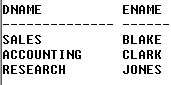
**Output:**



**5)** **List dept name and Manager name.**

**SQL>**: select dname,ename from emp,dept where emp.deptno=dept.deptnoandjob='MANAGER';

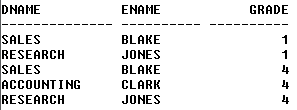
**Output:**



**6)** **List managers of various depts.. Along with grade sorted on grade.**

SQL>: select dname,ename,grade from emp,dept,salgrade where emp.deptno=dept.deptno and job='MANAGER' and sal between losaland hisal order by grade;

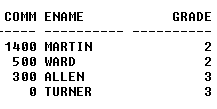
**Output:**



**7) List employees having commission along with grade.**

**SQL>:** select comm,ename,grade from emp,salgrade where sal between losal and hisal and comm is not null;

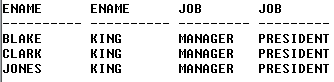
**Output:**



**8) List employees names with job manager along their manager names to whom they have to report.**

**SQL>**: select e1.ename,e2.ename,e1.job,e2.job from emp e1,emp e2 wheree1.mgr=e2.empno and e1.job='MANAGER';

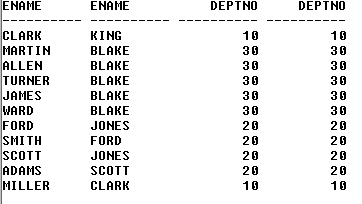
**Output:**



**9) List names of employees who are working in the same dept of their manager.**

**SQL>**: select e1.ename,e2.ename,e1.deptno,e2.deptno from emp e1,emp e2 where e1.mgr=e2.empno and e1.deptno=e2.deptno;

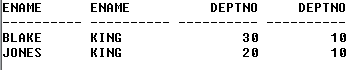
**Output:**



**10) List names of employees who are not working in the same dept of their manager.**

**SQL>:** select e1.ename,e2.ename,e1.deptno,e2.deptno from emp e1,emp e2 where e1.mgr=e2.empno and e1.deptno!=e2.deptno;

**Output:**



**11) List names of employees having first character in their name first character in their dept name same.**

**SQL>**: select ename,dname from emp,dept where emp.deptno=dept.deptno and substr(ename,1,1)=substr(dname,1,1);

**Output:**



**12) List employees who joined in the present month in any year and having grade and last digit in the year are same.**

**SQL>**: select ename,to\_char(hiredate,'month'),grade from emp,salgrade where sal between losal and hisal and to\_char(hiredate,'month')=to\_char(sysdate,'month') and to\_char (hiredate,'Y')=grade;

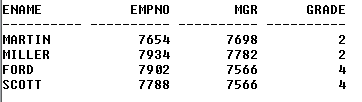
**Output:**



**13) List names of employees whose empno, mgr and grade given the same remainder when divided by 2.**

**SQL>:** select ename,empno,mgr,grade from emp,salgrade where sal between losal and hisal and mod(mgr,2)=mod(empno,2) and mod(mgr,2)=mod(grade,2);

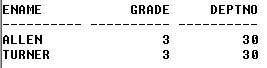
**Output:**



**14) List the names of employees having grade and tens position in the deptno same.**

**SQL>:** select ename,grade,deptno from emp,salgrade where sal between losal and hisal and grade=substr(deptno,1,1);

**Output:**



**15) List the names of employees having grade and tens position in the deptno same.**

**SQL>:** select ename,grade,deptno from emp,salgrade where sal between losal and hisal and grade=s**ubstr(deptno,1,1);**

**Output:**



**16) List employee name, deptname and dept location of those employees having any of these three same length.**

**SQL>:** select ename,dname,loc from emp,dept where emp.deptno=dept.deptno and length(ename)=length(dname) and length(dname)=length(loc);

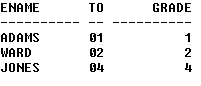
**Output:**



**17) List names of employees having month number of hiredate and grade same.**

**SQL>:** select ename,to\_char(hiredate,'MM'),grade from emp,salgrade where sal between losal and hisal and to\_char(hiredate,'MM')=grade;

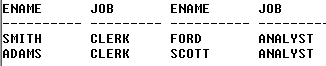
**Output:**



**18) List names of clerks who are reporting to analyst.**

**SQL>:** select e1.ename,e1.job,e2.ename,e2.job from emp e1,emp e2 where e1.mgr=e2.empno and e1.job='CLERK' and e2.job='ANALYST';

