Table Name: PPROGRAMMER

Name	NULL?	Type	Remark
NAME	Not Null	Varchar2(8)	(Name)
DOB	Not Null	Date	(Date of Birth)
DOJ	Not Null	Date	(Date of Joining)
SEX	Not Null	Varchar2(1)	(Male or Female)
PROF1		Varchar2(8)	(Known Language 1)
PROF2		Varchar2(8)	(Known Language 2)
SALARY	Not Null	Number(4)	(Salary)

Table Name: SOFTWARE

Name	NULL?	Type	Remark
NAME	Not Null	Varchar2(8)	(Name)
TITLE	Not Null	Varchar2(20)	(Developed Project Name)
DEV_IN	Not Null	Varchar2(8)	(Language Developed)
SCOST		Number(7,2)	(Software Cost)
DCOST		Number(5)	(Development Cost)
SOLD		Number(3)	(Number of Software sold)

Table Name: STUDIES

NAME	NULL?	TYPE	REMARK
NAME	Not Null	Varchar2(8)	(Name)
SPLACE	Not Null	Varchar2(9	(Studied Place)
COURSE	Not Null	Varchar2(5)	(Course Studied)
CCOST	Not Null	Number(5)	(Course Cost)

Data in table: PROGRAMMER

NAME	DOB	DOJ	S	PROF1	PROF2	SALARY
ANAND	21-APR-66	21-APR-92	M	PASCAL	BASIC	3200
ALTAF	02-JUL-64	13-NOV-90	M	CLIPPER	COBOL	2800
JAGDESH	06-OCT-70	04-OCT-94	M	ORACLE	JAVA	4100
JULIANA	31-JAN-68	21-APR-90	F	COBOL	DBASE	3000
KAMALA	30-OCT-68	02-JAN-92	F	C	DBASE	2900
MARY	24-JUN-70	01-FEB-91	F	C++	ORACLE	4500
PATRICK	19-NOV-65	21-APR-90	M	PASCAL	CLIPPER	2800
QUADIR	31-AUG-65	21-APR-93	M	ASSEMBLY	C	3000
RAMESH	03-MAY-67	28-FEB-91	M	PASCAL	DBASE	3200
REBECCA	01-JAN-67	01-DEC-90	F	BASIC	COBOL	2500
REMITHA	19-APR-70	20-APR-93	F	C	ASSEMBLY	3600
REVATHI	02-DEC-69	02-JAN-92	F	PASCAL	BASIC	3700
VIJAYA	14-DEC-65	02-MAY-92	F	FOXPRO	C	3500

Data in table: SOFTWARE

NAME	TITLE	DEV_IN	SCOST	DCOST	SOLD
ANAND	PARACHUTES	BASIC	399.95	6000	43
ANAND	VIDEO TILING PACK	PASCAL	7500	16000	9
JAGDESH	SERIAL LINK UTILITY	JAVA	800	7500	9
JAGDESH	SHARES MANAGEMENT	ORACLE	3000	12000	14
JULIANA	INVENTORY CONTROL	COBOL	3000	3500	0
KAMALA	PAYROLL PACKAGE	DBASE	9000	20000	7
MARY	ACC S/W	ORACLE	18000	85000	4
MARY	CODE GENERATOR	C	4500	20000	23
MARY	READ ME	C++	300	1200	84
PATRICK	GRAPHIC EDITOR	PASCAL	750	5000	11

QUADIR	FILE UTILITY	ASSEMBLY	499.95	530	114
QUADIR	TALLY	C	1900	3400	21
RAMESH	HOTEL MANAGEMENT	DBASE	12000	35000	4
RAMESH	READ ME	PASCAL	99.95	4500	73
REMITHA	PC UTILITIES	C	725	5000	51
REMITHA	ISR HELP MANAGEMENT	ASSEMBLY	2500	6000	6
REVATHI	HOTEL MANAGEMENT	PASCAL	1100	75000	2
REVATHI	QUIZ MASTER	BASIC	3200	2100	15
VIJAYA	ISK EDITOR	C	900	700	6

Data in tables: STUDIES

NAME	SPLACE	COURS	CCOST
ALTAF	CCIT	DCA	7200
JAGDESH	SSIL	DCA	3500
JULIANA	BDPS	DCA	22000
KAMALA	PRAGATHI	DCP	5000
MARY	SABHARI	PGDCA	4500
NELSON	PRAGATHI	DAP	6200
PATRICK		DCAP	5200
QUADIR	APPLE	HDCP	14000
RAMESH	SABHARI	PGDCA	4500
REBECCA	BRILLIANT	DCA	11000
REMITHA	BDPS	DCS	6000
REVATHI	SABHARI	DAP	5000
VIJAYA	BDPS	DCA	48000
ANAND	SABHARI	PGDCA	4500

Write SQL queries for the following:
Use appropriate column names whenever necessary
Output should be sorted wherever applicable
Only relevant columns should be displayed when specific columns are not mentioned.

```
CREATE TABLE PROGRAMMER
(
 NAME          VARCHAR2(8) NOT NULL,
 DOB           DATE NOT NULL,
 DOJ           DATE NOT NULL,
 SEX           VARCHAR2(1) NOT NULL,
 PROF1         VARCHAR2(8),
 PROF2         VARCHAR2(8),
 SALARY        NUMBER(4) NOT NULL
);

CREATE TABLE SOFTWARE
(
 NAME          VARCHAR2(8) NOT NULL,
 TITLE         VARCHAR2(20) NOT NULL,
 DEV_IN        VARCHAR2(8) NOT NULL,
 SCOST         NUMBER(7,2),
 DCOST         NUMBER(5),
 SOLD          NUMBER(3)
);

CREATE TABLE STUDIES
(
 NAME          VARCHAR2(8) NOT NULL,
 SPLACE              VARCHAR2(9) NOT NULL,
 COURSE        VARCHAR2(5) NOT NULL,
 CCOST         NUMBER(5) NOT NULL
);
```

```
INSERT INTO PROGRAMMER
VALUES('ANAND',TO_DATE('21-APR-1966','DD-MON-YYYY'),TO_DATE('21-APR-1992','DD-MON-YYYY'),
'M','PASCAL','BASIC',3200);

INSERT INTO PROGRAMMER
VALUES('ALTAF',TO_DATE('02-JUL-1964','DD-MON-YYYY'),TO_DATE('13-NOV-1990','DD-MON-YYYY'),
'M','CLIPPER','COBOL',2800);

INSERT INTO PROGRAMMER
VALUES('JAGDESH',TO_DATE('06-OCT-1970','DD-MON-YYYY'),TO_DATE('04-OCT-1994','DD-MON-YYYY'),
'M','ORACLE','JAVA',4100);

INSERT INTO PROGRAMMER
VALUES('JULIANA',TO_DATE('31-JAN-1968','DD-MON-YYYY'),TO_DATE('21-APR-1990','DD-MON-YYYY'),
'F','COBOL','DBASE',3000);

INSERT INTO PROGRAMMER
VALUES('KAMALA',TO_DATE('30-OCT-1968','DD-MON-YYYY'),TO_DATE('02-JAN-1992','DD-MON-YYYY'),
'F','C','DBASE',2900);

INSERT INTO PROGRAMMER
VALUES('MARY',TO_DATE('24-JUN-1970','DD-MON-YYYY'),TO_DATE('01-FEB-1991','DD-MON-YYYY'),
'F','C++','ORACLE',4500);

INSERT INTO PROGRAMMER
VALUES('NELSON',TO_DATE('11-SEP-1965','DD-MON-YYYY'),TO_DATE('11-OCT-1989','DD-MON-YYYY'),
'M', 'COBOL','DBASE',2500);

INSERT INTO PROGRAMMER
VALUES('PATRICK',TO_DATE('19-NOV-1965','DD-MON-YYYY'),TO_DATE('21-APR-1990','DD-MON-YYYY'),
'M','PASCAL','CLIPPER',2800);

INSERT INTO PROGRAMMER
VALUES('QUADIR',TO_DATE('31-AUG-1965','DD-MON-YYYY'),TO_DATE('21-APR-1993','DD-MON-YYYY'),
'M','ASSEMBLY','C',3000);

INSERT INTO PROGRAMMER
VALUES('RAMESH',TO_DATE('03-MAY-1967','DD-MON-YYYY'),TO_DATE('28-FEB-1991','DD-MON-YYYY'),
'M','PASCAL','DBASE',3200);

INSERT INTO PROGRAMMER
VALUES('REBECCA',TO_DATE('01-JAN-1967','DD-MON-YYYY'),TO_DATE('01-DEC-1990','DD-MON-YYYY'),
'F','BASIC','COBOL',2500);

INSERT INTO PROGRAMMER
VALUES('REMITHA',TO_DATE('19-APR-1970','DD-MON-YYYY'),TO_DATE('20-APR-1993','DD-MON-YYYY'),
'F','C','ASSEMBLY',3600);

INSERT INTO PROGRAMMER
VALUES('REVATHI',TO_DATE('02-DEC-1969','DD-MON-YYYY'),TO_DATE('02-JAN-1992','DD-MON-YYYY'),
'F','PASCAL','BASIC',3700);

INSERT INTO PROGRAMMER
VALUES('VIJAYA',TO_DATE('14-DEC-1965','DD-MON-YYYY'),TO_DATE('02-MAY-1992','DD-MON-YYYY'),
'F','FOXPRO','C',3500);

INSERT INTO SOFTWARE
VALUES('ANAND','PARACHUTES','BASIC',399.95,6000,43);

INSERT INTO SOFTWARE
VALUES('ANAND','VIDEO TILING PACK','PASCAL',7500,16000,9);
```

```
INSERT INTO SOFTWARE
VALUES('JAGDESH','SERIAL LINK UTILITY','JAVA',800,7500,9);

INSERT INTO SOFTWARE
VALUES('JAGDESH','SHARES MANAGEMENT','ORACLE',3000,12000,14);

INSERT INTO SOFTWARE
VALUES('JULIANA','INVENTORY CONTROL','COBOL',3000,3500,0);

INSERT INTO SOFTWARE
VALUES('KAMALA','PAYROLL PACKAGE','DBASE',9000,20000,7);

INSERT INTO SOFTWARE
VALUES('MARY','ACC S/W','ORACLE',18000,85000,4);

INSERT INTO SOFTWARE
VALUES('MARY','CODE GENERATOR','C',4500,20000,23);

INSERT INTO SOFTWARE
VALUES('MARY','READ ME','C++',300,1200,84);

INSERT INTO SOFTWARE
VALUES('PATRICK','GRAPHIC EDITOR','PASCAL',750,5000,11);

INSERT INTO SOFTWARE
VALUES('QUADIR','FILE UTILITY','ASSEMBLY',499.95,530,114);

INSERT INTO SOFTWARE
VALUES('QUADIR','TALLY','C',1900,3400,21);

INSERT INTO SOFTWARE
VALUES('RAMESH','HOTEL MANAGEMENT','DBASE',12000,35000,4);

INSERT INTO SOFTWARE
VALUES('RAMESH','READ ME','PASCAL',99.95,4500,73);

INSERT INTO SOFTWARE
VALUES('REMITHA','PC UTILITIES','C',725,5000,51);

INSERT INTO SOFTWARE
VALUES('REMITHA','ISR HELP MANAGEMENT','ASSEMBLY',2500,6000,6);

INSERT INTO SOFTWARE
VALUES('REVATHI','HOTEL MANAGEMENT','PASCAL',1100,75000,2);

INSERT INTO SOFTWARE
VALUES('REVATHI','QUIZ MASTER','BASIC',3200,2100,15);

INSERT INTO SOFTWARE
VALUES('VIJAYA','ISK EDITOR','C',900,700,6);


INSERT INTO STUDIES
VALUES('ANAND','SABHARI','PGDCA',4500);

INSERT INTO STUDIES
VALUES('ALTAF','CCIT','DCA',7200);

INSERT INTO STUDIES
VALUES('JAGDESH','SSIL','DCA',3500);
```

```
INSERT INTO STUDIES
VALUES('JULIANA','BDPS','DCA',22000);

INSERT INTO STUDIES
VALUES('KAMALA','PRAGATHI','DCP',5000);

INSERT INTO STUDIES
VALUES('MARY','SABHARI','PGDCA',4500);

INSERT INTO STUDIES
VALUES('NELSON','PRAGATHI','DAP',6200);

INSERT INTO STUDIES
VALUES('PATRICK','PRAGATHI','DCAP',5200);

INSERT INTO STUDIES
VALUES('QUADIR','APPLE','HDCP',14000);

INSERT INTO STUDIES
VALUES('RAMESH','SABHARI','PGDCA',4500);

INSERT INTO STUDIES
VALUES('REBECCA','BRILLIANT','DCA &P',11000);

INSERT INTO STUDIES
VALUES('REMITHA','BDPS','DCS',6000);

INSERT INTO STUDIES
VALUES('REVATHI','SABHARI','DAP',5000);

INSERT INTO STUDIES
VALUES('VIJAYA','BDPS','DCA',48000);
```

Queries – I

1. Find out the SELLING COST AVERAGE for packages developed in PASCAL.

```
SELECT AVG(SCOST) "SELLING COST AVERAGE"
FROM SOFTWARE
WHERE DEV_IN='PASCAL';

SELLING COST AVERAGE
--------------------
           2362.4875
```

2. Display the NAMES and AGES of all the programmers.

```
SELECT NAME,ROUND((SYSDATE-DOB)/365) "AGE"
FROM PROGRAMMER;

NAME            AGE
-------- ---------
ANAND            28
ALTAF            30
JAGDESH          23
JULIANA          26
KAMALA           25
MARY             24
PATRICK          28
QUADIR           28
RAMESH           27
REBECCA          27
REMITHA          24
REVATHI          24
VIJAYA           28
```

3. Display the NAMES of those who have done DAP course.

```
SELECT NAME
FROM STUDIES
WHERE COURSE='DAP';

NAME
--------
NELSON
REVATHI
```

