190031249 **P.MOHITH**

PL/SQL Programs on Case Study 2 & 5
(EMERGENCY ROOM INFORMATION SYSTEM) & (TOUR OPERATING SYSTEM)
PRE-LAB

1) Declare
   fvar number := null; svar number := 5
   Begin
   goto << fproc>>
   if fvar is null then
   << fproc>>
   svar := svar + 5
   end if;
   End;

What will be the value of svar after the execution ?

Ans) Ouput : Syntax Error.

2) What is a stored procedure?

Ans) A stored procedure is a prepared sql code that you can save, so the code can be reused over and over again.

3) What are the different datatypes supported in PL/SQL

Ans) pl/sql provides many pre-defined data-types like integer, floating point , charcater , Boolean, date , collection,refrence and large object (lob) types.

4) What is the Result of the following 'VIK'||NULL||'RAM' ?

Ans) VIKRAM

5) A database is an extensive collection of records. In what form are they stored?

Ans) Database is a collection of data and records. They are stored in form of simple tables. Tables are related if they contain common fields.

6) In the index allocation scheme of blocks to a file, the maximum possible size of the life depends on _____

Ans) the size of the blocks, the number of blocks used for the index and size of address of blocks.

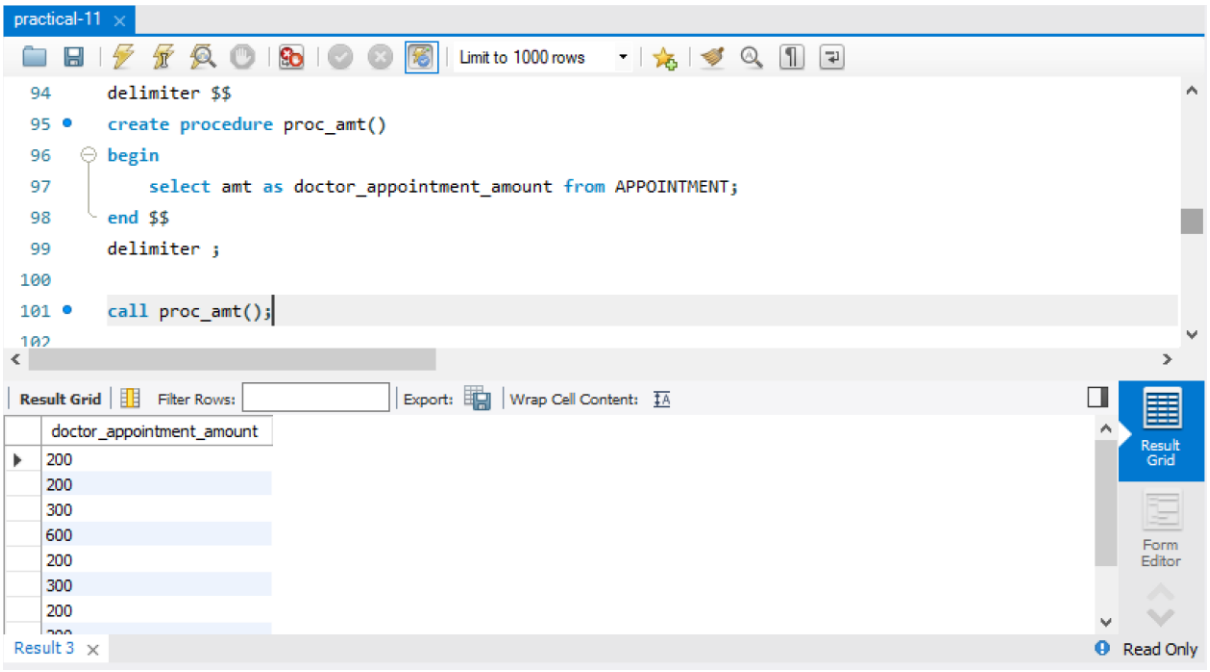
7) How many Clustered indexes can be created on table and why?

Ans) There can only be one clustered index per table , because the data row themselves can be stored in only one order. The only time the data rows in a table are stored in sorted order is when the table contains a clustered index.

