Titulación: Grado en Ingeniería Informática e Ingeniería

en Sistemas de Información

Curso: 2021-2022. Convocatoria Ordinaria de Junio

Asignatura: Bases de Datos Avanzadas – Laboratorio

Practica 3: Seguridad, Usuarios y

Transacciones.

ALUMNO 1:
Nombre y Apellidos:
DNI:
ALUMNO 2:
Nombre y Apellidos:
DNI:
Fecha:
Profesor Responsable:
Mediante la entrega de este fichero los alumnos aseguran que cumplen con la normativa de autoría de trabajos de la Universidad de Alcalá, y declaran éste como un trabajo original y propio.
En caso de ser detectada copia, se puntuará <u>TODA</u> la asignatura como <u>Suspenso –</u> Cero.

Plazos

Tarea en laboratorio: Semana 25 de Abril, Semana 2 de Mayo, Semana 9 de Mayo

y semana 16 de Mayo.

Entrega de práctica: Día 22 de Mayo. Aula Virtual

Documento a entregar: Este mismo fichero con las respuestas a las cuestiones

planteadas, con el código SQL utilizado en cada uno de los apartados. Así mismo se debe de entregar el fichero de configuración postgresql.conf usado en la práctica, así como los ficheros de log de postgres y de pgcli. Se entregará en un

ZIP comprimido: DNI 'sdelosAlumnos PECL3.zip

AMBOS ALUMNOS DEBEN ENTREGAR EL FICHERO EN LA PLATAFORMA.

Introducción

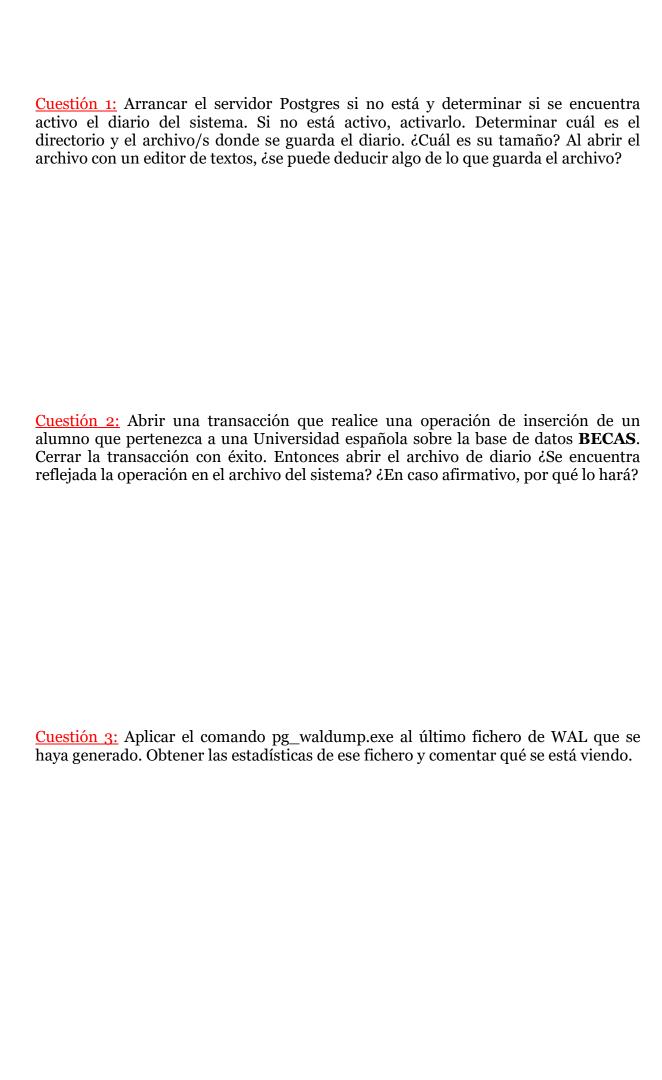
El contenido de esta práctica versa sobre el manejo de las transacciones en sistemas de bases de datos, así como el control de la concurrencia y la recuperación de la base de datos frente a una caída del sistema. Las transacciones se definen como una unidad lógica de procesamiento compuesta por una serie de operaciones simples que se ejecutan como una sola operación. Entre las etiquetas BEGIN y COMMIT del lenguaje SQL se insertan las operaciones simples a realizar en una transacción. La sentencia ROLLBACK sirve para deshacer todos los cambios involucrados en una transacción y devolver a la base de datos al estado consistente en el que estaba antes de procesar la transacción. También se verá el registro diario o registro histórico del sistema de la base de datos (en PostgreSQL se denomina WAL: Write Ahead Loggin) donde se reflejan todas las operaciones sobre la base de datos y que sirve para recuperar ésta a un estado consistente si se produjera un error lógico o de hardware. La versión de postgres a utilizar deberá ser la versión 14.