4. What is the highest number of copies sold of a package?

```
SELECT MAX(SOLD) "NO.OF COPIES"
FROM SOFTWARE;

NO.OF COPIES
------------
         114
```

5. Display the names and date of birth of all the programmers born in January.

```
SELECT NAME,DOB
FROM PROGRAMMER
WHERE SUBSTR(DOB,4,3)='JAN';

NAME     DOB
-------- ---------
JULIANA  31-JAN-68
REBECCA  01-JAN-67
```

6. Display the lowest course fee.

```
SELECT MIN(CCOST) "LOWEST COURSE FEES"
FROM STUDIES;

LOWEST COURSE FEES
------------------
              3500
```

7. How many programmers have done the PGDCA course.

```
SELECT COUNT(NAME) "NO.OF PROGRAMMERS"
FROM STUDIES
WHERE COURSE='PGDCA';

NO.OF PROGRAMMERS
-----------------
                3
```

8. How much revenues have been earned through sale of packages in 'C'.

```
SELECT SUM(SCOST*SOLD) "REVENUE"
FROM SOFTWARE
WHERE DEV_IN='C';

  REVENUE
---------
   185775
```

9. Display the details of the software developed by 'RAMESH'

```
SELECT TITLE,DEV_IN,SCOST,DCOST,SOLD
FROM SOFTWARE
WHERE NAME='RAMESH';

TITLE                DEV_IN       SCOST     DCOST      SOLD
-------------------- -------- --------- --------- ---------
HOTEL MANAGEMENT     DBASE        12000     35000         4
READ ME              PASCAL       99.95      4500        73
```

10. How many programmers studied at SABHARI.

```
SELECT COUNT(NAME) "NO.OF PROGRAMMERS"
FROM STUDIES
```

```
WHERE SPLACE='SABHARI';

NO.OF PROGRAMMERS
-----------------
                4
```

11. Display the details of packages whose sales crossed the 2000 mark.

```
SELECT TITLE,DEV_IN,SCOST,DCOST,SOLD
FROM SOFTWARE
WHERE SCOST*SOLD>2000;

TITLE                DEV_IN       SCOST     DCOST      SOLD
-------------------- -------- --------- --------- ---------
PARACHUTES           BASIC       399.95      6000        43
VIDEO TILING PACK    PASCAL        7500     16000         9
SERIAL LINK UTILITY  JAVA           800      7500         9
SHARES MANAGEMENT    ORACLE        3000     12000        14
PAYROLL PACKAGE      DBASE         9000     20000         7
ACC S/W              ORACLE       18000     85000         4
CODE GENERATOR       C             4500     20000        23
READ ME              C++            300      1200        84
GRAPHIC EDITOR       PASCAL         750      5000        11
FILE UTILITY         ASSEMBLY    499.95       530       114
TALLY                C             1900      3400        21
HOTEL MANAGEMENT     DBASE        12000     35000         4
READ ME              PASCAL       99.95      4500        73
PC UTILITIES         C              725      5000        51
ISR HELP MANAGEMENT  ASSEMBLY      2500      6000         6
HOTEL MANAGEMENT     PASCAL        1100     75000         2
QUIZ MASTER          BASIC         3200      2100        15
ISK EDITOR           C              900       700         6
```

12. Find out the number of copies, which should be sold in order to recover the DEVELOPMENT COST of each package.

```
SELECT CEIL(DCOST/SCOST) "NO.OF COPIES"
FROM SOFTWARE;

NO.OF COPIES
------------
          16
           3
          10
           4
           2
           3
           5
           5
           4
           7
           2
           2
           3
          46
           7
           3
          69
           1
           1
```

13. Display the details of packages for, which development cost has been recovered.

```
SELECT TITLE,DEV_IN,SCOST,DCOST,SOLD
FROM SOFTWARE
WHERE SCOST*SOLD>=DCOST;
```

```
TITLE                DEV_IN       SCOST     DCOST      SOLD
-------------------- -------- --------- --------- ---------
PARACHUTES           BASIC       399.95      6000        43
VIDEO TILING PACK    PASCAL        7500     16000         9
SHARES MANAGEMENT    ORACLE        3000     12000        14
PAYROLL PACKAGE      DBASE         9000     20000         7
CODE GENERATOR       C             4500     20000        23
READ ME              C++            300      1200        84
GRAPHIC EDITOR       PASCAL         750      5000        11
FILE UTILITY         ASSEMBLY    499.95       530       114
TALLY                C             1900      3400        21
HOTEL MANAGEMENT     DBASE        12000     35000         4
READ ME              PASCAL       99.95      4500        73
PC UTILITIES         C              725      5000        51
ISR HELP MANAGEMENT  ASSEMBLY      2500      6000         6
QUIZ MASTER          BASIC         3200      2100        15
ISK EDITOR           C              900       700         6
```

14. What is the price of the costliest software developed in BASIC?

```
SELECT MAX(SCOST) "PRICE"
FROM SOFTWARE
WHERE DEV_IN='BASIC';

    PRICE
---------
     3200
```

15. How many packages were developed in DBASE?

```
SELECT COUNT(TITLE) "NO.OF PACKAGES"
FROM SOFTWARE
WHERE DEV_IN='DBASE';

NO.OF PACKAGES
--------------
             2
```

16. How many programmers studied at PRAGATHI.

```
SELECT COUNT(NAME) "NO.OF PROGRAMMERS"
FROM STUDIES
WHERE SPLACE='PRAGATHI';

NO.OF PROGRAMMERS
-----------------
                3
```

17. How many programmers paid 5000 to 10000 for their course.

```
SELECT COUNT(NAME) "NO.OF PROGRAMMERS"
FROM STUDIES
WHERE CCOST BETWEEN 5000 AND 10000;

NO.OF PROGRAMMERS
-----------------
                6
```

18. What is the average course fee?

```
SELECT AVG(CCOST) "AVERAGE COURSE FEES"
FROM STUDIES;

AVERAGE COURSE FEES
-------------------
          10471.429
```

19. Display the details of programmers knowing C.

```
SELECT *
FROM PROGRAMMER
```

```
WHERE PROF1='C' OR PROF2='C';

NAME     DOB       DOJ       S PROF1    PROF2       SALARY
-------- --------- --------- - -------- -------- ---------
KAMALA   30-OCT-68 02-JAN-92 F C        DBASE          2900
QUADIR   31-AUG-65 21-APR-93 M ASSEMBLY C              3000
REMITHA  19-APR-70 20-APR-93 F C        ASSEMBLY       3600
VIJAYA   14-DEC-65 02-MAY-92 F FOXPRO   C              3500
```

20. How many programmers know either COBOL or PASCAL.

```
SELECT COUNT(NAME) "NO.OF PROGRAMMERS"
FROM PROGRAMMER
WHERE PROF1 IN('COBOL','PASCAL') OR PROF2 IN('COBOL','PASCAL');

NO.OF PROGRAMMERS
-----------------
                7
```

21. How many programmers don't know PASCAL & C?

```
SELECT COUNT(NAME) "NO.OF PROGRAMMERS"
FROM PROGRAMMER
WHERE PROF1 NOT IN('PASCAL','C') AND PROF2 NOT IN('PASCAL','C');

NO.OF PROGRAMMERS
-----------------
                5
```

22. How old is the oldest male programmer?

```
SELECT MAX(FLOOR(MONTHS_BETWEEN(SYSDATE,DOB)/12)) "AGE"
FROM PROGRAMMER
WHERE SEX = 'M'

      AGE
---------
       38
```

23. What is the AVERAGE age of female programmers?

```
SELECT AVG(MONTHS_BETWEEN(SYSDATE,DOB)/12) "AVERAGE AGE"
FROM PROGRAMMER
WHERE SEX = 'F'

AVERAGE AGE
-----------
  34.586661
```

24. Calculate the experience in years for each programmers and display along with the names in descending order.

```
SELECT NAME,FLOOR(MONTHS_BETWEEN(SYSDATE,DOJ)/12) EXPERIENCE
FROM PROGRAMMER
ORDER BY EXPERIENCE DESC

NAME     EXPERIENCE
-------- ----------
ALTAF            12
JULIANA          12
PATRICK          12
RAMESH           12
MARY             12
REBECCA          12
KAMALA           11
REVATHI          11
ANAND            10
VIJAYA           10
QUADIR            9
REMITHA           9
```

```
JAGDESH           8
```

25. Who are the programmers who celebrated their birthdays during the current month?

```
SELECT NAME
FROM PROGRAMMER
WHERE SUBSTR(SYSDATE,4,3)=SUBSTR(DOB,4,3);

NAME
--------
JULIANA
REBECCA
```

26. How many female programmers are there.

```
SELECT COUNT(NAME) "NO.OF FEMALE PROGRAMMERS"
FROM PROGRAMMER
WHERE SEX='M';

NO.OF FEMALE PROGRAMMERS
------------------------
                       6
```

27. What are the languages known by the male programmers?

```
SELECT PROF1,PROF2
FROM PROGRAMMER
WHERE SEX='M';

PROF1    PROF2
-------- --------
PASCAL   BASIC
CLIPPER  COBOL
ORACLE   JAVA
PASCAL   CLIPPER
ASSEMBLY C
PASCAL   DBASE
```

28. What is the average salary?

```
SELECT AVG(SALARY) "AVERGAE SALARY"
FROM PROGRAMMER;

AVERGAE SALARY
--------------
     3292.3077
```

29. How many people draw 2000 to 4000 salary?

```
SELECT COUNT(NAME) "NO.OF PROGRAMMERS"
FROM PROGRAMMER
WHERE SALARY BETWEEN 2000 AND 4000;

NO.OF PROGRAMMERS
-----------------
               11
```

30. Display the details of those who don't know CLIPPER, COBOL or PASCAL.

```
SELECT *
FROM PROGRAMMER
WHERE PROF1 NOT IN('CLIPPER','COBOL','PASCAL') OR
      PROF2 NOT IN('CLIPPER','COBOL','PASCAL')

NAME     DOB       DOJ       S PROF1    PROF2       SALARY
-------- --------- --------- - -------- -------- ---------
ANAND    21-APR-66 21-APR-92 M PASCAL   BASIC          3200
JAGDESH  06-OCT-70 04-OCT-94 M ORACLE   JAVA           4100
JULIANA  31-JAN-68 21-APR-90 F COBOL    DBASE          3000
KAMALA   30-OCT-68 02-JAN-92 F C        DBASE          2900
MARY     24-JUN-70 01-FEB-91 F C++      ORACLE         4500
QUADIR   31-AUG-65 21-APR-93 M ASSEMBLY C              3000
```

```
RAMESH   03-MAY-67 28-FEB-91 M PASCAL   DBASE         3200
REBECCA  01-JAN-67 01-DEC-90 F BASIC    COBOL         2500
REMITHA  19-APR-70 20-APR-93 F C        ASSEMBLY      3600
REVATHI  02-DEC-69 02-JAN-92 F PASCAL   BASIC         3700
VIJAYA   14-DEC-65 02-MAY-92 F FOXPRO   C             3500
```

31. How many female programmers knowing C are above 24 years of age.

```
SELECT COUNT(NAME) "NO.OF PROGRAMMERS"
FROM PROGRAMMER
WHERE SEX='F' AND (PROF1='C' OR PROF2='C') AND
MONTHS_BETWEEN(SYSDATE,DOB)/12 > 24

NO.OF PROGRAMMERS
-----------------
                2
```

32. Who are the programmers who will be celebrating their birthdays within a week

```
SELECT NAME
FROM PROGRAMMER
WHERE SUBSTR(DOB,4,3)=SUBSTR(SYSDATE,4,3) AND
   SUBSTR(DOB,1,2) BETWEEN SUBSTR(SYSDATE,1,2) AND SUBSTR(SYSDATE,1,2)+7;

NAME
--------
JULIANA
```

33. Display the details of those with less than a year's experience.

```
SELECT *
FROM PROGRAMMER
WHERE MONTHS_BETWEEN(SYSDATE,DOJ)/12 < 1;

NAME     DOB       DOJ       S PROF1    PROF2       SALARY
-------- --------- --------- - -------- -------- ---------
JAGDESH  06-OCT-70 04-OCT-94 M ORACLE   JAVA           4100
QUADIR   31-AUG-65 21-APR-93 M ASSEMBLY C              3000
REMITHA  19-APR-70 20-APR-93 F C        ASSEMBLY       3600
```

34. Display the details of those who will be completing 2 years of service this year.

```
SELECT *
FROM PROGRAMMER
WHERE CEIL(MONTHS_BETWEEN(SYSDATE,DOJ)/12)=2

NAME     DOB       DOJ       S PROF1    PROF2       SALARY
-------- --------- --------- - -------- -------- ---------
ANAND    21-APR-66 21-APR-92 M PASCAL   BASIC          3200
VIJAYA   14-DEC-65 02-MAY-92 F FOXPRO   C              3500
```

35. Calculate the amount to be recovered for those packages whose development cost has not yet been recovered.

```
SELECT TITLE, DCOST-SCOST*SOLD AMT
FROM SOFTWARE
WHERE DCOST-SCOST*SOLD > 0;

TITLE                        AMT
-------------------- ---------
SERIAL LINK UTILITY          300
INVENTORY CONTROL           3500
ACC S/W                    13000
HOTEL MANAGEMENT           72800
```

36. List the packages, which have not been sold so for.

```
SELECT TITLE
FROM SOFTWARE
WHERE SOLD=0;

TITLE
```

```
--------------------
INVENTORY CONTROL
```

37. Find out the cost of the software developed by Mary.

```
SELECT DCOST
FROM SOFTWARE
WHERE NAME = 'MARY';

    DCOST
---------
    85000
    20000
     1200
```

38. Display the institute names from the studies table without duplicates.

```
SELECT DISTINCT SPLACE
FROM STUDIES;

SPLACE
---------
APPLE
BDPS
BRILLIANT
CCIT
PRAGATHI
SABHARI
SSIL
```

39. How many different courses are mentioned in the studies table.

```
SELECT DISTINCT COURSE
FROM STUDIES;

COURS
-----
DAP
DCA
DCA
DCAP
DCP
DCS
HDCP
PGDCA
```

