

**MODELOS Y BASES DE DATOS**  
**PL/ SQL Básico**  
**2022-1**  
**Guía autoestudio 5/6**

**OBJETIVO**

1. Conocer herramientas que facilitan el trabajo del desarrollador de una base de datos específicamente la herramienta SQL Developer
2. Desarrollar competencias para definir e implementar los servicios que puede ofrecer una base de datos cumpliendo requisitos de concurrencia, seguridad y recuperación.

**TÓPICOS OBJETIVO**

1. Transacciones para concurrencia y recuperación
2. Permisos como mecanismos de seguridad

**ENTREGA**

Publicar las respuestas en el espacio correspondiente en un archivo **.zip** , el nombre de este archivo debe ser la concatenación en orden alfabético de los primeros apellidos de cada uno de los miembros.

**TRANSACCIONES**

**A. Transacciones**

1. ¿Cómo se define el comienzo y fin de una transacción en ORACLE?
2. ¿Cuáles son los diferentes tipos de aislamiento que soporta ORACLE? Para cada uno de ellos detalle, ¿cómo maneja los bloqueos? ¿qué problemas resuelve?
3. ¿Cuál es el tipo de aislamiento por defecto en ORACLE?

**B. Seguridad**

1. El mecanismo de control de acceso discrecional, ¿cómo se define en ORACLE?  
Explicite acciones, objetos y personas autorizadas.

**C. Vistas**

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

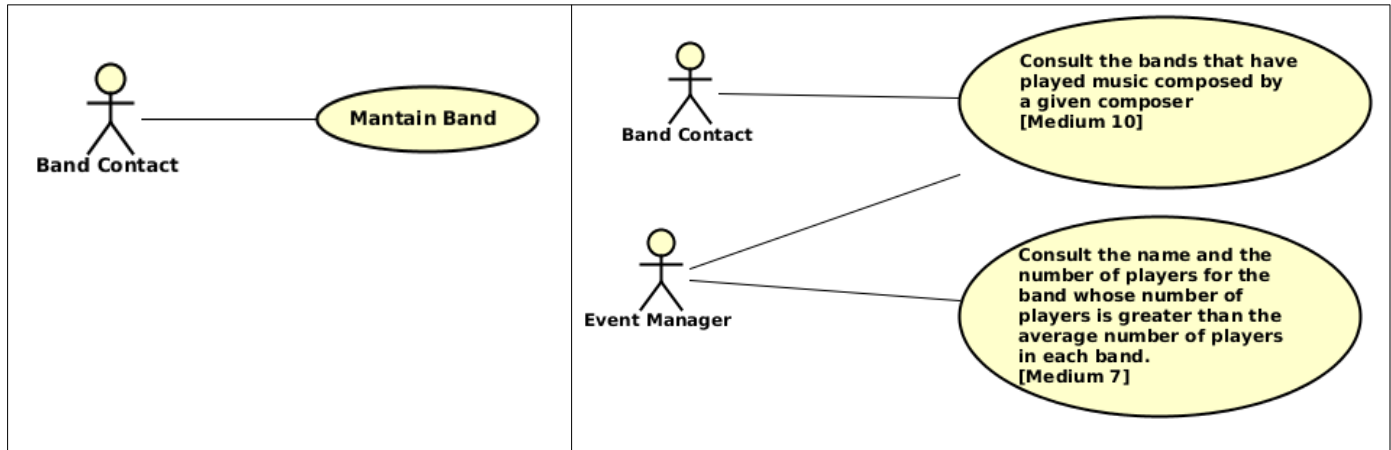
**D. Modularidad Paquetes**

1. ¿Para qué sirve un paquete?
2. ¿Cuáles son los mecanismos para la creación, invocación, modificación y borrado de paquetes en ORACLE?

**E. SYS\_REFCURSOR**

1. ¿Qué es un SYS\_REFCURSOR? ¿Para qué sirve?
2. ¿Cómo se define, se asigna y se retorna?

## PRACTICANDO.



### NOTAS

- El caso de uso Mantain Band está definido en el autoestudio 4.  
En la consulta (ayuda: SYS\_REFCURSOR Ver moodle)
- Las consultas operativas están definidas en autoestudio 5 y en SQLZoo.  
Estas consultas retornen un CURSOR (ayuda: SYS\_REFCURSOR Ver moodle)

Consultar [especificaciones de entrega](#) [Componentes](#), [Seguridad](#) y [Pruebas](#)

### A. Ofreciendo servicios

1. Implemente los paquetes de componentes necesario para ofrecer las funciones básicas y consultas del ciclo actual del sistema (CRUD).

#### **PC\_BANDS [Consultar diseño al final]**

(CRUDE (la especificación) , CRUDI (la implementación) )

2. Proponga un caso de prueba exitoso por subprograma.  
(CRUDOK)
3. Proponga tres casos en los que el subprograma no se puede ejecutar.  
(CRUDNoOK)
4. Escriba las instrucciones necesarias para eliminar los paquetes.  
(CRUDX)

#### PC\_BANDS

```
ad(name : VARCHAR, home : NUMBER, type : VARCHAR, contact : NUMBER) : NUMBER
up_type(band : NUMBER, type : VARCHAR) : void
up_ad_player(band : NUMBER, player : NUMBER) : void
co_band(band : NUMBER) : SYS_REFCURSOR
co_players(band : NUMBER) : SYS_REFCURSOR
de(band : NUMBER) : void
co_byComposer(composer : NUMBER) : SYS_REFCURSOR
co_theBigOnes() : SYS_REFCURSOR
```

ad Retoma el identificador asignado  
co\_band y co\_players es el consultar  
de mantener banda  
co\_byComposer es medium 10  
co\_theBigOnes es medium 7