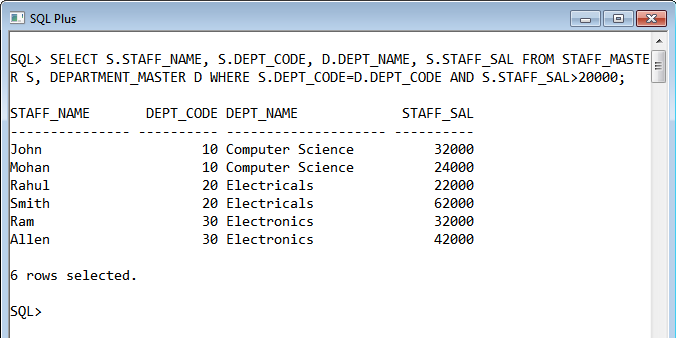
**Lab 3. Joins and Subqueries**

**1. Write a query which displays Staff Name, Department Code, Department Name,**

**and Salary for all staff who earns more than 20000.**

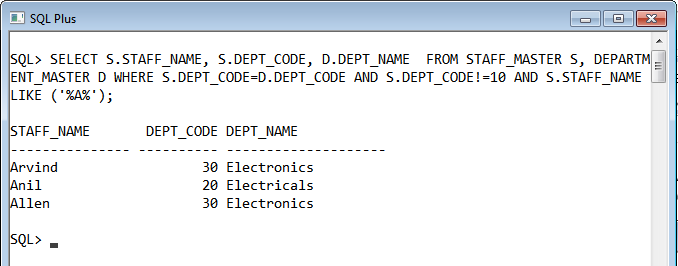
SELECT S.STAFF\_NAME, S.DEPT\_CODE, D.DEPT\_NAME, S.STAFF\_SAL FROM STAFF\_MASTERS, DEPARTMENT\_MASTER D WHERE S.DEPT\_CODE=D.DEPT\_CODE AND S.STAFF\_SAL>20000;



**2. Write a query to display Staff Name, Department Code, and Department Name for**

**all staff who do not work in Department code 10 and have ‘A’ in their name.**

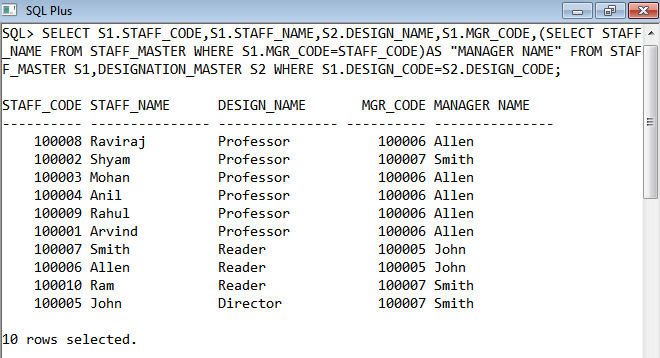
SELECT S.STAFF\_NAME, S.DEPT\_CODE, D.DEPT\_NAME FROM STAFF\_MASTER S, DEPARTMENT\_MASTER D WHERE S.DEPT\_CODE=D.DEPT\_CODE AND S.DEPT\_CODE!=10 AND S.STAFF\_NAME LIKE ('%A%');



**3. Display Staff Code, Staff Name, Department Name, and his manager’s number and**

**name. Label the columns Staff#, Staff, Mgr#, Manager.**

SELECT S1.STAFF\_CODE,S1.STAFF\_NAME,S2.DESIGN\_NAME,S1.MGR\_CODE,(SELECT STAFF\_NAME FROM STAFF\_MASTER WHERE S1.MGR\_CODE=STAFF\_CODE)AS "MANAGER NAME" FROM STAFF\_MASTER S1,DESIGNATION\_MASTER S2 WHERE S1.DESIGN\_CODE=S2.DESIGN\_CODE;

