Assignment 6:

O/P:

SQL> create or replace procedure calc\_fine(rollno in number,diff in number,dateofreturn in DATE) is

total\_fine number;

begin

total\_fine := 0;

if diff<15 then

total\_fine := 0;

elsif diff>=15 and diff<30 then

total\_fine := (diff-15)\*5;

else

total\_fine := 75 + (diff-30)\*50;

end if;

insert into fine values(rollno,dateofreturn,total\_fine);

update stud\_details set status='R' where roll\_no=rollno;

end;

/

Procedure created.

SQL> declare

rollnumber number;

issue\_date DATE;

gap number;

begin

rollnumber := &rollnumber;

select date\_of\_issue into issue\_date from stud\_details where roll\_no=rollnumber;

gap := to\_date(sysdate) - to\_date(issue\_date);

calc\_fine(rollnumber,gap,sysdate);

end;

/

Enter value for rollnumber: 147 old 6: rollnumber := &rollnumber;

new 6: rollnumber := 147;

PL/SQL procedure

successfully completed.

SQL> select \*from fine;

ROLL\_NO DATE\_OF\_R AMOUNT

---------- --------- ----------

147 31-JUL-18 125

SQL> select \*from stud\_details;

ROLL\_NO NAME DATE\_OF\_I BOOK\_NAME STATU

---------- -------------------- --------- -------------------- -----

113 Siddhesh 30-JUN-18 ABC R

130 Atharva 02-JUL-18 PQR I

144 Manthan 25-JUL-18 XYZ I

SQL> declare

rollnumber number;

issue\_date DATE;

gap number;

begin

rollnumber := &rollnumber;

select date\_of\_issue into issue\_date from stud\_details where roll\_no=rollnumber;

gap := to\_date(sysdate) - to\_date(issue\_date);

calc\_fine(rollnumber,gap,sysdate);

end;

/

Enter value for rollnumber: 144 old 6: rollnumber := &rollnumber;

new 6:rollnumber := 144;

PL/SQL procedure successfully completed.

SQL> select \*from fine;

ROLL\_NO DATE\_OF\_R AMOUNT

---------- --------- ----------

147 31-JUL-18 125

144 31-JUL-18 70

SQL> select \*from stud\_details;

ROLL\_NO NAME DATE\_OF\_I BOOK\_NAME STATU

---------- -------------------- --------- -------------------- -----

113 Siddhesh 30-JUN-18 wings of fire R

130 Atharva 02-JUL-18 into the plex R

205 Nikhil 25-JUL-18 kane and abel I

--------------------------------------------------------------------------------

Q2.)

create or replace function check\_status(accno in number) return number is value number;

stat varchar(5);

begin select status into stat from acc\_dets where acc\_num=accno;

if stat='A' then

value := 1;

else

value := 0;

end if;

return value;

end;

----------------------

PL-SQL

declare acc\_no number;

acc\_value number;

begin acc\_no := &acc\_no;

acc\_value := check\_status(acc\_no);

if acc\_value = 1 then

insert into active\_acc\_dets values(acc\_no,'A');

else

dbms\_output.put\_line('Account inactive!');

end if;

end;

Q3.)

create or replace procedure decide\_class(Name in varchar,marks in number) is marks\_class varchar(30);

begin

if marks>=990 and marks<=1500 then

marks\_class := 'distinction';

elsif marks>=900 and marks<990 then

marks\_class := 'first class';

elsif marks>=825 and marks<900 then

marks\_class := 'higher second class';

end if;

insert into result values(Name,marks\_class);

end;

declare marks\_of\_student number;

name\_of\_student varchar(20);

begin

marks\_of\_student := &marks\_of\_student;

select name into name\_of\_student from stud\_marks where total\_marks=marks\_of\_student;

decide\_class(name\_of\_student,marks\_of\_student);

exception when NO\_DATA\_FOUND then

dbms\_output.put\_line('Student not in the table!');

end;