ADMINISTRACIÓN DE BASE DE DATOS.

Tema Nº10:Programación PL/SQL VI.

Indicador de logro Nº10:Construye programas, a través de Paquetes Almacenados.

**TEMA 01 Teoría de los**

Imagen que contiene Icono

Descripción generada automáticamente

**TEMA Nº10:**

Programación PL/SQL VI.

**Subtema 10.1:**

Implementación de paquetes almacenados debidamente organizados, incorporando procedimientos almacenados y funciones por usuarios.

**INTRODUCCIÓN A LOS PAQUETES ALMACENADOS.**

Un paquete es, en esencia, una sección declarativa NOMINADA. Cualquier cosa que pueda incluirse en la sección declarativa de un bloque, puede incluirse también en un paquete. Esto abarca procedimientos, funciones, cursores, tipos y variables. Una ventaja de incluir estos objetos en un paquete es la posibilidad de referenciarlos desde otros bloques PL/SQL, con lo que los paquetes permiten disponer de variables globales en PL/SQL.

Es una estructura PL/SQL que permite almacenar una serie de objetos relacionados. Tiene dos partes bien diferenciadas:

* La especificación
* El cuerpo

**ESPECIFICACION DEL PAQUETE:**

La especificación (o cabecera) del paquete contiene información acerca del contenido del paquete. Sin embargo, no contiene el código de los procedimientos.

Sintaxis:

IS

AS

CREATE

OR REPLACE

PACKAGE

Nombre\_Paquete

Especificación variable | Definición Tipo

Procedimiento | función

**CUERPO DEL PAQUETE:**

IS

AS

CREATE

OR REPLACE

PACKAGE BODY

Nombre\_Paquete

Cuerpo Procedimiento | función

**AMBITO:**

* Cualquier objeto (procedimiento o función) declarado en la cabecera de un package es visible fuera del paquete.
* El llamado se puede hacer desde cualquier bloque PL/SQL:

**Nombre\_Package . Nombre Procedimiento / Función**

**EJEMPLO 1: SIN USAR PARAMETOS.**

Implementar un PAQUETE **MI\_PAQUETE1** el cual debe contener lo siguiente:

* procedimiento1: Mostrara el siguiente mensaje: «**PRIMER PROCEDIMIENTO**».
* procedimiento2: Mostrara el siguiente mensaje: «**SEGUNDO PROCEDIMIENTO**».

**SOLUCION PASO A PASO:**

**PASO 1:** REALIZAR LA ESPECIFICACION.

**CREATE OR REPLACE PACKAGE** MI\_PAQUETE1

**IS**

**PROCEDURE P1;**

**PROCEDURE P2;**

**END;**

**PASO 2:** CUERPO DEL PAQUETE.

**CREATE OR REPLACE PACKAGE BODY** MI\_PAQUETE1

**IS**

**PROCEDURE P1**

IS

BEGIN

DBMS\_OUTPUT.PUT\_LINE ('PRIMER PROCEDIMIENTO');

END P1;

**PROCEDURE P2**

IS

BEGIN

DBMS\_OUTPUT.PUT\_LINE ('SEGUNDO PROCEDIMIENTO');

END P2;

**END PACK1;**

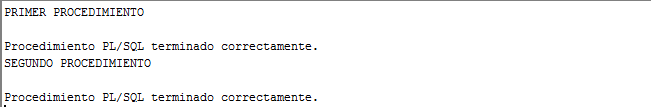
**PASO 3:** EJCUCION.

**SET SERVEROUTPUT ON;**

**EXEC** MI\_PAQUETE1.p1

**EXEC** MI\_PAQUETE1.p2

**RESULTADO:**



**EJEMPLO 2: CON USO DE PARAMETROS**

Implementar un PAQUETE **MI\_PAQUETE2** el cual debe contener lo siguiente:

* Procedimiento1: Mostrara la sumatoria de dos números.
* Procedimiento2: Mostrara la multiplicación de dos números.

**SOLUCION PASO A PASO:**

**PASO 1:** REALIZAR LA ESPECIFICACION.

**CREATE OR REPLACE PACKAGE** MI\_PAQUETE2

**IS**

**PROCEDURE P1 (A NUMBER, B NUMBER);**

**PROCEDURE P2 (X NUMBER, Y NUMBER);**

**END;**

**PASO 2:** CUERPO DEL PAQUETE.

**CREATE OR REPLACE PACKAGE BODY** MI\_PAQUETE2

**IS**

**PROCEDURE P1** (A NUMBER, B NUMBER)

IS

C NUMBER (5);

BEGIN

C:=A+B;

DBMS\_OUTPUT.PUT\_LINE ('LA SUMATORIA ES: ' || C);

END P1;

**PROCEDURE P2** (X NUMBER, Y NUMBER)

IS

Z NUMBER (5);

BEGIN

Z:=X\*Y;

DBMS\_OUTPUT.PUT\_LINE ('LA MULTIPLICACION ES: ' || Z);

END P2;

