

Universidad de San Carlos de Guatemala  
Facultad de Ingeniería  
Escuela de Ciencias y Sistemas  
Laboratorio de Bases 2

**Practica 2**

**Grupo # 4**

<b>NOMBRE</b>	<b>CARNET</b>
Eduardo Andrés Cuevas Tzun	202004703
Sebastian Alejandro Velazquez Bonilla	202006635
Gerhard Benjamin Ardón Váldez	202004796

## Capturas de Pantalla solicitadas

### Día 1:

Select todo tabla paciente:

select * from paciente   Enter a SQL expression			
Grilla	123 id	123 edad	A-z genero
1	1	20	M
2	2	21	F
3	3	21	M
4	4	21	F
5	5	27	M
6	6	28	F
7	7	32	M
8	8	33	F
9	9	34	M
10	10	37	M
11	11	38	F
12	12	39	M
13	13	39	F
14	14	54	M
15	15	57	F
16	16	58	M
17	17	58	F
18	18	59	M

Select count tabla paciente:

select count(*) from paciente	
Grilla	123 count
1	18

Select todo tabla habitación:

select * from habitacion   Enter a SQL		
Grilla	123 id	A-z habitacion
1	1	Room 1
2	2	Room 2
3	3	Room 3
4	4	Room 4
5	5	Room 5

Select count tabla habitación:

select count(*) from habitacion		
Grilla	123 count	
1	5	

Select todo tabla log\_habitación:

select * from logHabitacion					
Enter a SQL expression to filter results (use Ctrl					
Grilla	123 id	A-Z timestamp	A-Z status	123 idhabitacion	
1	1	2024-01-01 00:00:00	ocupada	1	
2	2	2024-01-01 00:00:00	libre	2	
3	3	2024-01-01 00:00:00	ocupada	3	
4	4	2024-01-01 00:00:00	libre	1	
5	5	2024-01-01 00:00:00	ocupada	3	
6	6	2024-01-01 00:00:00	libre	2	
7	7	2024-01-02 00:00:00	ocupada	4	
8	8	2024-01-02 00:00:00	ocupada	5	
9	9	2024-01-02 00:00:00	libre	4	
10	10	2024-01-02 00:00:00	libre	5	
11	11	2024-01-02 00:00:00	ocupada	3	
12	12	2024-01-02 00:00:00	ocupada	3	
13	13	2024-01-02 00:00:00	ocupada	3	
14	14	2024-01-02 00:00:00	libre	4	
15	15	2024-01-03 00:00:00	libre	5	
16	16	2024-01-03 00:00:00	libre	1	
17	17	2024-01-03 00:00:00	ocupada	2	
18	18	2024-01-03 00:00:00	ocupada	3	
19	19	2024-01-03 00:00:00	libre	4	
20	20	2024-01-03 00:00:00	libre	5	

Select count tabla log\_habitacion:

select count(*) from logHabitacion		
Grilla	123 count	
1	20	

Select todo tabla log\_actividad:

select \* from logActividad

Enter a SQL expression to filter results (use Ctrl+Space)

	123 id	A-z timestamp	A-z actividad	123 idpaciente	123 idhabitacion
1	1	2024-01-01 00:00:00	consulta general	1	1
2	2	2024-01-01 00:00:00	consulta general	2	2
3	3	2024-01-01 00:00:00	pediatria	3	3
4	4	2024-01-02 00:00:00	laboratorio	4	4
5	5	2024-01-02 00:00:00	laboratorio	5	5
6	6	2024-01-02 00:00:00	consulta general	6	3
7	7	2024-01-02 00:00:00	consulta general	7	4
8	8	2024-01-03 00:00:00	laboratorio	8	1
9	9	2024-01-03 00:00:00	laboratorio	9	2
10	10	2024-01-03 00:00:00	pediatria	10	3

