



Advanced Databases

Name =Akhil Pundir

Sap id=590016767

Course=BCA

Batch=3

Experiment-:1

Q1=Write a PL/SQL code to accept the value of A, B & C display which is greater.

Code-:

```
DECLARE
    a NUMBER := 10;
    b NUMBER := 25;
    c NUMBER := 15;
BEGIN
    IF a > b AND a > c THEN
        DBMS_OUTPUT.PUT_LINE('A is greatest: ' || a);
    ELSIF b > a AND b > c THEN
        DBMS_OUTPUT.PUT_LINE('B is greatest: ' || b);
    ELSIF c > a AND c > b THEN
        DBMS_OUTPUT.PUT_LINE('C is greatest: ' || c);
    ELSE
        DBMS_OUTPUT.PUT_LINE('Two or more numbers are equal and greatest.');

END IF;



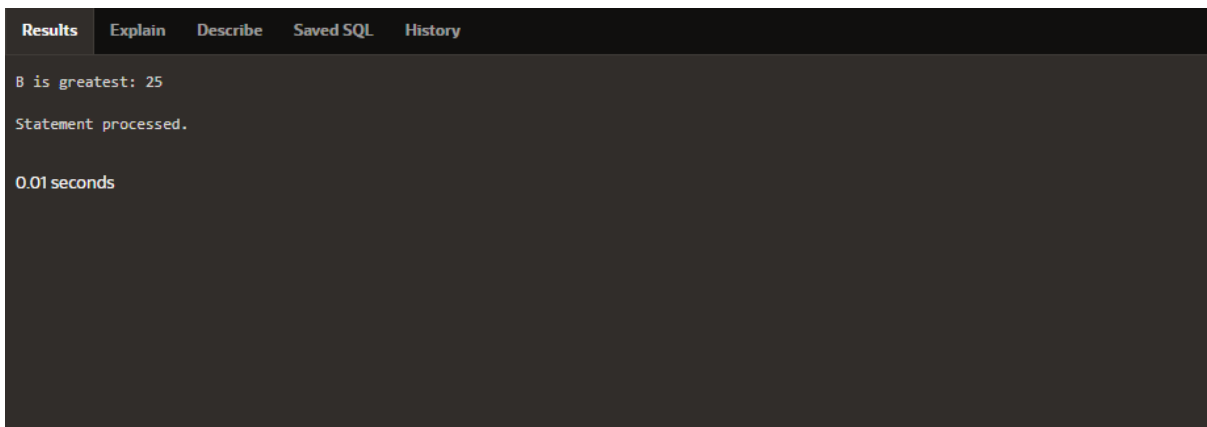
END;



/


```

Output-:



Results	Explain	Describe	Saved SQL	History
B is greatest: 25				
Statement processed.				
0.01 seconds				

Expt. No. _____	Date. ____/____/____
Page No. _____	
<u>Experiment ⇒ 1</u>	
1. Write a PL/SQL code to accept the value of A, B & C display which is greater.	
Code: DECLARE a NUMBER := 10; b NUMBER := 25; c NUMBER := 15; BEGIN IF a > b AND a > c THEN DBMS_OUTPUT.PUT_LINE('A is greatest: ' a); ELSIF b > a AND b > c THEN DBMS_OUTPUT.PUT_LINE('B is greatest: ' b); ELSIF c > a AND c > b THEN DBMS_OUTPUT.PUT_LINE('C is greatest: ' c); ELSE DBMS_OUTPUT.PUT_LINE('Two or more numbers are equal and greatest.'); END IF; END; /	

Q2=Using PL/SQL Statements create a simple loop that display message “Welcome to PL/SQL Programming” 20 times.

