

Tabla de contenido

1. Investigación.....	2
1.1. Acciones Referenciales	2
1.1.1. ¿Para qué sirven las acciones referenciales?	2
1.1.2. ¿Qué acciones soporta Oracle? ¿Qué permite hacer cada una de ellas?	2
1.2. PL/SQL	3
1.2.1. ¿Qué es PL/SQL?	3
1.2.2. ¿Qué motores lo soportan?	3
1.3. Datos e instrucciones en PL/SQL.....	3
1.3.1. ¿Cuáles son los tipos de datos que ofrece?	3
1.3.2. ¿Cuál es la forma de definir constantes y variables?.....	5
1.3.3. ¿Cómo se define una variable con un tipo tomado de la base de datos?	5
1.3.4. ¿Cuál es la forma de los diferentes tipos de asignación? (Son tres)	5
1.4. Cursores	6
1.4.1. ¿Qué es un cursor implícito? ¿Para qué sirve?	6
1.4.2. ¿Qué es un cursor explícito? ¿Para qué sirve?	6
1.4.3. ¿Cuáles son las excepciones propias de uso de estos cursores?.....	6
1.5. Modularidad	6
1.5.1. ¿Cuál es la estructura general de un bloque PL/SQL?	6
1.5.2. ¿Para qué sirven las diferentes estructuras modulares? (bloque anónimo, procedimiento, función y disparador)	6

1. Investigación

1.1. Acciones Referenciales

1.1.1. ¿Para qué sirven las acciones referenciales?

1.1.2. ¿Qué acciones soporta Oracle? ¿Qué permite hacer cada una de ellas?

Acciones definidas por restricciones de integridad referencial

Las restricciones de integridad referencial pueden especificar acciones particulares que se realizarán en las filas dependientes de una tabla secundaria si se modifica un valor de clave principal referenciado. Las acciones referenciales soportadas por las restricciones de integridad de FOREIGN KEY de Oracle son UPDATE , Delete No Action y Delete Cascade.

Delete No Action

La opción Sin acción (predeterminada) especifica que los valores clave referenciados no pueden actualizarse o eliminarse si los datos resultantes violan una restricción de integridad referencial.

Delete Cascade.

Un borrado cae en cascada cuando se eliminan las filas que contienen valores clave referenciados, lo que hace que todas las filas de las tablas secundarias con valores de clave externa dependientes también se eliminen.

Delete Set Null

Una eliminación establece nulo cuando se eliminan las filas que contienen valores de clave referenciados, lo que hace que todas las filas de las tablas secundarias con valores de clave externa dependientes establezcan esos valores en nulo.

Tomado de :

https://docs.oracle.com/cd/B19306_01/server.102/b14220/data_int.htm

1.2. PL/SQL

1.2.1. ¿Qué es PL/SQL?

PL/SQL es el lenguaje de programación que proporciona Oracle para extender el SQL estándar con otro tipo de instrucciones y elementos propios de los lenguajes de programación.

Con PL/SQL vamos a poder programar las unidades de programa de la base de datos Oracle:

- Procedimientos almacenados
- Funciones
- Triggers
- Scripts

Tomado de :

<http://www.juntadeandalucia.es/servicios/madeja/contenido/recurso/107>

1.2.2. ¿Qué motores lo soportan?

El lenguaje PL/SQL está incorporado en:

- Servidor de la base de datos.
- Herramientas de Oracle (Forms, Reports, ...)

Tomado de :

<https://es.wikipedia.org/wiki/PL/SQL>

1.3. Datos e instrucciones en PL/SQL

1.3.1. ¿Cuáles son los tipos de datos que ofrece?

Tipos de Variables

- ESCALARES
- BOOLEANAS
- COMPUESTAS
- LOB
- DE ENLACE (BIND)

Variables Escalares

- VARCHAR2 (longitud_máxima)
- NUMBER [(precisión, escala)]
- DATE

- CHAR [(longitud_máxima)]
- LONG
- LONG RAW
- BOOLEAN
- BINARY_INTEGER

Variables BOOLEANAS

- A una variable Boolean sólo se le pueden asignar los valores: **TRUE**, **FALSE** o **NULL**.
- Estas variables están conectadas por los operadores lógicos **AND**, **OR** y **NOT**.

Variables LOB

Permiten almacenar bloques de datos no estructurados, como pantallas de texto, imágenes gráficas, vídeo clips, y sonido, de hasta 4 Gb. de tamaño.

CLOB Character Large Object. Se utiliza para almacenar bloques grandes de datos de caracteres.

BLOB Binary Large Object. Se utiliza para almacenar objetos binarios grandes en la B.D.

BFILE Binary File. Se utiliza para almacenar objetos binarios grandes en archivos del sistema operativo, fuera de la B.D.

NCLOB National Language Character Large Object. Se utiliza para almacenar en la B.D. bloques grandes de datos NCHAR de un byte único o multi-bytes de ancho fijo.

Variables de Enlace (Bind)

Son variables de SQL*Plus.

Las pueden referenciar bloques de PL/SQL mediante el uso del ampersand (&).

Tomado de :

https://www.mundoracle.com/pl-sql.html?Pg=sql_plsql_11.htm

1.3.2. ¿Cuál es la forma de definir constantes y variables?