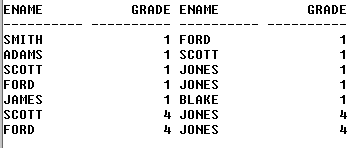
**Output:**



**19) List emp names and their manager names having same grade.**

**SQL>** select e1.ename,s1.grade,e2.ename,s2.grade from emp e1,emp e2, salgrade s1,salgrade s2 where e1.mgr=e2.empno and e1.sal between s1.losal and s1.hisal and e2.sal between s2.losal and s2.hisal and s1.grade=s2.grade;

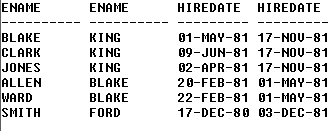
**Output:**



**20) List emp names of employees who joined before their manager’s joining date.**

**SQL>:** select e1.ename,e2.ename,e1.hiredate,e2.hiredate from emp e1,emp e2 where e1.mgr=e2.empno and e1.hiredate<=e2.hiredate;

**Output:**



**CREARTING TABLES**

**1. EMPLOYEE( FNAME, MlNIT, LNAME, SSN, SEX, SALARY, SUPERSSNIDNO)**

**CONSTRAINTS: FNAME, LNAME, SSN, DNO NOT NULL**

**PRIMARY KEY(SSN)**

**FOREIGN KEY (SUPERSSN)**

**REFERENCES EMPLOYEE(SSN) FOREIGN KEY(DNO)**

**REFERENCES DEPARTMENT(DNUMBER)**

create table employ (FNAME character(10) not null,MINIT character(5) not null,LNAME character(10) not null,SSN number(4) not null primary key,SEX character(3) not null,SALARY numbnot null,SUPERSSN number(4), DNO number(1) not null);

Table created.

insert into employ values('&fname','&minit','&lname',&ssn,'&sex',&salary,&superssnn,&dno)

Enter value for fname: JENNIFER

Enter value for minit: S

Enter value for lname: WALLACE

Enter value for ssn: 8765

Enter value for sex: F

Enter value for salary: 43000

Enter value for superssn: 8866

Enter value for dno: 4

old 2: values('&fname','&minit','&lname',&ssn,'&sex',&salary,&superssn,&dno)

new 2: values('JENNIFER','S','WALLACE',8765,'F',43000,8866,4)

1 row created.

insert into employ values('&fname','&minit','&lname',&ssn,'&sex',&salary,&superssn,&dno)

Enter value for fname: JENNIFER

Enter value for minit: S

Enter value for lname: WALLACE

Enter value for ssn: 8765

Enter value for sex: F

Enter value for salary: 43000

Enter value for superssn: 8866

Enter value for dno: 4

old 2: values('&fname','&minit','&lname',&ssn,'&sex',&salary,&superssn,&dno)

new 2: values('JENNIFER','S','WALLACE',8765,'F',43000,8866,4)

1 row created.

insert into employ values('&fname','&minit','&lname',&ssn,'&sex',&salary,&superssn,&dno)

Enter value for fname: ALICIA

Enter value for minit: J

Enter value for lname: ZELAYA

Enter value for ssn: 9988

Enter value for sex: F

Enter value for salary: 25000

Enter value for superssn: 8765

Enter value for dno: 4

old 2: values('&fname','&minit','&lname',&ssn,'&sex',&salary,&superssn,&dno)

new 2: values('ALICIA','J','ZELAYA',9988,'F',25000,8765,4)

1 row created.

insert into employ values('&fname','&minit','&lname',&ssn,'&sex',&salary,&superssn,&dno)

Enter value for fname: RAMESH

Enter value for minit: K

Enter value for lname: NARAYAN

Enter value for ssn: 6688

Enter value for sex: M

Enter value for salary: 38000

Enter value for superssn: 3344

Enter value for dno: 5

old 2: values('&fname','&minit','&lname',&ssn,'&sex',&salary,&superssn,&dno)

new 2: values('RAMESH','K','NARAYAN',6688,'M',38000,3344,5)

1 row created.

insert into employ values('&fname','&minit','&lname',&ssn,'&sex',&salary,&superssn,&dno)

Enter value for fname: JAMES

Enter value for minit: E

Enter value for lname: BORG

Enter value for ssn: 8866

Enter value for sex: M

Enter value for salary: 55000

Enter value for superssn: NULL

Enter value for dno: 1

old 2: values('&fname','&minit','&lname',&ssn,'&sex',&salary,&superssn,&dno)

new 2: values('JAMES','E','BORG',8866,'M',55000,NULL,1)

