CSE 4508 – RDBMS Programming Lab

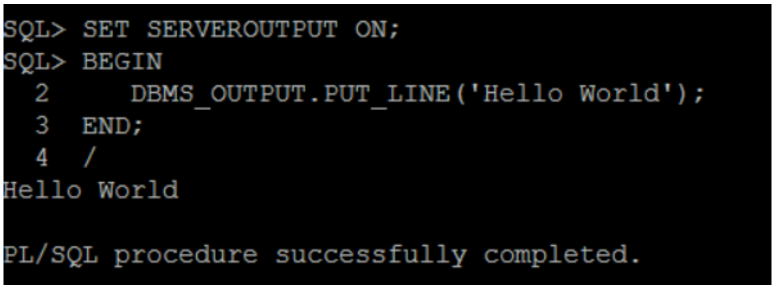
Lab 7

Prerequisites: Oracle 10g Express Edition, Any Text Editor

Instructor: Shahriar Ivan

PL/SQL is a block structured language where we can write code organized into blocks similar to Java/C/C++, although the coding style differs here due to not having any curly braces to define blocks, rather we define blocks using the BEGIN and END keywords. You can define PROCEDUREs (which execute some code without returning anything) and FUNCTIONs (which execute code and return some variable/record). And you can even define unnamed blocks in PL/SQL, otherwise known as **anonymous blocks** that can be called immediately upon defining using the / at the end.

Note: Remember to SET SERVEROUTPUT ON to see the results of the blocks.



```
SQL> SET SERVEROUTPUT ON;
SQL> BEGIN
  2     DBMS_OUTPUT.PUT_LINE('Hello World');
  3  END;
  4  /
Hello World

PL/SQL procedure successfully completed.
```

Refer to Lectures 6 & 7 from the Lectures pdf for pointers on Variables, Operators and Data Types.

Refer to Lectures 8 & 9 from the Lectures pdf for pointers on Control Structures and Loops.

A. Write a block of PL/SQL code which checks whether the current year is the starting year of a new decade (years such as 2000, 2010, 2020) and prints either “Yes” or “No”. After this, it should print the current decade (e.g. for 2000 to 2009, print ‘The 2000s’, for 2010 to 2019, print ‘The 2010s’).

B. Write a PL/SQL procedure(or function) called **prime_generator** which takes only one input: **s**. The function will keep generating prime numbers, starting from 2, until the sum of all the prime numbers generated so far is less than or equal to **s.** For example, if s = 20, the output will be: **2, 3, 5, 7** (Since 2+3+5+7 = 17. “11” is not included since that would make the sum greater than 20) Execute this function from a PL SQL block.

Task 1

```
-- Task 1
SET SERVEROUTPUT ON;


DECLARE
    CUR_YEAR VARCHAR(10) := TO_CHAR(SYSDATE, 'YYYY');
 -- cur_year INTEGER :=2092;
BEGIN
 --1st part
    IF (CUR_YEAR = 2000) OR (CUR_YEAR = 2010) OR (CUR_YEAR = 2020) OR
(CUR_YEAR = 2030) OR (CUR_YEAR = 2040) OR (CUR_YEAR = 2050) OR (CUR_YEAR =
2060) OR (CUR_YEAR = 2070) OR (CUR_YEAR = 2080) OR (CUR_YEAR = 2090) THEN
        DBMS_OUTPUT.PUT_LINE('Yes');
    ELSE
        DBMS_OUTPUT.PUT_LINE('No');
    END IF;
 -- 2nd part
    CASE
        WHEN CUR_YEAR < 2010 THEN
            DBMS_OUTPUT.PUT_LINE('The 2000s');
        WHEN CUR_YEAR < 2020 THEN
            DBMS_OUTPUT.PUT_LINE('The 2010s');
        WHEN CUR_YEAR < 2030 THEN

            DBMS_OUTPUT.PUT_LINE('The 2020s');

        WHEN CUR_YEAR < 2040 THEN

            DBMS_OUTPUT.PUT_LINE('The 2030s');

        WHEN CUR_YEAR < 2050 THEN

            DBMS_OUTPUT.PUT_LINE('The 2040s');

        WHEN CUR_YEAR < 2060 THEN

            DBMS_OUTPUT.PUT_LINE('The 2050s');

        WHEN CUR_YEAR < 2070 THEN

            DBMS_OUTPUT.PUT_LINE('The 2060s');

        WHEN CUR_YEAR < 2080 THEN

            DBMS_OUTPUT.PUT_LINE('The 2070s');

        WHEN CUR_YEAR < 2090 THEN

            DBMS_OUTPUT.PUT_LINE('The 2080s');

        ELSE

            DBMS_OUTPUT.PUT_LINE('The 2090s');

    END CASE;

END;

/
```

