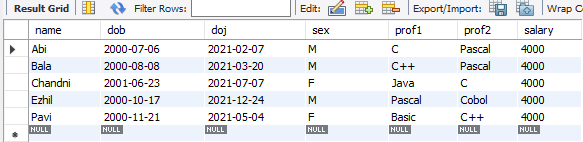
**SQL ASSIGNMENT**

**Query 3**

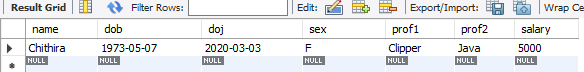
1. **Who is the highest paid C programmer?**

**Query:** SELECT \* FROM sql\_query.programmer WHERE salary=(SELECT MAX(salary) FROM sql\_query.programmer WHERE prof1 LIKE 'C' OR prof2 LIKE 'C');



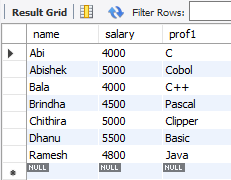
1. **Who is the highest paid female cobol programmer?**

**Query:** SELECT \* FROM sql\_query.programmer WHERE salary=(SELECT MAX(salary) FROM sql\_query.programmer WHERE (prof1 LIKE 'Cobol' OR prof2 LIKE 'Cobol')) AND sex LIKE 'F';



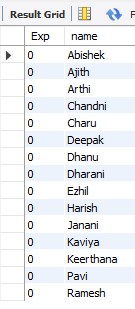
1. **Display the name of the HIGEST paid programmer for EACH language (prof1)**

**Query:** SELECT DISTINCT name, salary, prof1 FROM sql\_query.programmer WHERE (salary,prof1) IN (SELECT MAX(salary),prof1 FROM sql\_query.programmer GROUP BY prof1);



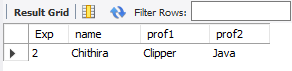
1. **Who is the LEAST experienced programmer?**

**Query:** SELECT FLOOR(datediff(current\_date,doj)/365)as Exp, name FROM sql\_query.programmer where FLOOR(datediff(current\_date,doj)/365)<1;



1. **Who is the MOST experienced programmer?**

**Query:** SELECT FLOOR(datediff(current\_date,doj)/365)as Exp,name,prof1,prof2 FROM sql\_query.programmer where FLOOR(datediff(current\_date,doj)/365)>1;



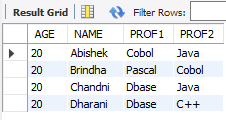
1. **Which language is known by ONLY ONE programmer?**

**Query:** SELECT PROF1 FROM PROGRAMMER GROUP BY PROF1 HAVING PROF1 NOT IN (SELECT PROF2 FROM PROGRAMMER) AND COUNT(PROF1)=1 UNION SELECT PROF2 FROM PROGRAMMER GROUP BY PROF2 HAVING PROF2 NOT IN (SELECT PROF1 FROM PROGRAMMER) AND COUNT(PROF2)=1;



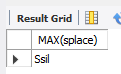
1. **Who is the YONGEST programmer knowing DBASE?**

**Query:** SELECT FLOOR(datediff(current\_date,DOB)/365) AGE, NAME, PROF1, PROF2 FROM PROGRAMMER WHERE FLOOR(datediff(current\_date,DOB)/365) = (SELECT MIN(FLOOR(datediff(current\_date,DOB)/365)) FROM PROGRAMMER WHERE PROF1 LIKE 'DBASE' OR PROF2 LIKE 'DBASE');



1. **Which institute has MOST NUMBER of students?**

**Query:** SELECT MAX(splace) FROM sql\_query.studies;

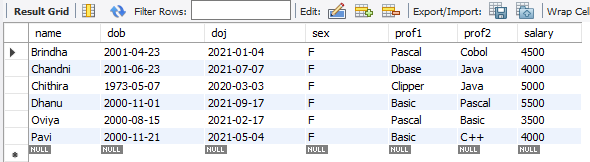


1. **Who is the above programmer?**

**Query:** SELECT NAME FROM STUDIES WHERE SPLACE=MAX(SPLACE);

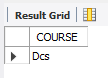
1. **Which female programmer earns MORE than 3000/- but DOES NOT know C, C++, Oracle or Dbase?**