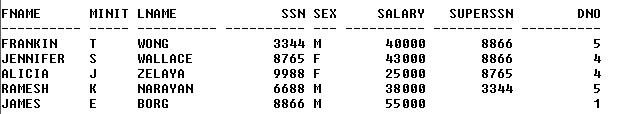
1 row created.

alter table employ add constraint employ\_SUPERSSN\_FK foreign key(SUPERSSN) references employ(ssn);

Table altered.

select \* from employ;

**Output:**



**2.DEPARTMENT(DNAME, DNUMBER, MGRSSN)**

**CONSTRAINTS: DNAME, DNUMBER, MGRSSN NOTNULL**

**PRIMARY KEY (DNUMBER)**

**UNIQUE (DNAME)**

**FOREIGN KEY(MGRSSN) REFERENCES EMPLOYEE(SSN)**

Query:

create table department(DNAME character(15) not null unique,DNUMBER number(1) not null,MGRSSN number(4) not null;

Table created.

INSERT INTO DEPARTMENT VALUES('&DNAME',&DNUMBER,&MGRSSN);

Enter value for dname: RESEARCH

Enter value for dnumber: 5

Enter value for mgrssn: 3344

old 1: INSERT INTO DEPARTMENT VALUES('&DNAME',&DNUMBER,&MGRSSN)

new 1: INSERT INTO DEPARTMENT VALUES('RESEARCH',5,3344)

1 row created.

INSERT INTO DEPARTMENT VALUES('&DNAME',&DNUMBER,&MGRSSN);

Enter value for dname: ADMINISTRATION

Enter value for dnumber: 4

Enter value for mgrssn: 8765

old 1: INSERT INTO DEPARTMENT VALUES('&DNAME',&DNUMBER,&MGRSSN)

new 1: INSERT INTO DEPARTMENT VALUES('ADMINISTRATION',4,8765)

1 row created.

INSERT INTO DEPARTMENT VALUES('&DNAME',&DNUMBER,&MGRSSN);

Enter value for dname: HEADQUATERS

Enter value for dnumber: 1

Enter value for mgrssn: 8866

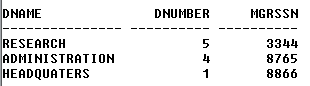
old 1: INSERT INTO DEPARTMENT VALUES('&DNAME',&DNUMBER,&MGRSSN)

new 1: INSERT INTO DEPARTMENT VALUES('HEADQUATERS',1,8866)

1 row created

SELECT \* FROM DEPARTMENT;

**Output:**

. 

**3.DEPT\_LOCATIONS(DNUMBER,DLOCATION)**

**CONSTRAINTS: DNUMBER.DLOCATION NOTNULL**

**PRIMARY KEY(DNUMBER,DLOCATION)**

**FOREIGN KEY(DNUMBER) REFERENCES DEPARTMENT(DNUMBER)**

**Query:**

create table dept\_locations(DNUMBER number(1) not null,DLOCATION character(10) not null)

Table created.

insert into dept\_locations values(&DNUMBER,'&DLOCATION');

Enter value for dnumber: 1

Enter value for dlocation: HOUSTON

old 1: insert into dept\_locations values(&DNUMBER,'&DLOCATION')

new 1: insert into dept\_locations values(1,'HOUSTON')

1 row created.

insert into dept\_locations values(&DNUMBER,'&DLOCATION');

Enter value for dnumber: 4

Enter value for dlocation: STAFORD

old 1: insert into dept\_locations values(&DNUMBER,'&DLOCATION')

new 1: insert into dept\_locations values(4,'STAFORD')

1 row created.

insert into dept\_locations values(&DNUMBER,'&DLOCATION');

Enter value for dnumber: 5

Enter value for dlocation: BEELLARIE

old 1: insert into dept\_locations values(&DNUMBER,'&DLOCATION')

new 1: insert into dept\_locations values(5,'BEELLARIE')

1 row created.

insert into dept\_locations values(&DNUMBER,'&DLOCATION');

Enter value for dnumber: 5

Enter value for dlocation: HOUSTON

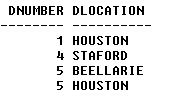
old 1: insert into dept\_locations values(&DNUMBER,'&DLOCATION')

new 1: insert into dept\_locations values(5,'HOUSTON')

1 row created.

select \* from dept\_locations;

**Output:**



**4. PROJECT(PNAME,PNUMBER,PLOCATIOIM,DNUM)**

**CONSTRAINTS: PNAME.PNUMBER.DNUM NOTNULL**

**PRIMARY KEY(PNUMBER) UNIQUE(PNAME)**

**FOREIGN KEY(DNUM) REFERENCES DEPARTMENT(DNUMBER)**

**Query:**

create table proj(PNAME character(15) not null unique,PNUMBER number(2) not null primary key,PLOCATION character(10) not null,DNUM number(1) not null)

