

# SQL ASSIGNMENT

**NAME:** SUJEETH K

**DATE:** 16-04-2024

## TABLE CREATION:

```
use sql_assignment;
```

```
create table Programmer(
```

```
NAME varchar(8) not null,
```

```
DOB date not null,
```

```
DOJ date not null,
```

```
SEX char(1) not null,
```

```
PROF1 varchar(8) not null,
```

```
PROF2 varchar(8) not null,
```

```
SALARY numeric(4) not null
```

```
);
```

```
alter table Programmer add primary key(Name);
```

```
alter table Programmer add constraint chk_sex check (sex in ('M', 'F'));
```

```
create table Software(
```

```
Name varchar(8) not null,
```

```
TITLE varchar(20) not null,
```

```
DEV_IN varchar(8) not null,
```

```
SCost decimal(7,2),
```

DCost decimal(5,0),

SOLD decimal(3,0)

);

alter table software add foreign key(Name) references programmer(Name);

create table Studies(

Name varchar(8) not null,

SPlace varchar(9) not null,

COURSE varchar(5),

CCost varchar(5)

);

alter table studies add foreign key(Name) references programmer(Name);

insert into programmer values('somedutt', '1966-04-21', '1992-04-21', 'M', 'pascal', 'basic', 3200);

insert into programmer values ("ANAND", "1966-04-12", "1992-04-21", "M", "PASCAL", "BASIC", 3200), ("ALTAF", "1964-07-02", "1990-11-13", "M", "CLIPPER", "COBOL", 2800), ("JULIANA", "1960-01-13", "1990-04-21", "F", "COBOL", "DBASE", 3000), ("KAMALA", "1968-10-30", "1992-01-02", "F", "C", "DBASE", 2900), ("MARY", "1970-06-24", "1991-02-01", "F", "CPP", "ORACLE", 4500), ("NELSON", "1985-09-11", "1989-03-11", "M", "COBOL", "DBASE", 2500), ("PATTRICK", "1965-11-10", "1990-04-21", "M", "PASCAL", " ", 2800), ("QADIR", "1965-08-31", "1990-04-21", "M", "ASSEMBLY", "C", 3000), ("RAMESH", "1967-05-03", "1991-02-26", "M", "PASCAL", "DBASE", 3200), ("REBECCA", "1997-01-01", "1990-12-01", "F", "BASIC", "COBOL", 2500), ("REMITHA", "1970-04-19", "1993-04-20", "F", "C", "ASSEMBLY", 3000), ("REVATHI", "1969-12-02", "1992-01-02", "F", "PASCAL", "BASIC", 3200), ("VIJAYA", "1965-12-14", "1992-05-02", "F", "FOXP", "C", 4500);

insert into software values('somedutt', 'parachutes', 'basic', 399.95, 6000, 43);

insert into software

```
values("MARY","README","CPP",100.00,1200,84),("ANAND","PARACHUTES","BASIC",399.95,6000,43),("ANAND","VIDEOTITLING","PASCAL",7500.00,16000,9),("JULIANA","INVENTORY","COBOL",3000.00,3500,0),("KAMALA","PAYROLLPRG","DBASE",9000.00,20000,7),("MARY","FINANCIALACC","ORACLE",18000.00,85000,4),("MARY","CODEGENERRATOP","C",4500.00,20000,23),("PATTRICK","README","CPP",300.00,1200,84),("QADIR","BOMBSAWAY","ASSEMBLY",750.00,5000,11),("QADIR","VACCINES","C",1900.00,3400,21),("RAMESH","HOTLIMGMT","DBASE",12000.00,35000,4),("RAMESH","DEADLEE","PASCAL",599.95,4500,73),("REMITHA","PCUTITLES","C",725.00,5000,51),("REMITHA","TSRHELPPKG","ASSEMBLY",2500.00,6000,7),("REVATHI","HOSPITALMGT","PASCAL",1100.00,75000,2),("VIJAYA","TSREDITOR","C",900.00,700,6);
```

insert into studies values('somdutt', 'sabhari', 'pgdca', '4500');