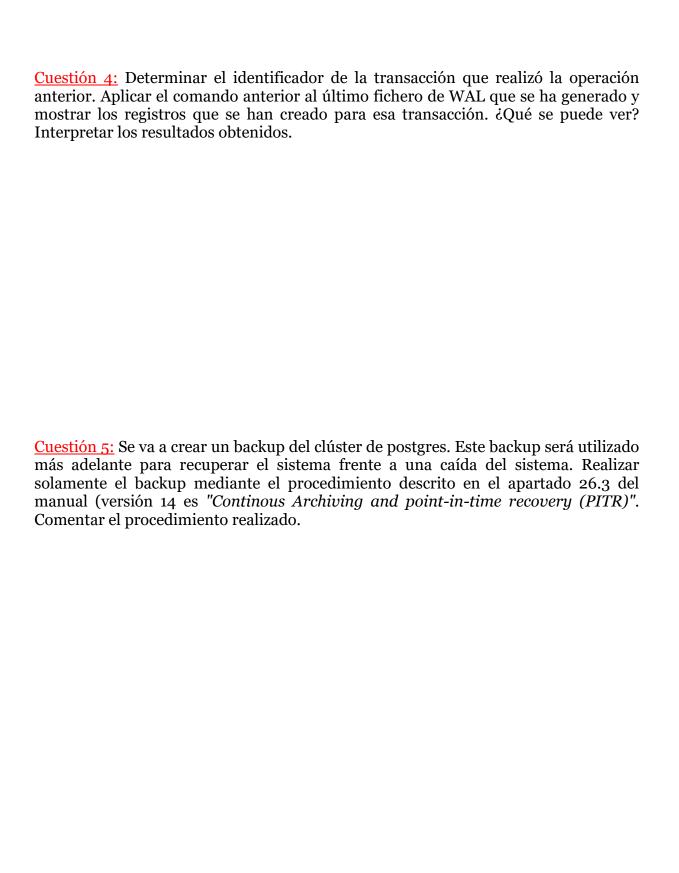
Actividades y Cuestiones

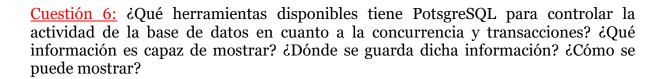
En esta parte la base de datos **BECAS** deberá de ser nueva y no contener datos. Además, consta de 5 actividades:

- Conceptos generales.
- Manejo de transacciones.
- Concurrencia.
- Registro histórico.
- Backup y Recuperación

<u>Cuestión o:</u> Configurar el fichero de Error Reporting and Logging de PostgreSQL para que aparezcan recogidas las sentencias SQL DDL (Lenguaje de Definición de Datos) + DML (Lenguaje de Manipulación de Datos) generadas en dicho fichero. No se pide activar todas las sentencias. No activar la duración de la consulta. También se debe de configurar el log para que en el comienzo de la línea de registro de la información del log ("line prefix") aparezca el DNI de los alumnos que realizan la práctica (ambos), el nombre del host con su puerto, y la fecha y hora de la operación que se ha realizado.