Table created.

insert into proj values('&PNAME',&PNUMBER,'&PLOCATION',&DNUM);

Enter value for pname: PRODUCT\_Y

Enter value for pnumber: 2

Enter value for plocation: SUGARLAND

Enter value for dnum: 5

old 1: insert into proj values('&PNAME',&PNUMBER,'&PLOCATION',&DNUM)

new 1: insert into proj values('PRODUCT\_Y',2,'SUGARLAND',5)

1 row created.

insert into proj values('&PNAME',&PNUMBER,'&PLOCATION',&DNUM);

Enter value for pname: PRODUCT\_X

Enter value for pnumber: 1

Enter value for plocation: BELLARIE

Enter value for dnum: 5

old 1: insert into proj values('&PNAME',&PNUMBER,'&PLOCATION',&DNUM)

new 1: insert into proj values('PRODUCT\_X',1,'BELLARIE',5)

1 row created.

insert into proj values('&PNAME',&PNUMBER,'&PLOCATION',&DNUM);

Enter value for pname: PRODUCT\_Z

Enter value for pnumber: 3

Enter value for plocation: HOUSTON

Enter value for dnum: 5

old 1: insert into proj values('&PNAME',&PNUMBER,'&PLOCATION',&DNUM);

new 1: insert into proj values('PRODUCT\_Z',3,'HOUSTON',5)

1 row created.

insert into proj values('&PNAME',&PNUMBER,'&PLOCATION',&DNUM);

Enter value for pname: COMPUTERIZATON

Enter value for pnumber: 10

Enter value for plocation: STAFORD

Enter value for dnum: 4

old 1: insert into proj values('&PNAME',&PNUMBER,'&PLOCATION',&DNUM)

new 1: insert into proj values('COMPUTERIZATON',10,'STAFORD',4)

1 row created.

insert into proj values('&PNAME',&PNUMBER,'&PLOCATION',&DNUM);

Enter value for pname: REORGANIZATION

Enter value for pnumber: 20

Enter value for plocation: HOUSTON

Enter value for dnum: 1

old 1: insert into proj values('&PNAME',&PNUMBER,'&PLOCATION',&DNUM)

new 1: insert into proj values('REORGANIZATION',20,'HOUSTON',1)

1 row created.

insert into proj values('&PNAME',&PNUMBER,'&PLOCATION',&DNUM);

Enter value for pname: NEWBENEFITS

Enter value for pnumber: 30

Enter value for plocation: STAFORD

Enter value for dnum: 4

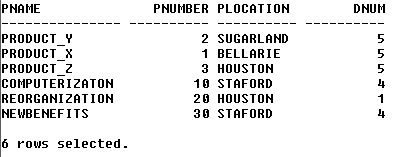
old 1: insert into proj values('&PNAME',&PNUMBER,'&PLOCATION',&DNUM)

new 1: insert into proj values('NEWBENEFITS',30,'STAFORD',4)

1 row created.

SELECT \* FROM PROJ;

**Output:**



**5. WORKS\_ON(ESSN,PNO,HOURS)**

**CONSTRAINTS: ESSN,PNO NOTNULL**

**PRIMARY KEY(ESSN,PNO)**

**FOREIGN KEY(ESSN) REFERENCES EMPLOYEE(SSN) FOREIGN KEY(PNO) REFERENCES PROJECT(PNUMBER)**

**Query:**

create table work\_on(ESSN number(4) not null,PNO number(2) not null,HOURS number(5));

SQL> insert into work\_on values(&ESSN,&PNO,&HOURS);

Enter value for essn: 2345

Enter value for pno: 1

Enter value for hours: 32.5

old 1: insert into work\_on values(&ESSN,&PNO,&HOURS)

new 1: insert into work\_on values(2345,1,32.5)

1 row created.

SQL> insert into work\_on values(&ESSN,&PNO,&HOURS);

Enter value for essn: 2345

Enter value for pno: 2

Enter value for hours: 7.5

old 1: insert into work\_on values(&ESSN,&PNO,&HOURS)

new 1: insert into work\_on values(2345,2,7.5)

1 row created.

SQL> insert into work\_on values(&ESSN,&PNO,&HOURS);

Enter value for essn: 6688

Enter value for pno: 3

Enter value for hours: 40

old 1: insert into work\_on values(&ESSN,&PNO,&HOURS)

new 1: insert into work\_on values(6688,3,40)

1 row created.