**Query:** SELECT \* FROM PROGRAMMER WHERE SEX = 'F' AND SALARY >3000 AND (PROF1 NOT IN('C','C++','ORACLE','DBASE') OR PROF2 NOT IN('C','C++','ORACLE','DBASE'));



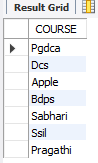
1. **Which is the COSTLIEST course?**

**Query:** SELECT COURSE FROM STUDIES WHERE CCOST = (SELECT MAX(CCOST) FROM STUDIES);



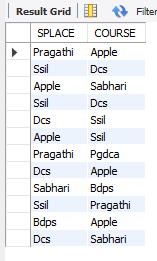
1. **Which course has been done by MOST of the students?**

**Query:** SELECT COURSE FROM studies group by COURSE HAVING COUNT(COURSE);



1. **Display name of the institute and course Which has below AVERAGE course fee?**

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



1. **Which institute conducts COSTLIEST course?**

**Query:** SELECT SPLACE FROM STUDIES WHERE CCOST = (SELECT MAX(CCOST) FROM STUDIES);

****

1. **Which course has below AVERAGE number of students?**

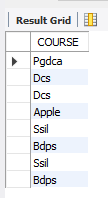
**Query:** SELECT COURSE FROM STUDIES GROUP BY COURSE HAVING COUNT(NAME)<(SELECT AVG(COUNT(NAME)) FROM STUDIES GROUP BY COURSE) ;

1. **Which institute conducts the above course?**

**Query:** SELECT SPLACE FROM STUDIES WHERE COURSE IN (SELECT COURSE FROM STUDIES GROUP BY COURSE HAVING COUNT(NAME) < (SELECT AVG(COUNT(NAME)) FROM STUDIES GROUP BY COURSE));

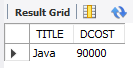
1. **Display names of the course WHOSE fees are within 1000(+ or -) of the AVERAGE fee.**

**Query:** SELECT COURSE FROM STUDIES WHERE CCOST < (SELECT AVG(CCOST)+1000 FROM STUDIES) AND CCOST > (SELECT AVG(CCOST)-1000 FROM STUDIES);



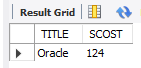
1. **Which package has the HIGEST development cost?**

**Query:** SELECT TITLE,DCOST FROM SOFTWARE WHERE DCOST = (SELECT MAX(DCOST) FROM SOFTWARE);

****

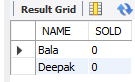
1. **Which package has the LOWEST selling cost?**

**Query:** SELECT TITLE,SCOST FROM SOFTWARE WHERE SCOST = (SELECT MIN(SCOST) FROM SOFTWARE);



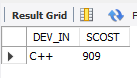
1. **Who developed the package, which has sold the LEAST number of copies?**

**Query:** SELECT NAME,SOLD FROM SOFTWARE WHERE SOLD = (SELECT MIN(SOLD) FROM SOFTWARE);



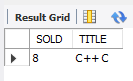
1. **Which language was used to develop the package WHICH has the HIGEST sales amount?**

**Query:** SELECT DEV\_IN,SCOST FROM SOFTWARE WHERE SCOST = (SELECT MAX(SCOST) FROM SOFTWARE);

****

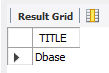
1. **How many copies of the package that has the LEAST DIFFRENCE between development and selling cost were sold?**

**Query:** SELECT SOLD,TITLE FROM SOFTWARE WHERE TITLE = (SELECT TITLE FROM SOFTWARE WHERE (DCOST-SCOST)=(SELECT MIN(DCOST-SCOST) FROM SOFTWARE));



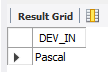
1. **Which is the COSTLIEAST package developed in PASCAL?**

**Query:** SELECT TITLE FROM SOFTWARE WHERE DCOST = (SELECT MAX(DCOST) FROM SOFTWARE WHERE DEV\_IN LIKE 'PASCAL');



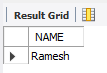
1. **Which language was used to develop the MOST NUMBER of package?**