Code-:

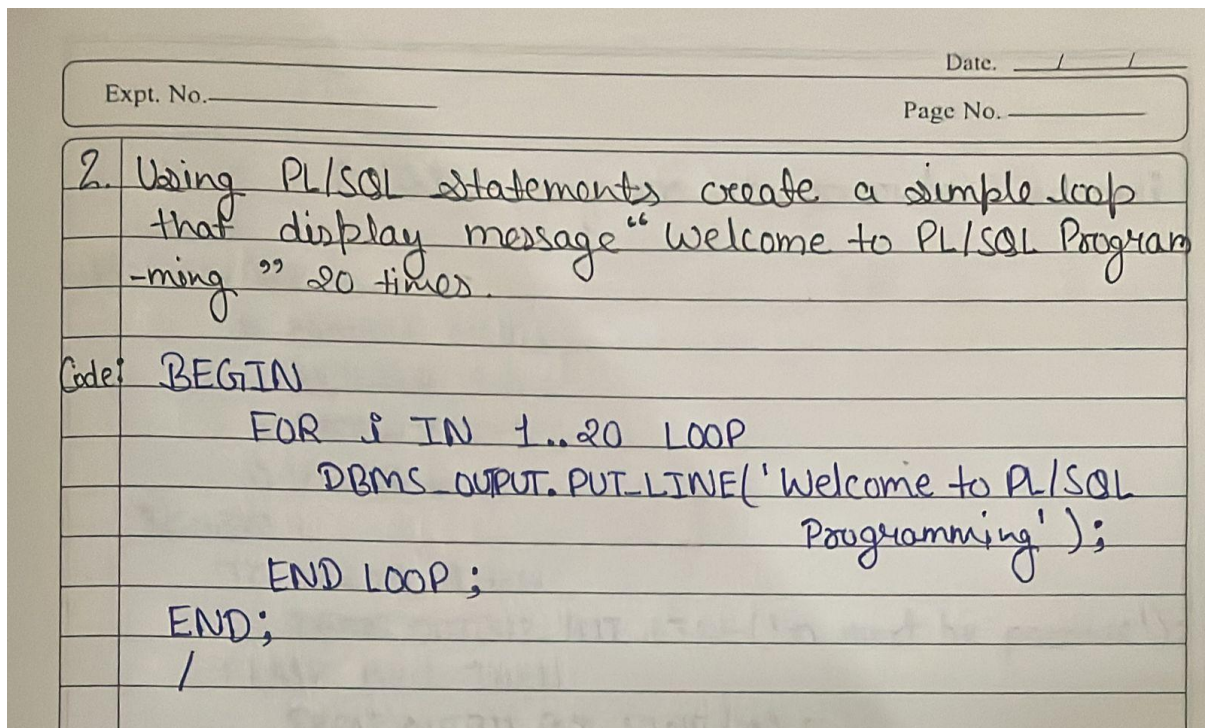
```

BEGIN
  FOR i IN 1..20 LOOP
    DBMS_OUTPUT.PUT_LINE('Welcome to PL/SQL Programming');
  END LOOP;
END;
/

```

Output:-

```
Results Explain Describe Saved SQL History
Welcome to PL/SQL Programming
Welcome to PL/SQL Programming
Welcome to PL/SQL Programming
Welcome to PL/SQL Programming
Welcome to PL/SQL Programming
Welcome to PL/SQL Programming
Welcome to PL/SQL Programming
Welcome to PL/SQL Programming
Welcome to PL/SQL Programming
Welcome to PL/SQL Programming
Welcome to PL/SQL Programming
Welcome to PL/SQL Programming
Welcome to PL/SQL Programming
Welcome to PL/SQL Programming
Welcome to PL/SQL Programming
Welcome to PL/SQL Programming
Welcome to PL/SQL Programming
Welcome to PL/SQL Programming
Welcome to PL/SQL Programming
Welcome to PL/SQL Programming
Statement processed.
```



Q3=Write a PL/SQL code block to find the factorial of a number.

Code:-

```
DECLARE
```

```
  n  NUMBER := 5;
```

```
  fact NUMBER := 1;
```

```
BEGIN
```



```
IF n < 0 THEN
```

```
    DBMS_OUTPUT.PUT_LINE('Factorial not defined for negative numbers');
```

```
ELSE
```

```
    FOR i IN 1..n LOOP
```

```
        fact := fact * i;
```

```
    END LOOP;
```

