CEGEP VANIER COLLEGE
CENTRE FOR CONTINUING EDUCATION
Developing Applications using Oracle
420-987-VA

Teacher: Samir Chebbine Lab 1: PL/SQL Programming Jun 27, 2022

Lab 1: Introduction to PL/SQL Programming

Complete all these following programs as explained in my **Lab 1 YouTube Video 1, Lab 1 YouTube Video 2.** All *missing* coding statements are presented in these videos with explanation.

1. **Create and Execute all the following PL/SQL programs, Save all these PL/SQL programs in a file called *Lab1_OraclePrograms.sql*:** A PL/SQL block consists of three sections:
   - A **declaration** section
   - An **executable** section
   - An **exception-handling** section

a) **Program PL/SQL # 1**: Edit PL/SQL program to display the message "Hello World!"

The general syntax	Example of PL/SQL

DECLARE
Declaration of constants, variables, cursors, and exception
BEGIN
PL/SQL statements (if, while…) and SQL statements
EXCEPTION
Action for error conditions
END;
	Oracle SQL*Plus
	File Edit Search Options Help
	SQL> DECLARE
	2 w_hello VARCHAR2(30) := 'Hello World!';
	3 BEGIN
	4 DBMS_OUTPUT.PUT_LINE('My first message with PL/SQL ');
	5 DBMS_OUTPUT.PUT_LINE(w_hello);
	6 END;
	7 /
	My first message with PL/SQL
	Hello World!
	
	PL/SQL procedure successfully completed.

You need to execute the following command in order to display the output

```
SQL> SET SERVEROUTPUT ON SIZE 4000
```

2. **Data Types: Scalar** (Character, Number, Boolean, Date), **Composite** (records, tables, varrays), **Reference**: objects, **LOB:** Large Object

a) **Program PL/SQL # 2**: Edit PL/SQL program to display the current date using DATE type.

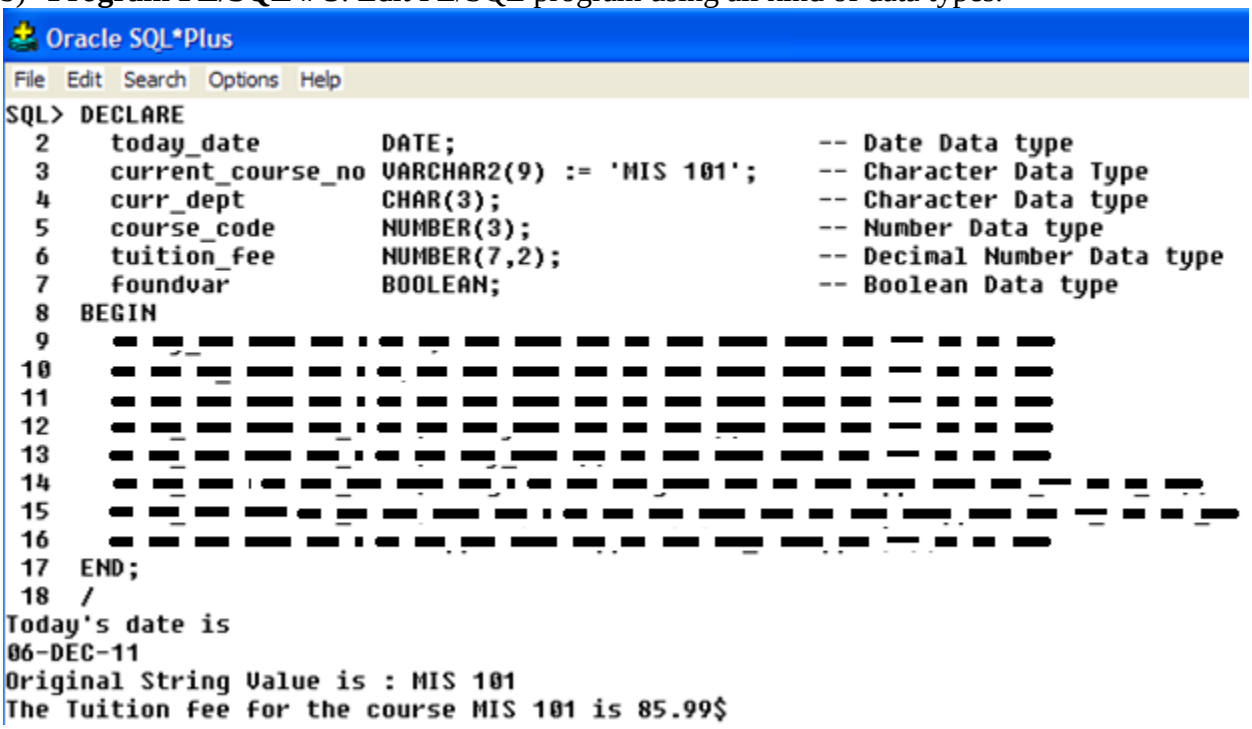
Scalar
- Character
- Number
- Date
- Boolean