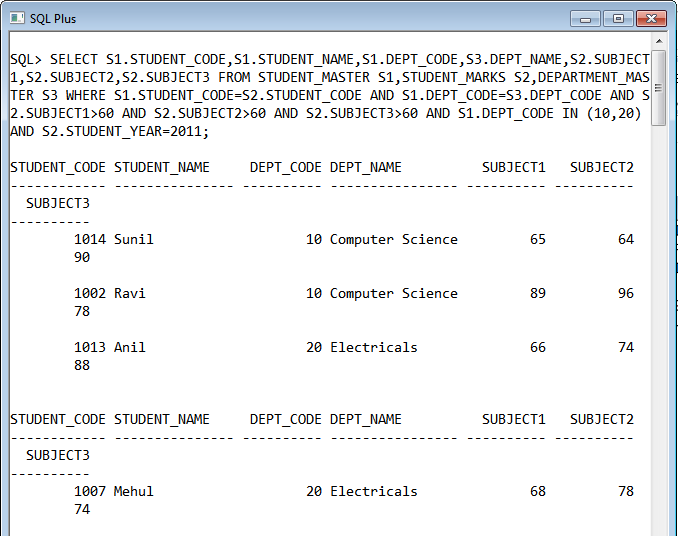


**4. Create a query that will display Student Code, Student Name, Department Name,**

**Subject1, Subject2, and Subject3 for all students who are getting 60 and above in each**

**subject from department 10 and 20.**

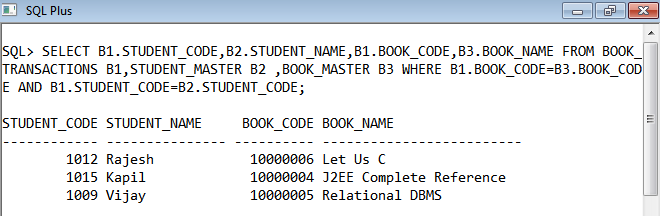
SELECT S1.STUDENT\_CODE,S1.STUDENT\_NAME,S1.DEPT\_CODE,S3.DEPT\_NAME,S2.SUBJECT1,S2.SUBJECT2,S2.SUBJECT3 FROM STUDENT\_MASTER S1,STUDENT\_MARKS S2,DEPARTMENT\_MASTER S3 WHERE S1.STUDENT\_CODE=S2.STUDENT\_CODE AND S1.DEPT\_CODE=S3.DEPT\_CODE AND S2.SUBJECT1>60 AND S2.SUBJECT2>60 AND S2.SUBJECT3>60 AND S1.DEPT\_CODE IN (10,20) AND S2.STUDENT\_YEAR=2011;



**5. Create a query that will display Student Code, Student Name, Book Code, and**

**Book Name for all students whose expected book return date is today.**

SELECT B1.STUDENT\_CODE,B2.STUDENT\_NAME,B1.BOOK\_CODE,B3.BOOK\_NAME FROM BOOK\_TRANSACTIONS B1,STUDENT\_MASTER B2 ,BOOK\_MASTER B3 WHERE B1.BOOK\_CODE=B3.BOOK\_CODE AND B1.STUDENT\_CODE=B2.STUDENT\_CODE;

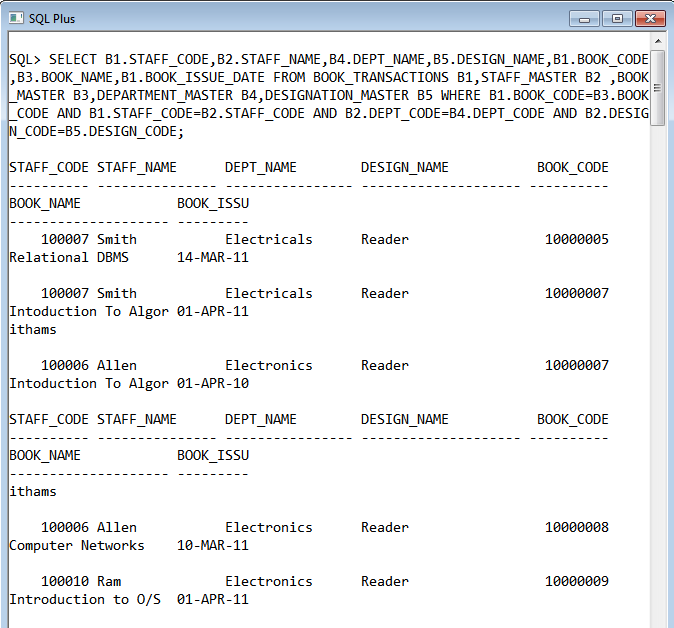


**6. Create a query that will display Staff Code, Staff Name, Department Name,**

**Designation, Book Code, Book Name, and Issue Date. For only those staff who have**

**taken any book in last 30 days.**

SQL> SELECT B1.STAFF\_CODE,B2.STAFF\_NAME,B4.DEPT\_NAME,B5.DESIGN\_NAME,B1.BOOK\_CODE,B3.BOOK\_NAME,B1.BOOK\_ISSUE\_DATE FROM BOOK\_TRANSACTIONS B1,STAFF\_MASTER B2 ,BOOK\_MASTER B3,DEPARTMENT\_MASTER B4,DESIGNATION\_MASTER B5 WHERE B1.BOOK\_CODE=B3.BOOK\_CODE AND B1.STAFF\_CODE=B2.STAFF\_CODE AND B2.DEPT\_CODE=B4.DEPT\_CODE AND B2.DESIGN\_CODE=B5.DESIGN\_CODE;