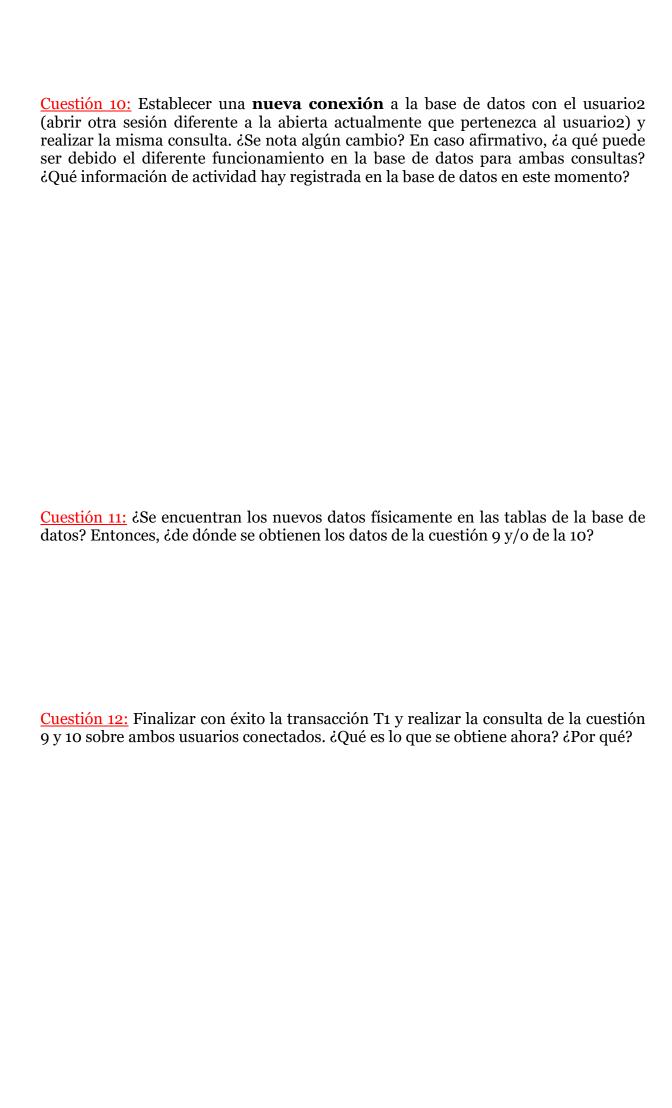
<u>Cuestión 7:</u> Crear tres usuarios en la base de datos que puedan acceder a la base de datos **BECAS** identificados como usuario1, usuario2 y usuario3 que tengan permisos de lectura/escritura a la base de datos **BECAS**, pero que no puedan modificar su estructura. Describir el proceso seguido.

<u>Cuestión 8:</u> Abrir una transacción que inserte una nueva asignatura de una Universidad localizada en Francia (NO cierre la transacción). Realizar una consulta SQL para mostrar los contenidos de la base de datos dentro de esa transacción. Consultar la información sobre lo que se encuentra actualmente activo en el sistema. ¿Qué conclusiones se pueden extraer?

<u>Cuestión 9:</u> Cierre la transacción anterior. Abrir una transacción T1 en el usuario1 que realice las siguientes operaciones sobre la base de datos **BECAS**. NO termine la transacción. Simplemente:

- Inserte un nuevo alumno para la Universidad localizada en Francia.
- Inserte una nueva asignatura de la Universidad anterior.

Realizar cualquier consulta SQL que muestre los datos anteriores insertados para ver que todo está correcto.



<u>Cuestión 13:</u> Sin ninguna transacción en curso, abrir una transacción en un usuario cualquiera y realizar las siguientes operaciones:

- Insertar una nueva Universidad localizada en Polonia.
- Insertar un nuevo alumno perteneciente a la Universidad anterior.
- Insertar una nueva asignatura perteneciente a la Universidad anterior.
- Borrar la Universidad anterior localizada en Polonia.
- Cerrar la transacción.

¿Cuál es el estado final de la base de datos? ¿Por qué?

<u>Cuestión 14:</u> Repetir la cuestión 9 con otra Universidad y asignatura. Realizar la misma consulta de la cuestión 10, pero ahora terminar la transacción con un ROLLBACK y repetir la consulta con los mismos dos usuarios. ¿Cuál es el resultado? ¿Por qué?

<u>Cuestión 15:</u> Crear las siguientes tres tablas:

- tabla_a con campo A de tipo real que sea PK.
- tabla_b con campo B de tipo real que sea PK.
- tabla_c con campo C de tipo real que sea PK.

Insertar una tupla en cada tabla con valor A=10, B=20 y C=30 respectivamente.

