PL/SQL

2.3:

DECLARE

cursor cur\_name IS

select staff\_code,staff\_name,staff\_sal from staff\_masters where dept\_code=&n order by dept\_code;

v\_sal number(7,2);

v\_code number(10);

v\_name varchar2(20);

BEGIN

OPEN cur\_name;

LOOP

FETCH cur\_name

INTO v\_code,v\_name,v\_sal;

EXIT

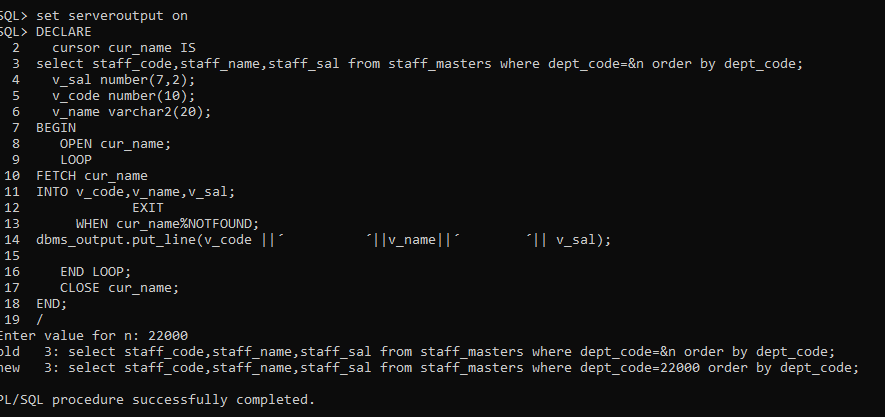
WHEN cur\_name%NOTFOUND;

dbms\_output.put\_line(v\_code ||´ ´||v\_name||´ ´|| v\_sal);

END LOOP;

CLOSE cur\_name;

END;



2.4:

DECLARE

cursor cur\_name2 IS

select staff\_name,staff\_sal from staff\_masters;

v\_sal number(10,2);

v\_code number(10);

v\_name varchar2(20);

BEGIN

OPEN cur\_name2;

LOOP

UPDATE staff\_masters set staff\_sal=staff\_sal+30\*(staff\_sal/100);

FETCH cur\_name2

INTO v\_name,v\_sal;

EXIT

WHEN cur\_name2%NOTFOUND;

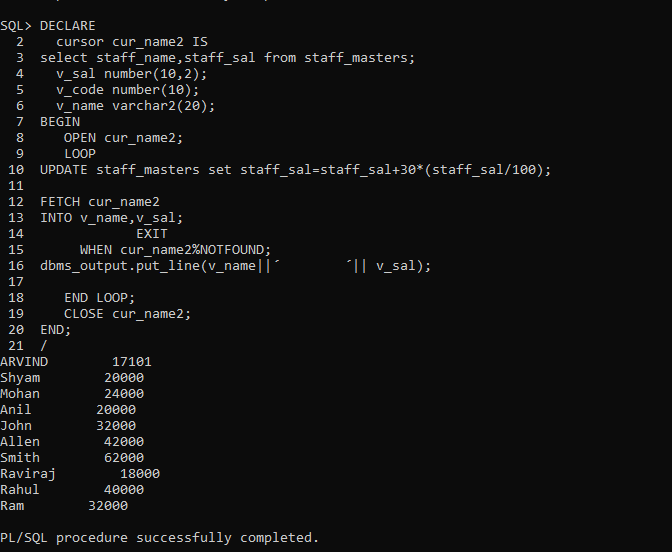
dbms\_output.put\_line(v\_name||´ ´|| v\_sal);

END LOOP;

CLOSE cur\_name2;

END;

