Q1.WRITE QUERIES FOR THE FOLLOWING

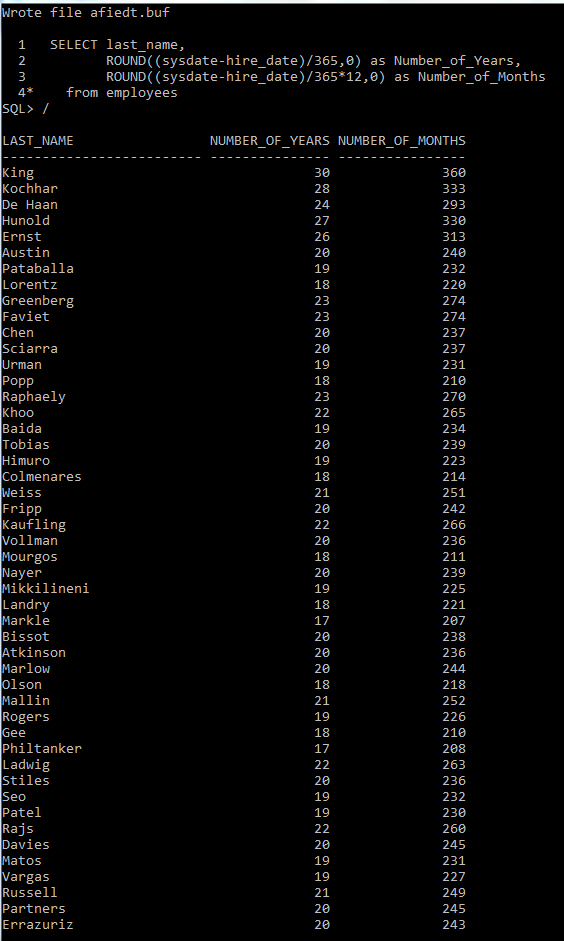
1.Display the last names of all employees together with the number of years and the number of months they have been employeed.

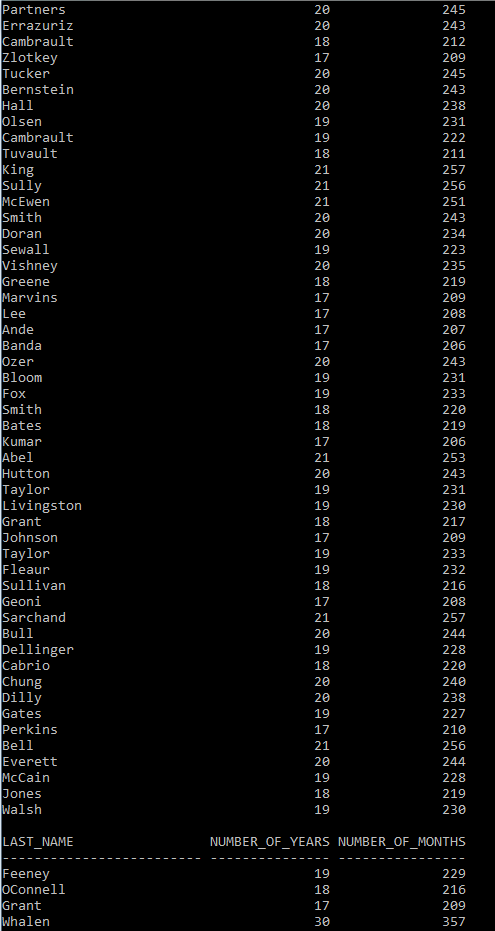
SOLUTION: SELECT last\_name,

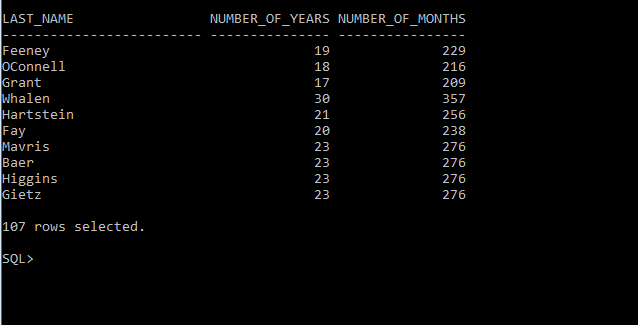
ROUND((sysdate-hire\_date)/365,0) as Year,