INLAB

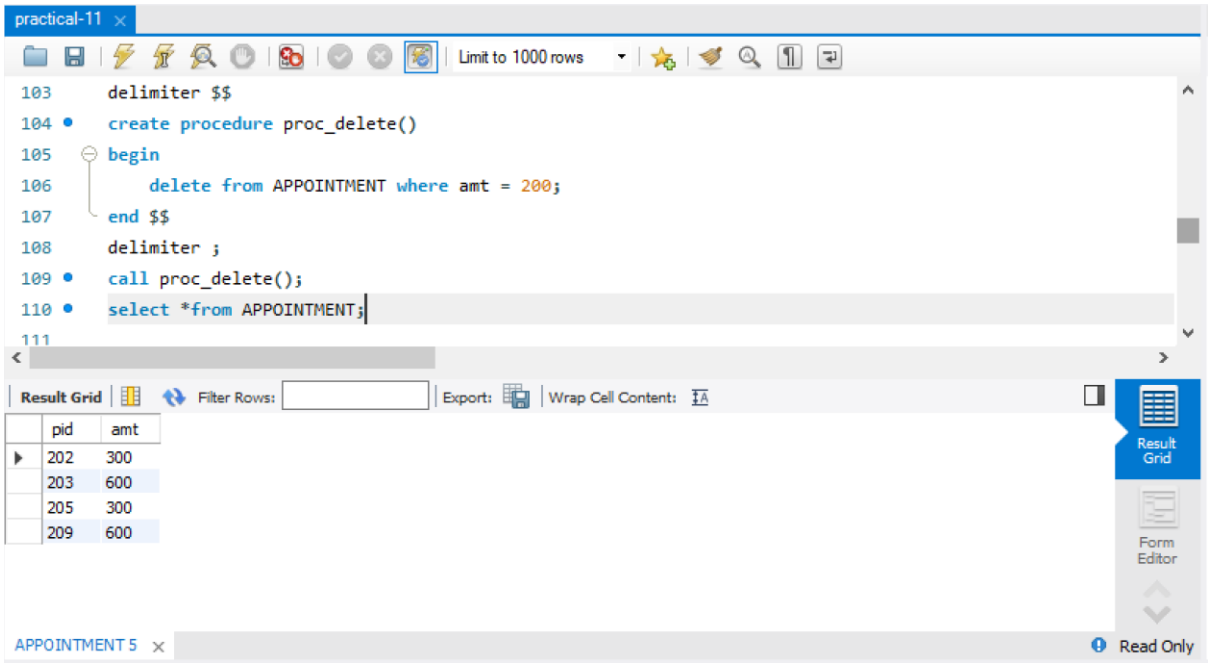
1) Write PL/SQL program to display doctor appointment fee amount value.

```
delimiter $$
create procedure proc_amt()
begin
        select amt as doctor_appointment_amount from APPOINTMENT;
end $$
delimiter ;

call proc_amt();
```



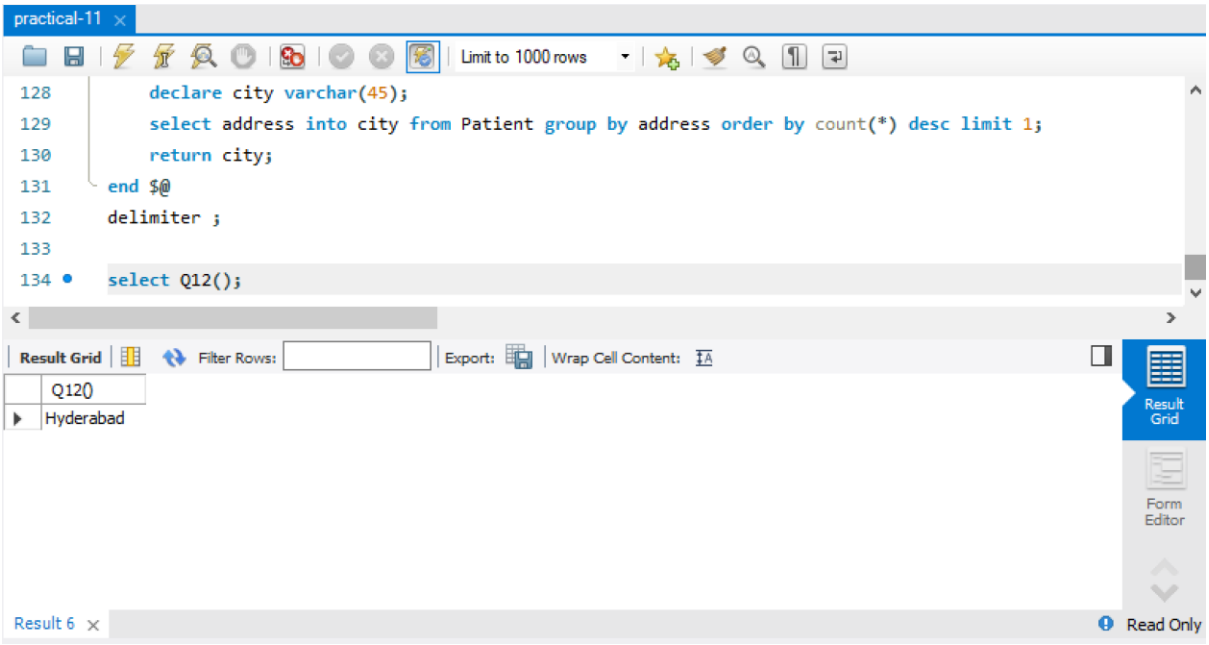
2) Write PL/SQL program to delete appointment amount value 200

```
delimiter $$
create procedure proc_delete()
begin
        delete from APPOINTMENT where amt = 200;
end $$
delimiter ;
call proc_delete();
select *from APPOINTMENT;
```