40. Display the names of the programmers whose names contain 2 occurrences of the letter ‘A’.

```
SELECT NAME
FROM PROGRAMMER
WHERE INSTR(NAME,'A',1,2)<>0;

NAME
--------
ANAND
ALTAF
JULIANA
KAMALA
VIJAYA
```

41. Display the names of programmers whose names contain upto 5 Characters.

```
SELECT NAME
ROM PROGRAMMER
WHERE LENGTH(NAME)<=5;

NAME
--------
ANAND
```

```
ALTAF
MARY
```

42. How many female programmers knowing COBOL have more than 2 years experience.

```
SELECT COUNT(NAME) "NO.OF PROGRAMMERS"
FROM PROGRAMMER
WHERE SEX = 'F' AND (PROF1='COBOL' OR PROF2='COBOL') AND
MONTHS_BETWEEN(SYSDATE,DOJ)>2;

NO.OF PROGRAMMERS
-----------------
                2
```

43. What is the length of the shortest name in the programmer table?

```
SELECT MIN(LENGTH(NAME)) "SHORTEST LENGTH"
FROM PROGRAMMER;

SHORTEST LENGTH
---------------
              4
```

44. What is the average development cost of a package developed in COBOL?

```
SELECT AVG(DCOST) "AVERAGE DEVELOPMENT COST"
FROM SOFTWARE
WHERE DEV_IN = 'PASCAL';

AVERAGE DEVELOPMENT COST
------------------------
                   25125
```

45. Display the name, sex, DOB(DD/MMM/YY format), DOJ(DD/MMM/YY format) for all the programmers without using conversion function.

```
SELECT NAME,SEX,
   SUBSTR(DOB,1,2)||'/'||SUBSTR(DOB,4,3)||'/'||SUBSTR(DOB,8,2) DOB,
   SUBSTR(DOJ,1,2)||'/'||SUBSTR(DOJ,4,3)||'/'||SUBSTR(DOJ,8,2) DOJ
FROM PROGRAMMER

NAME     S DOB       DOJ
-------- - --------- ---------
ANAND    M 21/APR/66 21/APR/92
ALTAF    M 02/JUL/64 13/NOV/90
JAGDESH  M 06/OCT/70 04/OCT/94
JULIANA  F 31/JAN/68 21/APR/90
KAMALA   F 30/OCT/68 02/JAN/92
MARY     F 24/JUN/70 01/FEB/91
PATRICK  M 19/NOV/65 21/APR/90
QUADIR   M 31/AUG/65 21/APR/93
RAMESH   M 03/MAY/67 28/FEB/91
REBECCA  F 01/JAN/67 01/DEC/90
REMITHA  F 19/APR/70 20/APR/93
REVATHI  F 02/DEC/69 02/JAN/92
VIJAYA   F 14/DEC/65 02/MAY/92
```

46. What is the amount paid in salaries of the male programmers who do not known Cobol?

```
SELECT SUM(SALARY) "AMT OF SALARY"
FROM PROGRAMMER
WHERE PROF1<>'COBOL' AND PROF2<>'COBOL' AND SEX='M';

AMT OF SALARY
-------------
        16300
```

47. Who are the programmers who were born on the last day of the month.

```
SELECT NAME
FROM PROGRAMMER
WHERE LAST_DAY(DOB)=DOB;
```

```
NAME
--------
JULIANA
QUADIR
```

48. Display the TITLE, SCOST, DCOST and difference between SCOST and DCOST in descending order of difference.

```
SELECT TITLE,SCOST,DCOST,DCOST-SCOST DIFFERENCE
FROM SOFTWARE
ORDER BY DIFFERENCE DESC;

TITLE                    SCOST     DCOST DIFFERENCE
-------------------- --------- --------- ----------
HOTEL MANAGEMENT          1100     75000      73900
ACC S/W                  18000     85000      67000
HOTEL MANAGEMENT         12000     35000      23000
CODE GENERATOR            4500     20000      15500
PAYROLL PACKAGE           9000     20000      11000
SHARES MANAGEMENT         3000     12000       9000
VIDEO TILING PACK         7500     16000       8500
SERIAL LINK UTILITY        800      7500       6700
PARACHUTES              399.95      6000    5600.05
READ ME                  99.95      4500    4400.05
PC UTILITIES               725      5000       4275
GRAPHIC EDITOR             750      5000       4250
ISR HELP MANAGEMENT       2500      6000       3500
TALLY                     1900      3400       1500
READ ME                    300      1200        900
INVENTORY CONTROL         3000      3500        500
FILE UTILITY            499.95       530      30.05
ISK EDITOR                 900       700       -200
QUIZ MASTER               3200      2100      -1100
```

49. Display the names of the packages whose names contain more than 1 word.

```
SELECT TITLE
FROM SOFTWARE
WHERE INSTR(TITLE,' ',1,1)<>0;

TITLE
--------------------
VIDEO TILING PACK
SERIAL LINK UTILITY
SHARES MANAGEMENT
INVENTORY CONTROL
PAYROLL PACKAGE
ACC S/W
CODE GENERATOR
READ ME
GRAPHIC EDITOR
FILE UTILITY
HOTEL MANAGEMENT
READ ME
PC UTILITIES
ISR HELP MANAGEMENT
HOTEL MANAGEMENT
QUIZ MASTER
ISK EDITOR
```

50. Display the name, DOB, DOJ of those whose month of birth and month of joining are same.

```
SELECT NAME,DOB,DOJ
FROM PROGRAMMER
WHERE SUBSTR(DOB,4,3)=SUBSTR(DOJ,4,3);
```

```
NAME     DOB       DOJ
-------- --------- ---------
ANAND    21-APR-66 21-APR-92
JAGDESH  06-OCT-70 04-OCT-94
REMITHA  19-APR-70 20-APR-93
```

Queries II:

1. Display the number of packages developed in each language.

```
SELECT DEV_IN,COUNT(TITLE)
FROM SOFTWARE
GROUP BY DEV_IN;

DEV_IN   COUNT(TITLE)
-------- ------------
ASSEMBLY            2
BASIC               2
C                   4
C++                 1
COBOL               1
DBASE               2
JAVA                1
ORACLE              2
PASCAL              4
```

2. Display the number of packages developed by each person.

```
SELECT NAME,COUNT(TITLE)
FROM SOFTWARE
GROUP BY NAME;

NAME     COUNT(TITLE)
-------- ------------
ANAND               2
JAGDESH             2
JULIANA             1
KAMALA              1
MARY                3
PATRICK             1
QUADIR              2
RAMESH              2
REMITHA             2
REVATHI             2
VIJAYA              1
```

3. Display the number of male and female programmers.

```
SELECT SEX,COUNT(NAME)
FROM PROGRAMMER
GROUP BY SEX;

S COUNT(NAME)
- -----------
F           7
M           6
```

4. Display the costliest package and highest selling package developed in each language.

```
CREATE VIEW SW_VW AS
SELECT DEV_IN,MAX(SCOST) PRICE, MAX(SOLD) COPIES
FROM SOFTWARE
GROUP BY DEV_IN;

SELECT SW_VW.DEV_IN,A.TITLE,B.TITLE
FROM SOFTWARE A, SOFTWARE B, SW_VW
WHERE A.DEV_IN = SW_VW.DEV_IN AND
```

```
B.DEV_IN = SW_VW.DEV_IN AND
A.SCOST = SW_VW.PRICE AND
B.SOLD = SW_VW.COPIES;

DEV_IN   TITLE                TITLE
-------- -------------------- --------------------
ASSEMBLY ISR HELP MANAGEMENT  FILE UTILITY
BASIC    QUIZ MASTER          PARACHUTES
C        CODE GENERATOR       PC UTILITIES
C++      READ ME              READ ME
COBOL    INVENTORY CONTROL    INVENTORY CONTROL
DBASE    HOTEL MANAGEMENT     PAYROLL PACKAGE
JAVA     SERIAL LINK UTILITY  SERIAL LINK UTILITY
ORACLE   ACC S/W              SHARES MANAGEMENT
PASCAL   VIDEO TILING PACK    READ ME
```

5. Display the number of people Born in each year.

```
SELECT SUBSTR(DOB,8,2) YEAR, COUNT(NAME)
FROM PROGRAMMER
GROUP BY SUBSTR(DOB,8,2);

YE COUNT(NAME)
-- -----------
64           1
65           3
66           1
67           2
68           2
69           1
70           3
```

6. Display the number of people joined in each year.

```
SELECT SUBSTR(DOJ,8,2) YEAR, COUNT(NAME)
FROM PROGRAMMER
GROUP BY SUBSTR(DOJ,8,2);

YE COUNT(NAME)
-- -----------
90           4
91           2
92           4
93           2
94           1
```

7. Display the number of people born in each month.

```
SELECT SUBSTR(DOB,4,3) MONTH, COUNT(NAME)
FROM PROGRAMMER
GROUP BY SUBSTR(DOB,4,3);

MON COUNT(NAME)
--- -----------
APR           2
AUG           1
DEC           2
JAN           2
JUL           1
JUN           1
MAY           1
NOV           1
OCT           2
```

8. Display the number of people joined in each month.

```
SELECT SUBSTR(DOJ,4,3) MONTH, COUNT(NAME)
FROM PROGRAMMER
```

```
GROUP BY SUBSTR(DOJ,4,3)

MON COUNT(NAME)
--- -----------
APR           5
DEC           1
FEB           2
JAN           2
MAY           1
NOV           1
OCT           1
```

9. Display the language wise count of prof1.

```
SELECT PROF1,COUNT(NAME)
FROM PROGRAMMER
GROUP BY PROF1;

PROF1    COUNT(NAME)
-------- -----------
ASSEMBLY           1
BASIC              1
C                  2
C++                1
CLIPPER            1
COBOL              1
FOXPRO             1
ORACLE             1
PASCAL             4
```

10. Display the language wise count of prof2.

```
SELECT PROF2,COUNT(NAME)
FROM PROGRAMMER
GROUP BY PROF2;

PROF2    COUNT(NAME)
-------- -----------
ASSEMBLY           1
BASIC              2
C                  2
CLIPPER            1
COBOL              2
DBASE              3
JAVA               1
ORACLE             1
```

11. Display the number of people in each salary group.

```
SELECT SALARY,COUNT(NAME) "NO.OF PEOPLE"
FROM PROGRAMMER
GROUP BY SALARY;

   SALARY NO.OF PEOPLE
--------- ------------
     2500            1
     2800            2
     2900            1
     3000            2
     3200            2
     3500            1
     3600            1
     3700            1
     4100            1
     4500            1
```

12. Display the number of people who studied in each institute.

```
SELECT SPLACE,COUNT(NAME) "NO.OF PEOPLE"
FROM STUDIES
GROUP BY SPLACE;

SPLACE    NO.OF PEOPLE
--------- ------------
APPLE                1
BDPS                 3
BRILLIANT            1
CCIT                 1
PRAGATHI             3
SABHARI              4
SSIL                 1
```

13. Display the number of people who studied in each course.

```
SELECT COURSE,COUNT(NAME)
FROM STUDIES
GROUP BY COURSE;

COURS COUNT(NAME)
----- -----------
DAP             2
DCA             4
DCA             1
DCAP            1
DCP             1
DCS             1
HDCP            1
PGDCA           3
```

14. Display the total development cost of the package developed in each language.

```
SELECT DEV_IN,SUM(DCOST)
FROM SOFTWARE
GROUP BY DEV_IN;

DEV_IN   SUM(DCOST)
-------- ----------
ASSEMBLY       6530
BASIC          8100
C             29100
C++            1200
COBOL          3500
DBASE         55000
JAVA           7500
ORACLE        97000
PASCAL       100500
```

15. Display the selling cost of the package developed in each language.

```
SELECT DEV_IN,SUM(SCOST)
FROM SOFTWARE
GROUP BY DEV_IN;

DEV_IN   SUM(SCOST)
-------- ----------
ASSEMBLY    2999.95
BASIC       3599.95
C              8025
C++             300
COBOL          3000
DBASE         21000
JAVA            800
ORACLE        21000
PASCAL      9449.95
```

16. Display the cost of the package developed by each programmer.

```
SELECT NAME,SUM(SCOST)
FROM SOFTWARE
GROUP BY NAME;
```

```
NAME     SUM(SCOST)
-------- ----------
ANAND       7899.95
JAGDESH        3800
JULIANA        3000
KAMALA         9000
MARY          22800
PATRICK         750
QUADIR       2399.95
RAMESH      12099.95
REMITHA        3225
REVATHI        4300
VIJAYA          900
```

17. Display the sales value of packages developed by each programmer.

```
SELECT NAME,SUM(SCOST*SOLD) "SALES VALUE"
FROM SOFTWARE
GROUP BY NAME;
```

```
NAME     SALES VALUE
-------- -----------
ANAND       84697.85
JAGDESH        49200
JULIANA            0
KAMALA         63000
MARY          200700
PATRICK         8250
QUADIR       96894.3
RAMESH      55296.35
REMITHA        51975
REVATHI        50200
VIJAYA          5400
```

18. Display the number of packages developed by each programmer.

```
SELECT NAME,COUNT(TITLE)
FROM SOFTWARE
GROUP BY NAME;
```

```
NAME     COUNT(TITLE)
-------- ------------
ANAND               2
JAGDESH             2
JULIANA             1
KAMALA              1
MARY                3
PATRICK             1
QUADIR              2
RAMESH              2
REMITHA             2
REVATHI             2
VIJAYA              1
```

19. Display the sales cost of the package developed by each programmer language wise.

```
BREAK ON DEV_IN

SELECT DEV_IN,NAME,SCOST
FROM SOFTWARE
ORDER BY DEV_IN,NAME
```

```
DEV_IN   NAME         SCOST
-------- -------- ---------
ASSEMBLY QUADIR      499.95
         REMITHA       2500
BASIC    ANAND       399.95
         REVATHI       3200
C        MARY          4500
         QUADIR        1900
         REMITHA        725
         VIJAYA         900
C++      MARY           300
COBOL    JULIANA       3000
DBASE    KAMALA        9000
         RAMESH       12000
JAVA     JAGDESH        800
ORACLE   JAGDESH       3000
         MARY         18000
PASCAL   ANAND         7500
         PATRICK        750
         RAMESH       99.95
         REVATHI       1100
```