insert into studies values

```
("ALTAFA","COIT","DCA",7200),("JULIANA","BITS","MCA",22000),("KAMALA","PRAGATHI","DCP",5000),("MARY","SABHARI","PGDCA",4500),("NELSON","PRAGATHI","DAP",6200),("PATTRICK","PRAGATHI","DCAP",5200),("QADIR","APPLE","HDCP",14000),("RAMESH","SABHARI","PGDCA",4500),("REBECCA","BRILLIANT","DCP",11000),("REMITHA","BDPS","DCS",6000),("VIJAYA","BDPS","DCA",48000);
```

select \*from programmer;

select \*from software;

select \*from studies;

## QUERIES I

-- 1 --

```
select avg(sold) as Cost from software where dev_in='pascal' ;
```

-- 2 --

```
select Name, floor(datediff(curdate(),DOB )/365) as Age from Programmer ;
```

-- 3 --

```
select Programmer.Name as name, floor(datediff(curdate(),DOB )/365) as Age from Programmer join  
studies on Programmer.Name = studies.Name where studies.course='DCS';
```

-- 4 --

```
select max(sold) as MaxSales from software;
```

-- 5 --

```
select name, DOB from programmer where month(DOB)=4;
```

-- 6 --

```
select min(ccost) as MinCost from studies;
```

-- 7 --

```
select count(name) as Count from studies where course = 'PGDCA';
```

-- 8 --

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

-- Q9 --

SELECT \* FROM SOFTWARE WHERE NAME = "RAMESH";

-- Q10 --

SELECT COUNT(NAME) FROM STUDIES WHERE splace = "SABHARI";

-- Q11 --

SELECT \* FROM SOFTWARE WHERE (DCOST\*SOLD) > 20000;

-- Q12 --

SELECT CEIL(DCost / SCost) AS copires FROM Software;

-- Q13 --

SELECT MAX(DCOST) FROM SOFTWARE WHERE dev\_in = "BASIC";

-- Q14 --

SELECT \* FROM SOFTWARE where SCOST\*SOLD >= DCOST;

-- Q15 --

SELECT COUNT(\*) FROM SOFTWARE WHERE DEV\_IN = "DBASE";

-- Q16 --

SELECT COUNT(\*) FROM STUDIES WHERE SPLACE = "PRAGATHI";

-- Q17 --

```
SELECT COUNT(*) FROM STUDIES WHERE CCOST >= 5000 AND CCOST <= 10000;
```

-- Q18 --

```
SELECT AVG(CCOST) FROM STUDIES;
```

-- Q19 --

```
SELECT * FROM PROGRAMMER WHERE PROF1 = 'C' OR PROF2 = 'C';
```

-- Q20 --

```
SELECT COUNT(NAME) FROM PROGRAMMER WHERE PROF1 = 'COBOL' OR PROF1 = 'PASCAL' OR PROF2  
= 'PASCAL' OR PROF2 = 'COBOL';
```

-- Q21 --

```
SELECT COUNT(NAME) FROM PROGRAMMER WHERE PROF1 <> 'PASCAL' AND PROF1 <> 'C' AND PROF2  
<> 'C' AND PROF2 <> 'PASCAL';
```

-- Q22 --

```
select max(floor(datediff(curdate(),DOB)/365)) as Age from Programmer where SEX='M' ;
```

-- Q23 --

```
select AVG(floor(datediff(curdate(),DOB)/365)) as Age from Programmer where SEX='F' ;
```

-- Q24 --

```
SELECT NAME,ABS(floor(datediff(curdate(),DOJ)/365)) AS EXP FROM PROGRAMMER ORDER BY EXP  
DESC;
```

-- Q25 --

```
SELECT NAME FROM PROGRAMMER WHERE MONTH(DOB) = MONTH(curdate());
```

-- Q26 --

```
SELECT COUNT(*) AS FEMALE_PROGRAMMERS FROM PROGRAMMER WHERE SEX = 'F';
```

-- Q27 --

```
SELECT PROF1 AS LANG FROM PROGRAMMER WHERE SEX = 'M' UNION DISTINCT SELECT PROF2 FROM PROGRAMMER WHERE SEX = 'M';
```