SQL> insert into work\_on values(&ESSN,&PNO,&HOURS);

Enter value for essn: 5345

Enter value for pno: 1

Enter value for hours: 20

old 1: insert into work\_on values(&ESSN,&PNO,&HOURS)

new 1: insert into work\_on values(5345,1,20)

1 row created.

SQL> insert into work\_on values(&ESSN,&PNO,&HOURS);

Enter value for essn: 5345

Enter value for pno: 2

Enter value for hours: 20

old 1: insert into work\_on values(&ESSN,&PNO,&HOURS)

new 1: insert into work\_on values(5345,2,20)

1 row created.

SQL> insert into work\_on values(&ESSN,&PNO,&HOURS);

Enter value for essn: 3344

Enter value for pno: 2

Enter value for hours: 10

old 1: insert into work\_on values(&ESSN,&PNO,&HOURS)

new 1: insert into work\_on values(3344,2,10)

1 row created.

SQL> insert into work\_on values(&ESSN,&PNO,&HOURS);

Enter value for essn: 3344

Enter value for pno: 3

Enter value for hours: 10

old 1: insert into work\_on values(&ESSN,&PNO,&HOURS)

new 1: insert into work\_on values(3344,3,10)

1 row created.

SQL> insert into work\_on values(&ESSN,&PNO,&HOURS);

Enter value for essn: 3344

Enter value for pno: 10

Enter value for hours: 10

old 1: insert into work\_on values(&ESSN,&PNO,&HOURS)

new 1: insert into work\_on values(3344,10,10)

1 row created.

SQL> insert into work\_on values(&ESSN,&PNO,&HOURS);

Enter value for essn: 3344

Enter value for pno: 20

Enter value for hours: 10

old 1: insert into work\_on values(&ESSN,&PNO,&HOURS)

new 1: insert into work\_on values(3344,20,10)

1 row created.

SQL> insert into work\_on values(&ESSN,&PNO,&HOURS);

Enter value for essn: 9988

Enter value for pno: 30

Enter value for hours: 30

old 1: insert into work\_on values(&ESSN,&PNO,&HOURS)

new 1: insert into work\_on values(9988,30,30)

1 row created.

SQL> insert into work\_on values(&ESSN,&PNO,&HOURS);

Enter value for essn: 9988

Enter value for pno: 10

Enter value for hours: 10

old 1: insert into work\_on values(&ESSN,&PNO,&HOURS)

new 1: insert into work\_on values(9988,10,10)

1 row created.

SQL> insert into work\_on values(&ESSN,&PNO,&HOURS);

Enter value for essn: 8798

Enter value for pno: 10

Enter value for hours: 35

old 1: insert into work\_on values(&ESSN,&PNO,&HOURS)

new 1: insert into work\_on values(8798,10,35)

1 row created.

SQL> insert into work\_on values(&ESSN,&PNO,&HOURS);

Enter value for essn: 8798

Enter value for pno: 20

Enter value for hours: 5

old 1: insert into work\_on values(&ESSN,&PNO,&HOURS)

new 1: insert into work\_on values(8798,20,5)

1 row created.

SQL> insert into work\_on values(&ESSN,&PNO,&HOURS);

Enter value for essn: 8765

Enter value for pno: 30

Enter value for hours: 15

old 1: insert into work\_on values(&ESSN,&PNO,&HOURS)

new 1: insert into work\_on values(8765,30,15)

1 row created.

SQL> insert into work\_on values(&ESSN,&PNO,&HOURS);

Enter value for essn: 8866

Enter value for pno: 30

Enter value for hours: NULL

old 1: insert into work\_on values(&ESSN,&PNO,&HOURS)

new 1: insert into work\_on values(8866,30,NULL)

1 row created.

SQL> insert into work\_on values(&ESSN,&PNO,&HOURS;

Enter value for essn: 8866

Enter value for pno: 1

Enter value for hours: NULL

old 1: insert into work\_on values(&ESSN,&PNO,&HOURS

new 1: insert into work\_on values(8866,1,NULL

insert into work\_on values(8866,1,NULL

\*

ERROR at line 1:

ORA-00917: missing comma

SQL> insert into work\_on values(&ESSN,&PNO,&HOURS);

Enter value for essn: 8866

Enter value for pno: 1

Enter value for hours: NULL

old 1: insert into work\_on values(&ESSN,&PNO,&HOURS)

new 1: insert into work\_on values(8866,1,NULL)

1 row created.