ROUND((sysdate-hire\_date)/365\*12,0) as Months

from employees;







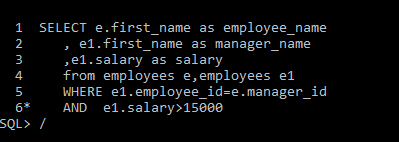
2. Display all employees whose managers earn salary higher than 15000. The query should display employee name, manager name and manager salary.

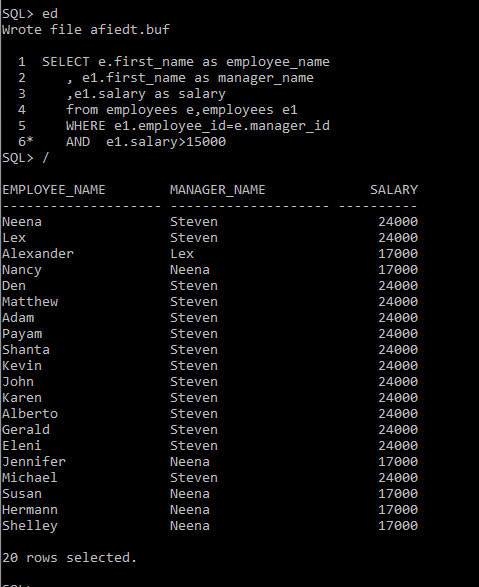
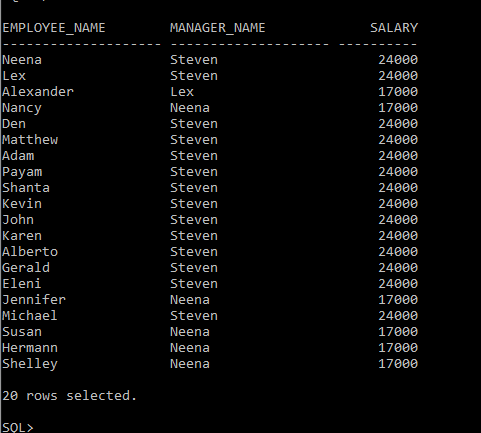
SOLUTION: SELECT e.first\_name as employee\_name,

e1.first\_name as manager\_name,

e1.salary as salary from employees e, employees e1

WHERE e1.employee\_id=e.manager\_id AND e1.salary>15000;



