

TRANSACCIONES Y SEGURIDAD

A. Transacciones

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

- El inicio se define procediendo con la siguiente sintaxis:

CREATE OR REPLACE

PROCEDURE TRANSACCION (description)

IS

BEGIN

- Para terminar una transacción se procede así:

END;

/

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

Hay 4 diferentes tipos de aislamiento, pero Oracle solo soporta 2 de ellos que son:

Lectura comprometida: solo se permiten lecturas de datos comprometidos.

Serializable: especifica que todas las transacciones ocurran de modo aislado, es decir, una Tras de otra.

B. Seguridad

1. El mecanismo de control de acceso discrecional, ¿cómo se define en ORACLE?

Explicite acciones, objetos y personas autorizadas.

Para los permisos se puede dar permisos y quitarlos o revocarlos, y así se definen en Oracle respectivamente:

Dar.

GRANT privilege

ON element

TO [user | role | PUBLIC]

[WITH GRANT OPTION]

Quitar.

REVOKE privilege

ON element

FROM [user | role | PUBLIC]

[RESTRICT | CASCADE]

Existen 4 acciones:

INSERT [(columnas)]

DELETE

UPDATE [(columnas)]

SELECT [(columnas)]

C. Vistas

1. ¿Para qué sirve una vista?

Es una ventana mediante la cual se pueden leer algunos datos de las tablas.

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

Para crear vistas se usa el siguiente mecanismo:

CREATE VIEW nombre_vista AS (consulta SQL);

Para borrar vistas se usa el siguiente mecanismo:

DROP VIEW nombre_vista;

3. ¿Cuáles son las restricciones de las vistas en ORACLE?

Hay 4 restricciones que son:

FORCE: crea la vista sin importar que la tabla base exista o no.

WITH CHECK OPTION: especifica que solamente las filas accesibles a la vista pueden ser insertadas o actualizadas.

CONSTRAINT: es el nombre que se le asigna a los checks.

WITH READ ONLY: Asegura que ninguna operación pueda realizarse sobre esta vista, solamente la lectura.

D. Modularidad Paquetes

1. ¿Para qué sirve un paquete?

Sirve para guardar subprogramas en los que cada uno de estos son las historias de casos de uso de cada usuario de la base de datos.

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

Los mecanismos para la creación:

```
CREATE OR REPLACE PACKAGE (NOMBRE)
```

Los de invocar:

```
EXECUTE Packete.Funcion (ARGUMENTOS)
```

Los de modificar:

```
ALTER PACKAGE (NOMBRE)
```

Los de Eliminar:

```
DROP PACKAGE Paquete;
```

E. Cursores

1. ¿Qué es un SYS_REFCURSOR? ¿Para qué sirve?

Un REF CURSOR es una variable, definida como un tipo de cursor, que apuntará o hará referencia a un resultado de un cursor.

2. ¿Cómo se define, se asigna y se retorna?

Define:

```
TYPE var_refcursor IS REF CURSOR;
```

Asignar y Retornar:

```
FUNCTION retorna_datos RETURN var_refcursor;
```

```
END ejemplo_package;
```

```
create or replace PACKAGE BODY ejemplo_package IS
```

```
FUNCTION retorna_datos RETURN var_refcursor IS  
var_cursorpackage var_refcursor;
```

```
BEGIN
```

```
OPEN var_cursorpackage FOR
```

```
SELECT STATEMENT ;
```

```
RETURN var_cursorpackage;  
END;  
END ejemplo_package;
```

Bibliografía:

- <https://magicplsqli.blogspot.com.co/2016/12/ref-cursor-de-oracle.html>
- https://docs.oracle.com/cd/E12839_01/bi.1111/b32122/orbr_refcur002.htm#RSBDR2044
- http://psoug.org/definition/SYS_REFCURSOR.htm
- https://www.lawebdelprogramador.com/foros/Oracle/982754-manejo-de-sys_refcursor.html
-