SQL> insert into work\_on values(&ESSN,&PNO,&HOURS);

Enter value for essn: 8866

Enter value for pno: 30

Enter value for hours: NULL

old 1: insert into work\_on values(&ESSN,&PNO,&HOURS)

new 1: insert into work\_on values(8866,30,NULL)

1 row created.

SQL> insert into work\_on values(&ESSN,&PNO,&HOURS);

Enter value for essn: 8866

Enter value for pno: 1

Enter value for hours: NULL

old 1: insert into work\_on values(&ESSN,&PNO,&HOURS)

new 1: insert into work\_on values(8866,1,NULL)

1 row created.

SQL> SELECT \* FROM WORK\_ON;

ESSN PNO HOURS

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

2345 1 33

2345 2 8

6688 3 40

5345 1 20

5345 2 20

3344 2 10

3344 3 10

3344 10 10

3344 20 10

9988 30 30

9988 10 10

ESSN PNO HOURS

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

8798 10 35

8798 20 5

8765 30 15

8866 30

8866 1

8866 30

8866 1

18 rows selected.

SQL> SET PAGESIZE 100;

SQL> SET LINE SIZE 100;

SP2-0268: linesize option not a valid number

SQL> SE LINESIZE 100;

SP2-0734: unknown command beginning "SE LINESIZ..." - rest of line ignor

SQL> SET LINESIZE 100;

SELECT \* FROM WORK\_ON;

ESSN PNO HOURS

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

2345 1 33

2345 2 8

6688 3 40

5345 1 20

5345 2 20

3344 2 10

3344 3 10

3344 10 10

3344 20 10

9988 30 30

9988 10 10

8798 10 35

8798 20 5

8765 30 15

8866 30

8866 1

8866 30

8866 1

**6. DEPENDENT (ESSN,D\_NAME,SEX,RELATIONSHIP)**

**CONSTRAINTS:**

**ESSN,D\_NAME NOTNULL**

**PRIMARY KEY(ESSN,D\_NAME)**

**Query:**

SQL> create table dependent

2 (ESSN number(4) not null,D\_NAME character(15) not null,SEX

3 character(3),RELATIONSHIP character(15));

Table created.

1 insert into dependent

2\* values(&ESSN,'&D\_NAME','&SEX','&RELATIONSHIP')

3 /

Enter value for essn: 3344

Enter value for d\_name: ALICE

Enter value for sex: F

Enter value for relationship: DAUGHTER

old 2: values(&ESSN,'&D\_NAME','&SEX','&RELATIONSHIP')

new 2: values(3344,'ALICE','F','DAUGHTER')

1 row created.

SQL> insert into dependent

2 values(&ESSN,'&D\_NAME','&SEX','&RELATIONSHIP')

3 /

Enter value for essn: 3344

Enter value for d\_name: THEODORE

Enter value for sex: M

Enter value for relationship: SON

old 2: values(&ESSN,'&D\_NAME','&SEX','&RELATIONSHIP')

new 2: values(3344,'THEODORE','M','SON')

1 row created.

SQL> insert into dependent

2 values(&ESSN,'&D\_NAME','&SEX','&RELATIONSHIP')

3 /

Enter value for essn: 3344

Enter value for d\_name: JOY

Enter value for sex: F

Enter value for relationship: SPOUSE

old 2: values(&ESSN,'&D\_NAME','&SEX','&RELATIONSHIP')

new 2: values(3344,'JOY','F','SPOUSE')

1 row created.

SQL> insert into dependent

2 values(&ESSN,'&D\_NAME','&SEX','&RELATIONSHIP')

3 /

Enter value for essn: 8765

Enter value for d\_name: ABNER

Enter value for sex: M

Enter value for relationship: SPOUSE

old 2: values(&ESSN,'&D\_NAME','&SEX','&RELATIONSHIP')

new 2: values(8765,'ABNER','M','SPOUSE')

1 row created.

SQL> insert into dependent

2 values(&ESSN,'&D\_NAME','&SEX','&RELATIONSHIP')

3 /

Enter value for essn: 2345

Enter value for d\_name: MICHAEL

Enter value for sex: M

Enter value for relationship: SON

old 2: values(&ESSN,'&D\_NAME','&SEX','&RELATIONSHIP')

new 2: values(2345,'MICHAEL','M','SON')

1 row created.

SQL> insert into dependent

2 values(&ESSN,'&D\_NAME','&SEX','&RELATIONSHIP')

3 /

Enter value for essn: 2345

Enter value for d\_name: ALICE

Enter value for sex: F

Enter value for relationship: DAUGHTER

old 2: values(&ESSN,'&D\_NAME','&SEX','&RELATIONSHIP')

new 2: values(2345,'ALICE','F','DAUGHTER')

1 row created.

