Assignment 9 : DBMS Lab

Name: Shishu
Reg_no:2020CA089

Q1). Write a PL/SQL program to print “HELLO WORLD”.

Code:

```
begin
dbms_output.put_line('Hello World');
end;
/
```

Output:

Q2). Write a PL/SQL code for inverting a number. (Example: 1234 – 4321)

Code:

```
DECLARE
or_no int(5);
REV int(5):=0;
R int(5):=0;
BEGIN
or_no:=12345;
WHILE or_no !=0
LOOP
R:=MOD(or_no,10);
REV:=REV*10+R;
or_no:=TRUNC(or_no/10);
END LOOP;
DBMS_OUTPUT.PUT_LINE('THE REVERSE OF A GIVEN
NUMBER IS '||REV);
END;
/
```

Output:

```
Statement processed.
THE REVERSE OF A GIVEN NUMBER IS 54321
```

Q4). Write a PL/SQL code to find the greatest number among three with Anonymous blocks.

Code:

```
DECLARE
A NUMBER(4,2):=20;
B NUMBER(4,2):=50;
C NUMBER(4,2):=70;
BEGIN
IF (A>B AND A>C) THEN
DBMS_OUTPUT.PUT_LINE('A IS GREATER '||''||A);
ELSIF B>C THEN
DBMS_OUTPUT.PUT_LINE('B IS GREATE '||''||B);
ELSE
DBMS_OUTPUT.PUT_LINE('C IS GREATER '||''||C);
END IF;
END;
/
```

Output:

```
Statement processed.
C IS GREATER 70
```

Q5).Write a PL/SQL code to calculate the area of a circle where radius takes values from 3 to 7.Store the calculated area in Table AREA. The schema of table is given below:
AREA (Radius, Area)

Code:

```
DECLARE
   area   NUMBER(6, 2)  ;
   perimeter   NUMBER(6, 2)  ;
   radius NUMBER(1) := 3;
   pi CONSTANT NUMBER(3, 2) := 3.14;
```

```
BEGIN
    area := pi * radius * radius;
    perimeter := 2 * pi * radius;
    dbms_output.Put_line('Area = ' || area);
    dbms_output.Put_line(' Perimeter = ' || perimeter);

END;
/
```

Output:

```
Statement processed.
Area = 28.26
 Perimeter = 18.84
```

Q6). Write a PL/SQL program to accept a number and find the factorial of the number

Code:

```
declare
fac number :=1;
n number := 5;

begin
while n > 0 loop
fac:=n*fac;
n:=n-1;
end loop;
dbms_output.put_line(fac);
end;
/
```

Output:

```
Statement processed.
120
```

Q7). Write a PL/SQL program to display the months between two dates of a year

Code:

```
DECLARE
 v_sysdate DATE := SYSDATE;
 v_systimestamp TIMESTAMP := SYSTIMESTAMP;
 v_date DATE;
 v_number NUMBER(10);
 BEGIN
 v_number := MONTHS_BETWEEN('13-JUN-1973', '23-JAN-
1973');
 DBMS_OUTPUT.PUT_LINE(v_number);
 END;
 /
```

Output:

Q8). Create an Account_Master table & insert the tuples as given the question. Write a PL/SQL code that will accept an account number from user. If the balance of account is less than minimum balance (i.e 1000) deducts Rs 100 from balance.
The schema of table is given below:
Acc_Master (acct_no, acct_holder_name , Balance);
Create table Account_Master(acct_no number(5) primary key,acct_holder_name varchar2(10),balance number(10));
Tuples to be inserted are:
insert into Account_Master values(1,'John',1000); insert into Account_Master values(2,'Denis',100); insert into Account_Master values(3,'Albert',1100); insert into Account_Master values(4,'Charles',700); insert into Account_Master values(5,'Darwin',1700);

Code:

```
create table acct_master(acct_no number(5) primary key,
                         acct_name varchar2(10),
                           balance number(10));
```

```
insert into acct_master values(1, 'aaa', 1000);
insert into acct_master values(2, 'bbb', 100);
insert into acct_master values(3, 'ccc', 1100);
insert into acct_master values(4, 'ddd', 700);
insert into acct_master values(5, 'eee', 1700) ;

DECLARE
xacct_no number(5);
xmin_bal number(5):=1000;
xbalance number(5);

BEGIN

xacct_no:=4;

select balance into xbalance
from acct_master
where acct_no=xacct_no;

IF(xbalance < xmin_bal) THEN
update acct_master
set balance=balance-100
where acct_no=xacct_no;

xbalance:=xbalance-100;
dbms_output.put_line('Rs 100 is deducted and current balance is
'||xbalance);

ELSE
dbms_output.put_line('Current balance is '||xbalance);

END IF;
END;
/
```

Output:

```
Table created.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

Statement processed.
Rs 100 is deducted and current balance is 600
```