/



2.5:

declare

cursor cur\_name IS

select \* from student\_masters order by student\_code;

cursor vk(par\_dept varchar2) IS

select subject1,subject2,subject3 from student\_marks where student\_code=par\_dept order by student\_code;

r\_code student\_masters%rowtype;

v\_name student\_marks.subject1%type;

v\_code student\_marks.subject2%type;

v\_code1 student\_marks.subject3%type;

v\_total number(10,2);

v\_perc number(10,2);

v\_grade varchar2(10);

BEGIN

dbms\_output.put\_line( ´code ´||´ name ´||´ subj1´||´ sub2´||´ sub3´||´ Total´||´ Percentage´ ||´grade´);

OPEN cur\_name;

LOOP

FETCH cur\_name

INTO r\_code;

EXIT

WHEN cur\_name%NOTFOUND;

OPEN vk(r\_code.student\_code);

LOOP

FETCH vk INTO v\_name,v\_code,v\_code1;

EXIT

WHEN vk%NOTFOUND;

v\_total:=v\_name+v\_code+v\_code1;

v\_perc:=(v\_total/3);

IF v\_perc >=80 THEN

v\_grade:=´A´;

ELSIF v\_perc >70 and v\_perc <=80 THEN

v\_grade:=´B´;

ELSIF v\_perc >60 and v\_perc <=70 THEN

v\_grade:=´C´;

ELSE

v\_grade:=´D´;

END IF;

dbms\_output.put\_line(r\_code.student\_code||´ ´||r\_code.student\_name||´ ´||v\_name||´ ´||v\_code||´ ´||v\_code1||´ ´||v\_total||´ ´||v\_perc||´ ´||v\_grade);

END LOOP;