Oracle SQL*Plus

File Edit Search Options Help

```
SQL> DECLARE
  2    today_date      DATE;
  3  BEGIN
  4    today_date := SYSDATE;
  5    DBMS_OUTPUT.PUT_LINE('Today''s date is ');
  6    DBMS_OUTPUT.PUT_LINE(today_date);
  7  END;
  8  /
Today's date is
05-DEC-11

PL/SQL procedure successfully completed.
```

b) **Program PL/SQL # 3**: Edit PL/SQL program using all kind of data types.



```
Oracle SQL*Plus
File  Edit  Search  Options  Help
SQL> DECLARE
  2    today_date        DATE;                          -- Date Data type
  3    current_course_no VARCHAR2(9) := 'MIS 101';      -- Character Data Type
  4    curr_dept         CHAR(3);                       -- Character Data type
  5    course_code       NUMBER(3);                     -- Number Data type
  6    tuition_fee       NUMBER(7,2);                   -- Decimal Number Data type
  7    foundvar          BOOLEAN;                       -- Boolean Data type
  8  BEGIN
  9    [redacted]
 10    [redacted]
 11    [redacted]
 12    [redacted]
 13    [redacted]
 14    [redacted]
 15    [redacted]
 16    [redacted]
 17  END;
 18  /
Today's date is
06-DEC-11
Original String Value is : MIS 101
The Tuition fee for the course MIS 101 is 85.99$
```

c) **Program PL/SQL # 4**: Edit PL/SQL program using *character functions* (of Block 1) to display the character code and numeric code of the variable *current_course_no* as shown hereafter.

```
 24  /
Today's date With a given Format is TUE, DECEMBER  06, 2011 07:54:56 P.M.
Original String Value is : MIS 101
The Tuition fee for the course MIS 101 is 85.99$
The Character code of the course captured in curr_dept variable is : MIS
The Numeric code of the course captured in course_code variable is : 101
The Character code of the course in lower case is : mis
Number of Characters in the course code is    : 7
The position of blank space in the course code is    : 4

PL/SQL procedure successfully completed.
```

3. Using your own wording, answer the following questions briefly:

1. What is the assignment operator used in PL/SQL to assign value to a given variable?
2. What is the comparison operator used in PL/SQL?
3. What is the concatenation operator used in PL/SQL to display two strings variables?
4. What is the exponential symbol used in PL/SQL program?
5. Why use two single quotations ' ' in line 12 of Program PL/SQL # 3
6. Is it possible to write the following PL/SQL statement within the DECLARE block DBMS_OUTPUT.PUT_LINE('My first message with PL/SQL ');
7. Is PL/SQL program a front-end program executed on client side or back-end program executed on the server side?
8. What is the name concept used in PL/SQL to describe DBMS_OUTPUT
9. What is the name concept used in PL/SQL to describe PUT_LINE
10. True or false and why:

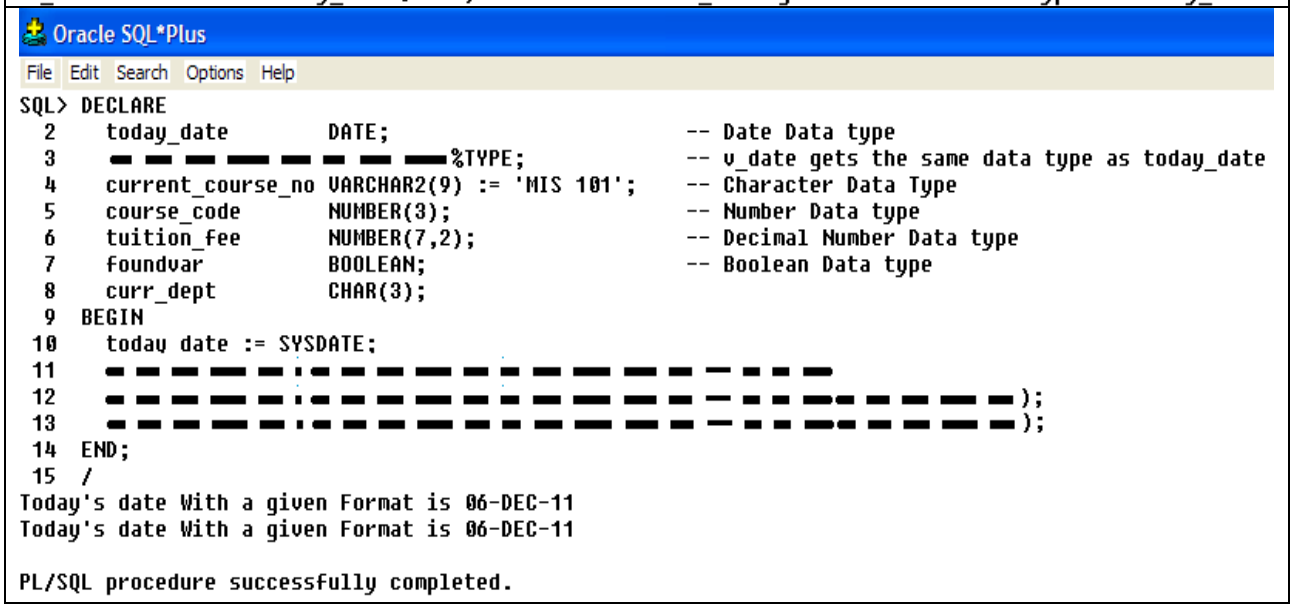
    a. DECLARE block is optional in PL/SQL program
    b. You cannot declare PL/SQL variable within BEGIN block.

c. Semi colon is mandatory at the end of every PL/SQL statement.
d. Line indentation makes PL/SQL program more readable.
e. PL/SQL program is compiled using a PL/SQL compiler.
f. You cannot assign PL/SQL variable within DECLARE block.
g. To declare PL/SQL variable, you need to specify date type following the name of variable such as **product price NUMBER (6,2);**
h. Multi-line comment in PL/SQL is double dash --
i. Every PL/SQL program must be terminated by EXCEPTION block.

4. Data Types: Anchored Declaration

PL/SQL uses %TYPE attribute to anchor a variable's data type. A variable gets the same data type as an existing one.

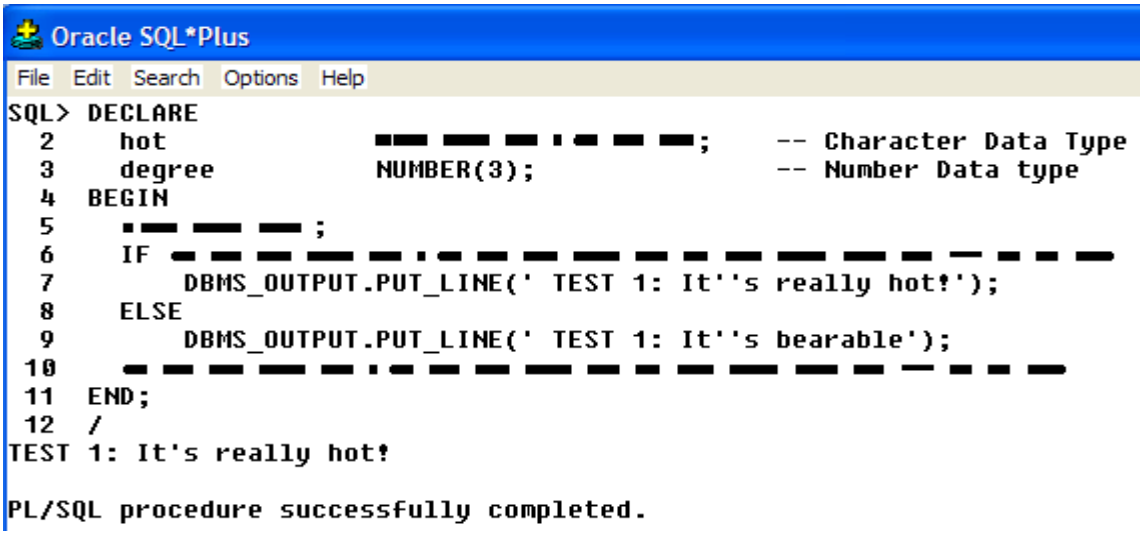
```
today_date        DATE;                          -- Date Data type
v_date            today_date%TYPE;               -- v_date gets the same data type as today_date
```