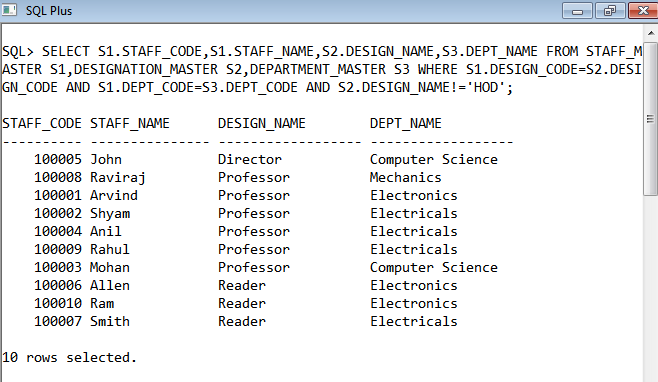


**7. Generate a report which contains the following information.**

**Staff Code Staff Name Designation Department Name Department Head**

**For all staff excluding HOD (List should not contain the details of Department head).**

SQL> SELECT S1.STAFF\_CODE,S1.STAFF\_NAME,S2.DESIGN\_NAME,S3.DEPT\_NAME FROM STAFF\_MASTER S1,DESIGNATION\_MASTER S2,DEPARTMENT\_MASTER S3 WHERE S1.DESIGN\_CODE=S2.DESIGN\_CODE AND S1.DEPT\_CODE=S3.DEPT\_CODE AND S2.DESIGN\_NAME!='HOD';



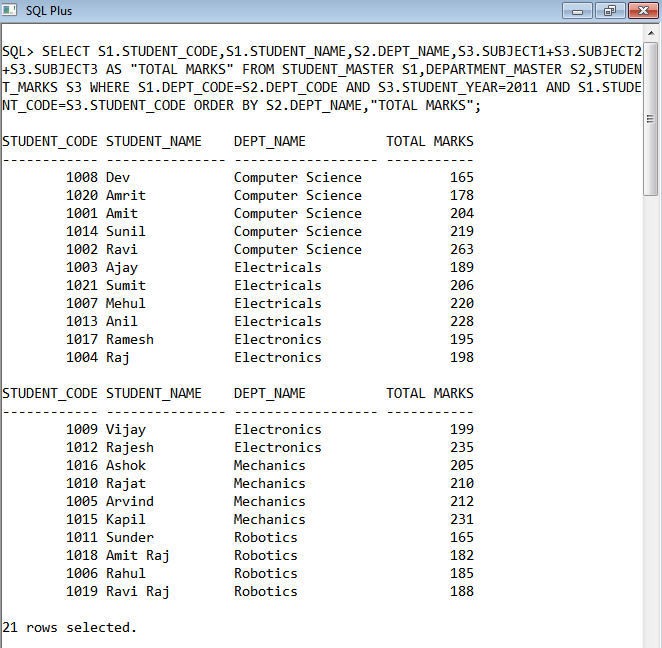
**8. Generate a report which contains the following information**

**Student Code Student Name Department Name Total Marks**

**HOD Name**

**Sort the output on Department Name and Total Marks.**

SELECT S1.STUDENT\_CODE,S1.STUDENT\_NAME,S2.DEPT\_NAME,S3.SUBJECT1+S3.SUBJECT2+S3.SUBJECT3 AS "TOTAL MARKS" FROM STUDENT\_MASTER S1,DEPARTMENT\_MASTER S2,STUDENT\_MARKS S3 WHERE S1.DEPT\_CODE=S2.DEPT\_CODE AND S3.STUDENT\_YEAR=2011 AND S1.STUDENT\_CODE=S3.STUDENT\_CODE ORDER BY S2.DEPT\_NAME,"TOTAL MARKS";