20. Display each programmers name, costliest package and cheapest packages developed by him/her.

```
CREATE VIEW PROGRAMMER_VW AS
SELECT NAME,MAX(SCOST) HIGH, MIN(SCOST) LOW
FROM SOFTWARE
GROUP BY NAME;

SELECT PROGRAMMER_VW.NAME,A.TITLE,B.TITLE
FROM PROGRAMMER_VW, SOFTWARE A, SOFTWARE B
WHERE PROGRAMMER_VW.NAME = A.NAME AND
   PROGRAMMER_VW.NAME = B.NAME AND
   PROGRAMMER_VW.HIGH = A.SCOST AND PROGRAMMER_VW.LOW = B.SCOST;

NAME     TITLE                TITLE
-------- -------------------- --------------------
ANAND    VIDEO TILING PACK    PARACHUTES
JAGDESH  SHARES MANAGEMENT    SERIAL LINK UTILITY
JULIANA  INVENTORY CONTROL    INVENTORY CONTROL
KAMALA   PAYROLL PACKAGE      PAYROLL PACKAGE
MARY     ACC S/W              READ ME
PATRICK  GRAPHIC EDITOR       GRAPHIC EDITOR
QUADIR   TALLY                FILE UTILITY
RAMESH   HOTEL MANAGEMENT     READ ME
REMITHA  ISR HELP MANAGEMENT  PC UTILITIES
REVATHI  QUIZ MASTER          HOTEL MANAGEMENT
VIJAYA   ISK EDITOR           ISK EDITOR
```

21. Display each language name with average development cost, average selling cost and average price per copy.

```
SELECT DEV_IN,AVG(DCOST),AVG(SCOST) "AVERAGE COST PER COPY"
FROM SOFTWARE
GROUP BY DEV_IN;

DEV_IN   AVG(DCOST) AVERAGE COST PER COPY
-------- ---------- --------------------
ASSEMBLY       3265              1499.975
BASIC          4050              1799.975
C              7275               2006.25
C++            1200                   300
```

```
COBOL          3500                  3000
DBASE         27500                 10500
JAVA           7500                   800
ORACLE        48500                 10500
PASCAL        25125              2362.4875
```

22. Display each institute name with number of courses, average cost per course.

```
SELECT SPLACE,COUNT(COURSE),AVG(CCOST)
FROM STUDIES
GROUP BY SPLACE;

SPLACE    COUNT(COURSE) AVG(CCOST)
--------- ------------- ----------
APPLE                 1      14000
BDPS                  3  25333.333
BRILLIANT             1      11000
CCIT                  1       7200
PRAGATHI              3  5466.6667
SABHARI               4       4625
SSIL                  1       3500
```

23. Display each institute name with number of students.

```
SELECT SPLACE,COUNT(NAME)
FROM STUDIES
GROUP BY SPLACE;

SPLACE    COUNT(NAME)
--------- -----------
APPLE               1
BDPS                3
BRILLIANT           1
CCIT                1
PRAGATHI            3
SABHARI             4
SSIL                1
```

24. Display the names of male and female programmers.

```
BREAK ON SEX

SELECT SEX,NAME
FROM PROGRAMMER
ORDER BY SEX;

S NAME
- --------
F JULIANA
  KAMALA
  MARY
  REVATHI
  VIJAYA
  REMITHA
  REBECCA
M ANAND
  QUADIR
  RAMESH
  PATRICK
  ALTAF
  JAGDESH
```

25. Display the programmers name and their packages.

```
BREAK ON NAME

SELECT NAME,TITLE
FROM SOFTWARE;
```

```
NAME     TITLE
-------- --------------------
ANAND    PARACHUTES
         VIDEO TILING PACK
JAGDESH  SERIAL LINK UTILITY
         SHARES MANAGEMENT
JULIANA  INVENTORY CONTROL
KAMALA   PAYROLL PACKAGE
MARY     ACC S/W
         CODE GENERATOR
         READ ME
PATRICK  GRAPHIC EDITOR
QUADIR   FILE UTILITY
         TALLY
RAMESH   HOTEL MANAGEMENT
         READ ME
REMITHA  PC UTILITIES
         ISR HELP MANAGEMENT
REVATHI  HOTEL MANAGEMENT
         QUIZ MASTER
VIJAYA   ISK EDITOR
```

26. Display the number of packages in each language.

```
SELECT DEV_IN,COUNT(TITLE)
FROM SOFTWARE
GROUP BY DEV_IN;

DEV_IN   COUNT(TITLE)
-------- ------------
ASSEMBLY            2
BASIC               2
C                   4
C++                 1
COBOL               1
DBASE               2
JAVA                1
ORACLE              2
PASCAL              4
```

27. Display the number of packages in each language for which development cost is less than 1000.

```
SELECT DEV_IN,COUNT(TITLE)
FROM SOFTWARE
WHERE DCOST<1000
GROUP BY DEV_IN

DEV_IN   COUNT(TITLE)
-------- ------------
ASSEMBLY            1
C                   1
```

28. Display the average difference Between SCOST and DCOST for each language.

```
SELECT DEV_IN,AVG(DCOST-SCOST)
FROM SOFTWARE
GROUP BY DEV_IN;

DEV_IN   AVG(DCOST-SCOST)
-------- ----------------
ASSEMBLY         1765.025
BASIC            2250.025
C                 5268.75
```

```
C++                    900
COBOL                  500
DBASE                17000
JAVA                  6700
ORACLE               38000
PASCAL           22762.513
```

29. Display the total SCOST, DCOST and amount to be recovered for each programmer for whose DCOST has not yet been recovered

```
SELECT NAME,SUM(SCOST),SUM(DCOST),SUM(DCOST-SCOST*SOLD)
FROM SOFTWARE
WHERE DCOST-SCOST*SOLD>0
GROUP BY NAME

NAME     SUM(SCOST) SUM(DCOST) SUM(DCOST-SCOST*SOLD)
-------- ---------- ---------- --------------------
JAGDESH         800       7500                  300
JULIANA        3000       3500                 3500
MARY          18000      85000                13000
REVATHI        1100      75000                72800
```

30. Display highest, lowest and average salaries for those earning more than 2000.

```
SELECT MAX(SALARY),MIN(SALARY),AVG(SALARY)
FROM PROGRAMMER
WHERE SALARY>2000

MAX(SALARY) MIN(SALARY) AVG(SALARY)
----------- ----------- -----------
       4500        2500   3292.3077
```

Queries III:

1. Who is the highest paid C programmer?

```
SELECT NAME
FROM PROGRAMMER
WHERE (PROF1='C' OR PROF2='C') AND
         SALARY >= ALL (SELECT SALARY
                        FROM PROGRAMMER
                        WHERE PROF1='C' OR PROF2='C');
NAME
--------
REMITHA
```

2. Who is the highest paid female COBOL programmer?

```
SELECT NAME
FROM PROGRAMMER
WHERE SEX ='F' AND (PROF1='COBOL' OR PROF2='COBOL') AND
SALARY >= ALL (SELECT SALARY FROM PROGRAMMER
               WHERE SEX = 'F' AND
               (PROF1='COBOL' OR PROF2='COBOL'));

NAME
--------
JULIANA
```

3. Display the names of the highest paid programmer for each language (prof1).

```
CREATE VIEW PROG_VW AS
SELECT PROF1,MAX(SALARY) HIGH
FROM PROGRAMMER
GROUP BY PROF1;

SELECT PROGRAMMER.PROF1,NAME
FROM PROGRAMMER, PROG_VW
WHERE PROGRAMMER.PROF1 = PROG_VW.PROF1 AND
   PROGRAMMER.SALARY = PROG_VW.HIGH;
```

```
PROF1    NAME
-------- --------
ASSEMBLY QUADIR
BASIC    REBECCA
C        REMITHA
C++      MARY
CLIPPER  ALTAF
COBOL    JULIANA
FOXPRO   VIJAYA
ORACLE   JAGDESH
PASCAL   REVATHI
```

4. Who is the least experienced programmer?

```
SELECT NAME
FROM PROGRAMMER
WHERE MONTHS_BETWEEN(SYSDATE,DOJ)/12 <=ALL
   (SELECT MONTHS_BETWEEN(SYSDATE,DOJ)/12 FROM PROGRAMMER);

NAME
--------
JAGDESH
```

5. Who is the most experienced programmer knowing PASCAL?

```
SELECT NAME
FROM PROGRAMMER
WHERE (PROF1='PASCAL' OR PROF2='PASCAL') AND
   MONTHS_BETWEEN(SYSDATE,DOJ)/12 >=ALL
   (SELECT MONTHS_BETWEEN(SYSDATE,DOJ)/12
   FROM PROGRAMMER
   WHERE PROF1='PASCAL' OR PROF2='PASCAL');

NAME
--------
PATRICK
```

6. Which language is known by only one programmer?

```
CREATE VIEW PROF_VW AS
SELECT PROF1 PROF
FROM PROGRAMMER
WHERE PROF1 NOT IN (SELECT PROF2 FROM PROGRAMMER)
GROUP BY PROF1
HAVING COUNT(NAME)=1
UNION
SELECT PROF2 PROF
FROM PROGRAMMER
WHERE PROF2 NOT IN (SELECT PROF1 FROM PROGRAMMER)
GROUP BY PROF2
HAVING COUNT(NAME)=1;

SELECT * FROM PROF_VW;

PROF
--------
C++
FOXPRO
JAVA
```

7. Who is the above programmer?

```
SELECT NAME
FROM PROGRAMMER,PROF_VW
WHERE PROGRAMMER.PROF1 = PROF_VW.PROF OR
   PROGRAMMER.PROF2 = PROF_VW.PROF;

NAME
```

```
--------
JAGDESH
MARY
VIJAYA
```

8. Who is the youngest programmer knowing DBASE

```
SELECT NAME
FROM PROGRAMMER
WHERE (PROF1='DBASE' OR PROF2='DBASE') AND
   MONTHS_BETWEEN(SYSDATE,DOB)<=ALL(SELECT MONTHS_BETWEEN(SYSDATE,DOB)
                                    FROM PROGRAMMER
                                    WHERE PROF1='DBASE'OR PROF2='DBASE');

NAME
--------
KAMALA
```

9. Which female programmer earns more than 3000/- but does not know C, C++, Oracle or DBASE?

```
SELECT NAME
FROM PROGRAMMER
WHERE SEX='F' AND PROF1 NOT IN('C','C++','ORACLE','DBASE') AND
PROF2 NOT IN('C','C++','ORACLE','DBASE');

NAME
--------
REBECCA
REVATHI
```

10. Which institute has most number of students?

```
SELECT SPLACE
FROM STUDIES
GROUP BY SPLACE
HAVING COUNT(NAME)>=ALL(SELECT COUNT(NAME)
                        FROM STUDIES
                        GROUP BY SPLACE);

SPLACE
---------
SABHARI
```

11. Which course has been done by most of the students?

```
SELECT COURSE
FROM STUDIES
GROUP BY COURSE
HAVING COUNT(NAME)>=ALL(SELECT COUNT(NAME)
                        FROM STUDIES
                        GROUP BY COURSE);

COURS
-----
DCA
```

12. Display name of the institute and course, which has below average course fees

```
SELECT SPLACE,COURSE
FROM STUDIES
WHERE CCOST <= (SELECT AVG(CCOST)
                FROM STUDIES);

SPLACE    COURS
--------- -----
CCIT      DCA
SSIL      DCA
PRAGATHI  DCP
SABHARI   PGDCA
```

```
PRAGATHI  DAP
PRAGATHI  DCAP
SABHARI   PGDCA
BDPS      DCS
SABHARI   DAP
SABHARI   PGDCA
```

13. Which is the costliest course?

```
SELECT COURSE
FROM STUDIES
WHERE CCOST >= ALL (SELECT CCOST
                    FROM STUDIES);

COURS
-----
DCA
```

14. Which institute conducts the costliest course?

```
SELECT SPLACE
FROM STUDIES
WHERE CCOST = (SELECT MAX(CCOST) FROM
               STUDIES);

SPLACE
---------
BDPS
```

15. Which course has below average number of students?

```
CREATE VIEW COURSE_VW AS
SELECT COURSE
FROM STUDIES
GROUP BY COURSE
HAVING COUNT(NAME) < (SELECT AVG(COUNT(NAME))
                      FROM STUDIES
                      GROUP BY COURSE);

SELECT * FROM COURSE_VW;

COURS
-----
DCAP
DCP
DCS
HDCP
```

16. Which institute conducts the above course?

```
SELECT SPLACE
FROM STUDIES,COURSE_VW
WHERE STUDIES.COURSE = COURSE_VW.COURSE;

SPLACE
---------
PRAGATHI
PRAGATHI
BDPS
APPLE
```

17. Display names of the courses whose fees are within 1000(+or-) of the average fee.

```
SELECT COURSE
FROM STUDIES
WHERE (CCOST>=1000 AND CCOST <= (SELECT AVG(CCOST) FROM STUDIES)) OR
(CCOST>= (SELECT AVG(CCOST) FROM STUDIES) AND CCOST <=1000)

COURS
-----
```

```
DCA
DCA
DCP
PGDCA
DAP
DCAP
PGDCA
DCS
DAP
PGDCA
```

18. Which package has the highest development cost?

```
SELECT TITLE
FROM SOFTWARE
WHERE DCOST>=ALL(SELECT DCOST
                 FROM SOFTWARE);

TITLE
--------------------
ACC S/W
```

19. Which package has the lowest selling cost?

```
SELECT TITLE
FROM SOFTWARE
WHERE SCOST <= ALL (SELECT SCOST
                    FROM SOFTWARE);

TITLE
--------------------
READ ME
```