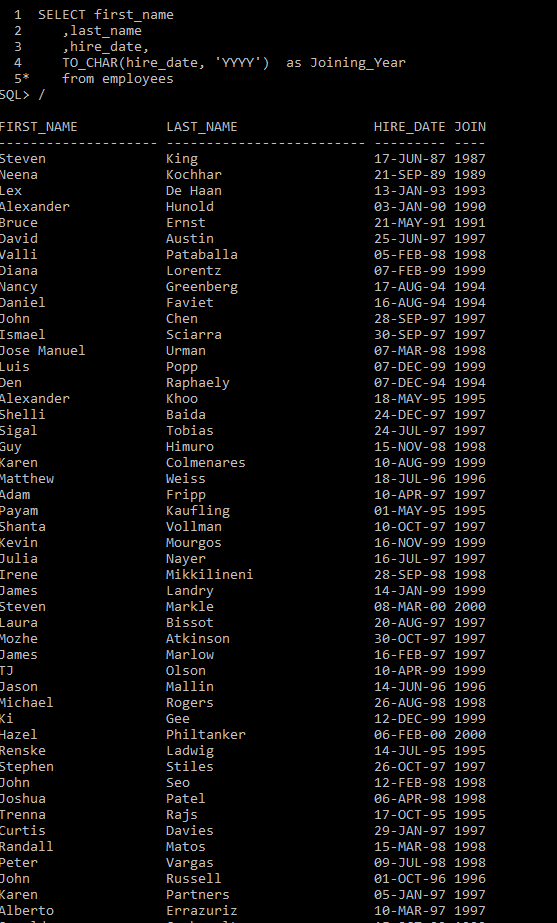


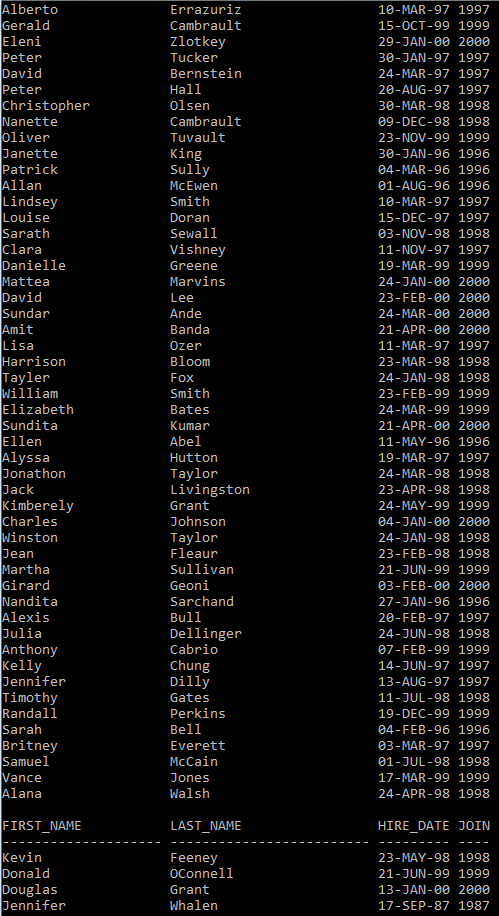
3.Display the first\_name, last\_name, hire\_date, and year of joining for all employees.

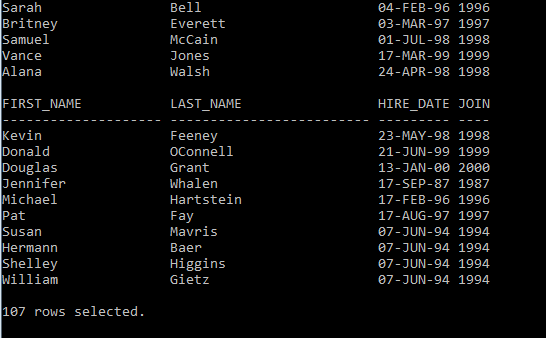
SOLUTION: SELECT first\_name,last\_name,

hire\_date, TO\_CHAR(hire\_date, 'YYYY') as Joining\_Year

from employees;