Declaración de Variables

```
identificador [CONSTANT] tipo_dato [NOT NULL]  
[:= | DEFAULT expresión];
```

● Reglas para nombres:

- Dos variables pueden tener el mismo nombre, si están en bloques diferentes.
- El nombre de la variable (identificador) no debería ser el mismo que el de una columna de una tabla utilizada en el bloque.

Declaración de Constantes

La declaración de una constante es similar a la declaración de una variable, añadiendo la palabra **CONSTANT** y asignándole a continuación un valor a la constante.

Ejemplo : `PI CONSTANT REAL := 3.14159;`

Tomado de :

<https://elbauldelprogramador.com/plsql-declaracion-de-variables/>

1.3.3. ¿Cómo se define una variable con un tipo tomado de la base de datos?

Variables de Enlace (Bind)

- Son variables de SQL*Plus.
- Las pueden referenciar bloques de PL/SQL mediante el uso del ampersand (&).
- Ejemplo:

```
CLEAR SCREEN  
ACCEPT g_nemp NUMBER(4) PROMPT 'Número empleado: '  
DECLARE  
    v_sal emp.sal%TYPE;  
BEGIN  
    SELECT sal INTO v_sal FROM emp  
    WHERE empno = &g_nemp;  
END;
```

1.3.4. ¿Cuál es la forma de los diferentes tipos de asignación? (Son tres)

- Zonas de declaraciones
- Zona de instrucciones
- Zona de tratamiento de excepciones

1.4. Cursores

1.4.1. ¿Qué es un cursor implícito? ¿Para qué sirve?

Este tipo de cursores se utiliza para operaciones **SELECT INTO**. Se usan cuando la consulta devuelve un único registro.

1.4.2. ¿Qué es un cursor explícito? ¿Para qué sirve?

Son los cursores que son declarados y controlados por el programador. Se utilizan cuando la consulta devuelve un conjunto de registros. Ocasionalmente también se utilizan en consultas que devuelven un único registro por razones de eficiencia. Son más rápidos.

1.4.3. ¿Cuáles son las excepciones propias de uso de estos cursores?

ACCESS_INTO_NULL, COLLECTION_IS_NULL , CURSOR_ALREADY_ ,
DUP_VAL_ON_ , INVALID_CURSOR, INVALID_NUMBER , NO_DATA_FOUND
, NOT_LOGGED_ON , entre otras.

Tomado de : <https://elbauldelprogramador.com/plsql-cursos/>

1.5. Modularidad

1.5.1. ¿Cuál es la estructura general de un bloque PL/SQL?

```
DECLARE - Opcional
Variables, cursores, excepciones definidas por el usuario
BEGIN - Obligatorio. Indica que empieza el programa
  Sentencias SQL y Sentencias de Control PL/SQL
EXCEPTION - Opcional
  Acciones a realizar cuando se produzcan errores
END; - Obligatorio
```

1.5.2. ¿Para qué sirven las diferentes estructuras modulares? (bloque anónimo, procedimiento, función y disparador)

Procedimiento

Un procedimiento [almacenado] es un subprograma que ejecuta una acción específica y que no devuelve ningún valor por si mismo, como sucede con las funciones. Un procedimiento tiene un nombre, un conjunto de parámetros (opcional) y un bloque de código. Para crear un procedimiento (stored procedure: procedimiento almacenado) usaremos la siguiente sintaxis:

```

CREATE {OR REPLACE} PROCEDURE nombre_proc( param1 [IN | OUT | IN OUT] tipo,
IS
    -- Declaración de variables Locales
BEGIN
    -- Instrucciones de ejecución
[EXCEPTION]
    -- Instrucciones de excepción
END;

```

Función

- Una función es un bloque nombrado PL/SQL que devuelve un valor.
- Una función puede estar almacenada en la B.D., como objeto de la B.D., para repetidas ejecuciones.
- Una función puede ser llamada como parte de una expresión.

Sintaxis Creación Funciones

```

CREATE [OR REPLACE] FUNCTION nombre_func
(argumento1 [modo] tipo_de_dato,
argumento2 [modo] tipo_de_dato,
...)
RETURN tipo_de_dato
IS | AS
    bloque_PL/SQL;

```

Bloque anónimo

Un bloque anónimo es aquel bloque que no tiene asignado un nombre.

```
SET SERVEROUTPUT ON;

DECLARE
    A VARCHAR(10) := '';
BEGIN
    SELECT TO_CHAR(SYSDATE) INTO A FROM DUAL;
    DBMS_OUTPUT.PUT_LINE('LA FECHA ACTUAL ES : ' || A);
EXCEPTION
    WHEN OTHERS THEN DBMS_OUTPUT.PUT_LINE('HOLA');
END;
```

Disparadores

Los disparadores (o triggers) son bloques de código PL/SQL asociados a una tabla y que se ejecutan automáticamente como reacción a una operación DML específica (INSERT, UPDATE o DELETE) sobre dicha tabla.

```
CREATE {OR REPLACE} TRIGGER nombre_disp
    [BEFORE|AFTER]
    [DELETE|INSERT|UPDATE {OF columnas}] [ OR [DELETE|INSERT|UPDATE {OF columnas}]
    ON tabla
    [FOR EACH ROW [WHEN condicion disparo]]
[DECLARE]
    -- Declaración de variables Locales
BEGIN
    -- Instrucciones de ejecución
[EXCEPTION]
    -- Instrucciones de excepción
END;
```