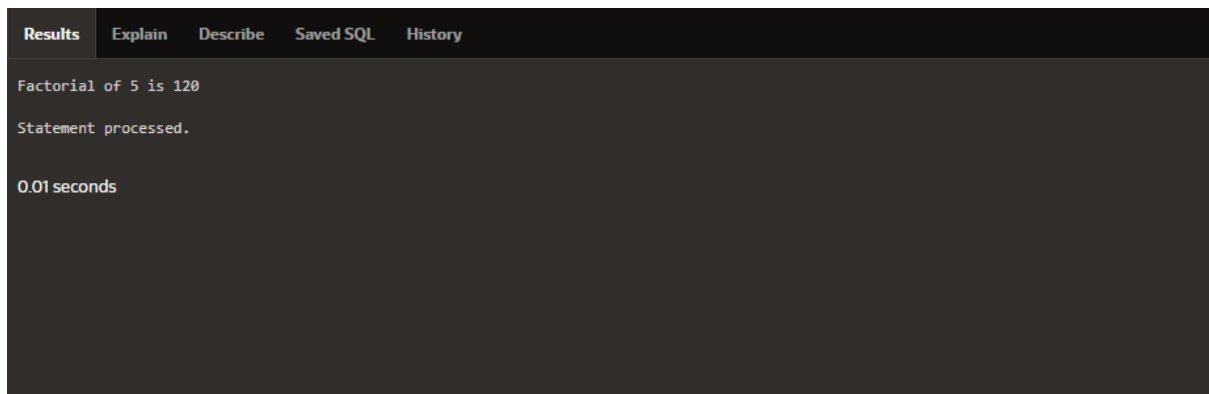
```
    DBMS_OUTPUT.PUT_LINE('Factorial of ' || n || ' is ' || fact);
```

```
END IF;
```

```
END;
```

```
/
```

Output:-

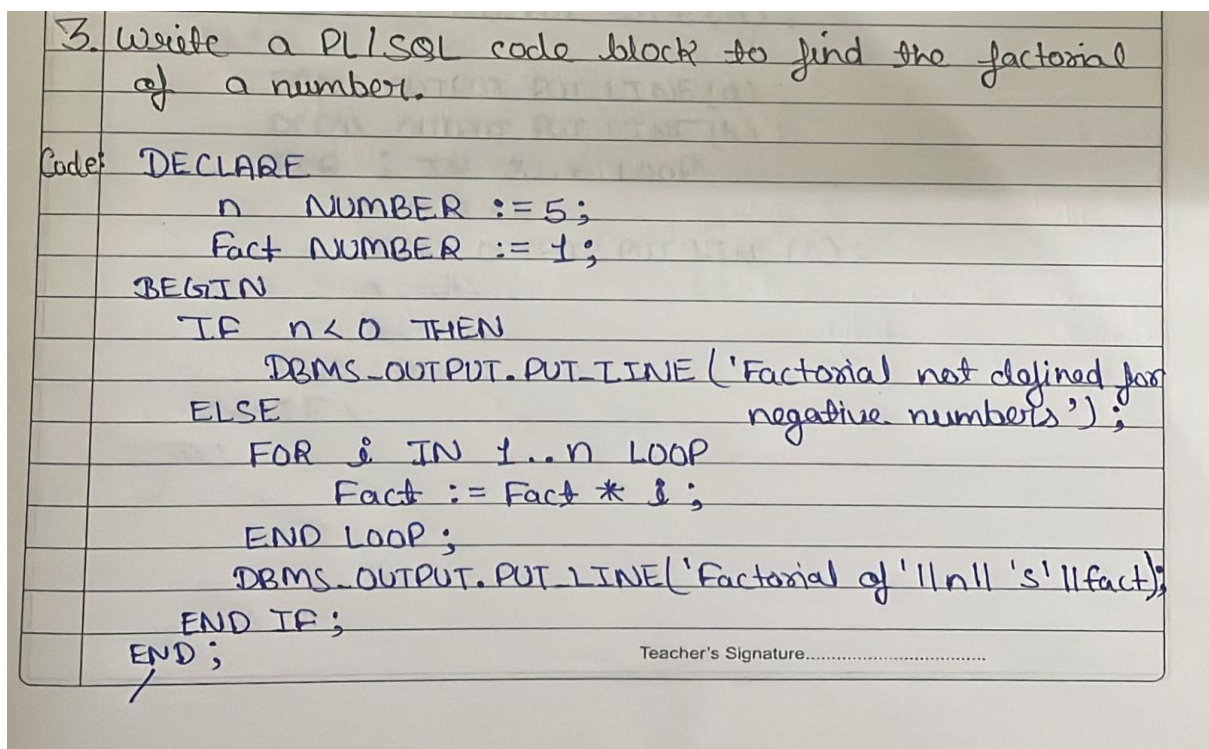


```
Results Explain Describe Saved SQL History
```

Factorial of 5 is 120

Statement processed.

