

# DOCUMENTO RECOPIULATORIO EXAMEN FINAL

CFGS. Desarrollo aplicaciones web y Desarrollo de  
aplicaciones multiplataforma

M02B. Gestión de base de datos



**DOCUMENTO RECOPIULATORIO EXAMEN FINAL**

Cada UF del Examen constará de 20 preguntas de tipo test que serán el **100%** de la nota del Examen por cada UF. Las preguntas pueden ser de un formato puramente teórico o pueden estar ligadas a un ejercicio de código. Recordad que cada UF está valorada sobre **10** y se deben **aprobar ambas por separado.**

## PREGUNTAS EXAMEN

### Unidad Formativa 3: Lenguajes SQL: DCL y extensión procedimental

- **Usuarios y tablespace:** create user, como le definimos un tablespace, que cuestión tiene ese tablespace en esa base de datos, cual es esa cuestión, porque se realiza esa cuestión, cual es la importancia de los usuarios (sus limitaciones, como sys, system...), create user, identified by, asignar tablespace, si es un usuario de sistema, ¿sys y system es un usuario dentro de la base de datos?
- Procedimientos y Funciones y diferencias
- Cursores (implícitos y explícitos) y Estructuras de control (EXCEPTION)
- Disparadores triggers
- SetServerOutPutOn para mostrar el mensaje por pantalla
- Tablas
- Bucles con la peculiaridad del loop, también el for, el if, el while, end if, then (¿then en lugar de corchetes?)
- Errores, predefinidos por el usuario, o por el lenguaje  
Bloque de código anónimo, les hemos añadido bloques exception

#### Pistas preguntas

- GRANT (modificar permisos, sesiones, privilegios...)
- Ruta del tablespace
- Sentencia DROP vs DELETE
- ALTER USER, TABLESPACE (tamaño, fichero asignado, todo lo que podemos alterar)
- Usuarios de sistema sys, system
- Las tablas de usuario no vana entrar.

## Unidad Formativa 4: Bases de datos objeto-relacionales

- Colecciones de datos (Array)
- Objetos
  - Tablas con objetos
  - Cuerpo(Body)
  - Mejoras de estos en la BBDD

### Pistas preguntas

- Herencia, polimorfismo y encapsulamiento.
- Sentencia de creación de arrays. [Le gusta preguntar](#)
- [varrays](#), se pueden orientar tanto a tipos de variables como a objetos. Tipos de colecciones con tipo varchar, o con tipo persona por ejemplo, puede almacenar ambos tipos de datos. Definimos tamaño máximo, que no será necesario ocupar en su totalidad.
- [Objetos: CREATE TYPE](#), colecciones o no, hay que declarar el cuerpo a parte, BODY. [Saberse la sintáxis, métodos, funciones, restricciones.](#)
- [UNDER](#). [aparece en las PACS opcionales.](#) Herencia
- Inserción de datos en objetos. [NEW](#)
- Modelo-objeto relacional (Cada registro se considera un objeto).
- Creación de un objeto y su cuerpo (BODY).
- Un objeto puede estar compuesto por nombre, atributos y métodos.
- [Diferencia entre VARCHAR y VARCHAR2](#): Si a VARCHAR le asigno un tamaño de 50 posiciones y solo ocupo 30, los otros 20 me los va a ocupar automáticamente con espacios en blanco, mientras que en VARCHAR2 50 se refiere tan solo al tamaño máximo, sin ocupar en memoria los espacios restantes con espacios en blanco y va a ser un poco más rápido en recursos.