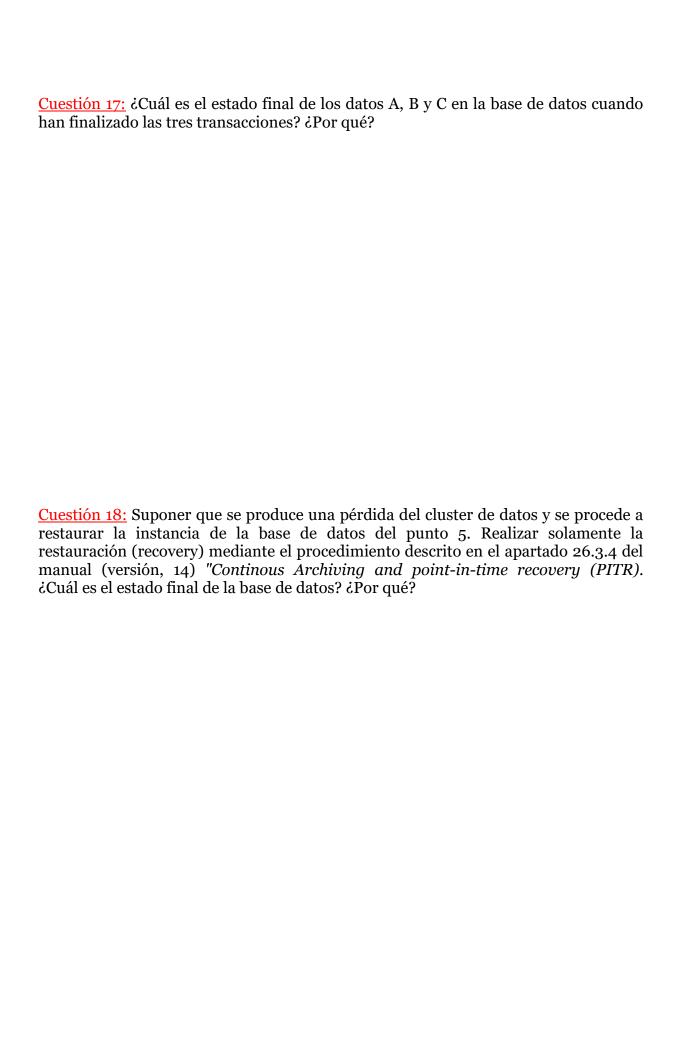
<u>Cuestión 16:</u> Se suministra la tabla siguiente donde se muestra la planificación que hay que seguir por tres transacciones concurrentes asociadas cada una a un usuario diferente que van a acceder en el orden marcado a la tabla **valores**. Se debe de abrir tres consolas pgcli con cada uno de los usuarios y seguir el orden de cada instrucción teniendo en cuenta que un READ es una sentencia SELECT y un WRITE es una sentencia UPDATE. Rellenar la tabla con los valores que ven cada uno de los usuarios en su ejecución de las instrucciones de cada transacción.

Secuencia	T1 Usuario1	T2 Usuario2	T3 Usuario3	Valores Usuario1	Valores Usuario2	Valores Usuario3	Sentencia SQL
1		USUATIOZ	USUATIOS	OSUATIOI	USUATIOZ	USUATIOS	
_	READ(C)						
2	C=C/100						
3	WRITE(C)						
4		READ(B)					
5		B=B+200					
6			READ(A)				
7			A=A*20				
8		WRITE(B)					
9			READ(B)				
10			A=A+B				
11			WRITE(A)				
12	READ(A)						
13	A=A-100						
14	WRITE(A)						
15		READ(C)					
16		C=B+C					
17			READ(C)				
18			B=A*C				

19			WRITE(B)		
20			COMMIT		
21		WRITE(C)			
22	C=C-A				
23	WRITE(C)				
24		READ(C)			
25	COMMIT				
26		C=B*30			
27		WRITE(C)			
28		COMMIT			

Rellenar también la siguiente tabla con el estado del servidor y comentarios de lo que está ocurriendo en cada secuencia de ejecución de las instrucciones de cada transacción en el servidor postgres.

Estado del servidor	Comentarios



Cuestión 19: A la vista de los resultados obtenidos en las cuestiones anteriores, ¿Qué tipo de sistema de recuperación tiene implementado postgreSQL? ¿Qué protocolo de gestión de la concurrencia tiene implementado? ¿Por qué? ¿Genera siempre planificaciones secuenciables? ¿Genera siempre planificaciones recuperables? ¿Tiene rollbacks en cascada? Justificar las respuestas.

Bibliografía

- Capítulo 9: System Information Functions and Operators, System Administration Functions.
- Capítulo 13: Concurrency Control.
- Capítulo 20: Server Configuration.
- Capítulo 26: Backup and Restore.
- Capítulo 28: Monitoring Database Activity.
- Capítulo 30: Reliability and the Write-Ahead log.