20. Who developed the package, which has sold the least number of copies?

```
SELECT NAME
FROM SOFTWARE
WHERE SOLD<=ALL (SELECT SOLD
                 FROM SOFTWARE);

NAME
--------
JULIANA
```

21. Which language was used to develop the package, which has the highest sales amount.

```
SELECT DEV_IN
FROM SOFTWARE
WHERE SCOST*SOLD >=ALL (SELECT SCOST*SOLD FROM SOFTWARE)

DEV_IN
--------
C
```

22. How many copies of the package were sold that has the least difference between development and selling cost?

```
SELECT SOLD
FROM SOFTWARE
WHERE DCOST-SCOST <= ALL (SELECT DCOST-SCOST FROM SOFTWARE);

     SOLD
---------
       15
```

23. Which is the costliest package developed in PASCAL?

```
SELECT TITLE
FROM SOFTWARE
WHERE DEV_IN = 'PASCAL' AND
SCOST>=ALL(SELECT SCOST FROM SOFTWARE WHERE DEV_IN='PASCAL' );
```

```
TITLE
--------------------
VIDEO TILING PACK
```

24. Which language was used to develop the most number of packages?

```
SELECT DEV_IN
FROM SOFTWARE
GROUP BY DEV_IN
HAVING COUNT(TITLE)>= ALL(SELECT COUNT(TITLE)
                          FROM SOFTWARE
                          GROUP BY DEV_IN);

DEV_IN
--------
C
PASCAL
```

25. Which programmer has developed the highest number of packages?

```
SELECT NAME
FROM SOFTWARE
GROUP BY NAME
HAVING COUNT(TITLE) >=ALL(SELECT COUNT(TITLE)
                          FROM SOFTWARE
                          GROUP BY NAME);

NAME
--------
MARY
```

26. Who is the author of costliest package?

```
SELECT NAME
FROM SOFTWARE
WHERE SCOST >= ALL (SELECT SCOST
                    FROM SOFTWARE);

NAME
--------
MARY
```

27. Display names of packages, which have been sold less than the average number of copies.

```
SELECT TITLE
FROM SOFTWARE
WHERE SOLD < (SELECT AVG(SOLD) FROM SOFTWARE);

TITLE
--------------------
VIDEO TILING PACK
SERIAL LINK UTILITY
SHARES MANAGEMENT
INVENTORY CONTROL
PAYROLL PACKAGE
ACC S/W
CODE GENERATOR
GRAPHIC EDITOR
TALLY
HOTEL MANAGEMENT
ISR HELP MANAGEMENT
HOTEL MANAGEMENT
QUIZ MASTER
ISK EDITOR
```

28. Who are the authors of packages, which have recovered more than double the development cost?

```
SELECT NAME
```

```
FROM SOFTWARE
WHERE SCOST*SOLD > 2*DCOST;

NAME
--------
ANAND
JAGDESH
KAMALA
MARY
QUADIR
REMITHA
REVATHI
VIJAYA
```

29. Display programmer names and cheapest package developed by them in each language.

```
CREATE VIEW CHEAP_SW_VW AS
SELECT DEV_IN,MIN(SCOST) MIN
FROM SOFTWARE
GROUP BY DEV_IN;

SELECT NAME,TITLE
FROM SOFTWARE,CHEAP_SW_VW
WHERE SOFTWARE.DEV_IN = CHEAP_SW_VW.DEV_IN AND
   SOFTWARE.SCOST = CHEAP_SW_VW.MIN;

NAME     TITLE
-------- --------------------
QUADIR   FILE UTILITY
ANAND    PARACHUTES
REMITHA  PC UTILITIES
MARY     READ ME
JULIANA  INVENTORY CONTROL
KAMALA   PAYROLL PACKAGE
JAGDESH  SERIAL LINK UTILITY
JAGDESH  SHARES MANAGEMENT
RAMESH   READ ME
```

30. Display language used by each programmer to develop the highest selling and lowest selling package.

```
CREATE VIEW SW_VW_HIGHLOW AS
SELECT MIN(SOLD) MIN, MAX(SOLD) MAX
FROM SOFTWARE;

SELECT A.DEV_IN,B.DEV_IN
FROM SOFTWARE A, SOFTWARE B, SW_VW_HIGHLOW
WHERE A.SOLD = SW_VW_HIGHLOW.MAX AND B.SOLD = SW_VW_HIGHLOW.MIN

DEV_IN   DEV_IN
-------- --------
ASSEMBLY COBOL
```

31. Who is the youngest male programmer born in 1965?

```
SELECT NAME
FROM PROGRAMMER
WHERE SEX = 'M' AND SUBSTR(DOB,8,2)=65 AND
MONTHS_BETWEEN(SYSDATE,DOB)/12 >=ALL
(SELECT MONTHS_BETWEEN(SYSDATE,DOB)/12
FROM PROGRAMMER
WHERE SEX='M' AND SUBSTR(DOB,8,2)=65);
NAME
--------
QUADIR
```

32. Who is the oldest female programmer who joined in 1992?

```
SELECT NAME
FROM PROGRAMMER
WHERE SEX = 'F' AND SUBSTR(DOJ,8,2)=92 AND
MONTHS_BETWEEN(SYSDATE,DOB)/12 >=ALL
                                (SELECT MONTHS_BETWEEN(SYSDATE,DOB)/12
                                FROM PROGRAMMER
                                WHERE SEX='F' AND SUBSTR(DOJ,8,2)=92);
```

```
NAME
--------
VIJAYA
```

33. In which year were most of the programmers born?

```
SELECT SUBSTR(DOB,8,2) "YEAR"
FROM PROGRAMMER
GROUP BY SUBSTR(DOB,8,2)
HAVING COUNT(NAME) >=ALL (SELECT COUNT(NAME)
                          FROM PROGRAMMER
                          GROUP BY SUBSTR(DOB,8,2))
```

```
YE
--
65
70
```

34. In which month did most number of programmers join?

```
SELECT SUBSTR(DOJ,4,3) "MONTH"
FROM PROGRAMMER
GROUP BY SUBSTR(DOJ,4,3)
HAVING COUNT(NAME) >=ALL (SELECT COUNT(NAME)
                          FROM PROGRAMMER
                          GROUP BY SUBSTR(DOJ,4,3));
```

```
MON
---
APR
```

35. In which language most of the programmers are proficient?

```
SELECT PROF1 A
FROM PROGRAMMER
GROUP BY PROF1
HAVING COUNT(NAME)>=ALL(SELECT COUNT(NAME) FROM PROGRAMMER
GROUP BY PROF1)
UNION
SELECT PROF2 A
FROM PROGRAMMER
GROUP BY PROF2
HAVING COUNT(NAME)>=ALL(SELECT COUNT(NAME) FROM PROGRAMMER
GROUP BY PROF2)
```

```
A
--------
DBASE
PASCAL
```

36. Who are the male programmers earning below the average salary of female programmers?

```
SELECT NAME
FROM PROGRAMMER
WHERE SEX = 'M' AND SALARY < (SELECT AVG(SALARY)
                              FROM PROGRAMMER
                              WHERE SEX='F');
```

```
NAME
--------
ANAND
ALTAF
PATRICK
QUADIR
RAMESH
```

37. Who are the female programmers earning more than the highest paid male programmers?

```
SELECT NAME
FROM PROGRAMMER
WHERE SEX = 'F' AND SALARY > (SELECT MAX(SALARY)
                              FROM PROGRAMMER
                              WHERE SEX='M');
NAME
--------
MARY
```

38. Which language has been stated as PROF1 by most of the programmers?

```
SELECT PROF1
FROM PROGRAMMER
GROUP BY PROF1
HAVING COUNT(NAME)>=ALL(SELECT COUNT(NAME)
                        FROM PROGRAMMER
                        GROUP BY PROF1);

PROF1
--------
PASCAL
```

Queries IV:

1. Display the details of those who are drawing the same salary.

```
SELECT A.*
FROM PROGRAMMER A, PROGRAMMER B
WHERE A.SALARY=B.SALARY AND A.NAME<>B.NAME;

NAME     DOB       DOJ       S PROF1    PROF2       SALARY
-------- --------- --------- - -------- -------- ---------
PATRICK  19-NOV-65 21-APR-90 M PASCAL   CLIPPER       2800
ALTAF    02-JUL-64 13-NOV-90 M CLIPPER  COBOL         2800
QUADIR   31-AUG-65 21-APR-93 M ASSEMBLY C             3000
JULIANA  31-JAN-68 21-APR-90 F COBOL    DBASE         3000
RAMESH   03-MAY-67 28-FEB-91 M PASCAL   DBASE         3200
ANAND    21-APR-66 21-APR-92 M PASCAL   BASIC         3200
```

2. Display the details of software developed by male programmers earning more than 3000.

```
SELECT SOFTWARE.*
FROM SOFTWARE,PROGRAMMER
WHERE SOFTWARE.NAME = PROGRAMMER.NAME AND SEX = 'M' AND SALARY>3000;

NAME     TITLE                DEV_IN       SCOST     DCOST      SOLD
-------- -------------------- -------- --------- --------- ---------
ANAND    PARACHUTES           BASIC        399.95      6000        43
ANAND    VIDEO TILING PACK    PASCAL         7500     16000         9
JAGDESH  SERIAL LINK UTILITY  JAVA            800      7500         9
JAGDESH  SHARES MANAGEMENT    ORACLE         3000     12000        14
RAMESH   HOTEL MANAGEMENT     DBASE         12000     35000         4
RAMESH   READ ME              PASCAL        99.95      4500        73
```

3. Display the details of packages developed in PASCAL by female programmers.

```
SELECT SOFTWARE.*
FROM SOFTWARE, PROGRAMMER
WHERE SOFTWARE.NAME = PROGRAMMER.NAME AND SEX='F' AND DEV_IN='PASCAL';
```

```
NAME     TITLE                DEV_IN       SCOST     DCOST      SOLD
-------- -------------------- -------- --------- --------- ---------
REVATHI  HOTEL MANAGEMENT     PASCAL        1100     75000         2
```

4. Display the details of programmers who joined before 1992.

```
SELECT *
FROM PROGRAMMER
WHERE SUBSTR(DOJ,8,2)<92;

NAME     DOB       DOJ       S PROF1    PROF2       SALARY
-------- --------- --------- - -------- -------- ---------
ALTAF    02-JUL-64 13-NOV-90 M CLIPPER  COBOL          2800
JULIANA  31-JAN-68 21-APR-90 F COBOL    DBASE          3000
MARY     24-JUN-70 01-FEB-91 F C++      ORACLE         4500
PATRICK  19-NOV-65 21-APR-90 M PASCAL   CLIPPER        2800
RAMESH   03-MAY-67 28-FEB-91 M PASCAL   DBASE          3200
REBECCA  01-JAN-67 01-DEC-90 F BASIC    COBOL          2500
```

5. Display details of software developed in C by female programmers of PRAGATHI.

```
SELECT SOFTWARE.*
FROM PROGRAMMER,SOFTWARE,STUDIES
WHERE PROGRAMMER.NAME = SOFTWARE.NAME AND
   SOFTWARE.NAME = STUDIES.NAME AND
   DEV_IN='DBASE' AND SEX ='F' AND SPLACE='PRAGATHI'

NAME     TITLE                DEV_IN       SCOST     DCOST      SOLD
-------- -------------------- -------- --------- --------- ---------
KAMALA   PAYROLL PACKAGE      DBASE         9000     20000         7
```

6. Display number of packages, number of copies sold and sales value of each programmer institute wise.

```
SPLACE,SOFTWARE.NAME,COUNT(SOFTWARE.NAME),SUM(SOLD),SUM(SCOST*SOLD)
FROM SOFTWARE,STUDIES
WHERE SOFTWARE.NAME(+) = STUDIES.NAME
GROUP BY SPLACE,SOFTWARE.NAME;

SPLACE    NAME     COUNT(SOFTWARE.NAME) SUM(SOLD) SUM(SCOST*SOLD)
--------- -------- -------------------- --------- ---------------
APPLE     QUADIR                      2       135         96894.3
BDPS      JULIANA                     1         0               0
          REMITHA                     2        57           51975
          VIJAYA                      1         6            5400
BRILLIANT                             0
CCIT                                  0
PRAGATHI  KAMALA                      1         7           63000
          PATRICK                     1        11            8250
                                      0
SABHARI   ANAND                       2        52        84697.85
          MARY                        3       111          200700
          RAMESH                      2        77        55296.35
          REVATHI                     2        17           50200
SSIL      JAGDESH                     2        23           49200
```

7. Display details of software developed in DBASE by male programmers who belong to the institute on which most number of programmers studied.

```
SELECT SOFTWARE.*
FROM SOFTWARE,PROGRAMMER,STUDIES
WHERE SOFTWARE.NAME = PROGRAMMER.NAME AND
PROGRAMMER.NAME = STUDIES.NAME AND
SEX = 'M' AND
DEV_IN='DBASE'  AND
SPLACE = (SELECT SPLACE
         FROM STUDIES
```

```
GROUP BY SPLACE
HAVING COUNT(NAME)>=ALL(SELECT COUNT(NAME)
                        FROM STUDIES
                        GROUP BY SPLACE));
```

NAME	TITLE	DEV_IN	SCOST	DCOST	SOLD
RAMESH	HOTEL MANAGEMENT	DBASE	12000	35000	4

8. Display the details of the software that was developed by male programmers born after 1970 and female programmers born before 1975.

```
SELECT SOFTWARE.*
FROM PROGRAMMER,SOFTWARE
WHERE PROGRAMMER.NAME = SOFTWARE.NAME AND
((SEX = 'M' AND TO_CHAR(DOB,'YY')>70) OR
(SEX = 'F' AND TO_CHAR(DOB,'YY')<70));
```

NAME	TITLE	DEV_IN	SCOST	DCOST	SOLD
JULIANA	INVENTORY CONTROL	COBOL	3000	3500	0
KAMALA	PAYROLL PACKAGE	DBASE	9000	20000	7
REVATHI	HOTEL MANAGEMENT	PASCAL	1100	75000	2
REVATHI	QUIZ MASTER	BASIC	3200	2100	15
VIJAYA	ISK EDITOR	C	900	700	6