-- Q28 --

```
SELECT AVG (SALARY) AS AVERAGE FROM PROGRAMMER;
```

-- Q29 --

```
SELECT COUNT(NAME) AS COUNT FROM PROGRAMMER WHERE SALARY >= 2000 AND SALARY <= 4000;
```

-- Q30 --

```
SELECT * FROM PROGRAMMER WHERE PROF1!= 'COBOL' AND PROF2!= 'COBOL' AND PROF1!= 'CLIPPER' AND PROF2!= 'CLIPPER' AND PROF1!= 'PASCAL' AND PROF2!= 'PASCAL';
```

-- Q31 --

```
SELECT COUNT(NAME) FROM PROGRAMMER WHERE (PROF1='C' OR PROF2='C') AND (floor(datediff(curdate(),DOB)/365))>24;
```

-- Q32 --

```
SELECT NAME, DOB FROM PROGRAMMER WHERE DATEDIFF((DATE_ADD(DOB, INTERVAL CEIL(datediff(curdate(),DOB)/365.25) YEAR)),CURDATE()) BETWEEN 0 AND 6;
```

```
SELECT DATEDIFF( (DATE_ADD(DOB, INTERVAL CEIL(datediff(curdate(),DOB )/365.25) YEAR)),CURDATE())  
FROM PROGRAMMER;
```

-- Q33 --

```
SELECT NAME FROM PROGRAMMER WHERE ABS(floor(datediff(curdate(),DOJ )/365)) < 1;
```

-- Q34--

```
SELECT NAME FROM PROGRAMMER WHERE ABS(floor(datediff(curdate(),DOJ )/365)) = 2;
```

-- Q35--

```
SELECT DCOST-(SCOST*SOLD) AS AMOUNT FROM SOFTWARE WHERE DCOST-(SCOST*SOLD) >0;
```

-- Q36 --

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

-- Q37 --

```
SELECT SUM(DCOST) AS TOTAL FROM SOFTWARE WHERE NAME = 'MARY';
```

-- Q38 --

```
SELECT DISTINCT SPLACE FROM STUDIES;
```

-- Q39 --

```
SELECT COUNT(DISTINCT COURSE) FROM STUDIES;
```

-- 40 --

```
SELECT NAME FROM PROGRAMMER WHERE NAME LIKE '%A%A%';
```



-- 41 --

```
SELECT NAME FROM PROGRAMMER WHERE CHAR_LENGTH(NAME) <=5;
```

-- 42--

```
SELECT NAME FROM PROGRAMMER WHERE SEX='F' AND (PROF1='COBOL' OR PROF2='COBOL') AND  
ABS(floor(datediff(curdate(),DOJ )/365)) > 2;
```

-- 43 --

```
SELECT MIN(CHAR_LENGTH(NAME)) FROM PROGRAMMER;
```

-- 44 --

```
SELECT AVG(DCOST) AS AVG FROM SOFTWARE WHERE DEV_IN='COBOL';
```

-- 45 --

-- 46 --

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

-- 47 --

```
SELECT NAME, SALARY , SEX FROM PROGRAMMER WHERE SEX='M' AND (PROF1 <> 'COBOL' AND PROF2  
<> 'COBOL');
```

-- 48 --

```
SELECT TITLE ,SCOST,DCOST,ABS(SCOST-DCOST) AS COST FROM SOFTWARE ORDER BY COST DESC;
```

-- 49 --

SELECT NAME,DOB,DOJ FROM PROGRAMMER WHERE MONTH(DOB)=MONTH(DOJ);

-- 50 --

SELECT TITLE FROM SOFTWARE WHERE LOCATE(' ', TITLE) >0;

## QUERIES II

-- 1 --

SELECT DEV\_IN , COUNT(\*) AS PACKAGES FROM SOFTWARE GROUP BY DEV\_IN;

