

# Planificación UF1468 – Almacenamiento de la información e introducción a los SGBD

Dr. Juan Pedro Cerro

En esta unidad formativa hemos ofrecido una visión centrada en cómo se accede a la información almacenada en ficheros externos utilizando diferentes tecnologías para, posteriormente, presentar como alternativas diferentes SGBD y sus características principales como una primera aproximación antes del siguiente módulo en el que se procederá a instalar SGBD Relacionales.

- Descargar e instalar el servidor XAMPP y configurarlo para que pueda ser utilizado en modo local.
  - Descargar también un editor de código como Brackets o VS Code para poder trabajar con el lenguaje PHP.
  - Hacer una página web PHP sencilla para explicar el funcionamiento del lado servidor.
  - Explicar los diferentes modelos de almacenamiento de ficheros: secuencial, directo, indexado y calculado (hash). Usar la presentación “*25.- Modelos de almacenamiento de información en ficheros.pptx*”
  - Hacer pruebas de cada uno de ellos usando ejemplos de PHP.
  - Explicar la diferencia del almacenamiento etiquetado XML en comparación con los archivos de texto plano. Compararlo también con los modos de acceso a ficheros explicados anteriormente.
  - Hacer un ejemplo en PHP de parser de ficheros XML, usando el fichero **cursos.xml**.
  - Hacer web scrapping con PHP (“*webScraping.php*”).
  - Ir un salto más allá usando la web “<https://www.ionos.com/digitalguide/websites/web-development/web-scrapping-with-python/>” y hacer un caso práctico. (“*webScraping.py*”)
- (Por si es necesario consultar BeautifulSoup referencia básica: <https://www.crummy.com/software/BeautifulSoup/bs4/doc/>)
- Abrir el manual **WebScraping.pdf** de la UOC.
  - Hacer el **ejercicio de evaluación continua 1**.
  - Explicar el funcionamiento del protocolo LDAP como servicio de directorio. Ponerlo en marcha en el Linux virtualizado.
  - Explicar cómo usar el comando “`slappasswd -h {SHA}`” para conseguir la contraseña encriptada del usuario anterior en Linux.
  - Poner un ejemplo de directorio explicando el contenido del archivo *Empleados.ldif*
  - En el anfitrión usar la aplicación de escritorio **ApacheDirectoryStudio**. Para ello hay que configurar una nueva conexión (LDAP Connection):

Network Parameter Authentication Browser Options Edit Options

Connection name: LINUX

Network Parameter

Hostname: 192.168.1.123

Port: 389

Connection timeout (s): 30

Encryption method: No encryption

Network Parameter Authentication Browser Options Edit Options

Authentication Method

Simple Authentication


Authentication Parameter

Bind DN or user: cn=admin, dc=EMPRESA, dc=COM

Bind password: ••••••

☒ Save password

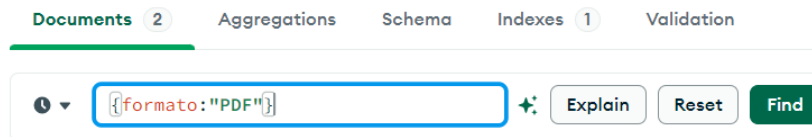
Check Authentication

- Distinguir los diferentes tipos de SGBD existentes: **Jerárquicos, en Red, Relacionales, Orientados a Objetos, Documentales**
  - Comenzar explicando los **Jerárquicos** (árboles dónde cada nodo sólo tiene un padre, pero un padre puede tener varios hijos)  
Esto se podría implementar también con XML anidando etiquetas, pero puede llevarse a cabo con diferentes estrategias de almacenamiento en disco usando lenguajes de programación y algoritmos.
  - Explicar la diferencia con los SGBD en **Red** (no se refiere a que estén distribuidos los servidores, sino que es la forma de conectar las entradas/registros de la BBDD), donde un nodo puede tener varios padres.
  - Explicar la diferencia con los SGBD **Orientado a objetos**: Explicar en qué consiste el paradigma de la programación orientada a objetos como uno de los paradigmas de programación más utilizados. Utilizar la presentación “**28.- Programación Orientada a Objetos.pptx**”
  - Descargar **BlueJ** y hacer un pequeño programa de ejemplo. Mostrando cómo se pueden instanciar objetos en de forma dinámica en BlueJ (Banco de objetos) e interactuar con sus métodos.
- Explicar cómo se representan los objetos mediante el formato JSON. Explicar las reglas de sintaxis:
  - Los nombres de las claves deben estar entre comillas dobles.
  - Los valores de las claves pueden ser strings, números, booleanos, null, objetos o arrays (listas).
  - Los arrays (listas) y objetos pueden estar anidados.
  - Los datos se separan con comas.
  - Los objetos se delimitan con llaves {} y los arrays (listas) con corchetes [].
- Crear un programa en JAVA que transforme un objeto en formato JSON.
- Crear un programa JAVA que pase un json a objeto en la memoria. Para ello se requieren librerías externas, usaremos JACKSON (descargarlas de campus y configurarlas en BlueJ)
- Crear un programa derivado que permita grabar en un fichero de texto varios objetos.
- Crear otro programa que recupere los objetos almacenados en el fichero de texto en formato json.
- Instalar BBDD no SQL **Documentales**, como MondoDB.
  - En la instalación también instalar MongoDB Compass para monitorizar el sistema de BBDD.
  - Luego abrir MongoDB Compass y crear una conexión nueva a localhost:27017 para ver las BBDD por defecto. En el asistente ir a “Advanced Connection Options” y ver cómo un sistema de autenticación posible es el protocolo LDAP.
  - Crear una BBDD (EJEMPLO) documental manualmente  con una colección (colección) inicial.