9. Display the details of the software that was developed in the language that is not the programmers first proficiency.

```
SELECT SOFTWARE.*
FROM SOFTWARE,PROGRAMMER
WHERE SOFTWARE.NAME = PROGRAMMER.NAME AND PROF1<>DEV_IN;
```

NAME	TITLE	DEV_IN	SCOST	DCOST	SOLD
ANAND	PARACHUTES	BASIC	399.95	6000	43
JAGDESH	SERIAL LINK UTILITY	JAVA	800	7500	9
KAMALA	PAYROLL PACKAGE	DBASE	9000	20000	7
MARY	ACC S/W	ORACLE	18000	85000	4
MARY	CODE GENERATOR	C	4500	20000	23
QUADIR	TALLY	C	1900	3400	21
RAMESH	HOTEL MANAGEMENT	DBASE	12000	35000	4
REMITHA	ISR HELP MANAGEMENT	ASSEMBLY	2500	6000	6
REVATHI	QUIZ MASTER	BASIC	3200	2100	15
VIJAYA	ISK EDITOR	C	900	700	6

10. Display details of software that was developed in the language which is neither first nor second proficiency of the programmer.

```
SELECT SOFTWARE.*
FROM SOFTWARE,PROGRAMMER
WHERE SOFTWARE.NAME = PROGRAMMER.NAME
AND PROF1<>DEV_IN AND PROF2<>DEV_IN
```

NAME	TITLE	DEV_IN	SCOST	DCOST	SOLD
MARY	CODE GENERATOR	C	4500	20000	23

11. Display details of software developed by male students of SABHARI.

```
SELECT SOFTWARE.*
FROM SOFTWARE,PROGRAMMER,STUDIES
WHERE SOFTWARE.NAME = PROGRAMMER.NAME AND
   SOFTWARE.NAME = STUDIES.NAME AND SEX = 'M' AND
   SPLACE = 'SABHARI';
```

NAME	TITLE	DEV_IN	SCOST	DCOST	SOLD

```
ANAND    PARACHUTES           BASIC       399.95      6000        43
ANAND    VIDEO TILING PACK    PASCAL        7500     16000         9
RAMESH   HOTEL MANAGEMENT     DBASE        12000     35000         4
RAMESH   READ ME              PASCAL       99.95      4500        73
```

12. Display the names who have not developed any package.

```
SELECT NAME
FROM PROGRAMMER
WHERE NAME NOT IN (SELECT NAME FROM SOFTWARE);

NAME
--------
ALTAF
REBECCA
```

13. What is the total cost of the software developed by the programmers of APPLE?

```
SELECT SUM(DCOST) "TOTAL COST"
FROM SOFTWARE,STUDIES
WHERE SOFTWARE.NAME = STUDIES.NAME AND SPLACE = 'APPLE';

TOTAL COST
----------
      3930
```

14. Who are the programmers who joined on the same day?

```
CREATE VIEW DOJ_VW AS
SELECT DOJ
FROM PROGRAMMER
GROUP BY DOJ
HAVING COUNT(DOJ)>1;

BREAK ON DOJ

SELECT PROGRAMMER.DOJ,NAME
FROM DOJ_VW,PROGRAMMER
WHERE PROGRAMMER.DOJ = DOJ_VW.DOJ;

DOJ       NAME
--------- --------
21-APR-90 JULIANA
          PATRICK
02-JAN-92 KAMALA
          REVATHI
```

15. Who are the programmers who have the same PROF2?

```
CREATE VIEW PROF2_VW AS
SELECT PROF2
FROM PROGRAMMER
GROUP BY PROF2
HAVING COUNT(NAME)>1;

SELECT PROGRAMMER.PROF2,NAME
FROM PROF2_VW,PROGRAMMER
WHERE PROGRAMMER.PROF2 = PROF2_VW.PROF2;

PROF2    NAME
-------- --------
BASIC    ANAND
         REVATHI
C        QUADIR
         VIJAYA
COBOL    ALTAF
         REBECCA
DBASE    JULIANA
```

```
RAMESH
KAMALA
```

16. Display the total sales value of software, institute wise?

```
SELECT SPLACE,SUM(SCOST*SOLD) "TOTAL SALES VALUE"
FROM SOFTWARE,STUDIES
WHERE SOFTWARE.NAME = STUDIES.NAME
GROUP BY SPLACE;

SPLACE    TOTAL SALES VALUE
--------- -----------------
APPLE               96894.3
BDPS                  57375
PRAGATHI              71250
SABHARI            390894.2
SSIL                  49200
```

17. In which institute did the person who developed the costliest package study?

```
SELECT SPLACE
FROM STUDIES
WHERE NAME IN(SELECT NAME
              FROM SOFTWARE
              WHERE SCOST >=ALL(SELECT SCOST
                                FROM SOFTWARE));

SPLACE
---------
SABHARI
```

18. Which language listed in prof1 and prof2 has not been used to develop any package?

```
SELECT PROF1 LANGUAGE
FROM PROGRAMMER
WHERE PROF1 NOT IN(SELECT DEV_IN FROM SOFTWARE)
UNION
SELECT PROF1 LANGUAGE
FROM PROGRAMMER
WHERE PROF2 NOT IN(SELECT DEV_IN FROM SOFTWARE)

LANGUAGE
--------
CLIPPER
FOXPRO
PASCAL
```

19. How much does the person who developed the highest selling package earn, and what course did he/she undergo?

```
SELECT SALARY,COURSE
FROM SOFTWARE,PROGRAMMER,STUDIES
WHERE SOFTWARE.NAME = PROGRAMMER.NAME
AND SOFTWARE.NAME = STUDIES.NAME
AND SOLD = (SELECT MAX(SOLD) FROM SOFTWARE)

   SALARY COURS
--------- -----
     3000 HDCP
```

20. How many months will it take for each programmer to recover the cost of the course underwent?

```
SELECT PROGRAMMER.NAME, CEIL(CCOST/SALARY) MONTHS
FROM PROGRAMMER,STUDIES
WHERE PROGRAMMER.NAME = STUDIES.NAME

NAME        MONTHS
-------- ---------
ALTAF             3
```

```
ANAND            2
JAGDESH          1
JULIANA          8
KAMALA           2
MARY             1
PATRICK          2
QUADIR           5
RAMESH           2
REBECCA          5
REMITHA          2
REVATHI          2
VIJAYA          14
```

21. Which is the costliest package developed by a person with under 3 years experience?

```
SELECT TITLE
FROM SOFTWARE,PROGRAMMER
WHERE SOFTWARE.NAME = PROGRAMMER.NAME AND
(SYSDATE-DOJ)/365 < 3 AND
SCOST >= ALL (SELECT SCOST
         FROM SOFTWARE,PROGRAMMER
         WHERE SOFTWARE.NAME = PROGRAMMER.NAME
         AND (SYSDATE-DOJ)/365 < 3)

TITLE
--------------------
HOTEL MANAGEMENT
```

22. What is the average salary for those whose software's sales value is more than 50000?

```
SELECT AVG(SALARY) "AVERAGE SALARY"
FROM PROGRAMMER,SOFTWARE
WHERE PROGRAMMER.NAME = SOFTWARE.NAME AND SCOST*SOLD>50000;

AVERAGE SALARY
--------------
          3620
```

23. How many packages were developed by the students who studied in the institute that charge the lowest course fee?

```
SELECT COUNT(TITLE) "NO.OF PACKAGES"
FROM SOFTWARE,STUDIES
WHERE SOFTWARE.NAME = STUDIES.NAME AND
   CCOST = (SELECT MIN(CCOST)
              FROM STUDIES);

NO.OF PACKAGES
--------------
             2
```

24. How many packages were developed by the person who developed the cheapest package? Where did he/ she study?

```
SELECT COUNT(TITLE) "NO.OF PACKAGES",SPLACE
FROM SOFTWARE,STUDIES
WHERE SOFTWARE.NAME = STUDIES.NAME AND
   SOFTWARE.NAME IN (SELECT NAME
                     FROM SOFTWARE
                     WHERE SCOST<=ALL(SELECT SCOST
                                      FROM SOFTWARE))
GROUP BY SPLACE;

NO.OF PACKAGES SPLACE
-------------- ---------
             2 SABHARI
```

25. How many packages were developed by female programmers earning more than the highest paid male programmer?

```
SELECT COUNT(TITLE) "NO.OF PACKAGES"
FROM SOFTWARE,PROGRAMMER
WHERE SOFTWARE.NAME = PROGRAMMER.NAME AND SEX='F' AND
        SALARY > (SELECT MAX(SALARY)
                    FROM PROGRAMMER
                    WHERE SEX='M');

NO.OF PACKAGES
--------------
             3
```

26. How many packages were developed by the most experienced programmers of BDPS.

```
SELECT COUNT(TITLE) "NO.OF PACKAGES"
FROM SOFTWARE,PROGRAMMER,STUDIES
WHERE SOFTWARE.NAME = PROGRAMMER.NAME AND
   PROGRAMMER.NAME = STUDIES.NAME AND
   SPLACE = 'BDPS' AND DOJ = (SELECT MIN(DOJ)
                              FROM PROGRAMMER,STUDIES
                              WHERE PROGRAMMER.NAME = STUDIES.NAME
                              AND SPLACE='BDPS');

NO.OF PACKAGES
--------------
             1
```

27. List the programmers (from SOFTWARE table) and institutes they studied including those who didn't develop any package.

```
SELECT PROGRAMMER.NAME,SPLACE,TITLE
FROM SOFTWARE,PROGRAMMER, STUDIES
WHERE PROGRAMMER.NAME = SOFTWARE.NAME(+) AND
PROGRAMMER.NAME = STUDIES.NAME(+);

NAME     SPLACE    TITLE
-------- --------- --------------------
ALTAF    CCIT
ANAND    SABHARI   PARACHUTES
ANAND    SABHARI   VIDEO TILING PACK
JAGDESH  SSIL      SERIAL LINK UTILITY
JAGDESH  SSIL      SHARES MANAGEMENT
JULIANA  BDPS      INVENTORY CONTROL
KAMALA   PRAGATHI  PAYROLL PACKAGE
MARY     SABHARI   ACC S/W
MARY     SABHARI   CODE GENERATOR
MARY     SABHARI   READ ME
PATRICK  PRAGATHI  GRAPHIC EDITOR
QUADIR   APPLE     FILE UTILITY
QUADIR   APPLE     TALLY
RAMESH   SABHARI   HOTEL MANAGEMENT
RAMESH   SABHARI   READ ME
REBECCA  BRILLIANT
REMITHA  BDPS      PC UTILITIES
REMITHA  BDPS      ISR HELP MANAGEMENT
REVATHI  SABHARI   HOTEL MANAGEMENT
REVATHI  SABHARI   QUIZ MASTER
VIJAYA   BDPS      ISK EDITOR
```

28. List each prof1 with the number of programmers having that prof1 and the number of packages developed in that prof1.

```
SELECT PROF1,COUNT(DISTINCT PROGRAMMER.NAME) "NO.OF PROGRAMMERS",
        COUNT(DISTINCT TITLE) "NO.OF PACKAGES"
FROM PROGRAMMER,SOFTWARE
WHERE PROGRAMMER.NAME = SOFTWARE.NAME
GROUP BY PROF1;
```

```
PROF1    NO.OF PROGRAMMERS NO.OF PACKAGES
-------- ----------------- --------------
ASSEMBLY                  1              2
C                         2              3
C++                       1              3
COBOL                     1              1
FOXPRO                    1              1
ORACLE                    1              2
PASCAL                    4              6
```

29. List programmer names (from programmer table) and number of packages each developed.

```
SELECT PROGRAMMER.NAME,COUNT(TITLE) "NO.OF PACKAGES"
FROM PROGRAMMER,SOFTWARE
WHERE PROGRAMMER.NAME = SOFTWARE.NAME
GROUP BY PROGRAMMER.NAME
```

```
NAME     NO.OF PACKAGES
-------- --------------
ANAND                 2
JAGDESH               2
JULIANA               1
KAMALA                1
MARY                  3
PATRICK               1
QUADIR                2
RAMESH                2
REMITHA               2
REVATHI               2
VIJAYA                1
```

30. List all the details of programmers who has done a course at SSIL

```
SELECT PROGRAMMER.*
FROM PROGRAMMER,STUDIES
WHERE PROGRAMMER.NAME = STUDIES.NAME AND SPLACE = 'SSIL';
```

```
NAME     DOB       DOJ       S PROF1    PROF2       SALARY
-------- --------- --------- - -------- -------- ---------
JAGDESH  06-OCT-70 04-OCT-94 M ORACLE   JAVA          4100
```

EMP - Contains information about the employees of the sample company.

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7369	SMITH	CLERK	7902	17-DEC-80	800		20
7499	ALLEN	SALESMAN	7698	20-FEB-81	1600	300	30
7521	WARD	SALESMAN	7698	22-FEB-81	1250	500	30
7566	JONES	MANAGER	7839	02-APR-81	2975		20
7654	MARTIN	SALESMAN	7698	28-SEP-81	1250	1400	30
7698	BLAKE	MANAGER	7839	01-MAY-81	2850		30
7782	CLARK	MANAGER	7839	09-JUN-81	2450		30
7788	SCOTT	ANALYST	7566	09-DEC-82	3000		20
7839	KING	PRESIDENT		17-NOV-81	5000		10
7844	TURNER	SALESMAN	7698	08-SEP-81	1500	0	30
7876	ADAMS	CLERK	7788	12-JAN-83	1100		20
7900	JAMES	CLERK	7698	03-DEC-81	950		30
7902	FORD	ANALYST	7566	03-DEC-81	3000		20
7934	MILLER	CLERK	7782	23-JAN-82	1300		10

DEPT - Contains information about the departments in the company.