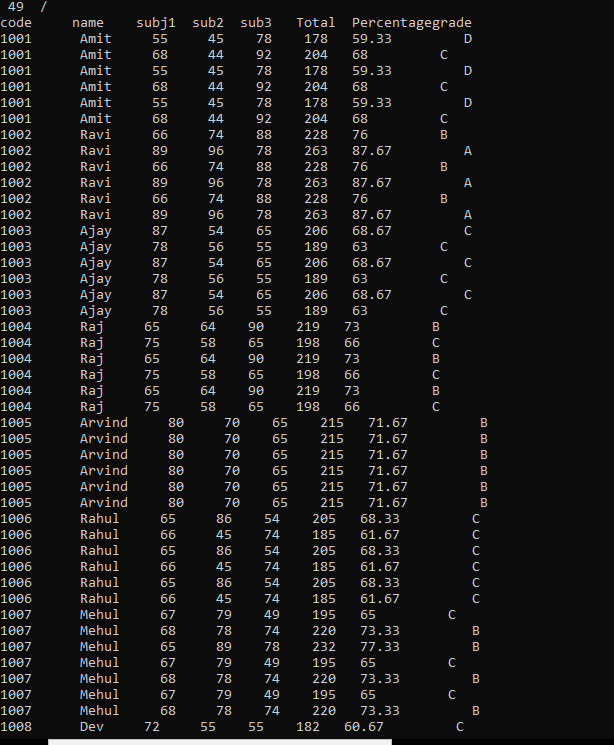
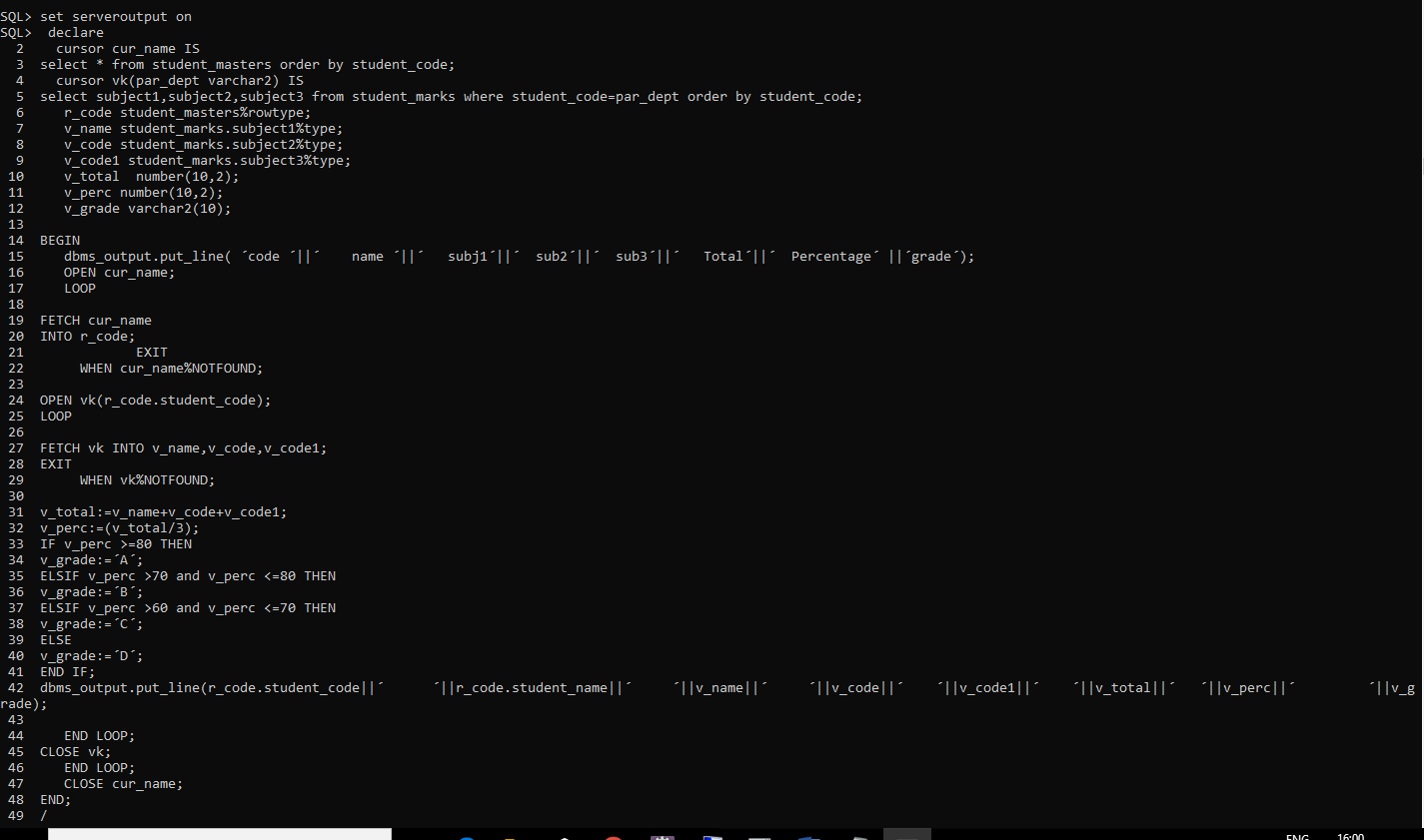
CLOSE vk;

END LOOP;

CLOSE cur\_name;

END;

/



2.6:

DECLARE

cursor prashanth IS

select \* from department\_masters order by dept\_code;

cursor prash(par\_dept varchar2) IS

select \* from staff\_masters where dept\_code=par\_dept and dept\_code=&n order by staff\_name;

r\_code department\_masters%rowtype;

v\_code staff\_masters%rowtype;

BEGIN

dbms\_output.put\_line( ´deptcode ´||´ deptname ´||´ staffcode´ ||´ staffname´);

OPEN prashanth;

LOOP

FETCH prashanth

INTO r\_code;

EXIT

WHEN prashanth%NOTFOUND;

OPEN prash(r\_code.dept\_code);

LOOP

FETCH prash INTO v\_code;

EXIT

WHEN prash%NOTFOUND;

dbms\_output.put\_line(r\_code.dept\_code||´ ´||r\_code.dept\_name||´ ´||v\_code.staff\_code||´ ´||v\_code.staff\_name);