5. Control Decisions:

a) Program PL/SQL # 7: Edit PL/SQL program using Selection control structure.

- IF… THEN… END IF
- IF… THEN… ELSE… END IF
- IF… THEN… ELSIF… END IF



b) Program PL/SQL # 8: Edit PL/SQL program using Looping Structure.

1. FOR… END LOOP
2. WHILE… END LOOP

Oracle SQL*Plus

File Edit Search Options Help

```
SQL> CREATE TABLE count_table
  2        (counter NUMBER (2));

Table created.

SQL>
SQL> DECLARE
  2     loop_count BINARY_INTEGER := 1;
  3  BEGIN
  4     FOR [redacted]
  5     LOOP
  6        [redacted]);
  7     END LOOP;
  8     [redacted]
  9  END;
 10  /

PL/SQL procedure successfully completed.
```

Oracle SQL*Plus

File Edit Search Options Help

```
SQL> CREATE TABLE count_table
  2        (counter NUMBER (2));

Table created.

SQL>
SQL> DECLARE
  2     loop_count BINARY_INTEGER := 1;
  3  BEGIN
  4     WHILE [redacted]
  5     LOOP
  6        [redacted] count);
  7        [redacted]
  8     END LOOP;
  9     [redacted]
 10  END;
 11  /

PL/SQL procedure successfully completed.
```

```
SQL> select * from count_table;

   COUNTER
----------
         1
         2
         3
         4
         5
```

3. c) LOOP…EXIT WHEN… END LOOP

```
DECLARE
   loop_count BINARY_INTEGER := 1;
BEGIN
   LOOP
      INSERT INTO [redacted];
      [redacted]
      [redacted]
   END LOOP;
   COMMIT;
END;
```

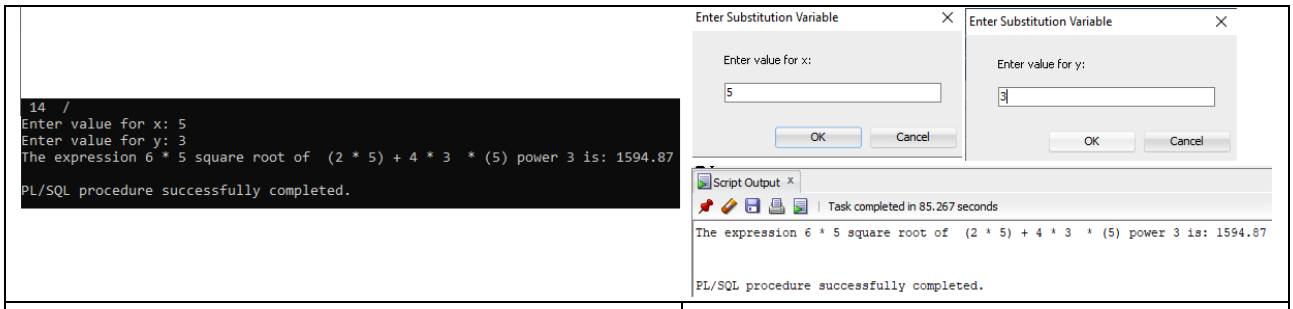
```
SQL> select * from count_table;

   COUNTER
----------
         1
         2
         3
         4
         5
```

IMPORTANT:

- PL/SQL supports only DML and DCL statements such as SELECT, INSERT, UPDATE, DELETE, COMMIT, ROLLBACK.
- PL/SQL does not support DDL statements such as CREATE TABLE, ALTER TABLE, DROP TABLE.