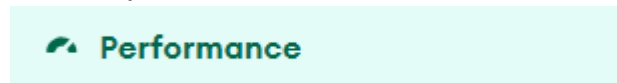
- Añadir dentro un documento con los siguientes documentos:

```
{
  "_id": {
    "$oid": "6628b96fe7ae63ec7f019599"
  },
  "URI": "c:\\datos\\info.txt",
  "formato": "TXT"
}
```

```
{
  "_id": {
    "$oid": "6628bac6e7ae63ec7f01959c"
  },
  "URI": "c:\\datos\\info.pdf",
  "formato": "PDF"
}
```



- Crear un filtro
- Crear un programa JAVA para crear una colección.
- Añadir un archivo pdf usando otra clase JAVA.
- Finalmente, listar la colección con los PDF insertados usando otra clase.
- Volver a ejecutar el listado anterior monitorizando el estado del servidor con la opción “Performance” y ver cómo actúa.



- Explicar cómo borrar elementos de la bbdd creando la clase “BorrarPDF.java”
- Finalmente, recomendar este tutorial: <https://ualmtorres.github.io/howtos/MongoDBJava/>

- Explicar los comandos más habituales de MongoDB (Mongo Query Language) usando la línea de comandos Mongo:

- Crear BBDD documental: use empresa
- Crear colecciones:
 

```
db.createCollection("empleados")
db.createCollection("departamentos")
```

- Ver BBDD y colecciones:

```
show databases
show collections
```

- Añadir documentos:

```
db.departamentos.insertOne({"codigo":"d1","nombre":"Administración"})
db.departamentos.insertOne({"codigo":"d2","nombre":"Comercial"})
db.departamentos.insertOne({"codigo":"d3","nombre":"Logístico"})
```

También se podría hacer con:

```
db.departamentos.insertMany([{"codigo":"d1","nombre":"Administración"}, {"codigo":"d2","nombre":"Comercial"}])
```

```
db.empleados.insertOne({"codigo":1,"nombre":"Juan","edad":40,"activo":true,"departamentos":["d1","d2"]})
db.empleados.insertOne({"codigo":2,"nombre":"María","edad":30,"activo":false,"departamentos":["d2","d3"]})
db.empleados.insertOne({"codigo":3,"nombre":"Pedro","edad":20,"activo":false,"departamentos":["d2"]})
db.empleados.insertOne({"codigo":4,"nombre":"Ana","edad":50,"activo":true,"departamentos":["d2","d3"]})
```

- Hacer una consulta global sobre una colección:

```
db.empleados.find()
```

- Consulta con criterios:

```
db.empleados.find({nombre:"Juan"})
db.empleados.find({activo:true})
db.empleados.find({activo:true,edad:{$lte:30}}) ($eq $ne $gt $gte $lt $lte $in $nin)
db.empleados.find({departamentos:{$exists:true}}) (Si tiene valor el campo "departamentos")
db.empleados.find({$or:[{departamentos:"d1"},{departamentos:"d2"}]})
db.empleados.find({$and:[{departamentos:"d1"},{departamentos:"d2"}]})
```

- Actualización de documentos:

```
db.departamentos.find()
db.departamentos.updateOne({codigo:"d2"},{$set:{nombre:"Ventas"}}) (Sólo modifica el primero, pero updateMany lo hace en todos los documentos)
```

- Borrar documentos:

```
db.empleados.deleteOne({activo:false})
db.empleados.find()
db.departamentos.drop()
```

- Hacer el **ejercicio de evaluación continua 2**.
- Hacer el ejercicio **"JSON to XML"** que está publicado en línea.

- Ahora explicar en qué consisten los SGBD Relacionales usando la presentación “*34.- Sistemas gestores de BBDD Relacionales.pptx*”