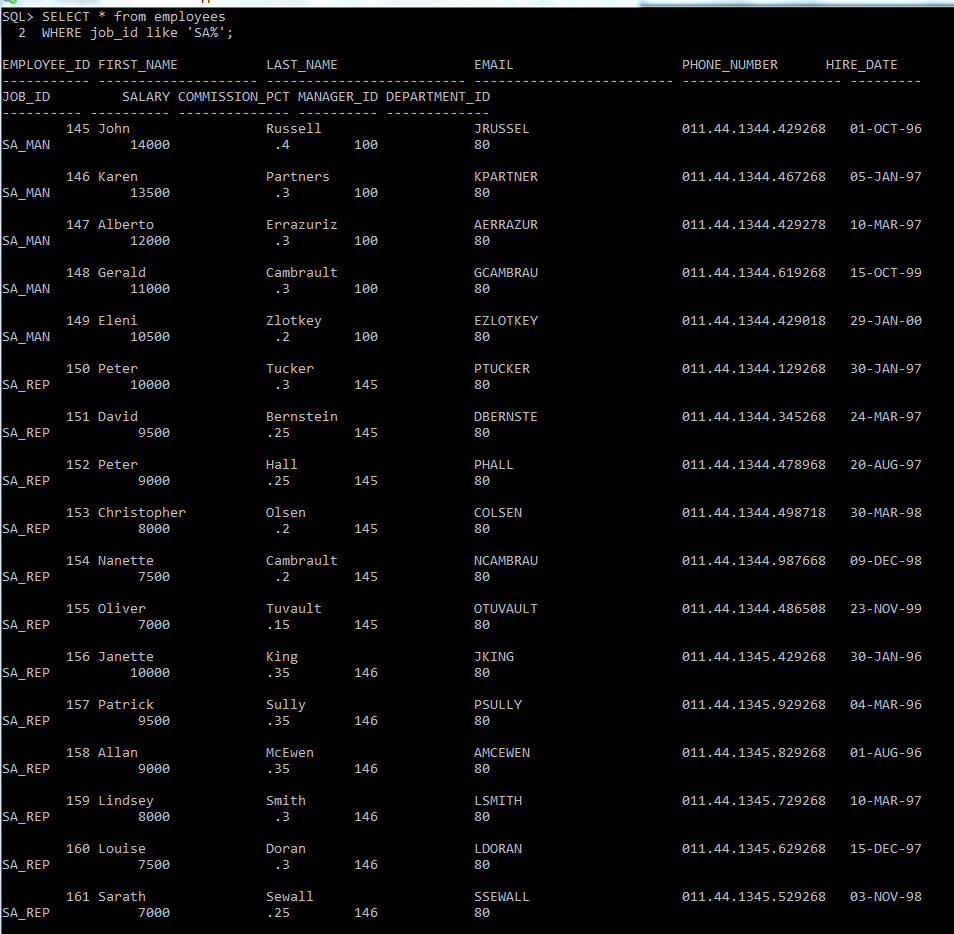


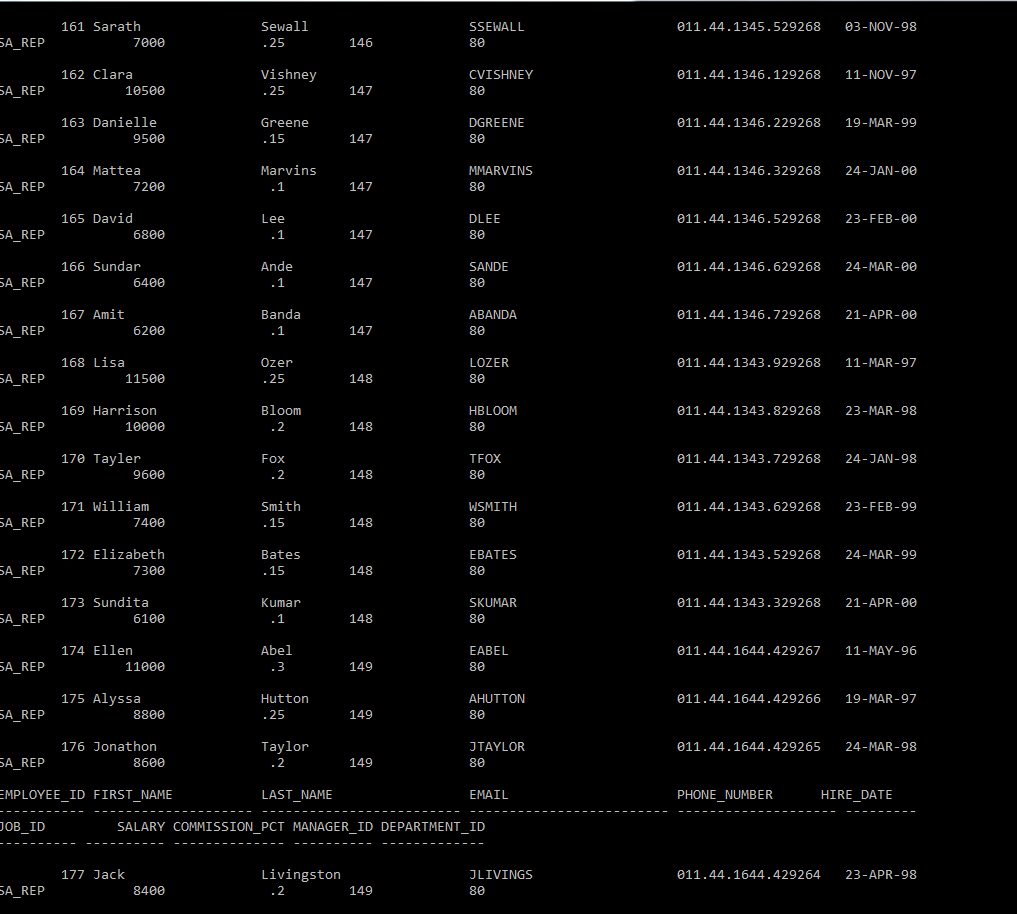


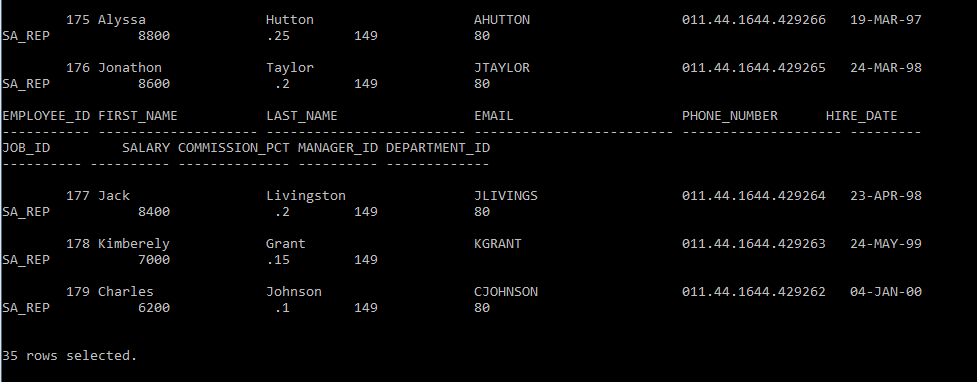
5.Display the employees those who have jobs ids starting with “SA”.

SOLUTION: SELECT \* from employees

WHERE job\_id LIKE 'SA%';







Q2.Create a procedure to increase salary of employees in department 30. The salary increase is 20% for employees earning less than 3000 and 10% for more than 3000.

CREATE OR REPLACE PROCEDURE proc\_sal

AS

CURSOR c1 IS SELECT salary,employee\_id from employees

WHERE department\_id=30; -- Declaring Cursor

v\_c1 c1%rowtype;

inc\_sal employees.salary%type;

BEGIN

OPEN c1; -- Opening Cursor

LOOP

FETCH c1 into v\_c1; -- Fetching the values from cursor

EXIT WHEN c1%NOTFOUND;

IF(v\_c1.salary>3000) THEN

inc\_sal:=v\_c1.salary+(v\_c1.salary\*0.10);

ELSE

inc\_sal:=v\_c1.salary+(v\_c1.salary\*0.20);

END IF;

UPDATE employees

SET salary=inc\_sal

WHERE employee\_id=v\_c1.employee\_id; -- Updating the Salary

END LOOP;

