**AUTOGUIA 5**

**MODELOS Y BASES DE DATOS**

**ESTUDIANTES:**

**JUAN SEBASTIAN PUENTES JULIO**

**CHRISTIAN ALFONSO ROMERO MARTÌNEZ**

**DOCENTE:**

**MARIA IRMA ROZO**

**ESCUELA COLOMBIANA DE INGENIERÍA JULIO GARAVITO**

**11/09/2024**

**A. Transacciones**

**1. ¿Cómo se define el comienzo y fin de una transacción en ORACLE?**

* **Comienzo de una transacción:** Una transacción comienza cuando se hace un INSERT, UPDATE y DELETE.
* **Fin de una transacción:** Una transacción termina cuando se ejecuta COMMIT, ROLLBACK o SET TRANSACTION.
  + **COMMIT:** Confirma y hace los cambios.
  + **ROLLBACK:** Borra los cambios realizados desde el inicio de la transacción o desde el último guardado.
  + **SET TRANSACTION:** Se finaliza una transacción si se cambia el estado.

**2. ¿Cuáles son los diferentes tipos de aislamiento que soporta ORACLE? Para cada uno de ellos detalle, ¿cómo maneja los bloqueos? ¿qué problemas resuelve?**

1. **Read Uncommitted (lectura no confirmada):**
   * **Bloqueos:** No se aplican bloqueos, una transacción puede hacer consultas de datos que pueden estar siendo modificados por otras transacciones.
   * **Problemas que resuelve:** Ninguno ya que se califica como lectura sucia.
2. **Read Committed (lectura confirmada):**
   * **Bloqueos:** En las filas se aplican mientras se lee o modifica una fila. Cada vez que una transacción consulta datos, solo puede leer datos de transacciones confirmadas.
   * **Problemas que resuelve:** Las consultas con las que trabaja ya son confirmadas.
3. **Repeatable Read (lectura repetible):**
   * **Bloqueos:** Las transacciones no cambian los datos hasta que terminen.
   * **Problemas que resuelve:** Al usar read committed se pueden tener lecturas diferentes durante la transacción, este evita eso.
4. **Serializable:**
   * **Bloqueos:** Aplica bloqueos estrictos a las filas y tablas, garantizando que ninguna otra transacción pueda modificar los datos reiteradamente.
   * **Problemas que resuelve:** Resuelve tanto las lecturas diferentes asegurando que las transacciones se ejecuten organizadamente.

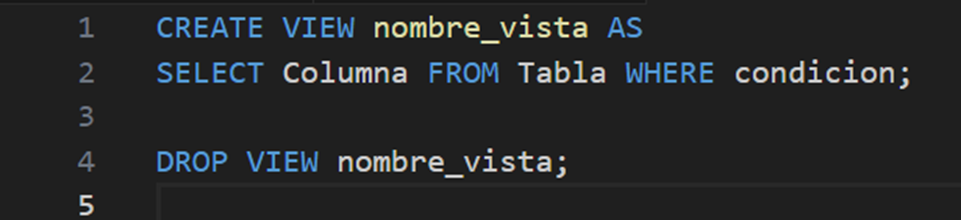
**3. ¿Cuál es el tipo de aislamiento por defecto en ORACLE?**

* El aislamiento por defecto de ORACLE es Read Committed.

### **B. Vistas**

**1. ¿Cuáles son los mecanismos para la creación y borrado de vistas en ORACLE?**

Se utiliza CREATE VIEW para la creación de la vista y DROP VIEW para eliminar la vista.



**2. ¿Cuáles son las restricciones de las vistas en ORACLE?**

* Las vistas no guardan datos solo definiciones de consultas.
* Cuando se tiene SUM, COUNT o funciones similares no se permite INSERT, UPDATE o DELETE.
* Cuando se tienen JOIN no se permite INSERT, UPDATE o DELETE.
* En vistas con DISTINCT las filas resultantes deben ser únicas
* En vistas con GRUP BY y similares se califican como listas complejas por lo que pueden tener restricciones en UPDATE

