# %TYPE Attribute

The %TYPE attribute lets you declare a constant, variable, field, or parameter to be of the same data type a previously declared variable, field, record, nested table, or database column. If the referenced item changes, your declaration is automatically updated.

Example:

lname employees.last\_name%TYPE;

fname employees.first\_name%TYPE;

# %ROWTYPE Attribute

CREATE TABLE employees\_temp (

empid NUMBER(6) PRIMARY KEY,

deptid NUMBER(6),

deptname VARCHAR2(30)

);

-- *Declaring and Assigning Values to Variables*

emprec employees\_temp%ROWTYPE;

……..

emprec.empid := NULL;

emprec.deptid := 50;

# Record Type

TYPE timerec IS RECORD (

hours SMALLINT,

minutes SMALLINT

);

# PL/SQL Block

DECLARE -- Declarative part (optional)

Declarations of local types, variables, & subprograms

BEGIN --Executable part (required)

Statements (which can use items declared in declarative part)

EXCEPTION -- Exception-handling part (optional)

Exception handlers for exceptions raised in executable part

END;

# *Declaring Variables in PL/SQL*

DECLARE

part\_number NUMBER(6); -- SQL data type

part\_name VARCHAR2(20); -- SQL data type

in\_stock BOOLEAN; -- PL/SQL-only data type

part\_price NUMBER(6,2); -- SQL data type

part\_description VARCHAR2(50); -- SQL data type

BEGIN

NULL;

END;

***Assigning Values to Variables With the Assignment Operator***

DECLARE

wages NUMBER;

country VARCHAR2(128);

emp\_rec1 employees%ROWTYPE;

emp\_rec2 employees%ROWTYPE;

BEGIN

wages := (hours\_worked \* hourly\_salary) + bonus;

country := UPPER('Canada');

emp\_rec1.first\_name := 'Antonio';

emp\_rec1.last\_name := 'Ortiz';

emp\_rec2 := emp\_rec1;

END;

***Assigning Values to Variables by SELECTing INTO***

DECLARE

bonus NUMBER(8,2);

emp\_id NUMBER(6):= 100;

BEGIN

SELECT salary \* 0.10 INTO bonus

FROM employees

WHERE employee\_id = emp\_id;

END;

**IF-THEN-ELSIF Statement**

DECLARE

sales NUMBER(8,2) := 20000;

bonus NUMBER(6,2);

emp\_id NUMBER(6) := 120;

BEGIN

IF sales > 50000 THEN

bonus := 1500;

ELSIF sales > 35000 THEN

bonus := 500;

ELSE

bonus := 100;

END IF;

UPDATE employees SET salary = salary + bonus

WHERE employee\_id = emp\_id;

END;