--WEEK 12--

CREATE OR REPLACE PROCEDURE display\_hello\_world IS

BEGIN

DBMS\_OUTPUT.PUT\_LINE('Hello World');

END;

--PART 2—

DECLARE

v\_input\_number NUMBER := 12345;

v\_sum\_of\_digits NUMBER := 0;

v\_num\_str VARCHAR2(100);

v\_digit CHAR(1);

BEGIN

-- Convert the number to a string to extract digits

v\_num\_str := TO\_CHAR(v\_input\_number);

-- Loop through each character in the string

FOR i IN 1..LENGTH(v\_num\_str) LOOP

-- Extract each digit character

v\_digit := SUBSTR(v\_num\_str, i, 1);

-- Convert the digit character back to a number and add it to the sum

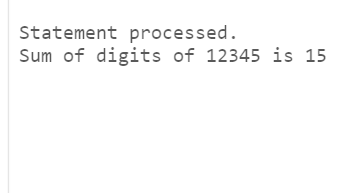
v\_sum\_of\_digits := v\_sum\_of\_digits + TO\_NUMBER(v\_digit);

END LOOP;

-- Output the result

DBMS\_OUTPUT.PUT\_LINE('Sum of digits of ' || v\_input\_number || ' is ' || v\_sum\_of\_digits);

END;

/ 

--PART3--

CREATE OR REPLACE PROCEDURE CountEvenOddNumbers (

p\_numbers\_array IN sys.odcinumberlist,

p\_even\_count OUT NUMBER,

p\_odd\_count OUT NUMBER

)

IS

BEGIN

-- Initialize counts

p\_even\_count := 0;

p\_odd\_count := 0;

-- Loop through each number in the array

FOR i IN 1..p\_numbers\_array.COUNT LOOP

-- Check if the number is even or odd

IF MOD(p\_numbers\_array(i), 2) = 0 THEN

p\_even\_count := p\_even\_count + 1;

ELSE

p\_odd\_count := p\_odd\_count + 1;

END IF;

END LOOP;

END;

/

DECLARE

v\_numbers sys.odcinumberlist := sys.odcinumberlist(1, 2, 3, 4, 5);

v\_even\_count NUMBER;

v\_odd\_count NUMBER;

BEGIN

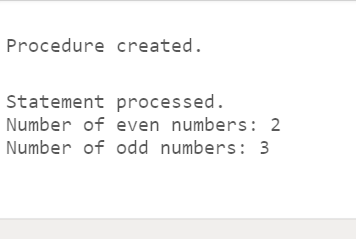
CountEvenOddNumbers(v\_numbers, v\_even\_count, v\_odd\_count);

DBMS\_OUTPUT.PUT\_LINE('Number of even numbers: ' || v\_even\_count);

DBMS\_OUTPUT.PUT\_LINE('Number of odd numbers: ' || v\_odd\_count);

END;

/



--PART 4—

CREATE OR REPLACE FUNCTION CompareAndDisplay (

p\_num1 IN NUMBER,

p\_num2 IN NUMBER,

p\_num3 IN NUMBER

) RETURN VARCHAR2

IS

v\_asc\_order VARCHAR2(100);

v\_desc\_order VARCHAR2(100);

BEGIN

-- Find ascending order

IF p\_num1 <= p\_num2 AND p\_num1 <= p\_num3 THEN

v\_asc\_order := TO\_CHAR(p\_num1) || ', ';

IF p\_num2 <= p\_num3 THEN

v\_asc\_order := v\_asc\_order || TO\_CHAR(p\_num2) || ', ' || TO\_CHAR(p\_num3);

ELSE

v\_asc\_order := v\_asc\_order || TO\_CHAR(p\_num3) || ', ' || TO\_CHAR(p\_num2);

END IF;

ELSIF p\_num2 <= p\_num1 AND p\_num2 <= p\_num3 THEN

v\_asc\_order := TO\_CHAR(p\_num2) || ', ';

IF p\_num1 <= p\_num3 THEN

v\_asc\_order := v\_asc\_order || TO\_CHAR(p\_num1) || ', ' || TO\_CHAR(p\_num3);

ELSE

v\_asc\_order := v\_asc\_order || TO\_CHAR(p\_num3) || ', ' || TO\_CHAR(p\_num1);

END IF;

ELSE

v\_asc\_order := TO\_CHAR(p\_num3) || ', ';

IF p\_num1 <= p\_num2 THEN

v\_asc\_order := v\_asc\_order || TO\_CHAR(p\_num1) || ', ' || TO\_CHAR(p\_num2);

ELSE

v\_asc\_order := v\_asc\_order || TO\_CHAR(p\_num2) || ', ' || TO\_CHAR(p\_num1);

END IF;

END IF;

-- Find descending order

v\_desc\_order := REPLACE(REPLACE(v\_asc\_order, ', ', ', '), ', ', ', ');

v\_desc\_order := REGEXP\_REPLACE(v\_desc\_order, '([^,]+),([^,]+),([^,]+)', '\3, \2, \1');

-- Return results

RETURN 'Ascending order: ' || v\_asc\_order || CHR(10) || 'Descending order: ' || v\_desc\_order;

END;

/

DECLARE

v\_num1 NUMBER := 10;

v\_num2 NUMBER := 5;

v\_num3 NUMBER := 8;

v\_result VARCHAR2(4000);

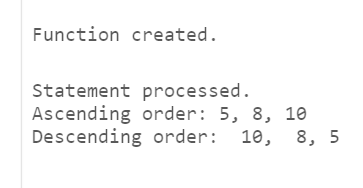
BEGIN

v\_result := CompareAndDisplay(v\_num1, v\_num2, v\_num3);

DBMS\_OUTPUT.PUT\_LINE(v\_result);

END;

/



--PART5—

CREATE OR REPLACE FUNCTION perform\_arithmetic(

num1 IN NUMBER,

num2 IN NUMBER,

operation IN VARCHAR2

) RETURN NUMBER IS

result NUMBER;

BEGIN

CASE operation

WHEN 'ADD' THEN

result := num1 + num2;

WHEN 'SUBTRACT' THEN

result := num1 - num2;

WHEN 'MULTIPLY' THEN

result := num1 \* num2;

WHEN 'DIVIDE' THEN

IF num2 != 0 THEN

result := num1 / num2;

ELSE

RAISE\_APPLICATION\_ERROR(-20001, 'Cannot divide by zero');

END IF;

ELSE

RAISE\_APPLICATION\_ERROR(-20001, 'Invalid operation');

END CASE;

RETURN result;

END;

DECLARE

v\_num1 NUMBER := 10;

v\_num2 NUMBER := 5;

v\_result VARCHAR2(4000);

BEGIN

v\_result := PerformArithmetic(v\_num1, v\_num2);

DBMS\_OUTPUT.PUT\_LINE(v\_result);

END;

/