-- 2 --

SELECT NAME,COUNT(\*) AS PACKAGES FROM SOFTWARE GROUP BY NAME;

-- 3 --

SELECT SEX, COUNT(\*) AS COUNT FROM PROGRAMMER GROUP BY SEX;

-- 4 --

SELECT DEV\_IN, MAX(SCOST\*SOLD) FROM SOFTWARE GROUP BY DEV\_IN;

-- 5 --

SELECT YEAR(DOB), COUNT(\*) AS TOTAL FROM PROGRAMMER GROUP BY YEAR(DOB) ORDER BY YEAR(DOB);

-- 6 --

```
SELECT YEAR(DOJ), COUNT(*) AS TOTAL FROM PROGRAMMER GROUP BY YEAR(DOJ);
```

-- 7 --

```
SELECT MONTH(DOB), COUNT(*) AS TOTAL FROM PROGRAMMER GROUP BY MONTH(DOB) ORDER BY  
MONTH(DOB);
```

-- 8 --

```
SELECT MONTH(DOJ), COUNT(*) AS TOTAL FROM PROGRAMMER GROUP BY MONTH(DOJ) ORDER BY  
MONTH(DOJ);
```

-- 9 --

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

-- 10 --

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

-- 11 --

```
SELECT SALARY, COUNT(NAME) AS TOTAL FROM PROGRAMMER GROUP BY SALARY ORDER BY SALARY;
```

-- 12 --

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

-- 13 --

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

-- 14 --

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

-- 15 --

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

-- 16 --

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

-- 17 --

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

-- 18 --

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

-- 19 --

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

-- 20 --

```
SELECT Name AS programmer_name, MAX(SCost) AS costliest_package, MIN(SCost) AS  
cheapest_package FROM SOFTWARE s GROUP BY Name;
```

-- 21 --

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

-- 22 --

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

-- 23 --

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

-- 24 --

SELECT SEX,NAME FROM PROGRAMMER WHERE SEX='M' UNION SELECT SEX,NAME FROM PROGRAMMER WHERE SEX='F';

-- 25 --

SELECT NAME, GROUP\_CONCAT(TITLE) FROM SOFTWARE GROUP BY NAME;

-- 26 --

SELECT DEV\_IN, COUNT(TITLE) FROM SOFTWARE GROUP BY DEV\_IN;

-- 27 --

SELECT DEV\_IN, COUNT(DCOST) FROM SOFTWARE WHERE DCOST<1000 GROUP BY DEV\_IN ;

-- 28 --

SELECT DEV\_IN, AVG(ABS(SCOST-DCOST))AS AVG FROM SOFTWARE GROUP BY DEV\_IN;

-- 29 --

SELECT NAME, SUM(SCOST), SUM(DCOST), SUM(DCOST-(SCOST\*SOLD)) AS RECOVERY FROM SOFTWARE GROUP BY NAME HAVING SUM(DCOST)>SUM(SCOST\*SOLD);

-- 30 --

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

## QUERY III

-- 1 --

```
SELECT NAME,SALARY FROM PROGRAMMER WHERE PROF1='C' OR PROF2='C' ORDER BY SALARY DESC  
LIMIT 1;
```

-- 2 --

```
SELECT NAME,SALARY FROM PROGRAMMER WHERE (PROF1='COBOL' OR PROF2='COBOL') AND SEX='F'  
ORDER BY SALARY DESC LIMIT 1;
```

-- 3 --

```
SELECT PROF1, MAX(SALARY), GROUP_CONCAT(NAME) FROM PROGRAMMER GROUP BY PROF1 ORDER  
BY PROF1;
```

-- 4 --

```
SELECT ABS(floor(datediff(curdate(),DOJ)/365)) AS EXP,GROUP_CONCAT(NAME) AS NAME FROM  
PROGRAMMER GROUP BY EXP ORDER BY EXP LIMIT 1 ;
```

-- 5 --

```
SELECT ABS(floor(datediff(curdate(),DOJ)/365)) AS EXP,GROUP_CONCAT(NAME) AS NAME FROM  
PROGRAMMER GROUP BY EXP ORDER BY EXP DESC LIMIT 1 ;
```