0.01 seconds



3. Write a PL/SQL code block to find the factorial of a number.

```
Code: DECLARE
    n NUMBER := 5;
    fact NUMBER := 1;
BEGIN
    IF n < 0 THEN
        DBMS_OUTPUT.PUT_LINE('Factorial not defined for negative numbers');
    ELSE
        FOR i IN 1..n LOOP
            fact := fact * i;
        END LOOP;
        DBMS_OUTPUT.PUT_LINE('Factorial of ' || n || ' is ' || fact);
    END IF;
END;
```

Teacher's Signature.....

Q4=Write a PL/SQL program to generate Fibonacci series.

Code-:

```
DECLARE
    n NUMBER := 10;
    a NUMBER := 0;
    b NUMBER := 1;
    c NUMBER;
BEGIN
    IF n <= 0 THEN
        DBMS_OUTPUT.PUT_LINE('n must be positive');
    ELSIF n = 1 THEN
        DBMS_OUTPUT.PUT_LINE(a);
    ELSE
        DBMS_OUTPUT.PUT_LINE(a);
        DBMS_OUTPUT.PUT_LINE(b);
        FOR i IN 3..n LOOP
            c := a + b;
            DBMS_OUTPUT.PUT_LINE(c);
            a := b;
            b := c;
        END LOOP;
    END IF;
END;
/
```

Output-:

0
1
1
2
3
5
8
13
21
34

Statement processed.

0.00 seconds

Expt. No. _____

Date. ____/____/____

Page No. _____

4 Write a PL/SQL program to generate Fibonacci series.

Code: DECLARE

n NUMBER := 10;

a NUMBER := 0;

b NUMBER := 1;

c NUMBER;

BEGIN

IF n <= 0 THEN

DBMS_OUTPUT.PUT_LINE('n must be positive');

ELSIF n = 1 THEN

DBMS_OUTPUT.PUT_LINE(a);

ELSE

DBMS_OUTPUT.PUT_LINE(a);

DBMS_OUTPUT.PUT_LINE(b);

FOR i IN 3..n LOOP

c := a + b;

DBMS_OUTPUT.PUT_LINE(c);

a := b;

b := c;

END LOOP;

END IF;

END;

/

Teacher's Signature.....

Q5=Write a PL/SQL code to find the sum of first N numbers.

Code-:

DECLARE

n NUMBER := 10;

s NUMBER := 0;

BEGIN

FOR i IN 1..n LOOP

s := s + i;

END LOOP;

DBMS_OUTPUT.PUT_LINE('Sum of first ' || n || ' numbers is ' || s);

END;

/

Output-:

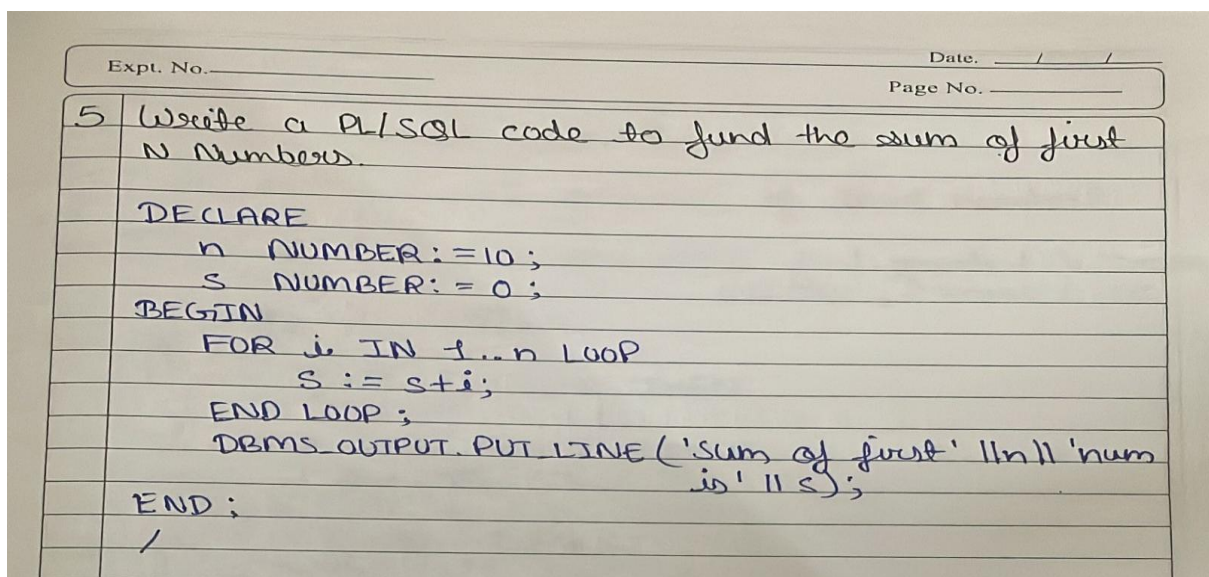


Results Explain Describe Saved SQL History

Sum of first 10 numbers is 55

Statement processed.

0.01 seconds



Expt. No. _____ Date, ____/____/____

Page No. _____

5 Write a PL/SQL code to find the sum of first N Numbers.

DECLARE

n NUMBER := 10;

s NUMBER := 0;

BEGIN

FOR i IN 1..n LOOP

s := s + i;

END LOOP;

DBMS_OUTPUT.PUT_LINE('Sum of first ' || n || ' numbers is ' || s);

END;

/

Experiment-:2

To understand concepts of Function and Procedure in PL/SQL

Q1=Find the greatest of A, B, C (Procedure).

Code-:

```
CREATE OR REPLACE PROCEDURE find_greatest(a NUMBER, b NUMBER, c NUMBER) IS
```

```
    greatest NUMBER;
```

```
BEGIN
```

```
    IF a > b AND a > c THEN
```

```
        greatest := a;
```

```
    ELSIF b > a AND b > c THEN
```

```
        greatest := b;
```

```
    ELSE
```

```
        greatest := c;
```

```
    END IF;
```

```
    DBMS_OUTPUT.PUT_LINE('Greatest number is: ' || greatest);
```

```
END;
```

```
/
```

```
BEGIN
```

```
    find_greatest(10, 25, 15);
```

```
END;
```

```
/
```



Expt. No. _____	Date. ____/____/____
Page No. _____	
<u>Experiment 2</u>	
1.	WAF to find the greatest of three number?
Code:-	<pre> CREATE OR REPLACE PROCEDURE find_greatest (a NUMBER, b NUMBER, c NUMBER) IS greatest NUMBER; BEGIN IF a > b AND a > c THEN greatest := a; ELSEIF b > a AND b > c THEN greatest := b; ELSE greatest := c; END IF; DBMS_OUTPUT.PUT_LINE ('Greatest number is: ' greatest); END; / BEGIN find_greatest (10, 15, 25); END; / </pre>
Teacher's Signature.....	

Q2= Display message 20 times (Procedure)

Code-:

```

CREATE OR REPLACE PROCEDURE print_message IS
BEGIN
    FOR i IN 1..20 LOOP
        DBMS_OUTPUT.PUT_LINE('Welcome to PL/SQL Programming');
    END LOOP;
END;
/

BEGIN
    print_message;