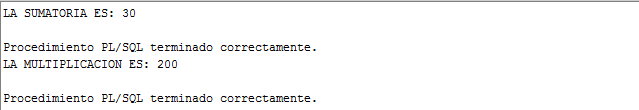
**END MI\_PAQUETE2;**

**PASO 3:** EJECUCION

**SET SERVEROUTPUT ON;**

**EXEC** MI\_PAQUETE2.p1 (10, 20);

**EXEC** MI\_PAQUETE2.p2 (10, 20);

**RESULTADO:**

**EJEMPLO 3: PROCEDURE Y FUNCTION COMBINADOS.**

Implementar un PAQUETE **MI\_PAQUETE3** el cual debe contener lo siguiente:

* Procedimiento1: Mostrara la sumatoria de dos números.
* Funcion1: Calcular la sumatoria de dos números.

**SOLUCION PASO A PASO:**

**PASO 1:** REALIZAR LA ESPECIFICACION.

**CREATE OR REPLACE PACKAGE MI\_PAQUETE3**

**IS**

**PROCEDURE P1 (A NUMBER, B NUMBER);**

**FUNCTION F1 (X NUMBER, Y NUMBER) RETURN NUMBER;**

**END;**

**PASO 2:** CUERPO DEL PAQUETE.

**CREATE OR REPLACE PACKAGE BODY MI\_PAQUETE3**

**IS**

**PROCEDURE P1** (A NUMBER, B NUMBER)

IS

C NUMBER (5);

BEGIN

C:=A+B;

DBMS\_OUTPUT.PUT\_LINE ('LA SUMATORIA ES: ' || C);

END P1;

**FUNCTION F1**(X NUMBER, Y NUMBER)

RETURN NUMBER

IS

Z NUMBER (5);

BEGIN

Z:=X\*Y;

RETURN Z;

END F1;

**END MI\_PAQUETE3;**

**PASO 3:** EJECUCION

--PROCEDIMIENTO

**SET SERVEROUTPUT ON;**

EXEC MI\_PAQUETE3.P1 (10, 20);

--FUNCTION

**SET SERVEROUTPUT ON;**

SELECT MI\_PAQUETE3.F1 (10,20) MULTIPLICA FROM DUAL;

**RESULTADO DEL PROCEDIMIENTO.**



**RESULTADO DE LA FUNCION.**

**EJEMPLO 4: USANDO DATOS DE TABLAS.**

Implemente un paquete llamado MI\_PAQUETE4, el cual contenga los siguientes procedimientos y funciones:

* Obtener el conteo de libros
* Totalización de libros
* El libro más caro
* El libro más barato

**SOLUCION PASO A PASO**

**PASO 1:** REALIZAR LA ESPECIFICACION.

**CREATE OR REPLACE PACKAGE MI\_PAQUETE4**

**IS**

**FUNCTION** FCONTEO\_LIBROS RETURN NUMBER;

**FUNCTION** FTOTALIZAR\_LIBROS RETURN NUMBER;

**FUNCTION** FLIBRO\_CARO RETURN VARCHAR2;

**FUNCTION** FLIBRO\_BARATO RETURN VARCHAR2;

**PROCEDURE** SP\_LLAMADA;

**END;**

**PASO 2:** CUERPO DEL PAQUETE.

**CREATE OR REPLACE PACKAGE BODY** MI\_PAQUETE4

**IS**

FUNCTION FCONTEO\_LIBROS

RETURN NUMBER

IS

CONTAR NUMBER:=0;

BEGIN

SELECT COUNT(CODLIB) INTO CONTAR FROM LIBRO;

RETURN CONTAR;

END FCONTEO\_LIBROS;

FUNCTION FTOTALIZAR\_LIBROS

RETURN NUMBER

IS

TOTAL NUMBER:=0;

BEGIN

SELECT SUM(CODLIB) INTO TOTAL FROM LIBRO;

RETURN TOTAL;

END FTOTALIZAR\_LIBROS;

FUNCTION FLIBRO\_CARO

RETURN Varchar2

IS

Precio Number;

Nombre VARCHAR2(200);

BEGIN

SELECT TITULOLIB,PRECIOLIB INTO Nombre, Precio

FROM (SELECT TITULOLIB, PRECIOLIB

FROM LIBRO ORDER BY PRECIOLIB DESC) WHERE ROWNUM<=1;

Nombre:= Nombre || ' ' || TO\_CHAR(Precio);

RETURN Nombre;

END FLIBRO\_CARO;

FUNCTION FLIBRO\_BARATO

RETURN Varchar2

IS

Precio Number;

Nombre VARCHAR2(200);

BEGIN

SELECT TITULOLIB,PRECIOLIB INTO Nombre, Precio

FROM (SELECT TITULOLIB,PRECIOLIB

FROM LIBRO ORDER BY PRECIOLIB) WHERE ROWNUM<=1;

Nombre:= Nombre || TO\_CHAR(Precio);

RETURN Nombre;

