Oracle MOOC: PL/SQL Fundamentals

Week 1

Homework for Lesson 1: Introducing Oracle PL/SQL

Homework is your chance to put what you've learned in this lesson into practice. This homework is not "graded" and you are encouraged to write additional code beyond what is asked.

Note:

- Ensure you completed the setup instructions provided on the course page, before attempting the homework.
- The solutions to the homework are NOT provided. We encourage you to try it out and discuss in the course forum for further learning.
- The homework is NOT mandatory to get the course completion award.
- Post your questions, comments, or suggestions (if any) in the course forum @ https://community.oracle.com/community/technology_network_community/moocs/plsql-fundamentals

Watch out for:

- Reference video that discussed the corresponding concept in this MOOC.
- Hints that can help you solve the assignment.

Assignment 1: Which of the following PL/SQL blocks executes successfully?

a.
```
BEGIN
   END;
```

b.
```
DECLARE
   v_amount INTEGER(10);
   END;
```

c.
```
DECLARE
   BEGIN
   END;
```

Oracle MOOC: PL/SQL Fundamentals

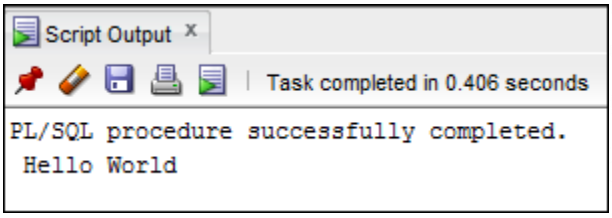
d.
```
DECLARE
   v_amount INTEGER(10);
   BEGIN
   DBMS_OUTPUT.PUT_LINE(v_amount);
   END;
```

See 1-4: Understanding PL/SQL Architecture and Block Structure for reference.

Assignment 2: Create and execute a simple anonymous block that prints “Hello World” and save this script as `soln_01_02.sql`.

See 1-4: Understanding PL/SQL Architecture and Block Structure for reference.

Expected output:



Assignment 3: Which of the below identifiers are invalid, and why?

a) `today`

b) `last_name`

c) `today’s_date`

d) `Number_of_days_in_February_this_year`

e) `Isleap$year`

f) `#number`

g) `NUMBER#`

h) `number1to7`

Oracle MOOC: PL/SQL Fundamentals

See 1-5: Using Variables in PL/SQL Blocks for reference.

Assignment 4: Identify which of the following declarations and initializations are invalid, and why?

```
a) number_of_copies    PLS_INTEGER;
b) PRINTER_NAME        constant VARCHAR2(10);
c) deliver_to          VARCHAR2(10):=Johnson;
d) by_when             DATE:= CURRENT_DATE+1;
```

See 1-5: Using Variables in PL/SQL Blocks for reference.

Assignment 5: Examine this anonymous block. Which statement is correct?

```
DECLARE
  v_fname VARCHAR2(20);
  v_lname VARCHAR2(15) DEFAULT 'fernandez';
BEGIN
  DBMS_OUTPUT.PUT_LINE(v_fname ||' ' ||v_lname);
END;
```

a. The block executes successfully and prints “fernandez.”
b. The block produces an error because the `fname` variable is used without initializing.
c. The block executes successfully and prints “null fernandez.”
d. The block produces an error because you cannot use the `DEFAULT` keyword to initialize a variable of type `VARCHAR2`.
e. The block produces an error because the `v_fname` variable is not declared.

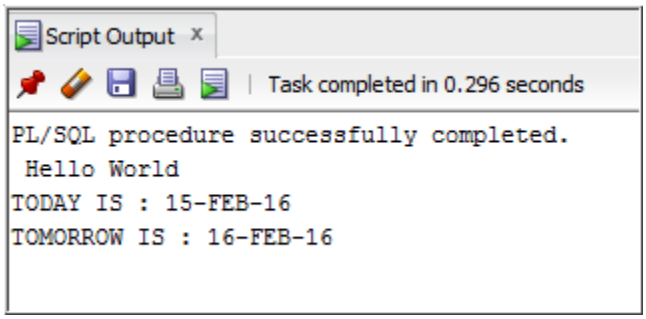
Oracle MOOC: PL/SQL Fundamentals

See 1-5: Using Variables in PL/SQL Blocks for reference.