```

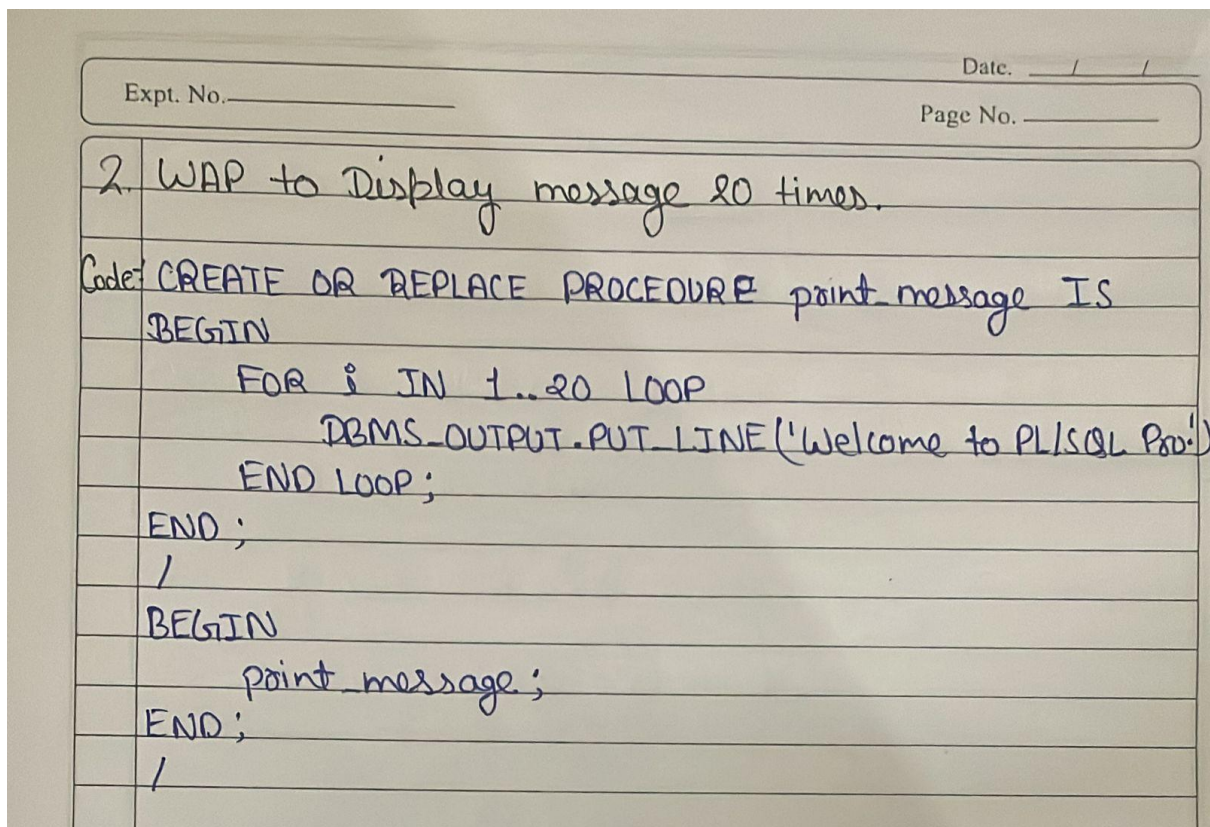
END;

/

Output:-



The screenshot shows the 'Results' tab in SQL Developer. It displays 20 lines of output, each reading 'Welcome to PL/SQL Programming'. Below the output, it states 'Statement processed.' The tabs at the top are 'Results', 'Explain', 'Describe', 'Saved SQL', and 'History'.



The image shows a handwritten lab report on lined paper. At the top right, there are fields for 'Date' and 'Page No.'. Below these, there is a box for 'Expt. No.'. The main content of the report is a handwritten PL/SQL program. The first line is '2. WAP to Display message 20 times.' followed by the code: 'Code: CREATE OR REPLACE PROCEDURE print_message IS BEGIN FOR i IN 1..20 LOOP DBMS_OUTPUT.PUT_LINE('Welcome to PL/SQL Pro.') END LOOP; END; / BEGIN print_message; END; /'.

Q3= Factorial of a number (Function)

Code:-

CREATE OR REPLACE FUNCTION fact(n NUMBER)

RETURN NUMBER IS

f NUMBER := 1;

BEGIN

FOR i IN 1..n LOOP

f := f * i;

END LOOP;

RETURN f;

END;