END LOOP;

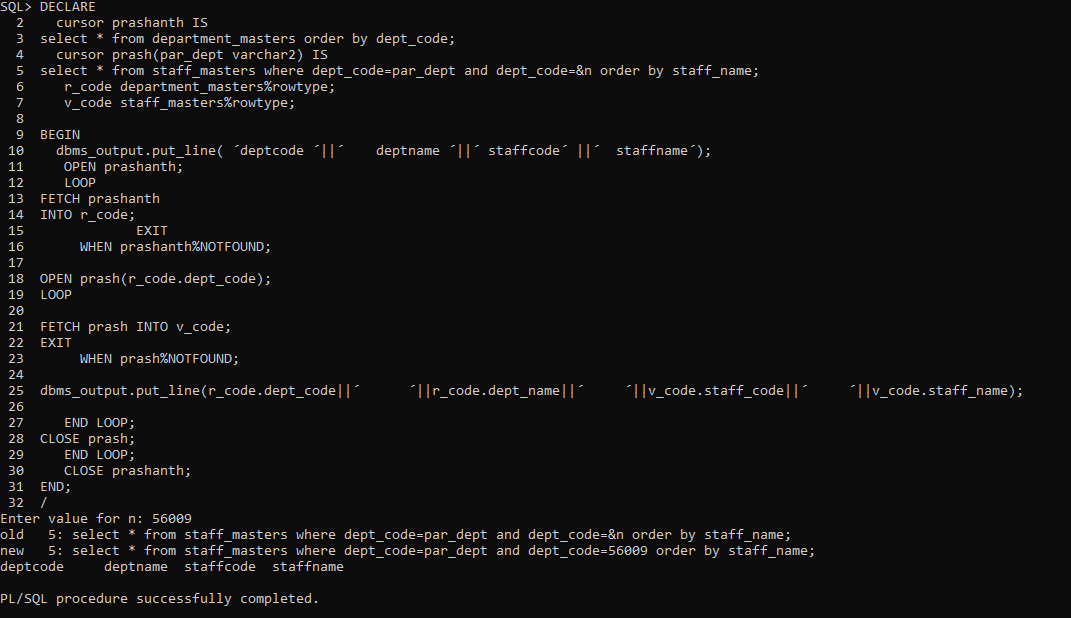
CLOSE prash;

END LOOP;

CLOSE prashanth;

END;

/



4.1:

DECLARE

cursor cur\_name3 IS

select \* from department\_masters order by dept\_code;

cursor vk3(par\_dept varchar2) IS

select max(staff\_sal) from staff\_masters where dept\_code=par\_dept;

r\_code department\_masters%rowtype;

v\_code number(10,2);

BEGIN

dbms\_output.put\_line( ´deptcode ´||´ maxsal ´);

OPEN cur\_name3;

LOOP

FETCH cur\_name3

INTO r\_code;

EXIT

WHEN cur\_name3%NOTFOUND;

OPEN vk3(r\_code.dept\_code);

LOOP

FETCH vk3 INTO v\_code;

EXIT

WHEN vk3%NOTFOUND;

dbms\_output.put\_line(r\_code.dept\_code||´ ´||v\_code);

END LOOP;