-- 6 --

```
SELECT lang FROM (
```

```
    SELECT PROF1 AS lang
```

```
    FROM PROGRAMMER
```

```
    UNION ALL
```

```
    SELECT PROF2 AS lang
```

```
    FROM PROGRAMMER
```

```
) languages
```

```
GROUP BY lang
```

```
HAVING COUNT(*) = 1;
```

```
-- 7 --
```

```
SELECT NAME,floor(datediff(curdate(),DOB)/365) AS AGE FROM PROGRAMMER WHERE PROF1='DBASE'  
OR PROF2='DBASE' ORDER BY AGE LIMIT 1 ;
```

```
-- 8 --
```

```
SELECT SPLACE, COUNT(*) AS TOTAL FROM STUDIES GROUP BY SPLACE ORDER BY TOTAL DESC LIMIT 1;
```

```
-- Q9 --
```

```
WITH TOTAL AS (
```

```
    SELECT SPlace
```

```
    FROM STUDIES
```

```
    GROUP BY SPlace
```

```
    ORDER BY COUNT(*) DESC
```

```
    LIMIT 1
```

```
)
```

```
SELECT DISTINCT NAME
```

FROM STUDIES

WHERE SPlace = (SELECT SPlace FROM TOTAL);

-- Q10 --

SELECT NAME , SEX FROM PROGRAMMER WHERE SEX='F' AND SALARY>3000 AND ((PROF1 <> 'C' AND  
PROF1 <> 'C++' AND PROF1<>'ORACLE' AND PROF1<>'DBASE') AND (PROF2 <> 'C' AND PROF2 <> 'C++'  
AND PROF2<>'ORACLE' AND PROF2<>'DBASE'));

-- Q11 --

SELECT SPLACE,CCOST FROM STUDIES WHERE CCOST=(SELECT MAX(CAST(CCOST AS DECIMAL(10,2)))  
FROM STUDIES);

-- Q12 --

SELECT COURSE, COUNT(\*) AS TOTAL FROM STUDIES GROUP BY COURSE ORDER BY TOTAL DESC LIMIT 1;

-- Q13 --

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

-- Q14 --

SELECT SPLACE, MAX(CCOST) FROM STUDIES GROUP BY SPLACE ORDER BY MAX(CCOST) DESC LIMIT 1;

-- Q15 --

SELECT COURSE, COUNT(\*) AS Num\_Students

FROM STUDIES

GROUP BY COURSE

HAVING COUNT(\*) < (

SELECT AVG(Num\_Students)



```
FROM (  
    SELECT COUNT(*) AS Num_Students  
    FROM STUDIES  
    GROUP BY COURSE  
    ) AS AvgNumStudents  
);
```

-- Q16 --

```
SELECT SPlace AS Institute  
FROM STUDIES  
WHERE COURSE IN (  
    SELECT COURSE  
    FROM STUDIES  
    GROUP BY COURSE  
    HAVING COUNT(*) < (  
        SELECT AVG(Num_Students)  
        FROM (  
            SELECT COUNT(*) AS Num_Students  
            FROM STUDIES  
            GROUP BY COURSE  
        ) AS AvgNumStudents  
    )  
);
```