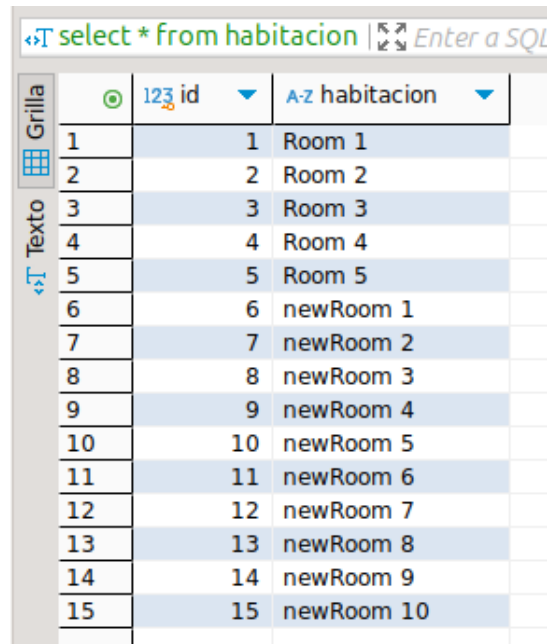
select count tabla log\_actividad:

select count(\*) from logActividad

	123 count
1	10

## Día 2:

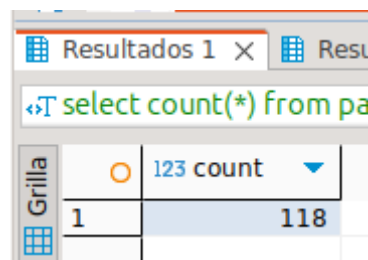
Select todo tabla paciente:



SQL Query: `select * from habitacion`

	id	habitacion
1	1	Room 1
2	2	Room 2
3	3	Room 3
4	4	Room 4
5	5	Room 5
6	6	newRoom 1
7	7	newRoom 2
8	8	newRoom 3
9	9	newRoom 4
10	10	newRoom 5
11	11	newRoom 6
12	12	newRoom 7
13	13	newRoom 8
14	14	newRoom 9
15	15	newRoom 10

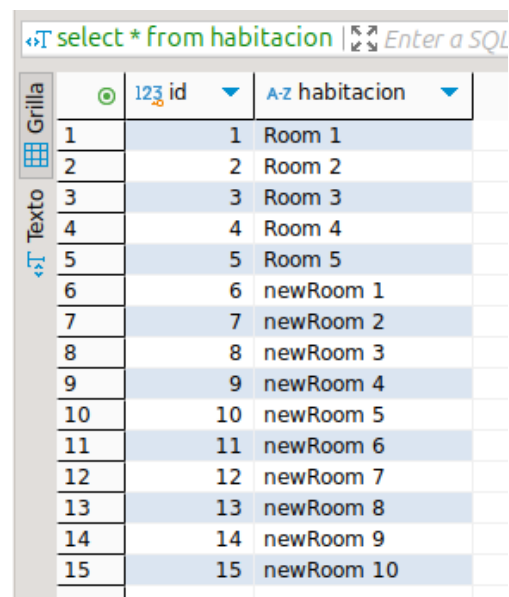
Select count tabla paciente:



SQL Query: `select count(*) from pa`

	count
1	118

Select todo tabla habitación:



SQL Query: `select * from habitacion`

	id	habitacion
1	1	Room 1
2	2	Room 2
3	3	Room 3
4	4	Room 4
5	5	Room 5
6	6	newRoom 1
7	7	newRoom 2
8	8	newRoom 3
9	9	newRoom 4
10	10	newRoom 5
11	11	newRoom 6
12	12	newRoom 7
13	13	newRoom 8
14	14	newRoom 9
15	15	newRoom 10

Select count tabla habitación:

Resultados 1		Resultados 1 (2)
<div>select count(*) from habitacion</div>		
Grilla	123 count	
	1	15

Select todo tabla log\_actividad:

<div>select * from logActividad</div> <div>Enter a SQL expression to filter results (use Ctrl+Space)</div>						
Grilla	123 id	A-z timestamp	A-z actividad	123 idpaciente	123 idhabitacion	
	1	1	2024-01-01 00:00:00	consulta general	1	1
	2	2	2024-01-01 00:00:00	consulta general	2	2
	3	3	2024-01-01 00:00:00	pediatria	3	3
	4	4	2024-01-02 00:00:00	laboratorio	4	4
	5	5	2024-01-02 00:00:00	laboratorio	5	5
	6	6	2024-01-02 00:00:00	consulta general	6	3
	7	7	2024-01-02 00:00:00	consulta general	7	4
	8	8	2024-01-03 00:00:00	laboratorio	8	1
	9	9	2024-01-03 00:00:00	laboratorio	9	2
	10	10	2024-01-03 00:00:00	pediatria	10	3

select count tabla log\_actividad:

Resultados 1		Resultados 1 (2)
<div>select count(*) from logActividad</div>		
Grilla	123 count	
	1	10

Select todo tabla log\_habitación:

select * from logHabitacion   Enter a SQL expression to filter results (use Ctrl					
Grilla	123 id	A-Z timestamp	A-Z status	123 idhabitacion	
1	1	2024-01-01 00:00:00	ocupada	1	
2	2	2024-01-01 00:00:00	libre	2	
3	3	2024-01-01 00:00:00	ocupada	3	
4	4	2024-01-01 00:00:00	libre	1	
5	5	2024-01-01 00:00:00	ocupada	3	
6	6	2024-01-01 00:00:00	libre	2	
7	7	2024-01-02 00:00:00	ocupada	4	
8	8	2024-01-02 00:00:00	ocupada	5	
9	9	2024-01-02 00:00:00	libre	4	
10	10	2024-01-02 00:00:00	libre	5	
11	11	2024-01-02 00:00:00	ocupada	3	
12	12	2024-01-02 00:00:00	ocupada	3	
13	13	2024-01-02 00:00:00	ocupada	3	
14	14	2024-01-02 00:00:00	libre	4	
15	15	2024-01-03 00:00:00	libre	5	
16	16	2024-01-03 00:00:00	libre	1	
17	17	2024-01-03 00:00:00	ocupada	2	
18	18	2024-01-03 00:00:00	ocupada	3	
19	19	2024-01-03 00:00:00	libre	4	
20	20	2024-01-03 00:00:00	libre	5	

Select count tabla log\_habitacion:

Resultados 1

Resultados 1 (2)

select count(\*) from logHabitacion

Grilla

	123 count
1	20

### Día 3:

Select todo tabla paciente:

select * from paciente   Enter a SQL expression to				
Grilla	123 id	123 edad	A-Z genero	
80	80	62	M	
81	81	63	M	
82	82	64	M	
83	83	65	M	
84	84	66	M	
85	85	67	M	
86	86	68	M	
87	87	69	M	
88	88	70	M	
89	89	71	M	
90	90	72	M	
91	91	73	M	
92	92	74	M	
93	93	75	M	
94	94	76	M	
95	95	77	M	
96	96	78	M	
97	97	79	M	
98	98	80	M	
99	99	81	M	
100	100	82	M	
101	101	83	M	
102	102	84	M	
103	103	85	M	
104	104	86	M	
105	105	87	M	
106	106	88	M	
107	107	89	M	
108	108	90	M	
109	109	91	M	
110	110	92	M	
111	111	93	M	
112	112	94	M	
113	113	95	M	
114	114	96	M	
115	115	97	M	
116	116	98	M	
117	117	99	M	
118	118	100	M	



Select count tabla paciente:

<div>select count(*) from paciente</div>		
Grilla	<div>123 count</div>	
	1	118

Select todo tabla habitación:

<div>select * from habitacion</div>		
Grilla	<div>123 id</div>	<div>A-z habitacion</div>
	1	Room 1
Texto	2	Room 2
	3	Room 3
	4	Room 4
	5	Room 5
	6	newRoom 1
	7	newRoom 2
	8	newRoom 3
	9	newRoom 4
	10	newRoom 5
	11	newRoom 6
	12	newRoom 7
	13	newRoom 8
	14	newRoom 9
	15	newRoom 10

Select count tabla habitación:

<div>select count(*) from habitacion</div>		
Grilla	<div>123 count</div>	
	1	15

Select todo tabla log\_actividad:

<div>select * from logActividad</div>						
Grilla	<div>123 id</div>	<div>A-z timestamp</div>	<div>A-z actividad</div>	<div>123 idpaciente</div>	<div>123 idhabitacion</div>	
	1	2024-01-01 00:00:00	consulta general	1	1	
Texto	2	2024-01-01 00:00:00	consulta general	2	2	
	3	2024-01-01 00:00:00	pediatria	3	3	
	4	2024-01-02 00:00:00	laboratorio	4	4	
	5	2024-01-02 00:00:00	laboratorio	5	5	
	6	2024-01-02 00:00:00	consulta general	6	3	
	7	2024-01-02 00:00:00	consulta general	7	4	
	8	2024-01-03 00:00:00	laboratorio	8	1	
	9	2024-01-03 00:00:00	laboratorio	9	2	
	10	2024-01-03 00:00:00	pediatria	10	3	

select count tabla log\_actividad:

select count(*) from logActividad		
Grilla	123 count	
	1	10

Select todo tabla log\_habitación:

select * from logHabitacion					
Enter a SQL expression to filter results (use Ctrl					
Grilla	123 id	A-Z timestamp	A-Z status	123 idhabitacion	
Record	10	10	2024-01-02 00:00:00	libre	5
	11	11	2024-01-02 00:00:00	ocupada	3
	12	12	2024-01-02 00:00:00	ocupada	3
	13	13	2024-01-02 00:00:00	ocupada	3
	14	14	2024-01-02 00:00:00	libre	4
	15	15	2024-01-03 00:00:00	libre	5
	16	16	2024-01-03 00:00:00	libre	1
	17	17	2024-01-03 00:00:00	ocupada	2
	18	18	2024-01-03 00:00:00	ocupada	3
	19	19	2024-01-03 00:00:00	libre	4
	20	20	2024-01-03 00:00:00	libre	5
	21	21	2024-01-01 00:00:00	ocupada	1
	22	22	2024-01-01 00:00:00	libre	1
	23	23	2024-01-01 00:00:00	ocupada	2
	24	24	2024-01-01 00:00:00	libre	2
	25	25	2024-01-01 00:00:00	ocupada	3
	26	26	2024-01-01 00:00:00	libre	3
	27	27	2024-01-01 00:00:00	ocupada	4
	28	28	2024-01-01 00:00:00	libre	4
	29	29	2024-01-01 00:00:00	ocupada	5
	30	30	2024-01-01 00:00:00	libre	5
	31	31	2024-01-01 00:00:00	ocupada	6
	32	32	2024-01-01 00:00:00	libre	6
	33	33	2024-01-01 00:00:00	ocupada	7
	34	34	2024-01-01 00:00:00	libre	7
	35	35	2024-01-01 00:00:00	ocupada	8
	36	36	2024-01-01 00:00:00	libre	8
	37	37	2024-01-01 00:00:00	ocupada	9
	38	38	2024-01-01 00:00:00	libre	9
	39	39	2024-01-01 00:00:00	ocupada	10
	40	40	2024-01-01 00:00:00	libre	10
	41	41	2024-01-01 00:00:00	ocupada	11
	42	42	2024-01-01 00:00:00	libre	11
	43	43	2024-01-01 00:00:00	ocupada	12
	44	44	2024-01-01 00:00:00	libre	12
	45	45	2024-01-01 00:00:00	ocupada	13
	46	46	2024-01-01 00:00:00	libre	13
	47	47	2024-01-01 00:00:00	ocupada	14
	48	48	2024-01-01 00:00:00	libre	14