Assignment 6: Modify `soln_01_02.sql` so that it prints "Hello World" followed by today's date and tomorrow's date. Save this script as `soln_01_06.sql`.

- Declare a variable to hold the value of today's date. You can initialize it to `SYSDATE`.
- Declare a variable to hold the value of tomorrow's date. You can use `%TYPE` attribute in its declaration.
- Increment today's date by 1 and assign it to tomorrow's date, in the executable section.
- Print the value of today's date and tomorrow's date after printing "Hello World".

Sample output:



See 1-5: Using Variables in PL/SQL Blocks for reference.

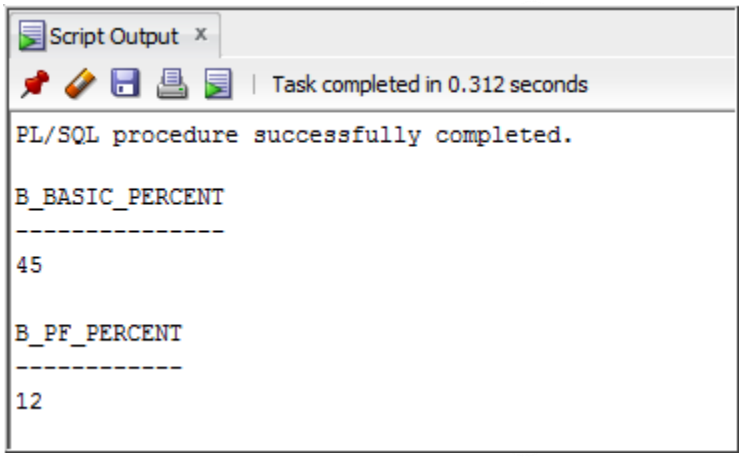
Assignment 7: Declare and assign values to bind variables. Print their values in the output.

- Create two bind variables, `b_basic_percent` and `b_pf_percent` and assign the values 45 and 12 respectively.
- Display the value of the bind variables by using the `PRINT` command.

Oracle MOOC: PL/SQL Fundamentals

- Execute and save your script as `soln_01_07.sql`

Sample output:



See 1-5: Using Variables in PL/SQL Blocks for reference.

Assignment 8: Evaluate this PL/SQL block:

```
DECLARE
    v_weight    NUMBER(3) := 600; -- position 1
    v_message   VARCHAR2(255) := 'Product 10012';
   BEGIN
     DECLARE
      v_weight   NUMBER(3) := 1; -- position 2
      v_message    VARCHAR2(255) := 'Product
   11001';
      v_new_locn  VARCHAR2(50) := 'Europe';
     BEGIN
      v_weight := v_weight + 1;
      v_new_locn := 'Western ' || v_new_locn;

     END;
    v_weight := v_weight + 1;
    v_message := v_message || ' is in stock';
```

Oracle MOOC: PL/SQL Fundamentals

```
    v_new_locn := 'Western ' || v_new_locn;

  END;
/
```

According to the rules of scoping, determine the value and data type of:

a. `v_weight` at position 1 is:
b. `v_new_locn` at position 1 is:
c. `v_weight` at position 2 is:
d. `v_message` at position 2 is:
e. `v_new_locn` at position 2 is:

See 1-7: Nesting Blocks in PL/SQL Programs for reference.

Assignment 9: Evaluate this PL/SQL block:

Oracle MOOC: PL/SQL Fundamentals

```
DECLARE
   v_customer     VARCHAR2(50) := 'Womansport';
   v_credit_rating    VARCHAR2(50) :=
'EXCELLENT';
BEGIN
    DECLARE
       v_customer  NUMBER(7) := 201;
       v_name VARCHAR2(25) := 'Unsorts';
    BEGIN
       v_credit_rating :='GOOD';
       …
    END;
  …
END;
/
```

Determine the value and data type of:

a. `v_customer` in the nested block
b. `v_name` in the nested block
c. `v_credit_rating` in the nested block
d. `v_customer` in the main block
e. `v_name` in the main block
   `v_credit_rating` in the main block

See 1-7: Nesting Blocks in PL/SQL Programs for reference.

Congratulations! You successfully practiced the concepts discussed in week 1.