**DBMS Lab Upload 4**

**Name:** AYUSH SINHA

**Reg. No. :** 18BCE0299

**Slot** – L37+38

**Q1:** **Swap two numbers without using third variable**.

Ans:

PROGRAM:

declare

a number;

b number;

begin

a:=&a;

b:=&b;

dbms\_output.put\_line('before swapping:');

dbms\_output.put\_line('a='||a||' b='||b);

a:=a+b;

b:=a-b;

a:=a-b;

dbms\_output.put\_line('after swapping:');

dbms\_output.put\_line('a='||a||' b='||b);

end;

/

OUTPUT:

SQL> @D:\ayu.sql;

Enter value for a: 2

old 6: a:=&a;

new 6: a:=2;

Enter value for b: 6

old 7: b:=&b;

new 7: b:=6;

before swapping:

a=2 b=6

after swapping:

a=6 b=2

PL/SQL procedure successfully completed.

**Q2 :Find the greatest of three numbers**

Ans : Program :

declare a number; b number; c number; begin

a := &a; b := &b; c := &c; if a>b then if a>c then dbms\_output.put\_line('a is greatest');

else dbms\_output.put\_line('c is greatest');

end if;

else if b>c then dbms\_output.put\_line('b is greatest');

else dbms\_output.put\_line('c is greatest');

end if;

end if; end; /

**OUTPUT:**

SQL> @C:\Users\18BCE0299\Desktop\cw.sql Enter value for a: 5 old 7: a := &a; new 7: a := 5; Enter value for b: 3 old 8: b := &b; new 8: b := 3; Enter value for c: 9

old 9: c := &c; new 9: c := 9; c is greatest

PL/SQL procedure successfully completed.

**Q3 : Find sum of given digits of a given number. Ans:** Program set serveroutput on;

DECLARE

n INTEGER;

temp\_sum INTEGER;

r INTEGER;

BEGIN n := 123456; temp\_sum := 0;

WHILE n <> 0 LOOP r := MOD(n, 10); temp\_sum := temp\_sum + r; n := Trunc(n / 10); END LOOP;

dbms\_output.Put\_line('sum of digits = '|| temp\_sum);

END;

/

**Output:**

SQL> @D:\ayu.sql; Enter value for n: 1234 old 6: n :=&n; new 6: n :=1234; sum of digits = 10

PL/SQL procedure successfully completed.

**Q4 : Reverse a given number.**

**Ans:** Declare num varchar2(5):='12345'; len number(2);

revnum varchar2(5); Begin len := length(num); for i in reverse 1.. len loop revnum := revnum || substr(num,i,1);

end loop;

dbms\_output.put\_line('given number ='|| num); dbms\_output.put\_line('reverse number ='|| revnum); end;

/

**OUTPUT:**

SQL> @D:\ayu.sql; Enter value for num: 123 old 6: num := &num; new 6: num := 123; given number =123 reverse number =321

PL/SQL procedure successfully completed.

**Q5: Find factorial of a given number.**

**Ans:** Program: declare n number; fac number:=1; i number;

begin n:=&n; for i in 1..n loop fac := fac\*i; end loop;

dbms\_output.put\_line(‘factorial =’ || fac); end;

/

**OUTPUT:**

SQL> @D:\ayu.sql; Enter value for n: 5 old 7: n:=&n; new 7: n:=5; factorial=120

PL/SQL procedure successfully completed.

**Q6: Write a progaram to find fibbonaci sequence.**

**Ans:** Program: declare first number := 0; second number := 1; temp number; n number; i number; begin n:=&n;

dbms\_output.put\_line('Series:'); for i in 2..n loop temp:=first+second;

first := second; second := temp;

dbms\_output.put\_line(temp);

end loop; end;

/

**OUTPUT:**

SQL> @D:\ayu.sql; Enter value for n: 10 old 9: n := &n; new 9: n := 10; Series:

1

2

3

5

8

13

21

34

55

PL/SQL procedure successfully completed.

**Q7 : Check if a given number is Prime or not.**

**Ans:** Program:

declare n number; i number; temp number;

begin n := &n; i := 2; temp := 1;

for i in 2..n/2 loop if mod(n, i) = 0 then temp := 0; exit;

end if;

end loop;

if temp = 1 then dbms\_output.put\_line('Prime');

else dbms\_output.put\_line('Not Prime');

end if;

end;

/

**OUTPUT:**

SQL> @D:\ayu.sql; Enter value for n: 23 old 7: n := &n; new 7: n := 23;

Prime

PL/SQL procedure successfully completed. **Q8 : check whether the number is Armstrong or number Ans:** Program: declare n number; s number:=0; r number; len number; m number; begin n:=&n; 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('armstrong number'); else dbms\_output.put\_line('not armstrong number'); end if;

end;

/

**OUTPUT:**

SQL> @D:\ayu.sql; Enter value for n: **26** old 9: n:=&n; new 9: n:=26; not armstrong number

PL/SQL procedure successfully completed.

SQL> @D:\ayu.sql; Enter value for n: **407** old 9: n:=&n; new 9: n:=407; armstrong number

PL/SQL procedure successfully completed.

**Q9: Find if a number is Palindrome or not.**

**Ans:** Program: declare n number; m number; temp number:=0; rem number;

begin n:=&n; m:=n;

while n>0 loop rem:=mod(n,10); temp:=(temp\*10)+rem; n:=trunc(n/10);

end loop;

if m = temp then dbms\_output.put\_line('The given string is Palindrome');

else dbms\_output.put\_line('The given string is not a Palindrome'); end if;

end;

/

**OUTPUT:**

SQL> @D:\ayu.sql; Enter value for n: **123** old 8: n:=&n; new 8: n:=123;

The given string is not a Palindrome

PL/SQL procedure successfully completed.

SQL> @D:\ayu.sql; Enter value for n: **1221** old 8: n:=&n; new 8: n:=1221;

The given string is Palindrome

PL/SQL procedure successfully completed.

**Q10: Make a Pyramid pattern using stars.**

**Ans:** Program: declare n number := 7; i number; j number;

begin for i in 1..n loop for j in 1..i loop dbms\_output.put('\*');

end loop;

dbms\_output.new\_line; end loop; end; /

**OUTPUT:**

SQL> @D:\ayu.sql;

\*

\*\*

\*\*\*

\*\*\*\*

\*\*\*\*\*

\*\*\*\*\*\*

\*\*\*\*\*\*\*

PL/SQL procedure successfully completed.

Q11: Swap two numbers without using third variable.

Ans:

PROGRAM:

declare

a number;

b number;

begin

a:=&a;

b:=&b;

dbms\_output.put\_line('before swapping:');

dbms\_output.put\_line('a='||a||' b='||b);

a:=a+b;

b:=a-b;

a:=a-b;

dbms\_output.put\_line('after swapping:');

dbms\_output.put\_line('a='||a||' b='||b);

end;

/

OUTPUT:

SQL> @D:\ayu.sql;

Enter value for a: 2

old 6: a:=&a;

new 6: a:=2;

Enter value for b: 6

old 7: b:=&b;

new 7: b:=6;

before swapping:

a=2 b=6

after swapping:

a=6 b=2

PL/SQL procedure successfully completed.