3) Write a PL/SQL program using functions to display the address details from where the number of patients are more than 3.

```
delimiter $@
create function Q12() returns varchar(100)
deterministic
begin
        declare city varchar(45);
   select address into city from Patient group by address order by count(*) desc limit 1;
   return city;
end $@
delimiter ;

select Q12();
```



4) Write an PL/SQL program using triggers to raise an exception on invalid patient ID

```
delimiter $$
create trigger trig_patient before insert on Patient
for each row
begin
        if new.pid not in  (select pid from Patient) then
                SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'Invalid Patient ID';
        end if;
end $$
delimiter ;

insert into Patient values(210,'Naveen','Kumar','jghfr@gmail.com',2099135327,'Hyderabad'
,STR_TO_DATE('22-04-2020','%d-%m-%Y'));
```

PL/SQL programs on TOUR OPERATING SYSTEM

1) Create a procedure to display the tourist details who are visiting the same place

```
delimiter @@
create procedure Q1 ()
begin
select customer.c_name,tour.tr_dest from customer inner join tour on customer.c_addr = tour.tr_start;
end @@
delimiter ;

call Q1();
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

c_name	tr_dest
Raju	vij
Hari	hyd
Kiran	tnl
Giri	gnt
jaya	Mumbai

Result 4 | Read Only

2) Create a cursor to display the details of the customers/tourists

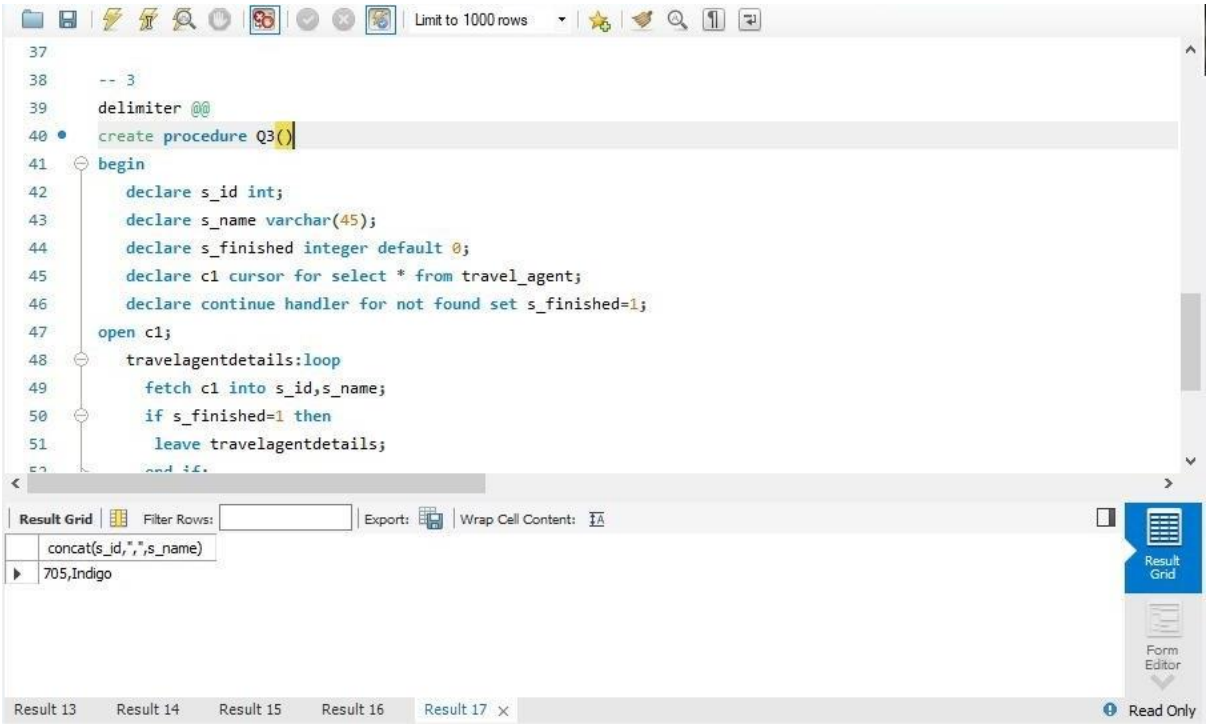
```
create procedure Q2()
begin
   declare s_id int;
   declare s_name varchar(45);
   declare s_addr varchar(45);
   declare s_mobile mediumtext;
   declare s_finished integer default 0;
   declare c1 cursor for select * from customer;
   declare continue handler for not found set s_finished=1;
open c1;
   customerdetails:loop
     fetch c1 into s_id,s_name,s_addr,s_mobile;
     if s_finished=1 then
      leave customerdetails;
     end if;
     select concat(s_id,",",s_name,",",s_addr,",",s_mobile);
    end loop customerdetails;
close c1;
end @@
delimiter ;

