Database Programming with PL/SQL
2-4: Using Scalar Data Types
Practice Activities

Vocabulary

Identify the vocabulary word for each definition below:

BOOLEAN	A datatype that stores one of the three possible values used for logical calculations: TRUE, FALSE, or NULL.
%TYPE	Attribute used to declare a variable according to another previously declared variable or database column.

Try It / Solve It

1. Declarations:

A. Which of the following variable declarations are valid?

	Declaration	Valid or Invalid
a	number_of_students PLS_INTEGER;	Valid
b	STUDENT_NAME VARCHAR2(10) = Johnson;	Invalid, must use ' ', and :=
c	stu_per_class CONSTANT NUMBER;	Invalid, must initialize
d	tomorrow DATE := SYSDATE+1;	Valid

B. For the invalid declarations above, describe why they are invalid.

Write an anonymous block in which you declare and print (on the screen) each of the variables in 1A above, correcting the invalid declarations and adding information as needed.

```
DECLARE
  number_of_students PLS_INTEGER;
  STUDENT_NAME VARCHAR2(10) := 'Johnson';
  stu_per_class CONSTANT Number := 4;
  tomorrow DATE := SYSDATE+1;
BEGIN
  DBMS_OUTPUT.PUT_LINE(number_of_students);
  DBMS_OUTPUT.PUT_LINE(STUDENT_NAME);
  DBMS_OUTPUT.PUT_LINE(stu_per_class);
  DBMS_OUTPUT.PUT_LINE(tomorrow);
END;
```

2. Evaluate the variables in the following code. Answer the following questions about each variable. Is it named well? Why or why not? If it is not named well, what would be a better name and why?

No, because they have the same name as the table columns. v_country_name and v_mediam_age would be better

```
DECLARE
    country_name   VARCHAR2(50);
    median_age     NUMBER(6, 2);
BEGIN
     SELECT country_name, median_age INTO country_name, median_age
        FROM countries
        WHERE country_name = 'Japan';
     DBMS_OUTPUT.PUT_LINE('The median age in '|| country_name || ' is '
        || median_age || '.');
END;
```

3. Change the declarations in #2 above so they use the %TYPE attribute.

v_country_name countries.country_name%TYPE
v_mediam_age countries.median_age%TYPE

4. In your own words, describe why using the %TYPE attribute is better than hard-coding data types. Can you explain how you could run into problems in the future by hard-coding the data types of the country_name and median_age variables in question 2?

%TYPE is more resistant against changes in the database. For instance, in the example from question 2, if the precision of median_age changes, than the declaration of v_mediam_age must also change.

5. Create the following anonymous block:

```
BEGIN
  DBMS_OUTPUT.PUT_LINE('Hello World');
END;
```

A. Add a declarative section to this PL/SQL block. In the declarative section, declare the following variables:

- A variable named TODAY of datatype DATE. Initialize TODAY with SYSDATE.
- A variable named TOMORROW with the same datatype as TODAY. Use the %TYPE attribute to declare this variable.

B. In the executable section, initialize the TOMORROW variable with an expression that calculates tomorrow's date (add 1 to the value in TODAY). Print the value of TODAY and TOMORROW after printing 'Hello World'.

```
1  DECLARE
2    today DATE := SYSDATE;
3    tomorrow today%TYPE := today + 1;
4  BEGIN
5    DBMS_OUTPUT.PUT_LINE(today);
6    DBMS_OUTPUT.PUT_LINE(tomorrow);
7  END;
```