**9. Generate a report which contains the following information.**

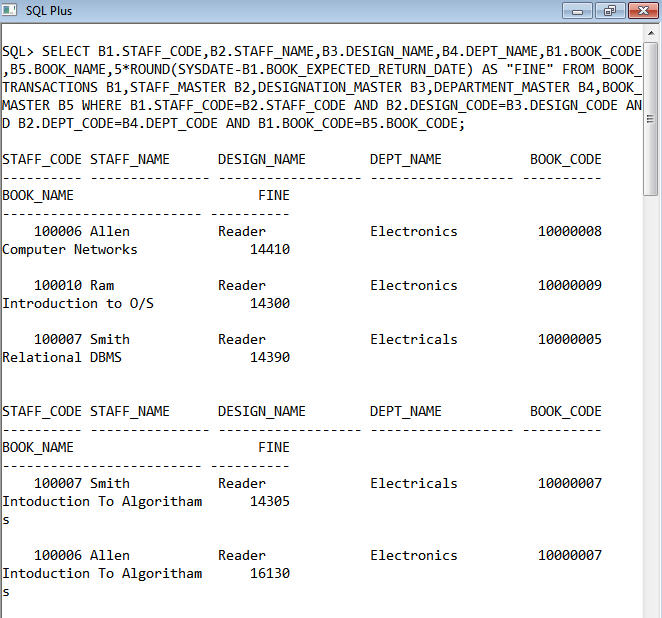
**Staff Code, Staff Name, Designation, Department, Book Code, Book Name,**

**Author, Fine**

**For the staff who have not return the book. Fine will be calculated as Rs. 5 per day.**

**Fine = 5 \* (No. of days = Current Date – Expected return date).**

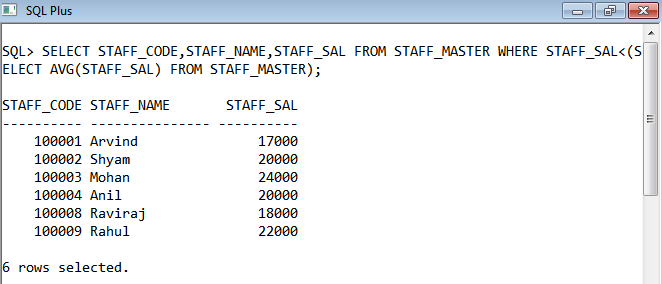
SELECT B1.STAFF\_CODE,B2.STAFF\_NAME,B3.DESIGN\_NAME,B4.DEPT\_NAME,B1.BOOK\_CODE,B5.BOOK\_NAME,5\*ROUND(SYSDATE-B1.BOOK\_EXPECTED\_RETURN\_DATE) AS "FINE" FROM BOOK\_TRANSACTIONS B1,STAFF\_MASTER B2,DESIGNATION\_MASTER B3,DEPARTMENT\_MASTER B4,BOOK\_MASTER B5 WHERE B1.STAFF\_CODE=B2.STAFF\_CODE AND B2.DESIGN\_CODE=B3.DESIGN\_CODE AN D B2.DEPT\_CODE=B4.DEPT\_CODE AND B1.BOOK\_CODE=B5.BOOK\_CODE;



**10. List Staff Code, Staff Name, and Salary for those who are getting less than the**

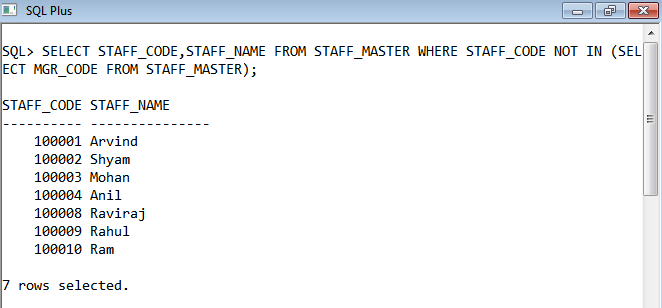
**average salary of organization.**

SELECT STAFF\_CODE,STAFF\_NAME,STAFF\_SAL FROM STAFF\_MASTER WHERE STAFF\_SAL<(SELECT AVG(STAFF\_SAL) FROM STAFF\_MASTER);



**11. List the Staff Code, Staff Name who are not Manager.**

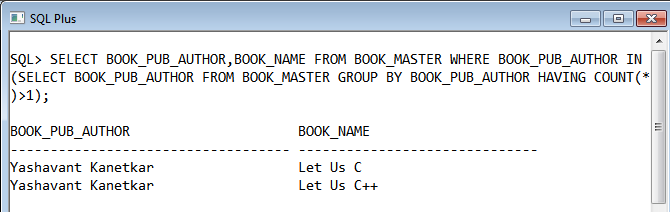
SELECT STAFF\_CODE,STAFF\_NAME FROM STAFF\_MASTER WHERE STAFF\_CODE NOT IN (SELECT MGR\_CODE FROM STAFF\_MASTER);