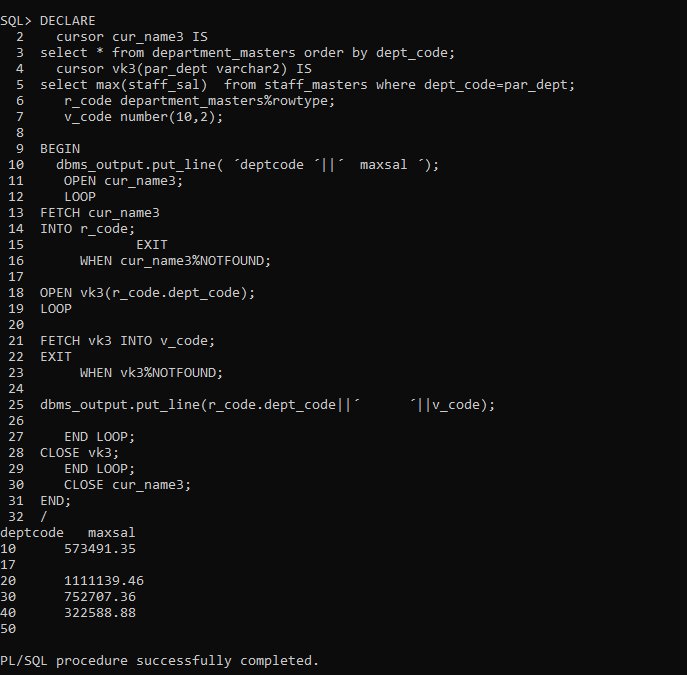
CLOSE vk3;

END LOOP;

CLOSE cur\_name3;

END;

/



4.3:

create or replace procedure upper\_procedure(staff\_c in number,staff\_n out varchar2)is

2 upper\_name varchar2(15);

3 missing\_name exception;

4 begin

5 select staff\_name into upper\_name from staff\_masters where staff\_code=staff\_n;

6 update staff\_masters set staff\_name=upper(upper\_name);

7 if upper\_name is null then

8 raise missing\_name;

9 end if;

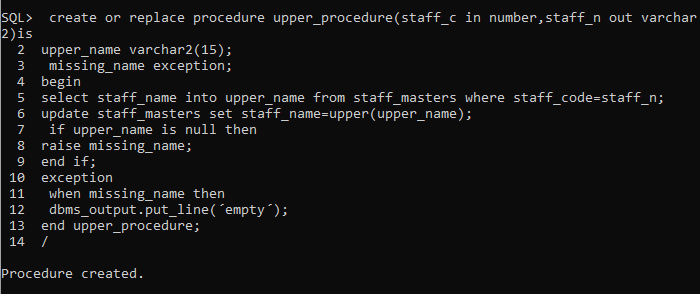
10 exception

11 when missing\_name then

12 dbms\_output.put\_line(´empty´);

13 end upper\_procedure;

14 /



SQL> set serveroutput on;

SQL> declare

2 cursor cur\_sal is select max(staff\_sal) from staff\_masters where dept\_code=50;

3 var1 number;

4 begin

5 open cur\_sal;

6 fetch cur\_sal into var1;

7 dbms\_output.put\_line(var1);

8 close cur\_sal;

9 end;

10 /

4.7:

create or replace procedure proc\_vk522(staf\_code in number) is

staff\_master\_back staff\_masters%rowtype;

new\_sal number;

exp number;

begin

select staff\_sal,(sysdate-hiredate) into new\_sal,exp from staff\_masters where staff\_code=staf\_code;

if exp>2 and exp<5 then

update staff\_masters set staff\_sal=0.2\*new\_sal+new\_sal;