END FLIBRO\_BARATO;

PROCEDURE SP\_LLAMADA

IS

BEGIN

DBMS\_OUTPUT.PUT\_LINE('=========================================');

DBMS\_OUTPUT.PUT\_LINE('DETALLE DE LIBROS:');

DBMS\_OUTPUT.PUT\_LINE('=========================================');

DBMS\_OUTPUT.PUT\_LINE('CONTEO DE LIBROS : ' || FCONTEO\_LIBROS);

DBMS\_OUTPUT.PUT\_LINE('TOTALIZACION DE LIBROS : ' || FTOTALIZAR\_LIBROS);

DBMS\_OUTPUT.PUT\_LINE('EL LIBRO MAS CARO : ' || FLIBRO\_CARO);

DBMS\_OUTPUT.PUT\_LINE('EL LIBRO MAS BARATO : ' || FLIBRO\_BARATO);

END SP\_LLAMADA;

END MI\_PAQUETE4;

SELECT TITULOLIB, PRECIOLIB INTO Nombre, Precio

FROM (SELECT TITULOLIB, PRECIOLIB

FROM LIBRO ORDER BY PRECIOLIB DESC) WHERE ROWNUM<=1;

Nombre:= Nombre || ' ' || TO\_CHAR(Precio);

RETURN Nombre;

END FLIBRO\_CARO;

FUNCTION FLIBRO\_BARATO

RETURN Varchar2

IS

Precio Number;

Nombre VARCHAR2(200);

BEGIN

SELECT TITULOLIB, PRECIOLIB INTO Nombre, Precio

FROM (SELECT TITULOLIB, PRECIOLIB

FROM LIBRO ORDER BY PRECIOLIB) WHERE ROWNUM<=1;

Nombre:= Nombre || TO\_CHAR(Precio);

RETURN Nombre;

END FLIBRO\_BARATO;

PROCEDURE SP\_LLAMADA

IS

BEGIN

DBMS\_OUTPUT.PUT\_LINE('=========================================');

DBMS\_OUTPUT.PUT\_LINE('DETALLE DE LIBROS:');

DBMS\_OUTPUT.PUT\_LINE('=========================================');

DBMS\_OUTPUT.PUT\_LINE('CONTEO DE LIBROS : ' || FCONTEO\_LIBROS);

DBMS\_OUTPUT.PUT\_LINE('TOTALIZACION DE LIBROS : ' || FTOTALIZAR\_LIBROS);

DBMS\_OUTPUT.PUT\_LINE('EL LIBRO MAS CARO : ' || FLIBRO\_CARO);

DBMS\_OUTPUT.PUT\_LINE('EL LIBRO MAS BARATO : ' || FLIBRO\_BARATO);

END SP\_LLAMADA;

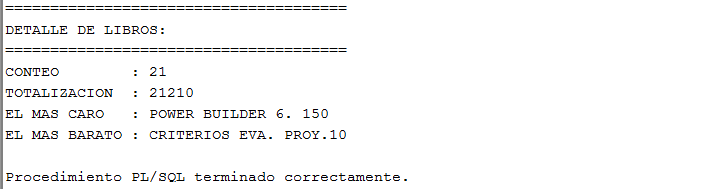
END MI\_PAQUETE4;

**PASO 3:** EJECUCION

**SET SERVEROUTPUT ON;**

**EXEC** MI\_PAQUETE4.SP\_LLAMADA;

**RESULTADO:**



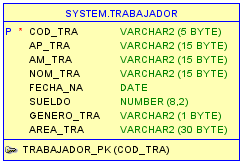
**CONCLUSIONES Y RECOMENDACIONES DE LA EXPERIENCIA**

La codificación PL/SQL de Oracle es tan amplia, que ofrece a los estudiantes una alta gama de alternativas para la resolución de problemas en paquetes, procedimientos y funciones, no difíciles de entender, con la prioridad de manejar los datos con suma responsabilidad por el programador Oracle.

**ACTIVIDAD VIRTUAL**

Revisar y analizar el material presentado del tema 10.

**SE TIENE LA SIGUIENTE TABLA:**



**LA ESTRUCTURA Y DATOS DE LA TABLA TRABAJADOR SE ENCUENTRAN**

**EN EL ARCHIVO SCRIPT DE BASE DE DATOS.**

**Semana10\_PL\_SQL\_2.sql,**

**FAVOR DE DESCARGAR DICHO ARCHIVO.**

De acuerdo al material presentado, responda a las siguientes Preguntas:

**RESOLVER:**

Crear las siguientes funciones en PL/SQL.

1. Crear un Paquete el cual permita calcular las operaciones básicas.
2. Implementar un paquete el cual permita determinar:
   1. Conteo de trabajadores
   2. Conteo de varones
   3. Conteo de damas
   4. Totalizar sueldos de Trabajadores varones
   5. Totalizar sueldos de trabajadores mujeres.
   6. Totalización de sueldos para hacer una planilla
   7. Obtener totalización por área elegida.