SQL> insert into dependent

2 values(&ESSN,'&D\_NAME','&SEX','&RELATIONSHIP')

3 /

Enter value for essn: 2345

Enter value for d\_name: ELIZABETH

Enter value for sex: F

Enter value for relationship: SPOUSE

old 2: values(&ESSN,'&D\_NAME','&SEX','&RELATIONSHIP')

new 2: values(2345,'ELIZABETH','F','SPOUSE')

1 row created.

select \* from dependent;



**PL/SQL PROGRAMS USING CONDITIONAL LOOPING STATEMENTS**

**1.Write a PL/SQL program to print employee number of an employee as well as the corresponding MGR NO.**

declare

e1 emp.empno%type;

e2 emp.mgr%type;

begin

e1:=&empno;

select mgr into e2 from emp where empno=e1;

dbms\_output.put\_line('empno:'||e1 ||'mgr:'||e2);

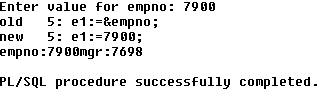
exception

when no\_data\_found then

dbms\_output.put\_line('wrong input');

end;

**Output:**



**2.Write a PL/SQL program using FOR/WHILE LOOPS to list out month names and month numbers.**

**Program:**

**Program using FOR LOOP**

declare

d date;

i number;

begin

for i in 0..11

loop

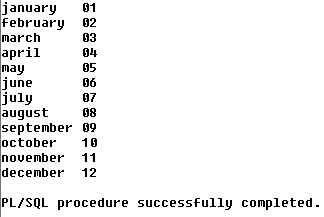
select add\_months(to\_date('01-jan-05'),i) into d from dual;

dbms\_output.put\_line(to\_char(d,'month')||' '|| to\_char(d,'mm'));

end loop;

end;

**Output:**



**Program using While loop:**

Declare

d date;

i number;

begin

i:=0;

while i<=11

loop

select add\_months(to\_date('01-jan-05'),i)into d from dual;

dbms\_output.put\_line(to\_char(d,'month')||' '||to\_char(d,'mm'));

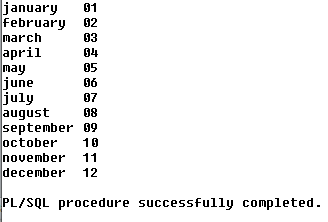
i:=i+1;

end loop;

end;

/

**Output:**



**3. Write a PL/SQL program to update commission of an employee (employee number as input) as per the following norms.**

**i) If commission is NULL, make it as 10% of salary**

**ii) If comm. < 200 make comm. = 200**

**iii) If comm. <300 make comm. =300**

**Program:**

declare

c1 emp.comm%type;

s1 emp.sal%type;

e1 emp.empno%type;

begin

e1:=&e1;

select comm,sal into c1,s1 from emp where empno=e1;

if c1 is null then

c1:=s1\*(0.1);

elsif c1>200 and c1<300 then

c1:=200;

else

c1:=c1+c1\*(0.1);

end if;

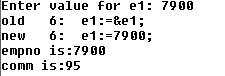
dbms\_output.put\_line('empno is:'||e1);

dbms\_output.put\_line('comm is:'||c1);

end;

/

**Output:**



**4. Write a PL/SQL program to list out .**

**DEPT NO,DNAME,NO OF EMPLOYEES, MAX(SAL),MIN(SAL), AVG(SAL)**

**In each dept. If a dept has no employees then display“employees are not there in this dept”.**

**Program:**

declare

d1 emp.deptno%type;

dn dept.dname%type;

cn number;

mi number;

mx number;

ag number(10,4);

cnt exception;

begin

d1:=&d1;

select dname into dn from dept where deptno=d1;

select count(empno),min(sal),max(sal),avg(sal) into

cn,mi,mx,ag from emp where deptno=d1;

if cn=0 then

raise cnt;

else

dbms\_output.put\_line('deptno:'||d1);

dbms\_output.put\_line('dname:'||dn);

dbms\_output.put\_line('no of employees:'||cn);

dbms\_output.put\_line('min sal:'||mi);

dbms\_output.put\_line('max sal:'||mx);

dbms\_output.put\_line('avg sal:'||ag);

end if;

exception

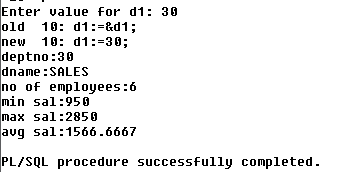
when cnt then

dbms\_output.put\_line('there is no employees in that dept');

end;

