

Práctica base de datos avanzada:

- La práctica es obligatoria y su valor es un 70% de la asignatura.
- Se puede realizar la práctica en grupos de hasta 3 personas.
- Se puede utilizar la máquina virtual del virtual donde todos los prototipos vistos durante la asignatura estarán operativos y funcionando.

1.- Diseño en PostgreSQL una base de datos

(2 puntos)

- Relaciones: Alumnos, asignaturas y profesores.
- Realizar todos los cambios necesarios para crear la relación entre alumnos / asignaturas y profesores teniendo en cuenta que:
 1. Un curso siempre lo imparte un profesor.
 2. Los alumnos pueden estar matriculados en varios cursos.
 3. Debemos crear datos aleatorios:
 - o Generar 150.000 cursos.
 - o Generar 1.000.000 de alumnos.
 - o Generar 10.000 profesores.
 - o Asociar todos los cursos a profesores
 - o Asociar 50 alumnos diferentes a cada curso.
 4. Debemos poder consultar estas consultas:
 - o Consulta1: Dado un profesor cuántos alumnos diferentes tiene y cuántos cursos tiene.
 - o Consulta 2: Mostrar curso, alumno y profesor
 - o Consulta 3: Dado un alumno, cuántos cursos tiene y cuántos profesores diferentes tienen.

Entrega:

- La entrega se hará en una carpeta con los 2 dag (apache airflow) correspondientes:
 - o Creación de la estructura PostgreSQL (tarea1) y optimización e índices (tarea2).
 - o Generación de los datos.
- Además de un fichero SQL aparte con las tres consultas del apartado 4.

2.- Generación de auditoría

(2 puntos)

- Creación de una base de datos de auditoría de las tablas del ejercicio 1.
- Cualquier modificación / creación / borrado se debe crear en la base de datos de auditoría.

Entrega:

- La entrega se hará en una carpeta con los 1 dag (apache airflow) correspondientes:
 - o Creación de la base de datos de Auditoría + creación de todas las tablas espejo y los triggers necesarios para hacerla funcionar.

3.- Análisis de la página web de UCJC

(1 punto)

- Realizar un proceso que descargue páginas web de la universidad UCJC.
- Guardar esa información en una base de datos elasticsearch.
- Realizar consultas sobre la búsqueda de esas páginas web buscando keywords introducidas por el usuario.

Entrega:

- La entrega se hará en una carpeta con los 1 dag (apache airflow) correspondientes: Extracción, transformación y carga de la información en la base de datos de elasticsearch.

4.- Diseño FLASK de una aplicación web que se conecte a varias bases de datos:

(5 punto)

Realizar una aplicación FLASK que se conecte a una base de datos PostgreSQL y que:

- Tenga usuarios y contraseñas (cifrada en md5) **(0.5 punto)**.
- Una vez logueado podamos hacer búsquedas sobre las tablas de la sección 1 **(0.5 puntos)**.
- Tengamos las consultas del apartado 1. 4 en diferentes formularios funcionando **(0.5 puntos)**.
- Podamos consultar las tablas de auditoría del apartado 2 **(0.5 punto)**.
- Podamos consultar mediante keywords la información del apartado 3. **(0.5 punto)**.
- Podamos loggear en una base de datos mongoDB toda la información que está realizando los usuarios con la aplicación FLASK. **(2.5 puntos)**.

Entrega:

- Aplicación completa ejecutable con manual de instrucciones.

Opcional

Si se ha entregado la práctica entera y está correcta se puede sustituir el examen por el siguiente punto:

1.- Mejora aplicación FLASK

(3 puntos)

Diseña, comenta e implementa mejoras a la aplicación FLASK haciendo uso de bases de datos y de sus aspectos legales.

Entrega:

- Dags correspondientes y / o ejecutables.
- Memoria explicativa de la mejora presentada.

La mejora tiene que ser única por grupo.

Se evaluará complejidad, utilidad e innovación.