**12. Display Author Name, Book Name for those authors who wrote more than one**

**book.**

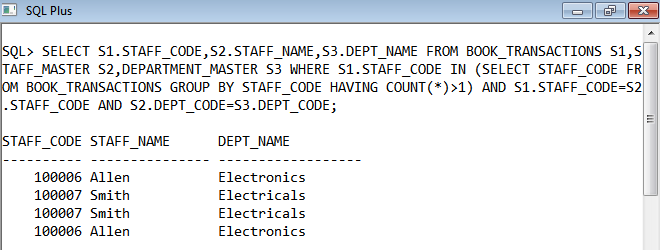
SELECT BOOK\_PUB\_AUTHOR,BOOK\_NAME FROM BOOK\_MASTER WHERE BOOK\_PUB\_AUTHOR IN(SELECT BOOK\_PUB\_AUTHOR FROM BOOK\_MASTER GROUP BY BOOK\_PUB\_AUTHOR HAVING COUNT(\*)>1);



**13. Display Staff Code, Staff Name, and Department Name for those who have taken**

**more than one book.**

SELECT S1.STAFF\_CODE,S2.STAFF\_NAME,S3.DEPT\_NAME FROM BOOK\_TRANSACTIONS S1,STAFF\_MASTER S2,DEPARTMENT\_MASTER S3 WHERE S1.STAFF\_CODE IN (SELECT STAFF\_CODE FROM BOOK\_TRANSACTIONS GROUP BY STAFF\_CODE HAVING COUNT(\*)>1) AND S1.STAFF\_CODE=S2.STAFF\_CODE AND S2.DEPT\_CODE=S3.DEPT\_CODE;



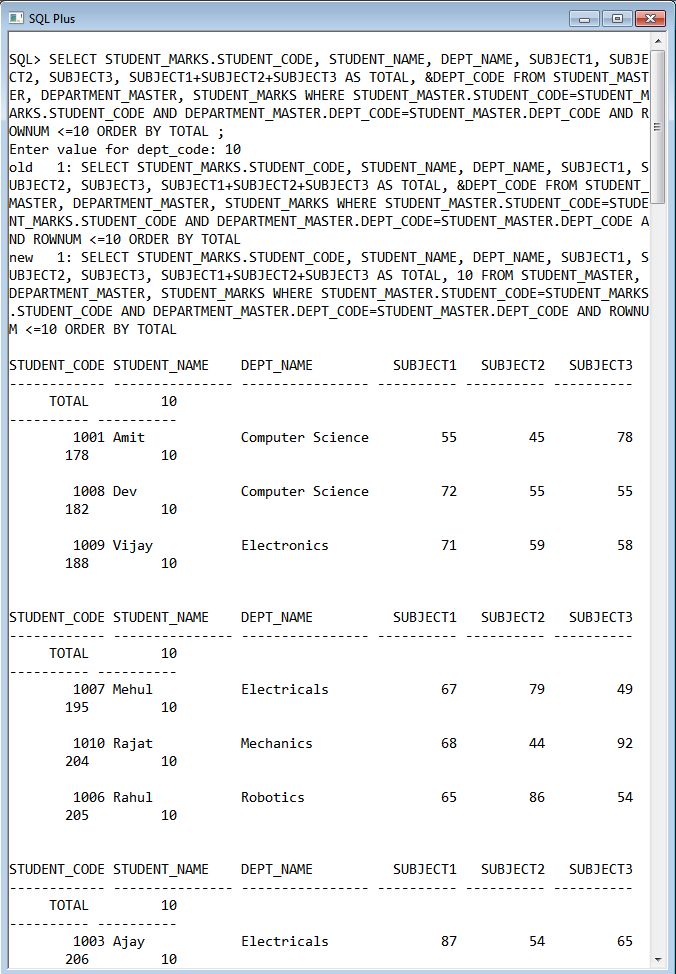
**14. Display top ten students for a specified department. Details are:**

**Student Code, Student Name, Department Name, Subject1, Subject2,**

**Subject3, Total.**

SQL> SELECT STUDENT\_MARKS.STUDENT\_CODE, STUDENT\_NAME, DEPT\_NAME, SUBJECT1, SUBJECT2, SUBJECT3, SUBJECT1+SUBJECT2+SUBJECT3 AS TOTAL, &DEPT\_CODE FROM STUDENT\_MASTER, DEPARTMENT\_MASTER, STUDENT\_MARKS WHERE STUDENT\_MASTER.STUDENT\_CODE=STUDENT\_MARKS.STUDENT\_CODE AND DEPARTMENT\_MASTER.DEPT\_CODE=STUDENT\_MASTER.DEPT\_CODE AND R