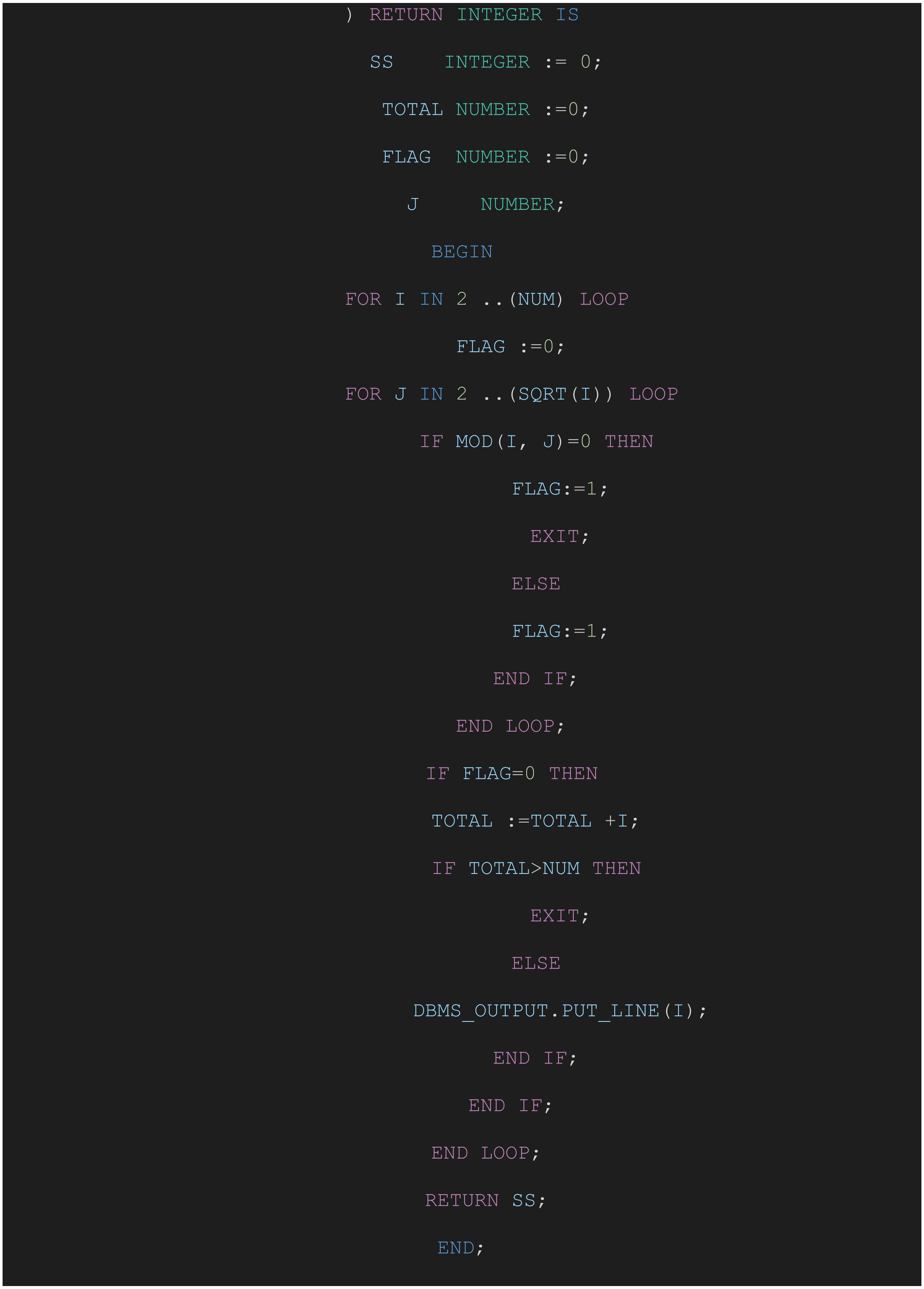
Task 2

```
-- Task 2

SET SERVEROUTPUT ON;


CREATE OR REPLACE FUNCTION GENERATE_PRIMES(

NUM INTEGER
```



```
) RETURN INTEGER IS
  SS    INTEGER := 0;
   TOTAL NUMBER :=0;
   FLAG  NUMBER :=0;
     J     NUMBER;
       BEGIN
FOR I IN 2 ..(NUM) LOOP
         FLAG :=0;
FOR J IN 2 ..(SQRT(I)) LOOP
      IF MOD(I, J)=0 THEN
             FLAG:=1;
               EXIT;
             ELSE
             FLAG:=1;
            END IF;
         END LOOP;
      IF FLAG=0 THEN
       TOTAL :=TOTAL +I;
       IF TOTAL>NUM THEN
               EXIT;
             ELSE
      DBMS_OUTPUT.PUT_LINE(I);
            END IF;
          END IF;
       END LOOP;
      RETURN SS;
       END;
```

```
/

DECLARE
S   INTEGER;
SS  INTEGER;
 BEGIN
  S:=  &S;
SS:=GENERATE_PRIMES(S);
 END;
  /
```