/

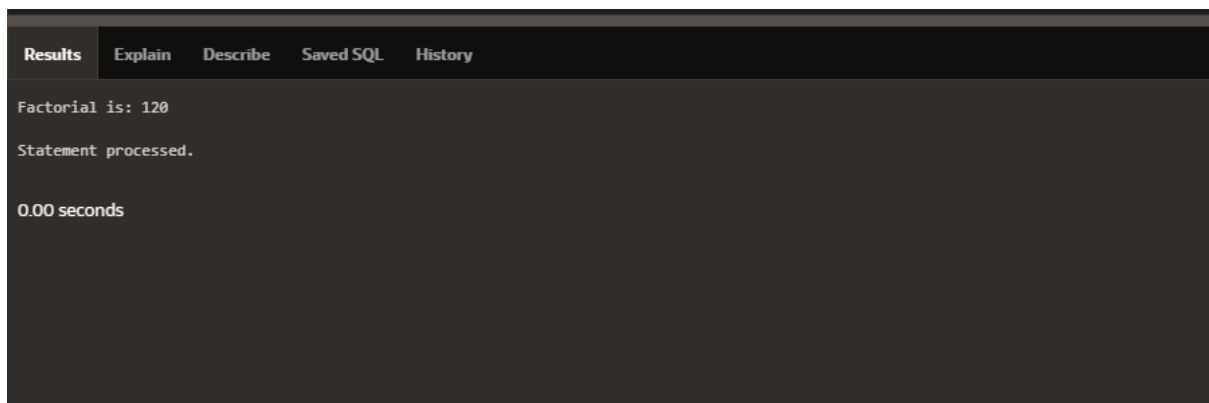
BEGIN

DBMS_OUTPUT.PUT_LINE('Factorial is: ' || fact(5));

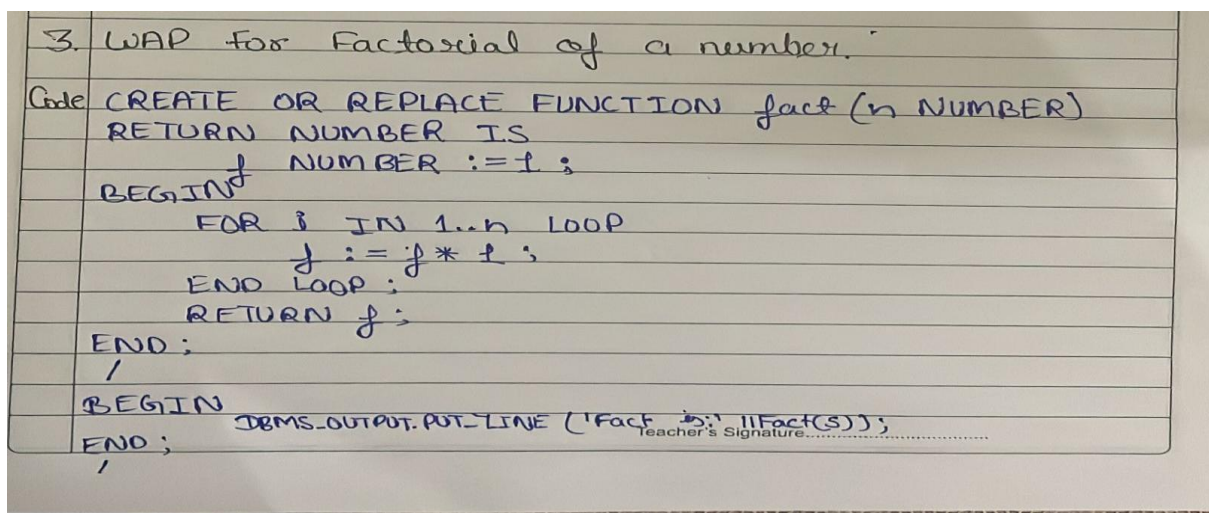
END;

/

Output-:



The screenshot shows a database interface with a dark theme. At the top, there are tabs: 'Results', 'Explain', 'Describe', 'Saved SQL', and 'History'. The 'Results' tab is selected. Below the tabs, the text 'Factorial is: 120' is displayed. Underneath that, it says 'Statement processed.' and at the bottom, '0.00 seconds'.



The image shows a handwritten code snippet on lined paper. The code is as follows:

```
3. WAP for Factorial of a number.  
Code CREATE OR REPLACE FUNCTION fact(n NUMBER)  
RETURN NUMBER IS  
    f NUMBER := 1;  
BEGIN  
    FOR i IN 1..n LOOP  
        f := f * i;  
    END LOOP;  
    RETURN f;  
END;  
/  
BEGIN  
    DBMS_OUTPUT.PUT_LINE('Fact is: ' || fact(5));  
END;  
/
```

There is a signature line at the bottom right that says 'Teacher's Signature.....'.

Q4= Fibonacci series (Procedure).