**Query:** SELECT DEV\_IN FROM SOFTWARE GROUP BY DEV\_IN HAVING MAX(DEV\_IN) = (SELECT MAX(DEV\_IN) FROM SOFTWARE);



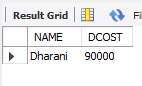
1. **Which programmer has developed the HIGEST NUMBER of package?**

**Query:** SELECT NAME FROM SOFTWARE GROUP BY NAME HAVING MAX(NAME) = (SELECT MAX(NAME) FROM SOFTWARE);



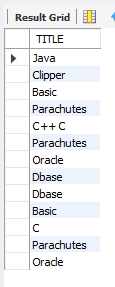
1. **Who is the author of the COSTLIEST package?**

**Query:** SELECT NAME,DCOST FROM SOFTWARE WHERE DCOST = (SELECT MAX(DCOST) FROM SOFTWARE);



1. **Display names of packages WHICH have been sold LESS THAN the AVERAGE number of copies?**

**Query:** SELECT TITLE FROM SOFTWARE WHERE SOLD < (SELECT AVG(SOLD) FROM SOFTWARE);



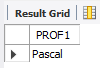
1. **Who are the female programmers earning MORE than the HIGEST paid male programmers?**

**Query:** SELECT NAME FROM PROGRAMMER WHERE SEX = 'F' AND SALARY > (SELECT(MAX(SALARY)) FROM PROGRAMMER WHERE SEX = 'M');



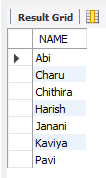
1. **Which language has been stated as prof1 by MOST of the programmers?**

**Query:** SELECT PROF1 FROM PROGRAMMER GROUP BY PROF1 HAVING PROF1 = (SELECT MAX(PROF1) FROM PROGRAMMER);



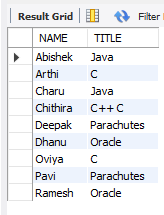
1. **Who are the authors of packages, WHICH have recovered MORE THAN double the development cost?**

**Query:** SELECT NAME FROM SOFTWARE WHERE SOLD\*SCOST > 2\*DCOST;

****

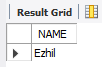
1. **Display programmer names and CHEAPEST package developed by them in EACH language?**

**Query:** SELECT NAME,TITLE FROM SOFTWARE WHERE DCOST IN (SELECT MIN(DCOST) FROM SOFTWARE GROUP BY DEV\_IN);



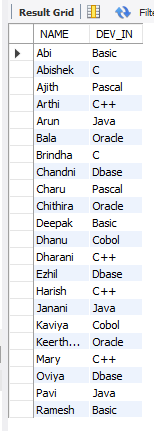
1. **Who is the YOUNGEST male programmer born in 1965?**

**Query:** SELECT NAME FROM PROGRAMMER WHERE DOB=(SELECT (MAX(DOB)) FROM PROGRAMMER WHERE date\_format(DOB,'%Y') LIKE '1965');



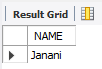
1. **Display language used by EACH programmer to develop the HIGEST selling and LOWEST selling package.**

**Query:** 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);



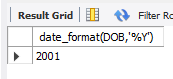
1. **Who is the OLDEST female programmer WHO joined in 1992?**

**Query:** SELECT NAME FROM PROGRAMMER WHERE DOJ=(SELECT (MIN(DOJ)) FROM PROGRAMMER WHERE date\_format(DOJ,'%Y') LIKE '1992');



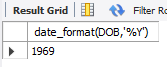
1. **In WHICH year where the MOST NUMBER of programmer born?**

**Query:** SELECT DISTINCT date\_format(DOB,'%Y') FROM PROGRAMMER WHERE date\_format(DOB,'%Y') = (SELECT MAX(date\_format(DOB,'%Y')) FROM PROGRAMMER);



1. **In WHICH month did MOST NUMBRER of programmer join?**

**Query:** SELECT DISTINCT date\_format(DOJ,'%M') FROM PROGRAMMER WHERE date\_format(DOJ,'%M') = (SELECT MAX(date\_format(DOJ,'%M')) FROM PROGRAMMER);



1. **In WHICH language are MOST of the programmer's proficient?**

**Query:** SELECT MAX(prof1) as Most\_proficient FROM PROGRAMMER;



1. **Who are the male programmers earning BELOW the AVERAGE salary of female programmers?**

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