call Q2();
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

concat(s_id,",",s_name,",",s_addr,",",s_mobile)
307,Kiran,Guntur,7322938936

Result 5 | Result 6 | Result 7 | Result 8 | Result 9 | Result 10 | Result 11 | Result 12 | Read Only

3) Create a cursor to display the details of the travel agents where the tourists booked their tours



POSTLAB

1) Write a PL/SQL PROGRAM to select the five highest paid employees from the emp table using CURSORS.

```
use practical11;
create table emp_table(
    emp_name varchar(20) not null,
    emp_no int not null,
    primary key (emp_no),
    salary int not null
);
insert into emp_table
values('KING',7839,5000),
      ('SCOTT',7788,3000),
      ('FORD',7902,3000),
      ('JONES',7566,2975),
      ('BLAKE',7698,2850),
      ('CLARK',7782,2450),
      ('ALLEN',7499,1600),
      ('TURNER',7844,1500),
      ('MILLER',7934,1300),
      ('WARD',7521,1250),
      ('MARTIN',7654,1250),
      ('ADAMS',7876,1100),
      ('JAMES',7900,950),
      ('SMITH',7369,800);
```

```
call practical11.emp_salary();
```

s_empno	s_empname	s_empsal
7839	KING	5000

2) Write a PL/SQL Program to check for an Armstrong Number

```
declare
        n number:=1634;
        s number:=0;
        r number;
        len number;
        m number;

begin
```

```
        m := n;

        len := length(to_char(n));

        while n>0
        loop
                r := mod(n , 10);
                s := s + power(r , len);
                n := trunc(n / 10);
        end loop;

        if m = s
        then
                dbms_output.put_line('yes');
        else
                dbms_output.put_line('no');
        end if;

end;
```

3) Write a PL/SQL program to check whether a given character is letter or digit.

```
DECLARE
   get_ctr CHAR(1) := '&input_a_character';
BEGIN
   IF ( get_ctr >= 'A'
       AND get_ctr <= 'Z' )
      OR ( get_ctr >= 'a'
          AND get_ctr <= 'z' ) THEN
    dbms_output.Put_line ('The given character is a letter');    ELSE
    dbms_output.Put_line ('The given character is not a letter');

    IF get_ctr BETWEEN '0' AND '9' THEN
     dbms_output.Put_line ('The given character is a number');      ELSE
     dbms_output.Put_line ('The given character is not a number');
    END IF;
   END IF;
END;
```

Output: A

The given Character is a letter.