-- Q17 --

```
SELECT COURSE FROM STUDIES WHERE ABS(CCOST - (SELECT AVG(CCOST) FROM STUDIES)) <= 1000;
```

```
-- Q18 --
```

```
SELECT TITLE, DCOST FROM SOFTWARE ORDER BY DCOST DESC LIMIT 1;
```

```
-- Q19 --
```

```
SELECT TITLE, SCOST FROM SOFTWARE ORDER BY SCOST LIMIT 1;
```

```
-- Q20 --
```

```
SELECT NAME,SOLD FROM SOFTWARE WHERE SOLD = (SELECT MIN(SOLD) FROM SOFTWARE);
```

```
-- Q21 --
```

```
SELECT DEV_IN,SCOST FROM SOFTWARE WHERE SCOST = (SELECT MAX(SCOST) FROM SOFTWARE);
```

```
-- Q22 --
```

```
SELECT SOLD,TITLE FROM SOFTWARE WHERE TITLE = (SELECT TITLE FROM SOFTWARE WHERE (DCOST-SCOST)=(SELECT MIN(DCOST-SCOST) FROM SOFTWARE));
```

```
-- Q23 --
```

```
SELECT TITLE FROM SOFTWARE WHERE DCOST = (SELECT MAX(DCOST)FROM SOFTWARE WHERE DEV_IN LIKE 'PASCAL');
```

```
-- Q24 --
```

```
SELECT DEV_IN FROM SOFTWARE GROUP BY DEV_IN HAVING DEV_IN = (SELECT MAX(DEV_IN) FROM SOFTWARE);
```

```
-- Q25 --
```

```
SELECT NAME FROM SOFTWARE GROUP BY NAME HAVING NAME = (SELECT MAX(NAME) FROM SOFTWARE);
```

-- Q26 --

```
SELECT NAME FROM SOFTWARE WHERE SCOST = ( SELECT MAX(SCOST) FROM SOFTWARE );
```

-- Q27 --

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

-- Q28 --

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

-- Q29 --

```
SELECT PROF1, COUNT(*) FROM PROGRAMMER GROUP BY PROF1 ORDER BY COUNT(*) DESC LIMIT 1;
```

-- Q30 --

```
SELECT DISTINCT NAME FROM SOFTWARE WHERE SOLD*SCOST > 2*DCOST;
```

-- Q31 --

```
SELECT NAME,TITLE FROM SOFTWARE WHERE DCOST IN (SELECT MIN(DCOST) FROM SOFTWARE GROUP BY DEV_IN);
```

-- Q32 --

```
select name,floor(DATEDIFF(CURDATE(), DOB)/365) as age from programmer where year(dob)='1965' and sex='M' limit 1;
```

-- Q33 --

```
SELECT NAME, DEV_IN FROM SOFTWARE WHERE SOLD IN (SELECT MAX(SOLD) FROM SOFTWARE GROUP BY NAME) UNION SELECT NAME, DEV_IN FROM SOFTWARE WHERE SOLD IN (SELECT MIN(SOLD) FROM SOFTWARE GROUP BY NAME);
```

-- Q34--

```
select name, 2024-year(dob) as age from programmer where year(doj)='1992' and sex='F' order by age desc limit 1;
```

-- Q35--

```
select year(dob) ,count(year(dob)) as count from programmer group by year(dob) order by count(*) desc limit 1;
```

-- Q36 --

```
select month(doj) as month,count(month(doj)) as count from programmer group by month(doj) order by count(*) desc limit 1;
```

-- Q37 --

```
select prof1,prof2,count(name) as count from programmer group by prof1,prof2 order by count desc limit 2;
```

-- Q38 --

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

## QUERY IV

-- 1 --

```
SELECT SALARY, GROUP_CONCAT(NAME) FROM PROGRAMMER WHERE SALARY IN ( SELECT SALARY
FROM PROGRAMMER GROUP BY SALARY HAVING COUNT(*) > 1) GROUP BY SALARY ORDER BY SALARY;
```

-- 2 --

```
SELECT S.NAME, S.TITLE,P.SEX,P.SALARY FROM SOFTWARE S JOIN PROGRAMMER P ON P.NAME =
S.NAME WHERE P.SEX='M' AND P.SALARY >3000;
```

-- 3 --

```
SELECT S.* FROM SOFTWARE S JOIN PROGRAMMER P ON S.NAME=P.NAME WHERE P.SEX='F' AND
S.DEV_IN='PASCAL';
```

-- 4 --

```
SELECT * FROM PROGRAMMER WHERE YEAR(DOJ)<1990;
```

-- 5 --

```
SELECT S.* FROM SOFTWARE S
JOIN PROGRAMMER P ON S.NAME = P.NAME
JOIN STUDIES ST ON p.NAME = ST.NAME
WHERE P.SEX = 'F' AND S.DEV_IN = 'C' AND ST.SPlace = 'PRAGATHI';
```

