LA-6(A)

NAME: YASH SIGCHI

REG NO: 23BAI1242

DATE: 17/02/2025

PL/SQL BLOCK FOR BELOW

- Factorial of a number

```
SQL> DECLARE
  2       n NUMBER;
  3       fact NUMBER := 1;
  4  BEGIN
  5       n := &n;
  6       FOR i IN 1..n LOOP
  7           fact := fact * i;
  8       END LOOP;
  9       DBMS_OUTPUT.PUT_LINE('BAI1242 - Factorial of ' || n ||
 10  END;
 11  /
Enter value for n: 6
old   5:       n := &n;
new   5:       n := 6;
BAI1242 - Factorial of 6 is 720

PL/SQL procedure successfully completed.
```

- Greatest of three numbers (input from user)

```
Run SQL Command Line

SQL*Plus: Release 11.2.0.2.0 Production on Mon Feb 17 17:05:51 2

Copyright (c) 1982, 2014, Oracle.  All rights reserved.

SQL> CONNECT SYSTEM
Enter password:
Connected.
SQL> SET SERVEROUTPUT ON;
SQL> DECLARE
  2       num1 NUMBER;
  3       num2 NUMBER;
  4       num3 NUMBER;
  5       greatest NUMBER;
  6   BEGIN
  7       num1 := &num1;
  8       num2 := &num2;
  9       num3 := &num3;
 10
 11       greatest := CASE
 12                       WHEN num1 >= num2 AND num1 >= num3 THEN
 13                       WHEN num2 >= num1 AND num2 >= num3 THEN
 14                       ELSE num3
 15                   END;
 16
 17       DBMS_OUTPUT.PUT_LINE('BAI1242 - Greatest of ' || num1 |
 18   END;
 19   /
Enter value for num1: 5
old   7:       num1 := &num1;
new   7:       num1 := 5;
Enter value for num2: 8
old   8:       num2 := &num2;
new   8:       num2 := 8;
Enter value for num3: 12
old   9:       num3 := &num3;
new   9:       num3 := 12;
BAI1242 - Greatest of 5, 8, and 12 is 12

PL/SQL procedure successfully completed.
```

- Find whether the given string is palindrome or not (input from user)

```
SQL> DECLARE
  2      input_string VARCHAR2(50);
  3      reversed_string VARCHAR2(50);
  4      is_palindrome BOOLEAN := FALSE;
  5  BEGIN
  6      input_string := '&Enter_string_here';
  7      reversed_string := '';
  8      FOR i IN REVERSE 1..LENGTH(input_string) LOOP
  9          reversed_string := reversed_string || SUBSTR(input_s
 10      END LOOP;
 11      IF input_string = reversed_string THEN
 12          is_palindrome := TRUE;
 13      END IF;
 14      IF is_palindrome THEN
 15          DBMS_OUTPUT.PUT_LINE('BAI1242 - ' || input_string |
 16      ELSE
 17          DBMS_OUTPUT.PUT_LINE('BAI1242 - ' || input_string |
 18      END IF;
 19  END;
 20  /
Enter value for enter_string_here: MADAM
old   6:      input_string := '&Enter_string_here';
new   6:      input_string := 'MADAM';
BAI1242 - MADAM is a palindrome.

PL/SQL procedure successfully completed.
```

```
SQL> DECLARE
  2      input_string VARCHAR2(50);
  3      reversed_string VARCHAR2(50);
  4      is_palindrome BOOLEAN := FALSE;
  5  BEGIN
  6      input_string := '&Enter_string_here';
  7      reversed_string := '';
  8      FOR i IN REVERSE 1..LENGTH(input_string) LOOP
  9          reversed_string := reversed_string || SUBSTR(input_s
 10      END LOOP;
 11      IF input_string = reversed_string THEN
 12          is_palindrome := TRUE;
 13      END IF;
 14      IF is_palindrome THEN
 15          DBMS_OUTPUT.PUT_LINE('BAI1242 - ' || input_string |
 16      ELSE
 17          DBMS_OUTPUT.PUT_LINE('BAI1242 - ' || input_string |
 18      END IF;
 19  END;
 20  /
Enter value for enter_string_here: YASH
old   6:      input_string := '&Enter_string_here';
new   6:      input_string := 'YASH';
BAI1242 - YASH is NOT a palindrome.

PL/SQL procedure successfully completed.
```

- Sum of 100 numbers

```
SQL> DECLARE
  2      sum_of_numbers NUMBER := 0;
  3  BEGIN
  4      -- Sum of first 100 numbers
  5      FOR i IN 1..100 LOOP
  6          sum_of_numbers := sum_of_numbers + i;
  7      END LOOP;
  8      DBMS_OUTPUT.PUT_LINE('BAI1242 - Sum of first 100 number
  9  END;
 10  /
BAI1242 - Sum of first 100 numbers is 5050

PL/SQL procedure successfully completed.
```

- Reverse a given number (input from user)

```
SQL> DECLARE
  2      number_to_reverse NUMBER;
  3      reversed_number NUMBER := 0;
  4  BEGIN
  5      number_to_reverse := &Enter_number_here;
  6      WHILE number_to_reverse > 0 LOOP
  7          reversed_number := reversed_number * 10 + MOD(numbe
  8          number_to_reverse := TRUNC(number_to_reverse / 10);
  9      END LOOP;
 10      DBMS_OUTPUT.PUT_LINE('BAI1242 - Reverse of the number i
 11  END;
 12  /
Enter value for enter_number_here: 96
old   5:      number_to_reverse := &Enter_number_here;
new   5:      number_to_reverse := 96;
BAI1242 - Reverse of the number is 69

PL/SQL procedure successfully completed.
```

- Print Fibonacci series of a user input

```
SQL> DECLARE
  2       fib_limit NUMBER;
  3       fib1 NUMBER := 0;
  4       fib2 NUMBER := 1;
  5       next_fib NUMBER;
  6   BEGIN
  7       fib_limit := &Enter_fibonacci_limit_here;
  8       DBMS_OUTPUT.PUT_LINE('BAI1242 - Fibonacci series:');
  9       DBMS_OUTPUT.PUT_LINE('BAI1242 - ' || fib1);
 10       DBMS_OUTPUT.PUT_LINE('BAI1242 - ' || fib2);
 11       FOR i IN 3..fib_limit LOOP
 12           next_fib := fib1 + fib2;
 13           DBMS_OUTPUT.PUT_LINE('BAI1242 - ' || next_fib);
 14           fib1 := fib2;
 15           fib2 := next_fib;
 16       END LOOP;
 17   END;
 18   /
Enter value for enter_fibonacci_limit_here: 9
old   7:       fib_limit := &Enter_fibonacci_limit_here;
new   7:       fib_limit := 9;
BAI1242 - Fibonacci series:
BAI1242 - 0
BAI1242 - 1
BAI1242 - 1
BAI1242 - 2
BAI1242 - 3
BAI1242 - 5
BAI1242 - 8
BAI1242 - 13
BAI1242 - 21

PL/SQL procedure successfully completed.
```