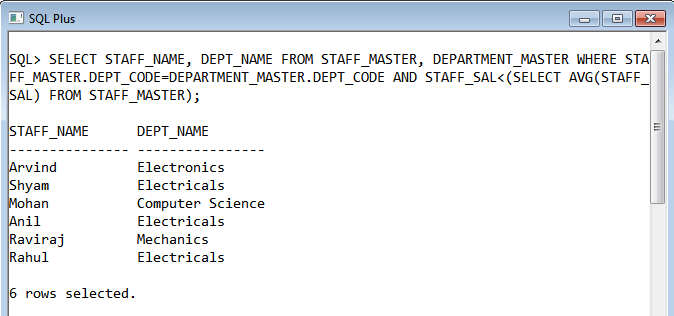
OWNUM <=10 ORDER BY TOTAL;



**15. Display the Staff Name, Department Name, and Salary for those staff who are**

**getting less than average salary in their own department**

SELECT STAFF\_NAME, DEPT\_NAME FROM STAFF\_MASTER, DEPARTMENT\_MASTER WHERE STAFF\_MASTER.DEPT\_CODE=DEPARTMENT\_MASTER.DEPT\_CODE AND STAFF\_SAL<(SELECT AVG(STAFF\_SAL) FROM STAFF\_MASTER);

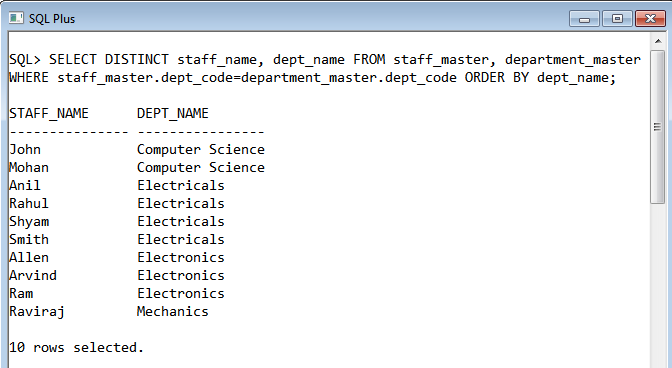


**16. Create a query that will display the Staff Name, Department Name, and all the**

**staff that work in the same department as a given staff. Give the column as**

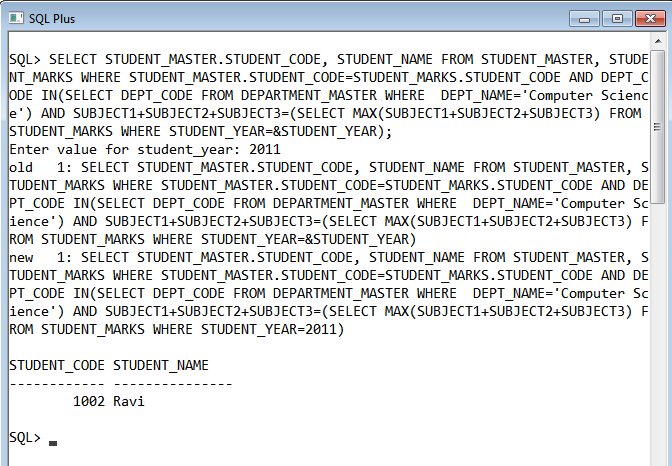
**appropriate label.**

SELECT DISTINCT STAFF\_NAME, DEPT\_NAME FROM STAFF\_MASTER, DEPARTMENT\_MASTER WHERE STAFF\_MASTER.DEPT\_CODE=DEPARTMENT\_MASTER.DEPT\_CODE ORDER BY DEPT\_NAME;



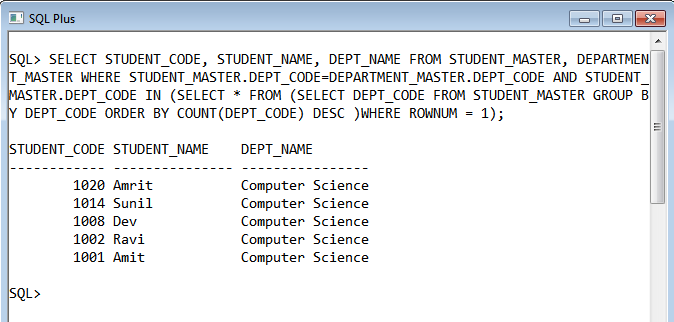
**17. List the Student Code, Student Name for that student who got highest marks in all three subjects in Computer Science department for current year.**