-- 6 --

```
SELECT S.NAME, ST.SPLACE, COUNT(S.TITLE),SUM(S.SOLD), SUM(S.SOLD * S.SCOST) FROM SOFTWARE S
JOIN PROGRAMMER P ON S.NAME = P.NAME
JOIN STUDIES ST ON S.NAME = ST.NAME
```

```
GROUP BY S.NAME, ST.SPLACE;
```

```
-- 7 --
```

```
SELECT S.* FROM SOFTWARE S
```

```
JOIN PROGRAMMER P ON S.NAME = P.NAME
```

```
JOIN STUDIES ST ON ST.NAME = S.NAME
```

```
WHERE S.DEV_IN='DBASE' AND P.SEX='M' AND ST.SPLACE=(SELECT SPLACE FROM STUDIES GROUP BY  
SPLACE ORDER BY COUNT(*) DESC LIMIT 1);
```

```
-- 8 --
```

```
SELECT S.* FROM SOFTWARE S
```

```
JOIN PROGRAMMER P ON S.NAME = P.NAME
```

```
WHERE (P.SEX='M' AND YEAR(P.DOB)<=1965) OR (P.SEX='F' AND YEAR(P.DOB)>1975);
```

```
-- 9 --
```

```
SELECT S.* FROM SOFTWARE S
```

```
JOIN PROGRAMMER P ON S.NAME = P.NAME
```

```
WHERE S.DEV_IN <> P.PROF1;
```

```
-- 10 --
```

```
SELECT S.* FROM SOFTWARE S
```

```
JOIN PROGRAMMER P ON S.NAME = P.NAME
```

```
WHERE S.DEV_IN <> P.PROF1 AND S.DEV_IN <> P.PROF2;
```

```
-- 11 --
```

```
SELECT S.* FROM SOFTWARE S
```

```
JOIN PROGRAMMER P ON S.NAME = P.NAME
```

```
JOIN STUDIES ST ON S.NAME = ST.NAME
```

```
WHERE P.SEX='M' AND ST.SPLACE='SABHARI';
```

```
-- 12 --
```

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

```
-- 13 --
```

```
SELECT SUM(S.SCOST) FROM SOFTWARE S
```

```
JOIN STUDIES ST ON S.NAME = ST.NAME
```

```
WHERE ST.SPLACE = 'APPLE';
```

```
-- 14 --
```

```
SELECT DOJ, GROUP_CONCAT(NAME) FROM PROGRAMMER GROUP BY DOJ HAVING COUNT(*) > 1;
```

```
-- 15 --
```

```
SELECT PROF2, GROUP_CONCAT(NAME) FROM PROGRAMMER GROUP BY PROF2 HAVING COUNT(*) > 1;
```

```
-- 16 --
```

```
SELECT ST.SPLACE, SUM(S.SCOST*S.SOLD) AS TOTAL FROM SOFTWARE S
```

```
JOIN STUDIES ST ON S.NAME = ST.NAME
```

```
GROUP BY ST.SPLACE;
```

```
-- 17 --
```

```
SELECT ST.SPLACE, P.NAME FROM STUDIES ST
```

```
JOIN PROGRAMMER P ON ST.NAME = P.NAME
```

```
JOIN SOFTWARE S ON ST.NAME = S.NAME
```

```
WHERE S.SCOST = (SELECT MAX(SCOST) FROM SOFTWARE);
```

```
-- 18 --
```