DEPTNO	DNAME	LOC
10	ACCOUNTING	NEW YORK
20	RESEARCH	DALLAS
30	SALES	CHICAGO
40	OPERATIONS	BOSTON

```
CREATE TABLE DEPT
(
 DEPTNO     NUMBER(2) NOT NULL,
 DNAME      CHAR(14),
 LOC        CHAR(13),
 CONSTRAINT DEPT_PRIMARY_KEY PRIMARY KEY (DEPTNO)
);

INSERT INTO DEPT VALUES (10,'ACCOUNTING','NEW YORK');
INSERT INTO DEPT VALUES (20,'RESEARCH','DALLAS');
INSERT INTO DEPT VALUES (30,'SALES','CHICAGO');
INSERT INTO DEPT VALUES (40,'OPERATIONS','BOSTON');

CREATE TABLE EMP
(
 EMPNO      NUMBER(4) NOT NULL,
 ENAME      CHAR(10),
 JOB        CHAR(9),
 MGR        NUMBER(4) CONSTRAINT EMP_SELF_KEY REFERENCES EMP (EMPNO),
 HIREDATE   DATE,
 SAL        NUMBER(7,2),
 COMM       NUMBER(7,2),
 DEPTNO     NUMBER(2) NOT NULL,
 CONSTRAINT EMP_FOREIGN_KEY FOREIGN KEY (DEPTNO) REFERENCES DEPT (DEPTNO),
 CONSTRAINT EMP_PRIMARY_KEY PRIMARY KEY (EMPNO)
);

INSERT INTO EMP VALUES (7839,'KING','PRESIDENT',NULL,'17-NOV-81',5000,NULL,10);
INSERT INTO EMP VALUES (7698,'BLAKE','MANAGER',7839,'1-MAY-81',2850,NULL,30);
INSERT INTO EMP VALUES (7782,'CLARK','MANAGER',7839,'9-JUN-81',2450,NULL,10);
INSERT INTO EMP VALUES (7566,'JONES','MANAGER',7839,'2-APR-81',2975,NULL,20);
INSERT INTO EMP VALUES (7654,'MARTIN','SALESMAN',7698,'28-SEP-81',1250,1400,30);
INSERT INTO EMP VALUES (7499,'ALLEN','SALESMAN',7698,'20-FEB-81',1600,300,30);
INSERT INTO EMP VALUES (7844,'TURNER','SALESMAN',7698,'8-SEP-81',1500,0,30);
INSERT INTO EMP VALUES (7900,'JAMES','CLERK',7698,'3-DEC-81',950,NULL,30);
INSERT INTO EMP VALUES (7521,'WARD','SALESMAN',7698,'22-FEB-81',1250,500,30);
INSERT INTO EMP VALUES (7902,'FORD','ANALYST',7566,'3-DEC-81',3000,NULL,20);
INSERT INTO EMP VALUES (7369,'SMITH','CLERK',7902,'17-DEC-80',800,NULL,20);
```

```
INSERT INTO EMP VALUES (7788,'SCOTT','ANALYST',7566,'09-DEC-82',3000,NULL,20);
INSERT INTO EMP VALUES (7876,'ADAMS','CLERK',7788,'12-JAN-83',1100,NULL,20);
INSERT INTO EMP VALUES (7934,'MILLER','CLERK',7782,'23-JAN-82',1300,NULL,10);
```

Oracle Assignment – I

Solve the following queries, which are based on the EMP and DEPT tables:

1. List all information about employees in the EMP table.

```
SELECT * FROM EMP;
```

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7369	SMITH	CLERK	7902	17-DEC-80	800		20
7499	ALLEN	SALESMAN	7698	20-FEB-81	1600	300	30
7521	WARD	SALESMAN	7698	22-FEB-81	1250	500	30
7566	JONES	MANAGER	7839	02-APR-81	2975		20
7654	MARTIN	SALESMAN	7698	28-SEP-81	1250	1400	30
7698	BLAKE	MANAGER	7839	01-MAY-81	2850		30
7782	CLARK	MANAGER	7839	09-JUN-81	2450		10
7788	SCOTT	ANALYST	7566	19-APR-87	3000		20
7839	KING	PRESIDENT		17-NOV-81	5000		10
7844	TURNER	SALESMAN	7698	08-SEP-81	1500	0	30
7876	ADAMS	CLERK	7788	23-MAY-87	1100		20
7900	JAMES	CLERK	7698	03-DEC-81	950		30
7902	FORD	ANALYST	7566	03-DEC-81	3000		20
7934	MILLER	CLERK	7782	23-JAN-82	1300		10

2. List all information about departments in the DEPT table.

```
SELECT * FROM DEPT;
```

DEPTNO	DNAME	LOC
10	ACCOUNTING	NEW YORK
20	RESEARCH	DALLAS
30	SALES	CHICAGO
40	OPERATIONS	BOSTON

3. List the employee number, name, job, title, and hire date of employees in department 10.

```
SELECT EMPNO,ENAME,JOB,HIREDATE
FROM EMP
WHERE DEPTNO=10;
```

EMPNO	ENAME	JOB	HIREDATE
7782	CLARK	MANAGER	09-JUN-81
7839	KING	PRESIDENT	17-NOV-81
7934	MILLER	CLERK	23-JAN-82

4. Select name and salary of employees who are clerks.

```
SELECT ENAME,SAL
FROM EMP
WHERE JOB='CLERK';
```

ENAME	SAL
SMITH	800
ADAMS	1100
JAMES	950
MILLER	1300

5. List the department number and name for all departments with department numbers greater than or equal to 20.

```
SELECT DEPTNO,DNAME
FROM DEPT
WHERE DEPTNO>=20;
```

DEPTNO	DNAME
20	RESEARCH

```
        30 SALES
        40 OPERATIONS
```

6. List the name of the employees having salary less than 2500.

```
SELECT ENAME
FROM EMP
WHERE SAL<=2500;

ENAME
----------
SMITH
ALLEN
WARD
MARTIN
CLARK
TURNER
ADAMS
JAMES
MILLER
```

7. Select name, salary and commission of employees whose commission is greater than their salary.

```
SELECT ENAME,SAL,COMM
FROM EMP
WHERE COMM>SAL;

ENAME             SAL       COMM
---------- --------- ---------
MARTIN          1250       1400
```

8. List the employee number and name of the president.

```
SELECT EMPNO,ENAME
FROM EMP
WHERE JOB='PRESIDENT'

    EMPNO ENAME
--------- ----------
     7839 KING
```

9. List the employees who do not get any commission.

```
SELECT *
FROM EMP
WHERE COMM IS NULL;

    EMPNO ENAME      JOB              MGR HIREDATE         SAL       COMM     DEPTNO
--------- ---------- --------- --------- --------- --------- --------- ---------
     7369 SMITH      CLERK          7902 17-DEC-80       800                   20
     7566 JONES      MANAGER        7839 02-APR-81      2975                   20
     7698 BLAKE      MANAGER        7839 01-MAY-81      2850                   30
     7782 CLARK      MANAGER        7839 09-JUN-81      2450                   10
     7788 SCOTT      ANALYST        7566 19-APR-87      3000                   20
     7839 KING       PRESIDENT           17-NOV-81      5000                   10
     7876 ADAMS      CLERK          7788 23-MAY-87      1100                   20
     7900 JAMES      CLERK          7698 03-DEC-81       950                   30
     7902 FORD       ANALYST        7566 03-DEC-81      3000                   20
     7934 MILLER     CLERK          7782 23-JAN-82      1300                   10
```

10. List all the employees in DEPTNO 10 other than KING.

```
SELECT *
FROM EMP
WHERE DEPTNO=20 AND ENAME<>'KING';

    EMPNO ENAME      JOB              MGR HIREDATE         SAL       COMM     DEPTNO
--------- ---------- --------- --------- --------- --------- --------- ---------
     7369 SMITH      CLERK          7902 17-DEC-80       800                   20
     7566 JONES      MANAGER        7839 02-APR-81      2975                   20
     7788 SCOTT      ANALYST        7566 19-APR-87      3000                   20
     7876 ADAMS      CLERK          7788 23-MAY-87      1100                   20
```

```
     7902 FORD       ANALYST        7566 03-DEC-81      3000                    20
```

11. Display names of the employees whose job is either analyst or clerk.

```
SELECT ENAME
FROM EMP
WHERE JOB IN('ANALYST','CLERK');

ENAME
----------
SMITH
SCOTT
ADAMS
JAMES
FORD
MILLER
```

12. Display different kind of jobs available.

```
SELECT DISTINCT JOB
FROM EMP;

JOB
---------
ANALYST
CLERK
MANAGER
PRESIDENT
SALESMAN
```

13. List names of all employees whose names are 4 letters long.

```
SELECT ENAME
FROM EMP
WHERE ENAME LIKE '____';

ENAME
----------
WARD
KING
FORD
```

14. List names of all employees whose names end with letter ‘R’.

```
SELECT ENAME
FROM EMP
WHERE ENAME LIKE '%R';

ENAME
----------
TURNER
MILLER
```

15. List names of all employees whose names start with ‘B’ or ‘M’.

```
SELECT ENAME
FROM EMP
WHERE ENAME LIKE 'B%' OR ENAME LIKE 'M%';

ENAME
----------
MARTIN
BLAKE
MILLER
```

16. If a new person HENRY joins the organization in place of TURNER on 9th December 1985 with EMPNO 7333, make necessary change in the EMP table.

```
UPDATE EMP
SET EMPNO=7333,ENAME='HENRY',HIREDATE='09-DEC-85'
WHERE ENAME='TURNER';
```

17. Retrieve the names and jobs of the employees working in the department number 20. Display the result the 'Employee-job' as column heading and arranging the columns with '-' in between like 'Smith-Clerk'.

```
SELECT ENAME||'-'||JOB "EMPLOYEE-JOB"
FROM EMP
WHERE DEPTNO=20;

EMPLOYEE-JOB
--------------------
SMITH-CLERK
JONES-MANAGER
SCOTT-ANALYST
ADAMS-CLERK
FORD-ANALYST
```

18. Show what length names appear in the EMP table. Eliminate the duplicate length from the rows returned.

```
SELECT DISTINCT(LENGTH(ENAME)) "LENGTH"
FROM EMP;

   LENGTH
---------
        4
        5
        6
```

19. Determine the average earning of an employee working in department 30.

```
SELECT AVG(SAL) "AVERAGE EARNINGS"
FROM EMP
WHERE DEPTNO=30;

AVERAGE EARNINGS
----------------
       1566.6667
```

20. Determine the new comer to the organization.

```
SELECT *
FROM EMP
WHERE HIREDATE = (SELECT MAX(HIREDATE) FROM EMP)

    EMPNO ENAME      JOB              MGR HIREDATE         SAL       COMM     DEPTNO
--------- ---------- --------- --------- --------- --------- --------- ---------
     7876 ADAMS      CLERK          7788 23-MAY-87      1100                    20
```

21. Find out the total salary of each department. Display the department number and the total salary.

```
SELECT DEPTNO,SUM(SAL) "TOTAL SAL"
FROM EMP
GROUP BY DEPTNO;

   DEPTNO TOTAL SAL
--------- ---------
       10      8750
       20     10875
       30      9400
```

Oracle Assignment – II

Solve the following queries, which are based on the EMP and DEPT tables:

1. List the names and hire dates of the employees in DEPTNO 20, display the hire date formatted as 21.03.87.

```
SELECT ENAME,TO_CHAR(HIREDATE,'DD.MM.YY') "HIREDATE"
FROM EMP;

ENAME      HIREDATE
```

```
---------- ---------------------------------------------------------
SMITH      17.12.80
ALLEN      20.02.81
WARD       22.02.81
JONES      02.04.81
MARTIN     28.09.81
BLAKE      01.05.81
CLARK      09.06.81
SCOTT      19.04.87
KING       17.11.81
HENRY      09.12.85
ADAMS      23.05.87
JAMES      03.12.81
FORD       03.12.81
MILLER     23.01.82
```

2. How many months did the president work for the company? Round to the nearest whole number of months.

```
SELECT ROUND(MONTHS_BETWEEN(SYSDATE,HIREDATE)) "NO.OF MONTHS"
FROM EMP
WHERE JOB='PRESIDENT'

NO.OF MONTHS
------------
         147
```

3. Find the day of the week on which SMITH joined.

```
SELECT TO_CHAR(HIREDATE,'FMDAY') "DAY OF WEEK"
FROM EMP
WHERE ENAME='SMITH'

DAY OF WEEK
---------------------------------------------------------------------------
WEDNESDAY
```

4. Find out time of the day (in HH24:HI:SSSS) on which FORD joined.

```
SELECT TO_CHAR(HIREDATE,'HH24:MI:SSSS')  "TIME"
FROM EMP
WHERE ENAME='FORD';

TIME
---------------------------------------------------------------------------
00:00:0000
```

5. Find out the day of the month on which JAMES joined.

```
SELECT TO_CHAR(HIREDATE,'DD') "DAY OF MONTH"
FROM EMP
WHERE ENAME='JAMES';

DAY OF MONTH
---------------------------------------------------------------------------
03
```

6. Find out the quarter of the year the employees joined.

```
SELECT ENAME,TO_CHAR(HIREDATE,'Q') "QUATER"
FROM EMP

ENAME      QUATER
---------- ---------------------------------------------------------
SMITH      4
ALLEN      1
WARD       1
JONES      2
MARTIN     3
```

```
BLAKE      2
CLARK      2
SCOTT      2
KING       4
HENRY      4
ADAMS      2
JAMES      4
FORD       4
MILLER     1
```

7. List the names, department of all employees whose immediate anniversary does not exist in the first quarter of the year.

```
SELECT ENAME,DNAME
FROM EMP,DEPT
WHERE DEPT.DEPTNO = EMP.DEPTNO AND TO_CHAR(HIREDATE,'Q') <> 1;

ENAME      DNAME
---------- --------------
SMITH      RESEARCH
JONES      RESEARCH
MARTIN     SALES
BLAKE      SALES
CLARK      ACCOUNTING
SCOTT      RESEARCH
KING       ACCOUNTING
HENRY      SALES
ADAMS      RESEARCH
JAMES      SALES
FORD       RESEARCH
```