Code-:

```
CREATE OR REPLACE PROCEDURE fibonacci(n NUMBER) IS
```

```
    a NUMBER := 0;
```

```
    b NUMBER := 1;
```

```
    c NUMBER;
```

```
BEGIN
```

```
    DBMS_OUTPUT.PUT_LINE('Fibonacci series:');
```

```
    DBMS_OUTPUT.PUT_LINE(a);
```

```
    DBMS_OUTPUT.PUT_LINE(b);
```

```
    FOR i IN 3..n LOOP
```

```
        c := a + b;
```

```
        DBMS_OUTPUT.PUT_LINE(c);
```

```
        a := b;
```

```
        b := c;
```

```
    END LOOP;
```

```
END;
```

```
/
```

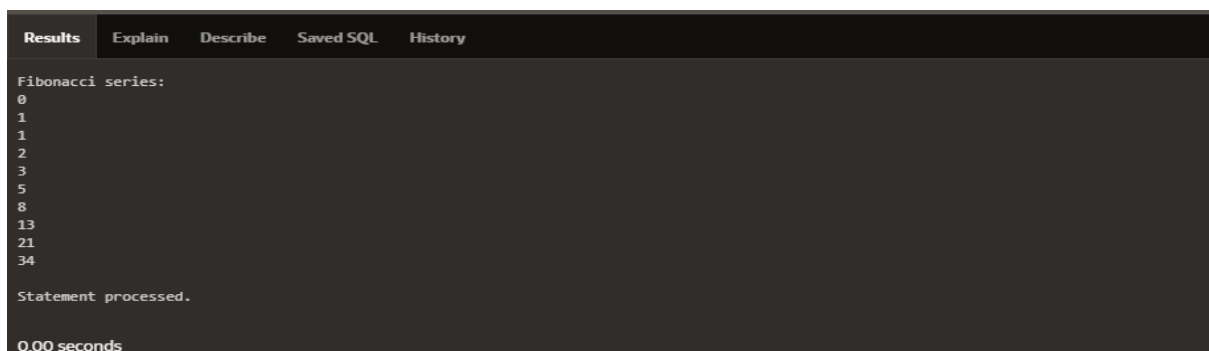
```
BEGIN
```

```
    fibonacci(10);
```

```
END;
```

```
/
```

Output-:



The screenshot shows a SQL IDE interface with a dark theme. At the top, there are tabs labeled 'Results', 'Explain', 'Describe', 'Saved SQL', and 'History'. The 'Results' tab is active, displaying the output of the SQL execution. The output text is as follows:

```
Fibonacci series:
0
1
1
2
3
5
8
13
21
34
Statement processed.
0.00 seconds
```


Expt. No. _____	Date. ____/____/____
Page No. _____	
4.	WAP for Fibonacci Series (Procedure).
Code:	<pre> CREATE OR REPLACE PROCEDURE Fibonacci(n NUMBER) IS a NUMBER := 0; b NUMBER := 1; c NUMBER; BEGIN DBMS_OUTPUT.PUT_LINE('Fibonacci series:'); DBMS_OUTPUT.PUT_LINE(a); DBMS_OUTPUT.PUT_LINE(b); FOR i IN 3..n LOOP c := a+b; DBMS_OUTPUT.PUT_LINE(c); a := b; b := c; END LOOP; END; / BEGIN Fibonacci(10); END; / </pre>

Q5= Sum of first N numbers (Function).

Code-:

CREATE OR REPLACE FUNCTION sum_n(n NUMBER)

RETURN NUMBER IS

s NUMBER := 0;

BEGIN

FOR i IN 1..n LOOP

s := s + i;

```

END LOOP;

RETURN s;

END;

/

BEGIN

    DBMS_OUTPUT.PUT_LINE('Sum of first 10 numbers is: ' || sum_n(10));

END;

/

```

Output-:



```

Results Explain Describe Saved SQL History
Sum of first 10 numbers is: 55
Statement processed.
0.01 seconds

```

Expt. No. _____ Date. ____/____/____

Page No. _____

S. WAP for sum of First N numbers.

Code: CREATE OR REPLACE FUNCTION sum_n(n NUMBER)
RETURN NUMBER IS
 s NUMBER := 0;
BEGIN
 FOR i IN 1..n LOOP
 s := s + i;
 END LOOP;
 RETURN s;
END;
/
BEGIN
 DBMS_OUTPUT.PUT_LINE('Sum of first 10 num: || sum
n(10));
END;
/