6. **Complete this part and save the following PL/SQL programs and their outputs in the same file called *Lab1_Practice_OraclePrograms.sql* (or in another sql file if you like):**

A. **Program PL/SQL # 4A**: Edit PL/SQL program using Substitution variable (&) to calculate the output of the following expression ($6x\sqrt{2x} + 4yx^3$).
You need to use oracle function **SQRT** as shown in complement presentation 1.
Reminder: use **Substitution (&)** to get an interactive user input such as:
x=&x; -- you will get end-user to input value for x such as: Enter value for x



Oracle Express XE	Oracle SQL Developer

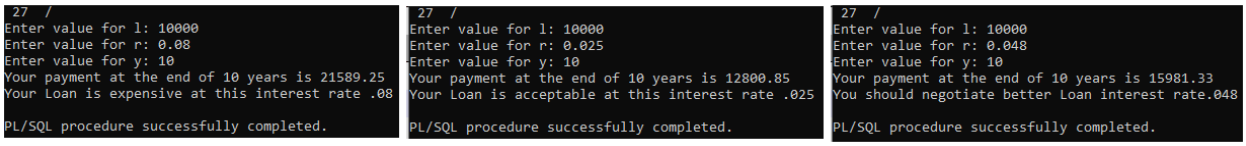
B. **Program PL/SQL # 4B**: Edit PL/SQL program using Substitution variable (&) to calculate the payment for a loan based on constant payments and a constant interest rate (the equivalent of PMT function in Excel)

The payment function for the sake of simplicity is calculated as such: PMT= $L(1+r)^y$ where L is the Loan, r is the Interest rate per year (2% is 0.02), y is the number of Years.

If (PMT/L) is higher than 2: display the following message "Your Loan is expensive at this interest rate"

If (PMT/L) is less than 1.4: display the following message "Your Loan is acceptable at this interest rate"
otherwise: display the following message " You should negotiate better Loan interest rate"

You need to use **control structure IF** and oracle function **ROUND** to get the following output:



Scenario 1: r=0.08(8%)	Scenario 2: r=0.025(2.5%)	Scenario 3: r=0.048(4.8%)

C. **Program PL/SQL # 4C**: Edit PL/SQL program using ***Oracle character functions*** to display the hospital name, virus name and number of hospitalisations of a given entered hospital information as shown hereafter.

Requirements: The **entered hospital info** of character data type includes the following rules:

- Hospital name as three sub strings separated by one space such as: Jewish General Hospital or Royal Victoria Hospital.

- Number of hospitalisations separated by one space from Hospital name such as **406 Jewish General Hospital**

```
hospitalInfo        -- Character Data Type
                    -- Number of hospitalisations Number (3)
                    -- Hospital Name Type Character(30)
```

```
 19  /
Enter value for hospitalinfo: 406 Jewish General Hospital
Today's date With a given Format is MON, OCTOBER   25, 2021 05:06:54 P.M.
Original hospitalInfo is : 406 Jewish General Hospital
Number of hospitalisations  is: 406
Hospital Name is: Jewish General Hospital

PL/SQL procedure successfully completed.

SQL>
```

D. **Program PL/SQL # 4D**: Taking into account the previous requirements, edit PL/SQL program using ***Oracle character functions*** (SUBSTR, and INSTR to locate the occurrence of multiple blank spaces) in order to display the number of hospitalisations, cause of hospitalisation, wave number, and hospital name of a given entered hospital info such as:

406 COVID19Delta 4 Jewish General Hospital

which stands for

406 as number of hospitalisations
COVID19Delta as cause of hospitalisation (here virus COVID19 Delta variant)
4 as wave number 4
Jewish General Hospital as hospital name

```
hospitalInfo        -- Character Data Type
                    -- Number of hospitalisations Number (3)
                    -- Hospital Name Type Character(30)
                    -- Virus type Character(15)
                    -- Wave number NUMBER(2)
```

```
 26  /
Enter value for hospitalinfo: 406 COVID19Delta 04 Jewish General Hospital
Today's date With a given Format is MON, OCTOBER   25, 2021 06:01:18 P.M.
Original hospitalInfo is : 406 COVID19Delta 04 Jewish General Hospital
Number of hospitalisations  is: 406
Cause of hospitalisation is: COVID19Delta
Wave number is: 4
Hospital Name is: Jewish General Hospital

PL/SQL procedure successfully completed.

SQL>
```