8. Write a query to count the number of people in DEPTNO 30 who receive a salary and the number of people who receive a commission.

```
CREATE VIEW EMPVW1 AS
SELECT COUNT(EMPNO) CNT1
FROM EMP
WHERE SAL<>0 AND COMM<>0 AND DEPTNO=30;

CREATE VIEW EMPVW2 AS
SELECT COUNT(EMPNO) CNT2
FROM EMP
WHERE SAL<>0 AND COMM=0 AND DEPTNO=30;

SELECT CNT1,CNT2
FROM EMPVW1,EMPVW2;

     CNT1      CNT2
--------- ---------
        3         1
```

9. Compute the average, minimum and maximum salaries of those groups of employees having the job of CLERK or MANAGER.

```
SELECT JOB,AVG(SAL) "AVG", MIN(SAL) "MIN", MAX(SAL) "MAX"
FROM EMP

WHERE JOB = 'CLERK' OR JOB='MANAGER'
GROUP BY JOB;

JOB              AVG       MIN       MAX
--------- --------- --------- ---------
CLERK       1141.25       880      1430
MANAGER   3034.1667      2695    3272.5
```

10. List the department no and the maximum salary earned in DEPTNO =20

```
SELECT DEPTNO,MAX(SAL) "MAXIMUM SAL"
```

```
FROM EMP
WHERE DEPTNO=20
GROUP BY DEPTNO

   DEPTNO MAXIMUM SAL
--------- -----------
       20        3000
```

11. Give a hike of 10% to each employee.

```
UPDATE EMP
SET SAL=SAL*1.1;
```

12. Select all employees whose names fall between 'A' and 'G' alphabetical range.

```
SELECT *
FROM EMP
WHERE ENAME BETWEEN 'A' AND 'G';
```

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7499	ALLEN	SALESMAN	7698	20-FEB-81	1760	300	30
7698	BLAKE	MANAGER	7839	01-MAY-81	3135		30
7782	CLARK	MANAGER	7839	09-JUN-81	2695		10
7876	ADAMS	CLERK	7788	23-MAY-87	1210		20
7902	FORD	ANALYST	7566	03-DEC-81	3300		20

13. Find out maximum salaries department wise excluding those who are less than 3000

```
SELECT DEPTNO,MAX(SAL) "MAX"
FROM EMP
WHERE SAL<3000
GROUP BY DEPTNO;

   DEPTNO       MAX
--------- ---------
       10      2695
       20      1210
       30      1760
```

14. Give commission equal to 1% of their salaries to employees having commission as NULL.

```
UPDATE EMP
SET SAL = SAL * 1.1
WHERE COMM IS NULL;
```

15. List employees under their own department name like

```
Department 10         Department 20                  Department 30
-------------------------------------------------------------------------------------------------------------------
Scott
                      Robert
                                                       John
```

Assume there are three departments only

```
SELECT DECODE(DEPTNO,10,ENAME,'') "DEPARTMENT 10",
   DECODE(DEPTNO,20,ENAME,'') "DEPARTMENT 20",
   DECODE(DEPTNO,30,ENAME,'') "DEPARTMENT 30"
FROM EMP;
DEPARTMENT 10        DEPARTMENT 20        DEPARTMENT 30
-------------------- -------------------- --------------------
                     SMITH
                                          ALLEN
                                          WARD
                     JONES
                                          MARTIN
                                          BLAKE
CLARK
                     SCOTT
KING
                                          HENRY
                     ADAMS
```

JAMES
FORD
MILLER

16. List employee name, sal and his income group as 'LOW' of 'HIGH' depending on the salary amount.(If the salary is less than 5000 then he is in 'LOW' income group or else in 'HIGH' income group)

ENAME SALARY LOW HIGH

SCOTT 2000 LOW
JOHN 6000 HIGH

```
SELECT ENAME, SAL, DECODE(ABS(5000-SAL), 5000-SAL, 'LOW', '') "LOW",
        DECODE(ABS(SAL-5000), SAL-5000,'HIGH','') "HIGH"
FROM EMP;
```

```
ENAME             SAL LOW HIGH
---------- --------- --- ----
SMITH              968 LOW
ALLEN             1760 LOW
WARD              1375 LOW
JONES           3599.75 LOW
MARTIN            1375 LOW
BLAKE           3448.5 LOW
CLARK           2964.5 LOW
SCOTT             3630 LOW
KING              6050     HIGH
HENRY             1650 LOW
ADAMS             1331 LOW
JAMES           1149.5 LOW
FORD              3630 LOW
MILLER            1573 LOW
```

17. List ename, salary for all employees showing salary in bar chart form

Ename Salary Graph

SCOTT 2000 ****
JOHN 6000 **********
Scale one for 500

```
SELECT ENAME,SAL,LPAD('*',SAL/500,'*') "GRAPH"
FROM EMP;
```

```
ENAME             SAL GRAPH
---------- --------- --------------------
SMITH              968 *
ALLEN             1760 ***
WARD              1375 **
JONES           3599.75 *******
MARTIN            1375 **
BLAKE           3448.5 ******
CLARK           2964.5 *****
SCOTT             3630 *******
KING              6050 ************
HENRY             1650 ***
ADAMS             1331 **
JAMES           1149.5 **
FORD              3630 *******
MILLER            1573 ***
```

18. What is the length of the longest employee name, and by how many characters is longer than its nearest one.

```
CREATE VIEW EMPVW_LENGTH2 AS
SELECT MAX(LENGTH(ENAME)) LENGTH2
FROM EMP;

CREATE VIEW EMPVW_LENGTH1 AS
SELECT MAX(LENGTH(ENAME)) LENGTH1
FROM EMP
WHERE LENGTH(ENAME) < (SELECT MAX(ENAME) FROM EMP);

SELECT LENGTH2, LENGTH2-LENGTH1 DIFFERENCE
FROM EMPVW_LENGTH2, EMPVW_LENGTH1;

  LENGTH2 DIFFERENCE
--------- ----------
        6          1
```

19. Find out the locations of the employees.

```
SELECT ENAME,LOC
FROM EMP,DEPT
WHERE EMP.DEPTNO = DEPT.DEPTNO;

ENAME      LOC
---------- -------------
SMITH      DALLAS
ALLEN      CHICAGO
WARD       CHICAGO
JONES      DALLAS
MARTIN     CHICAGO
BLAKE      CHICAGO
CLARK      NEW YORK
SCOTT      DALLAS
KING       NEW YORK
HENRY      CHICAGO
ADAMS      DALLAS
JAMES      CHICAGO
FORD       DALLAS
MILLER     NEW YORK
```

20. Who was the last employee hired in each department.

```
CREATE VIEW EMPVW_LAST AS
SELECT DEPTNO,MAX(HIREDATE) HIRE_DATE
FROM EMP
GROUP BY DEPTNO;

SELECT EMP.DEPTNO,ENAME
FROM EMP,EMPVW_LAST
WHERE EMP.DEPTNO = EMPVW_LAST.DEPTNO AND HIREDATE = HIRE_DATE

   DEPTNO ENAME
--------- ----------
       10 MILLER
       20 ADAMS
       30 HENRY
```

21. Display name and salary of the employee who is working in CHICAGO.

```
SELECT ENAME,SAL
FROM EMP,DEPT
WHERE EMP.DEPTNO = DEPT.DEPTNO AND LOC = 'CHICAGO';

ENAME            SAL
```

```
---------- ---------
ALLEN           1760
WARD            1375
MARTIN          1375
BLAKE           3135
HENRY           1650
JAMES           1045
```

22. How many employees work in New York.

```
SELECT COUNT(ENAME) "NO.OF EMPLOYEES"
FROM EMP,DEPT
WHERE EMP.DEPTNO = DEPT.DEPTNO AND LOC = 'NEW YORK';

NO.OF EMPLOYEES
---------------
              3
```

23. List the employee names and cities in which they work. Sort the list by city.

```
SELECT ENAME,LOC
FROM EMP,DEPT
WHERE EMP.DEPTNO = DEPT.DEPTNO
ORDER BY LOC;

ENAME      LOC
---------- -------------
ALLEN      CHICAGO
WARD       CHICAGO
MARTIN     CHICAGO
JAMES      CHICAGO
HENRY      CHICAGO
BLAKE      CHICAGO
SMITH      DALLAS
FORD       DALLAS
ADAMS      DALLAS
JONES      DALLAS
SCOTT      DALLAS
CLARK      NEW YORK
KING       NEW YORK
MILLER     NEW YORK
```

24. Display the names of the employees who are working in Sales or Research department.

```
SELECT ENAME
FROM EMP,DEPT
WHERE EMP.DEPTNO = DEPT.DEPTNO AND DNAME IN('SALES','RESEARCH')

ENAME
----------
SMITH
ALLEN
WARD
JONES
MARTIN
BLAKE
SCOTT
HENRY
ADAMS
JAMES
FORD
```

25. Find out the difference between the maximum salary earned by a person in DEPTNO 10 and minimum salary earned by a person in DEPTNO 30.

```
CREATE VIEW EMP_MAX AS
SELECT MAX(SAL) MAX_SAL
FROM EMP
```

```
WHERE DEPTNO = 10;

CREATE VIEW EMP_MIN AS
SELECT MIN(SAL) MIN_SAL
FROM EMP
WHERE DEPTNO = 30;

SELECT MAX_SAL-MIN_SAL DIFFERENCE
FROM EMP_MIN,EMP_MAX;

DIFFERENCE
----------
    4900.5
```

26. Find out the difference between average earnings of DEPTNO 30 and 40.

```
CREATE VIEW EMPVW_10 AS
SELECT AVG(SAL) AVG_SAL1
FROM EMP
WHERE DEPTNO=30;

CREATE VIEW EMPVW_40 AS
SELECT AVG(SAL) AVG_SAL2
FROM EMP
WHERE DEPTNO=40;

SELECT AVG_SAL1 - AVG_SAL2 "DIFFERENCE"
FROM EMPVW_30,EMPVW_40;

DIFFERENCE
----------
      1793
```

27. Find out the people whose salary is less than the average salary for DETPNO 20.

```
SELECT ENAME
FROM EMP
WHERE SAL < (SELECT AVG(SAL)
            FROM EMP
            WHERE DEPTNO=30);

ENAME
----------
SMITH
ALLEN
WARD
MARTIN
HENRY
ADAMS
JAMES
MILLER
```

28. List number, name and job of each employee and the person's manager and the manager's job.

```
SELECT A.EMPNO,A.ENAME,A.JOB,B.ENAME,B.JOB
FROM EMP B, EMP A
WHERE A.MGR = B.EMPNO;

    EMPNO ENAME      JOB       ENAME      JOB
--------- ---------- --------- ---------- ---------
     7369 SMITH      CLERK     FORD       ANALYST
     7499 ALLEN      SALESMAN  BLAKE      MANAGER
     7521 WARD       SALESMAN  BLAKE      MANAGER
     7566 JONES      MANAGER   KING       PRESIDENT
     7654 MARTIN     SALESMAN  BLAKE      MANAGER
```

```
      7698 BLAKE      MANAGER   KING       PRESIDENT
      7782 CLARK      MANAGER   KING       PRESIDENT
      7788 SCOTT      ANALYST   JONES      MANAGER
      7333 HENRY      SALESMAN  BLAKE      MANAGER
      7876 ADAMS      CLERK     SCOTT      ANALYST
      7900 JAMES      CLERK     BLAKE      MANAGER
      7902 FORD       ANALYST   JONES      MANAGER
      7934 MILLER     CLERK     CLARK      MANAGER
```

29. Display information about people who have no people reporting to them.

```
SELECT * FROM EMP
MINUS
SELECT * FROM EMP
WHERE EMPNO IN(SELECT DISTINCT MGR FROM EMP)
```

30. Display information about people who have maximum people reporting to them.

```
CREATE VIEW EMPVW_REPORT AS
SELECT MGR,COUNT(EMPNO) NO_OF_PEOPLE
FROM EMP
GROUP BY MGR;

SELECT EMP.*
FROM EMP,EMPVW_REPORT
WHERE EMP.EMPNO = EMPVW_REPORT.MGR AND
          NO_OF_PEOPLE = (SELECT MAX(NO_OF_PEOPLE)
                          FROM EMPVW_REPORT);

    EMPNO ENAME      JOB              MGR HIREDATE         SAL       COMM     DEPTNO
--------- ---------- --------- --------- --------- --------- --------- ---------
     7698 BLAKE      MANAGER        7839 01-MAY-81    3448.5                   30
```

31. List the names of the people who are reporting to BLAKE.

```
SELECT ENAME
FROM EMP
WHERE MGR = (SELECT EMPNO
               FROM EMP
               WHERE ENAME = 'BLAKE');

ENAME
----------
ALLEN
WARD
MARTIN
HENRY
JAMES
```

32. Find the employees who earn more than the average salary in their own department.

```
CREATE VIEW EMPVW_AVG
AS
SELECT DEPTNO,AVG(SAL) "AVG_SAL"
FROM EMP
GROUP BY DEPTNO;

SELECT ENAME
FROM EMP,EMPVW_AVG
WHERE EMP.DEPTNO = EMPVW_AVG.DEPTNO AND SAL > AVG_SAL;

ENAME
----------
KING
FORD
SCOTT
JONES
BLAKE
```

33. List the locations of all the departments and the employees working in them including the departments without employees.

```
SELECT LOC,ENAME
FROM EMP,DEPT
WHERE EMP.DEPTNO(+) = DEPT.DEPTNO
ORDER BY LOC
```

```
LOC           ENAME
------------- ----------
BOSTON
CHICAGO       ALLEN
CHICAGO       BLAKE
CHICAGO       MARTIN
CHICAGO       JAMES
CHICAGO       WARD
CHICAGO       HENRY
DALLAS        SMITH
DALLAS        ADAMS
DALLAS        SCOTT
DALLAS        FORD
DALLAS        JONES
NEW YORK      CLARK
NEW YORK      KING
NEW YORK      MILLER
```