

Package

PL/SQL Package là gì?

- Là một tập hợp các PL/SQL object được nhóm lại tạo thành một gói.
- Chúng có thể chứa:
 - Procedures, functions
 - Types, cursors, variables, constants
 - Exceptions

Cấu trúc của Package

- Có 2 phần:
 - Phần mô tả (specification):
 - Khai báo các **public** item (procedure, function, variable...).
 - Các chương trình khác có thể truy xuất đến chúng từ ngoài package.
 - Phần thân (body)
 - Hiện thực các public item. Đồng thời nó cũng có thể chứa các **private** item.
 - Các đơn vị **private** này chỉ có thể truy xuất trong tầm vực của package.

Ví dụ (package specification)

```
CREATE PACKAGE cust_sal AS

    PROCEDURE find_sal(c_id customers.id%type);

END cust_sal;
```

Ví dụ (Package Body)

```
CREATE OR REPLACE PACKAGE BODY cust_sal AS

    PROCEDURE find_sal(c_id customers.id%TYPE) IS
        c_sal customers.salary%TYPE;
    BEGIN
        SELECT salary INTO c_sal
        FROM customers
        WHERE id = c_id;
        dbms_output.put_line('Salary: ' || c_sal);
    END find_sal;

END cust_sal;
```

Using the Package Elements

- package elements (variables, procedures or functions) được truy cập theo cú pháp sau:

package_name.element_name;
- Ví dụ: gọi hàm find_sal trong package cust_sal

```
DECLARE
    code customers.id%type := 101;
BEGIN
    cust_sal.find_sal(code);
END;
```

Ví dụ (package specification)

```
CREATE OR REPLACE PACKAGE package_name
IS
  PROCEDURE sal_raise (amount NUMBER, depNo NUMBER);
  -----
  FUNCTION salDep_sum (depNo IN NUMBER)
                      RETURN NUMBER;
  -----
END package_name;
```

Ví dụ (Package Body)

```
CREATE OR REPLACE PACKAGE BODY package_name
IS
  PROCEDURE sal_raise (amount NUMBER, depNo NUMBER)
  IS
    BEGIN
      update staff set salary=salary+amount where deptId=depNo;
    END sal_raise;
  -----
  FUNCTION salDep_sum (depNo IN NUMBER)
  RETURN NUMBER
  IS
    sal      number;
  BEGIN
    select sum(salary) into sal from staff where deptId= depNo;
    RETURN sal ;
  END salDep_sum;
END package_name;
```

Xóa Package

- Để xóa toàn bộ package bao gồm specification và body sử dụng cú pháp:
 - **DROP PACKAGE *package_name***
- Chỉ xóa package body sử dụng cú pháp:
 - **DROP PACKAGE BODY *package_name***;

Ví dụ

- Tạo một package quản lý nhân viên bao gồm các thao tác: thêm nhân viên, xóa nhân viên, tăng lương cho nhân viên

```
CREATE OR REPLACE PACKAGE Employee_management AS -- package specification

  FUNCTION Hire_emp
    (Name VARCHAR2, Job VARCHAR2, Mgr NUMBER,
     Hiredate DATE, Sal NUMBER, Comm NUMBER, Deptno NUMBER)
    RETURN NUMBER;

  PROCEDURE fire_emp (emp_id IN NUMBER);

  PROCEDURE Sal_raise (emp_id IN NUMBER, Sal_incr IN NUMBER);
END Employee_management;
```

CREATE PACKAGE BODY **Employee_management** AS

```
FUNCTION Hire_emp (Name VARCHAR2, Job VARCHAR2, Mgr NUMBER, Hiredate DATE,
  New_empno  NUMBER(10);
BEGIN
  SELECT Emp_sequence.NEXTVAL INTO New_empno FROM dual;
  INSERT INTO Emp_tab VALUES (New_empno, Name, Job, Mgr, Hiredate, Sal, Comm, Deptno);
  RETURN (New_empno);
END Hire_emp;
```

```
PROCEDURE fire_emp (emp_id IN NUMBER) AS
BEGIN
  DELETE FROM Emp_tab WHERE Empno = Emp_id;
  IF SQL%NOTFOUND THEN
    Raise_application_error(-20011, 'Invalid Employee Number: ' || TO_CHAR(Emp_id));
  END IF;
END fire_emp;
```

```
PROCEDURE Sal_raise (Emp_id IN NUMBER, Sal_incr IN NUMBER) AS
BEGIN
  UPDATE Emp_tab SET Sal = Sal + Sal_incr WHERE Empno = Emp_id;
  IF SQL%NOTFOUND THEN
    Raise_application_error(-20011, 'Invalid Employee Number: ' || TO_CHAR(Emp_id));
  END IF;
END Sal_raise;
```

END **Employee_management**;

Calling a Subprogram in a Package

```
DECLARE
  Assigned_empno  NUMBER;
BEGIN
  Assigned_empno := Hire_emp('JSMITH', 'President', 1032, SYSDATE, 5000, NULL, 10);
  DBMS_OUTPUT.PUT_LINE (Assigned_empno);
END;
```