/

**Output:**



**5. write a PL/SQL program to get no.of employees whose salary is in between given range.** **Program:**

declare

losal emp.sal%type;

hisal emp.sal%type;

count1 number;

begin

losal:=&losal;

hisal:=&hisal;

select count(empno) into count1 from emp

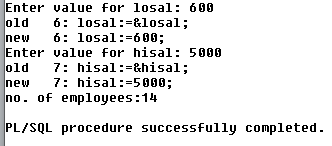
where sal between losal and hisal;

dbms\_output.put\_line('no. of employees:'|| count1);

end;

/

**Output:**



**FUNCTIONS**

**1. Write a program to check whether the given number is Prime or not.**

create or replace function isprime(x in number)

RETURN number

IS

i int;

cnt int;

BEGIN

cnt:=0;

for i in 2..x/2 LOOP

if mod(x,i)=0 then

cnt:=cnt+1;

end if;

end loop;

return cnt;

end;

/

**Output:**



declare

num number;

cnt number;

begin

num:=&num;

cnt:=isprime(num);

if cnt>=0 then

dbms\_output.put\_line(num||' '||'is a primenumber');

else

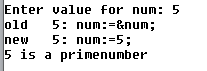
dbms\_output.put\_line(num|| ' '||'is not prime');

end if;

end;

/

**Output:**



**2. Create a Procedure to accept an Empno, and a salary increase amount, if Empno is not found or currentsalary is NULL then raise exceptions otherwise display total salary.**

create or replace function emps(e1 emp.empno%type,in1

emp.sal%type)

return number is

s1 emp.sal%type;

nusal exception;

nsal emp.sal%type;

begin

select sal into s1 from emp where empno=e1;

if s1 is null then

raise nusal;

else

nsal:=s1+in1;

end if;

return nsal;

exception

when nusal then

nsal:=1;

return nsal;

end emps;

/



declare

n number;

eno emp.empno%type;i

emp.sal%type;

begin

eno:=&eno;

i:=&i;

n:=emps(eno,i);

if n=1 then

dbms\_output.put\_line('emp. sal is null');

else

dbms\_output.put\_line(n);

end if;

exception

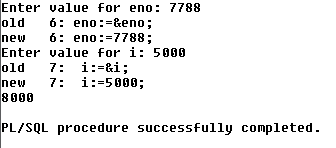
when no\_data\_found then

dbms\_output.put\_line('wrong empno');

end;

/

**Output:**



**3.Get the number of employees working under a given employee.**

create or replace function noe1(e1 emp.empno%type) return number as

cnt number;

begin

select count(empno) into cnt from emp where mgr=e1;

return cnt;

end noe1;

/

**Output:**

****

declare

n number;

e2 emp.empno%type;

en emp.ename%type;

nm exception;

begin

e2:=&e2;

n:=noe1(e2);

select ename into en from emp where empno=e2;

if n=0 then

raise nm;

else

dbms\_output.put\_line(e2||' '||en||' '||n);

end if;

exception

when nm then

dbms\_output.put\_line('he is not the manager');

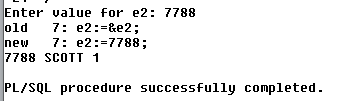
when no\_data\_found then dbms\_output.put\_line('wrong

empno');

end;

/

**Output:**



**4. Get the number of employees working in given department name.**

create or replace function noe(dn dept.dname%type) return number as

num number;

begin

select count(\*) into num from emp,dept where

emp.deptno=dept.deptno and dname=dn;

return num;

end noe;

/

**Output:**



declare

n dept.dname%type;

cnt number;

begin

n:='&n';

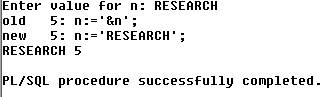
cnt:=noe(n);

dbms\_output.put\_line(n||' '||cnt);

end;

/

**Output:**



**PROCEDURES**

**1. Get the number of employees working under a given employee.**

create or replace procedure nos(e1 emp.empno%type) is cnt number;

begin

select count(\*) into cnt from emp where mgr=e1;

dbms\_output.put\_line(cnt);

exception

when no\_data\_found then

dbms\_output.put\_line('wrong empno');

end nos;

/



2. **Get the number of employees working in given department name.**

create or replace procedure nosd(d1 dept.dname%type) is cnt number;

begin

select count(\*) into cnt from emp,dept where

emp.deptno=dept.deptno and dname=d1;

dbms\_output.put\_line(cnt);

exception

when no\_data\_found then

dbms\_output.put\_line('wrong dept name');

end nosd;

/

Procedure created.