SELECT STUDENT\_MASTER.STUDENT\_CODE, STUDENT\_NAME FROM STUDENT\_MASTER, STUDENT\_MARKS WHERE STUDENT\_MASTER.STUDENT\_CODE=STUDENT\_MARKS.STUDENT\_CODE AND DEPT\_CODE IN(SELECT DEPT\_CODE FROM DEPARTMENT\_MASTER WHERE DEPT\_NAME='Computer Science') AND SUBJECT1+SUBJECT2+SUBJECT3=(SELECT MAX(SUBJECT1+SUBJECT2+SUBJECT3) FROM STUDENT\_MARKS WHERE STUDENT\_YEAR=&STUDENT\_YEAR);



**18. Display the Student Code, Student Name, and Department Name for that department in which there are maximum number of student are studying.**

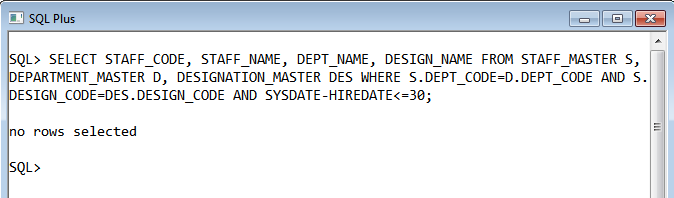
SELECT STUDENT\_CODE, STUDENT\_NAME, DEPT\_NAME FROM STUDENT\_MASTER, DEPARTMENT\_MASTER WHERE STUDENT\_MASTER.DEPT\_CODE=DEPARTMENT\_MASTER.DEPT\_CODE AND STUDENT\_MASTER.DEPT\_CODE IN (SELECT \* FROM (SELECT DEPT\_CODE FROM STUDENT\_MASTER GROUP BY DEPT\_CODE ORDER BY COUNT(DEPT\_CODE) DESC )WHERE ROWNUM = 1);



**19. Display Staff Code, Staff Name, Department Name, and Designation for those**

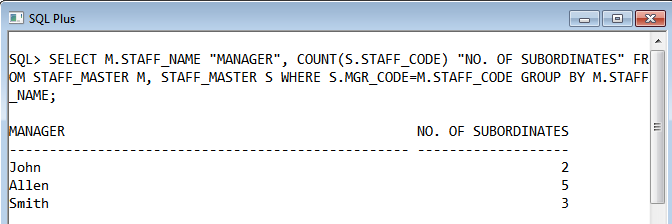
**who have joined most recently.**

SELECT STAFF\_CODE, STAFF\_NAME, DEPT\_NAME, DESIGN\_NAME FROM STAFF\_MASTER S, DEPARTMENT\_MASTER D, DESIGNATION\_MASTER DES WHERE S.DEPT\_CODE=D.DEPT\_CODE AND S.DESIGN\_CODE=DES.DESIGN\_CODE AND SYSDATE-HIREDATE<=30;



**20. Display the Manager Name and the total strength of his/her team.**

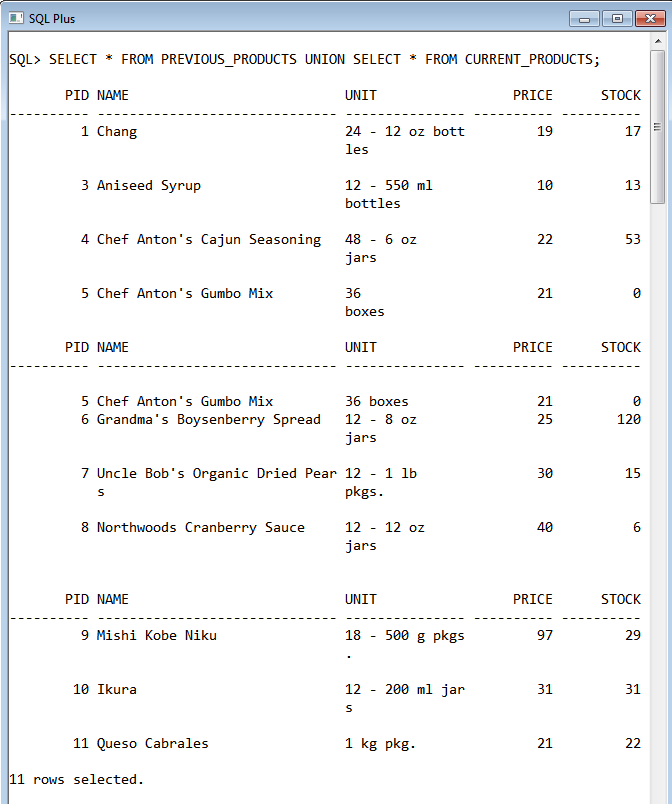
SELECT M.STAFF\_NAME "MANAGER", COUNT(S.STAFF\_CODE) "NO. OF SUBORDINATES" FROM STAFF\_MASTER M, STAFF\_MASTER S WHERE S.MGR\_CODE=M.STAFF\_CODE GROUP BY M.STAFF\_NAME;