**C. Modularidad Paquetes**

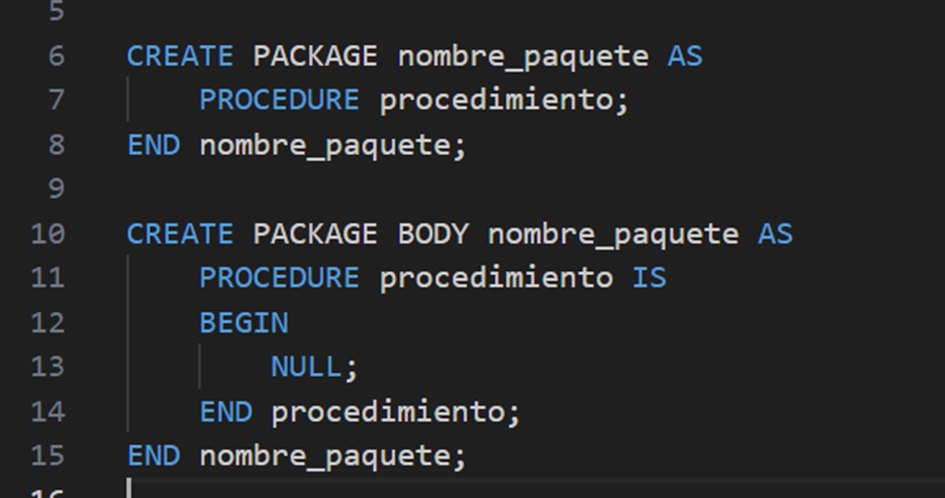
**1. ¿Para qué sirve un paquete?**

Sirven para organizar el código en módulos lógicos, facilitando la reutilización, mantenimiento y encapsulamiento.

Los paquetes son unidades de modularidad que agrupa procedimientos, funciones y variables. Estos almacenan la implementación de los procedimientos y funciones, y en su especificación define procedimientos, funciones y otros tipos de accesibilidad.

**2. ¿Cuáles son los mecanismos para la creación, invocación, modificación y borrado de paquetes en ORACLE?**

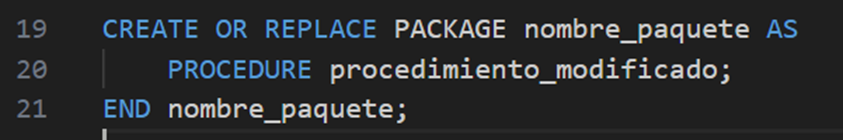
**Creación de un paquete:** Para crear un paquete se utiliza el comando CREATE PACKAGE para la especificación y CREATE PACKAGE BODY para el cuerpo.



**Invocación de un paquete:** Para invocar un paquete se usa EXEC.



**Modificación de un paquete:** Para modificar un paquete se utiliza el comando CREATE OR REPLACE PACKAGE o CREATE OR REPLACE PACKAGE BODY para actualizar su especificación o cuerpo.



**Borrado de un paquete:** Para eliminar un paquete se utiliza DROP PACKAGE

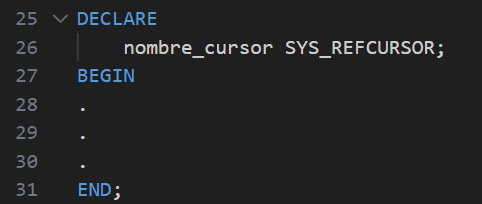


### **D. SYS\_REFCURSOR**

**1. ¿Qué es un SYS\_REFCURSOR? ¿Para qué sirve?**

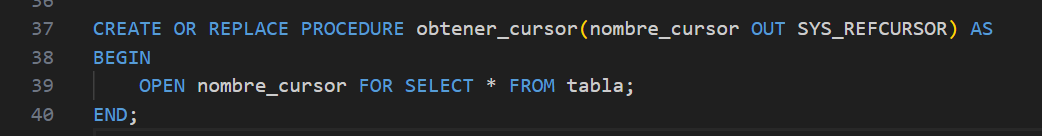
Un **SYS\_REFCURSOR** es un cursor en Oracle que se utiliza para retornar resultados desde un procedimiento. Permite manejar resultados de consultas de forma flexible. Es útil para procedimientos almacenados que necesitan devolver un conjunto de resultados a una aplicación cliente.

**2. ¿Cómo se define, se asigna y se retorna un SYS\_REFCURSOR?**

**Definición de un SYS\_REFCURSOR **

**Asignación de un SYS\_REFCURSOR **

**Retorno de un SYS\_REFCURSOR**

****