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

```
UNION
```

```
SELECT PROF2 FROM PROGRAMMER WHERE PROF2 NOT IN(SELECT DEV_IN FROM SOFTWARE);
```

```
-- 19 --
```

```
SELECT S.NAME,P.SALARY, ST.COURSE FROM SOFTWARE S
```

```
JOIN PROGRAMMER P ON S.NAME = P.NAME
```

```
JOIN STUDIES ST ON S.NAME = ST.NAME
```

```
WHERE S.SOLD=(SELECT MAX(SOLD) FROM SOFTWARE);
```

```
-- select p1.salary,s2.course from programmer p1,software s1,studies s2
```

```
-- where p1.pname=s1.pname and s1.pname=s2.pname and scost=(select max(scost) from software);
```

```
-- 20 --
```

```
SELECT P.NAME, CEIL(ST.CCOST/P.SALARY) AS TIMENEDED FROM PROGRAMMER P
```

```
JOIN STUDIES ST ON P.NAME=ST.NAME;
```

```
-- 21 --
```



```
SELECT S.* FROM SOFTWARE S  
  
JOIN PROGRAMMER P ON S.NAME = P.NAME  
  
WHERE FLOOR(DATEDIFF(CURDATE(), P.DOJ) / 365) < 3  
  
ORDER BY S.DCOST DESC LIMIT 1;
```

-- 22 --

```
SELECT AVG(P.SALARY)  
  
FROM SOFTWARE S  
  
JOIN PROGRAMMER P ON S.NAME = P.NAME  
  
WHERE S.SOLD * S.SCOST > 50000;
```

-- 23 --

```
SELECT COUNT(*)  
  
FROM SOFTWARE S  
  
JOIN PROGRAMMER P ON S.NAME = P.NAME  
  
JOIN STUDIES ST ON P.NAME = ST.NAME  
  
WHERE ST.CCOST = (  
  
    SELECT MIN(CCOST)  
  
    FROM STUDIES  
  
);
```

-- 24 --

```
SELECT COUNT(S.TITLE) AS TOTAL, ST.SPLACE, P.NAME  
  
FROM SOFTWARE S
```

```
JOIN PROGRAMMER P ON S.NAME = P.NAME
```

```
JOIN STUDIES ST ON P.NAME = ST.NAME
```

```
WHERE S.SCOST = (
```

```
    SELECT MIN(SCOST)
```

```
    FROM SOFTWARE
```

```
)
```

```
GROUP BY ST.SPLACE,P.NAME;
```

```
-- 25 --
```

```
SELECT COUNT(*)
```

```
FROM SOFTWARE S
```

```
JOIN PROGRAMMER P ON S.NAME = P.NAME
```

```
WHERE P.SEX = 'F' AND P.SALARY > (
```

```
    SELECT MAX(SALARY)
```

```
    FROM PROGRAMMER
```

```
    WHERE SEX = 'M'
```

```
);
```

```
-- 26 --
```

```
SELECT COUNT(*)
```

```
FROM SOFTWARE S
```

```
JOIN PROGRAMMER P ON S.NAME = P.NAME
```

```
JOIN STUDIES ST ON P.NAME = ST.NAME
```

```
WHERE FLOOR(DATEDIFF(CURDATE(), P.DOJ)) = (
```

```
SELECT MAX(FLOOR(DATEDIFF(CURDATE(), DOJ)))  
  
FROM PROGRAMMER  
  
) AND ST.SPlace = 'BDPS';
```

-- 27 --

```
SELECT NAME, SPLACE FROM STUDIES WHERE NAME NOT IN(SELECT NAME FROM SOFTWARE);
```

-- 28 --

```
SELECT P.PROF1, COUNT(DISTINCT P.NAME) AS NumberOfProgrammers, COUNT(S.TITLE) AS  
NumberOfPackages  
  
FROM PROGRAMMER P  
  
LEFT JOIN SOFTWARE S ON P.PROF1 = S.DEV_IN  
  
GROUP BY P.PROF1;
```

-- 29 --

```
SELECT S.NAME, COUNT(DEV_IN) FROM PROGRAMMER P, SOFTWARE S WHERE P.NAME = S.NAME  
GROUP BY S.NAME;
```

-- 30 --

```
SELECT P.* FROM PROGRAMMER P  
  
JOIN STUDIES ST ON P.NAME=ST.NAME  
  
WHERE ST.SPLACE = 'COIT'; -- ST.PLACE='S.S.I.L';
```