**21. Get the details of all products irrespective of the fact whether they are in previous**

**set or current set.**

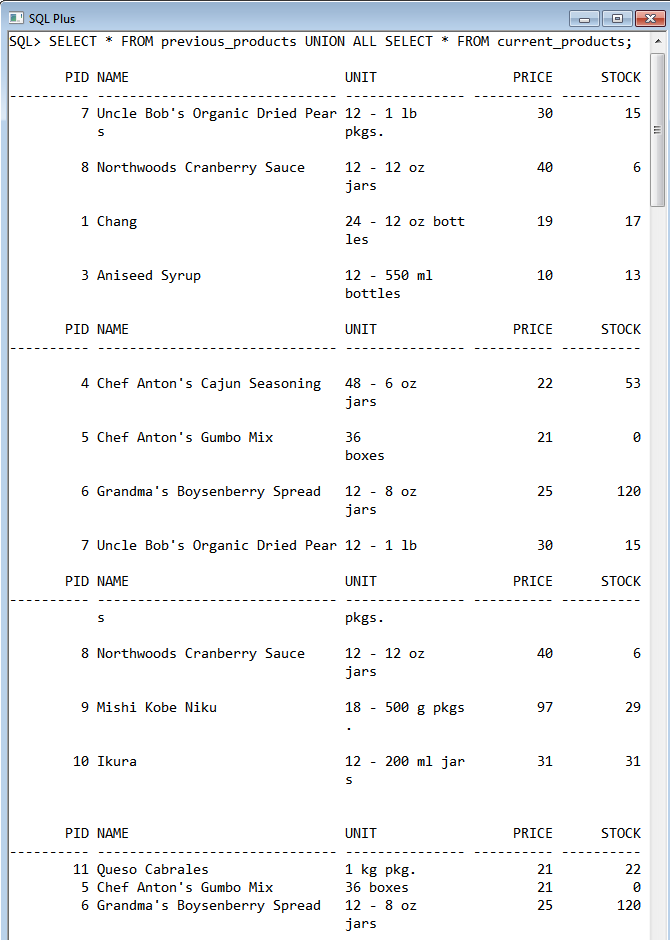
SELECT \* FROM PREVIOUS\_PRODUCTS UNION SELECT \* FROM CURRENT\_PRODUCTS;



**22. Get the details of all products along with the repetition of those that were present**

**both in the previous and current sets.**

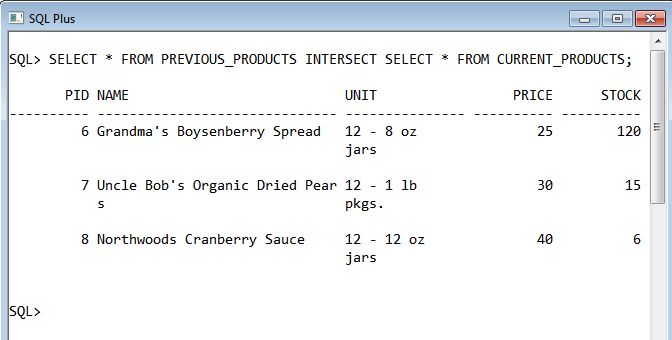
SELECT \* FROM PREVIOUS\_PRODUCTS UNION ALL SELECT \* FROM CURRENT\_PRODUCTS;



**23. Get the details of only those products which were present in the previous set and**

**are still continuing.**

SELECT \* FROM PREVIOUS\_PRODUCTS INTERSECT SELECT \* FROM CURRENT\_PRODUCTS;



**24. Get the details of all obsolete products (no longer continued).**

SELECT \* FROM PREVIOUS\_PRODUCTS MINUS SELECT \* FROM CURRENT\_PRODUCTS;