else

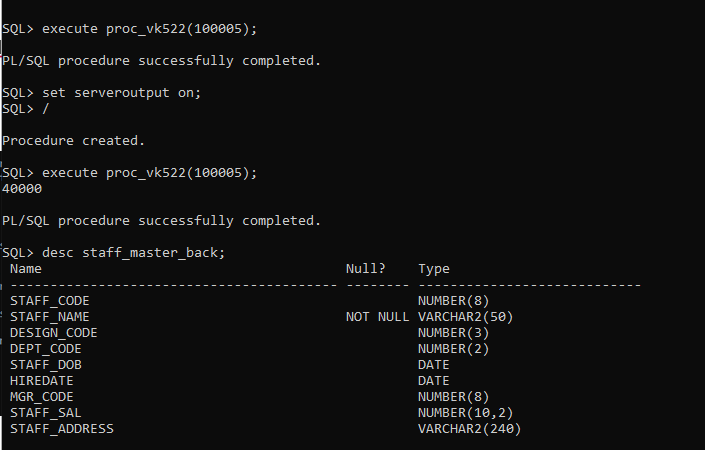
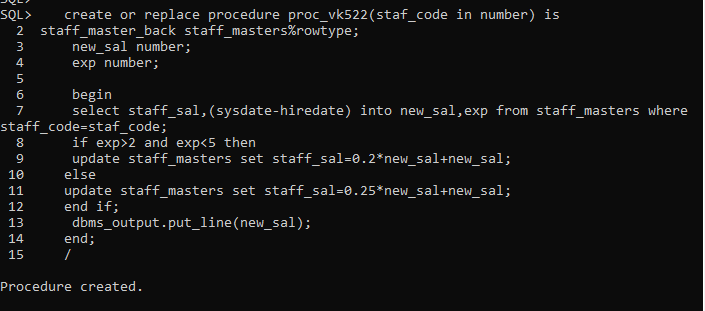
update staff\_masters set staff\_sal=0.25\*new\_sal+new\_sal;

end if;

dbms\_output.put\_line(new\_sal);

end;

/



4.8:

create or replace procedure proc\_vk(bcode in number,stucode out number,stuname out varchar2,publ\_issue\_date out date,exp\_return\_date out date) is

begin

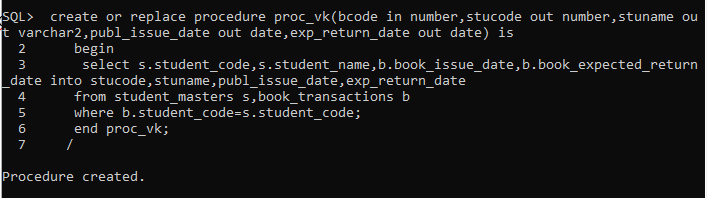
select s.student\_code,s.student\_name,b.book\_issue\_date,b.book\_expected\_return\_date into stucode,stuname,publ\_issue\_date,exp\_return\_date

from student\_masters s,book\_transactions b

where b.student\_code=s.student\_code;

end proc\_vk;

/



4.10:

reate or replace procedure book\_proc(bcode in number,scode in number,stcode in number) is

begin

if(to\_char(sysdate,´day´)=´saturday´ or to\_char(sysdate,´day´)=´sunday´) then

insert into books1 values(bcode,scode,stcode,sysdate,sysdate+12,sysdate);

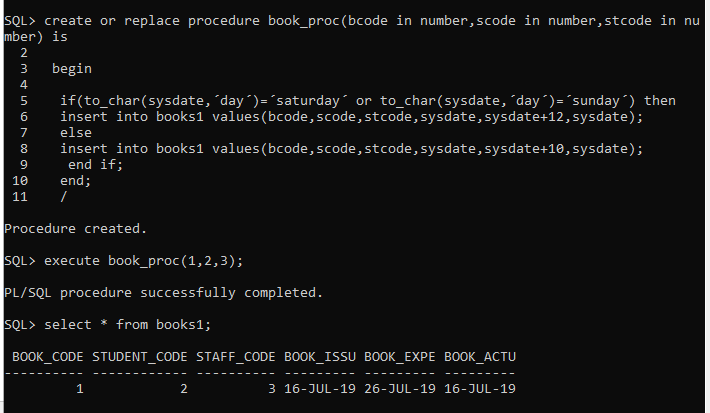
else

insert into books1 values(bcode,scode,stcode,sysdate,sysdate+10,sysdate);

end if;

end;

/



FUNCTIONS:

4.2) create or replace function age(dob in date) return number is agein number;

2 begin

3 agein:=round((sysdate-dob)/365);

4 return(agein);

5 end;

6 /

select age(´27-sep-1998´) from dual;