EXCEPTION -- Exception Block

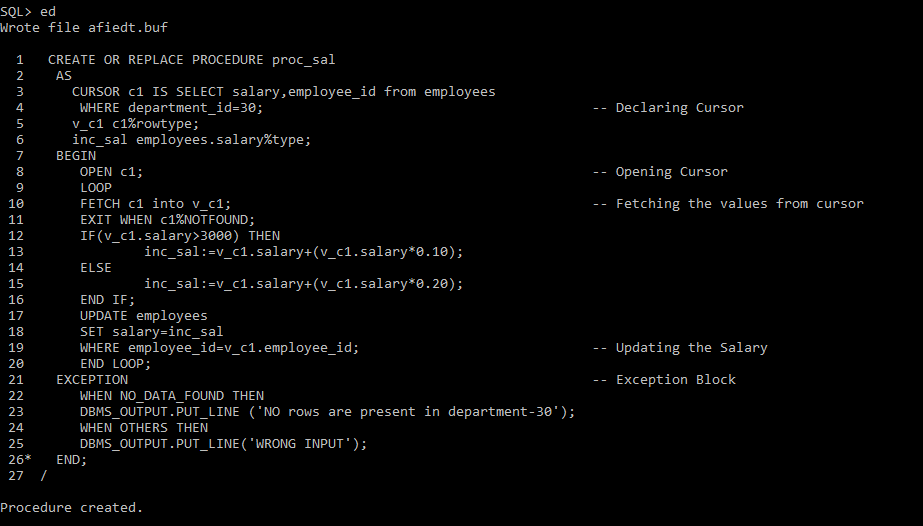
WHEN NO\_DATA\_FOUND THEN

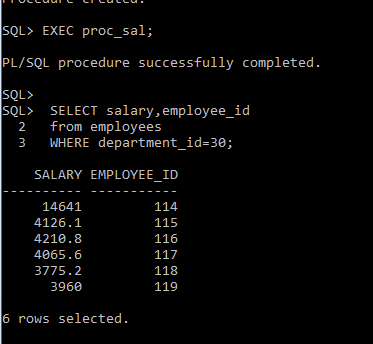
DBMS\_OUTPUT.PUT\_LINE ('NO rows are present in department-30');

WHEN OTHERS THEN

DBMS\_OUTPUT.PUT\_LINE(‘ Wrong Input’);

END;





Q3. Write a function to accept empno & update salary. Update salary as salary + commission only if the employee is a sales representative else salary should not be updated. If comm=0 or null update commission by 5% of salary and then update salary. Returm the calculated value.

SOLUTION: CREATE OR REPLACE FUNCTION UPDATE\_SAL(EMPNO IN NUMBER)

RETURN NUMBER AS

V\_COMM EMPLOYEES. COMMISSION\_PCT%type;

V\_SAL EMPLOYEES.SALARY%TYPE;

V\_JOB\_ID EMPLOYEES.JOB\_ID%TYPE;

V\_NEW\_SAL EMPLOYEES.SALARY%TYPE;

BEGIN

SELECT COMMISSION\_PCT, SAALRY, JOB\_ID

INTO V\_COMM, V\_SAL, V\_JOB\_ID FROM EMPLOYEES

WHERE EMPLOYEE\_ID = EMPNO;

IF (V\_COMM IS NULL OR V\_COMM= 0) THEN

V\_COMM:= V\_SAL\*0.5;

END IF;

V\_NEW\_SAL := V\_SAL + V\_COMM;

IF(V\_JOB\_ID= ‘SA\_REP’) THEN

UPDATE EMPLOYEES SET SALARY = V\_NEW\_SAL WHERE EMPLOYEE\_ID=EMPNO;

END IF;

RETURN V\_NEW\_SAL;

EXCEPTION

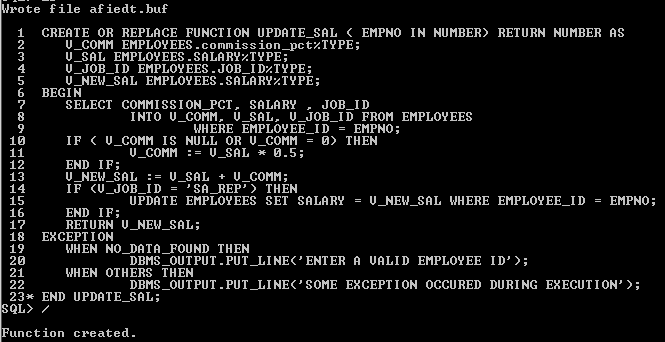
WHEN NO\_DATA\_FOUND THEN

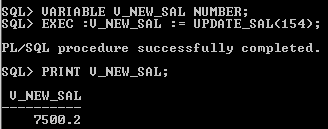
DBMS\_OUTPUT.PUT\_LINE (‘Enter a valid employee ID’);

WHEN OTHERS THEN

DBMS\_OUTPUT.PUT\_LINE(‘ Some exception while executing’);

END UPDATE\_SAL;





Q5.Design a report to display employee name , salry, total salary, total salary for every job within a deparment and the grand total salry . The headings of the column should bu formatted. The salary value should be displayed as $99,999.00

SOLUTION:

D:\module_MPT\SQL 1.1\Q5 Q1.PNG

