

Exemplo de criação de pacote

```
CREATE SEQUENCE emp_sequence
  START WITH 8000
  INCREMENT BY 10;

CREATE PACKAGE employee_management AS
  FUNCTION hire_emp (name VARCHAR2, job VARCHAR2,
    mgr NUMBER, hiredate DATE, sal NUMBER, comm NUMBER,
    deptno NUMBER) RETURN NUMBER;
  PROCEDURE fire_emp (emp_id NUMBER);
  PROCEDURE sal_raise (emp_id NUMBER, sal_incr NUMBER);
END employee_management;

CREATE PACKAGE BODY employee_management AS
  FUNCTION hire_emp (name VARCHAR2, job VARCHAR2,
    mgr NUMBER, hiredate DATE, sal NUMBER, comm NUMBER,
    deptno NUMBER) RETURN NUMBER IS

    new_empno    NUMBER(10);

  BEGIN
    SELECT emp_sequence.NEXTVAL INTO new_empno FROM dual;
    INSERT INTO emp 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 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
      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;
```

O pacote `EMPLOYEE_MANAGEMENT` pode inserir um novo funcionário através da função `HIRE_EMP`, remover um empregado através do procedimento `FIRE_EMP` e aumentar o salário do funcionário através do procedimento `SAL_RAISE`. Use-o como modelo para desenvolver o pacote `GERENCIA_FUNCIONARIO`.

1. Execute os comandos abaixo para criar e popular a tabela `FUNC`.

```
create table FUNC(  
  codigo      number(4,0),  
  nome        varchar2(30),  
  cargo       varchar2(20),  
  sal         number(9,2),  
  constraint pk_emp primary key (codigo),  
);
```

```
INSERT INTO func VALUES (01, 'VANESSA', 'ANALISTA', 5000);  
INSERT INTO func VALUES (11, 'EMMANUELE', 'DEVELOPER', 4800);  
INSERT INTO func VALUES (21, 'NUBIA', 'DBA', 6800);  
INSERT INTO func VALUES (31, 'ANA', 'GERENTE', 7200);  
INSERT INTO func VALUES (41, 'DAFNE', 'ESTAGIARIO', 1200);
```

2. Crie uma especificação e um corpo de pacote denominados `GERENCIA_FUNCIONARIO`. Crie a especificação do pacote incluindo os cabeçalhos das funções e dos procedures como estruturas públicas. Crie o corpo do pacote com as implementações de cada um dos subprogramas. O pacote deverá conter:

- a. Uma função denominada `PEGA_NOME` que receba o código do funcionário e devolva o nome do funcionário.
- b. Uma procedure denominada `PROX_FUNC` que obtém o maior código de funcionário e retorna esse número acrescido do valor 10.
- c. Uma procedure denominada `INC_FUNC` que inclui um novo funcionário na tabela. O código do novo funcionário deve ser obtido através da procedure `PROX_FUNC`. O nome, cargo e salário devem ser informados pelo usuário. Caso o nome não for informado, assumir o valor 'SEM NOME' para o campo. Caso o cargo não seja informado, assumir o valor 'SEM CARGO' para o campo. Caso o salário não seja informado, assumir o valor 980 para o campo.
- d. Uma procedure denominada `LISTA_FUNC` que exiba o código, nome, cargo e salário de todos os funcionários.