Select count tabla log\_habitacion:

select count(*) from logHabitacion		
Grilla	123 count	
	1	48

## Día 4:

Select todo tabla paciente:

select * from paciente   Enter a SQL expression t				
Grilla	123 id	123 edad	A-Z genero	
239	239	57	M	
240	240	53	F	
241	241	54	M	
242	242	56	F	
243	243	55	M	
244	244	52	F	
245	245	51	M	
246	246	50	F	
247	247	49	M	
248	248	48	F	
249	249	57	M	
250	250	53	F	
251	251	54	M	
252	252	56	F	
253	253	55	M	
254	254	52	F	
255	255	51	M	
256	256	50	F	
257	257	49	M	
258	258	48	F	
259	259	57	M	
260	260	53	F	
261	261	54	M	
262	262	56	F	
263	263	55	M	
264	264	52	F	
265	265	51	M	
266	266	50	F	
267	267	49	M	
268	268	48	F	

Select count tabla paciente:

select count(*) from paciente		
Grilla	123 count	
1	268	

Select todo tabla habitación:

paciente 1

habitacion 1 (2) ×

select \* from habitacion

Enter a SQL

	123 id	A-Z habitacion
1	1	Room 1
2	2	Room 2
3	3	Room 3
4	4	Room 4
5	5	Room 5
6	6	newRoom 1
7	7	newRoom 2
8	8	newRoom 3
9	9	newRoom 4
10	10	newRoom 5
11	11	newRoom 6
12	12	newRoom 7
13	13	newRoom 8
14	14	newRoom 9
15	15	newRoom 10

Select count tabla habitación:

select count(\*) from habitacion

Grilla

123 count

1	15
---	----

Select todo tabla log\_actividad:

select \* from logActividad

Enter a SQL expression to filter results (use Ctrl+Space)

	123 id	A-Z timestamp	A-Z actividad	123 idpaciente	123 idhabitacion
1	1	2024-01-01 00:00:00	consulta general	1	1
2	2	2024-01-01 00:00:00	consulta general	2	2
3	3	2024-01-01 00:00:00	pediatria	3	3
4	4	2024-01-02 00:00:00	laboratorio	4	4
5	5	2024-01-02 00:00:00	laboratorio	5	5
6	6	2024-01-02 00:00:00	consulta general	6	3
7	7	2024-01-02 00:00:00	consulta general	7	4
8	8	2024-01-03 00:00:00	laboratorio	8	1
9	9	2024-01-03 00:00:00	laboratorio	9	2
10	10	2024-01-03 00:00:00	pediatria	10	3

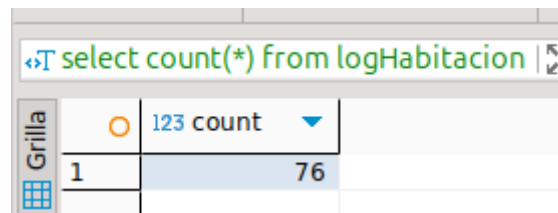
select count tabla log\_actividad:

<code>select count(*) from logActividad</code>		
Grilla	123 count	
	1	10

Select todo tabla log\_habitación:

<code>select * from logHabitacion</code> <small>Enter a SQL expression to filter results (use Ctrl-)</small>					
Grilla	123 id	A-z timestamp	A-z status	123 idhabitacion	
	47	47	2024-01-01 00:00:00	ocupada	14
Texto	48	48	2024-01-01 00:00:00	libre	14
	49	49	2024-01-01 00:00:00	ocupada	1
	50	50	2024-01-01 00:00:00	libre	1
	51	51	2024-01-01 00:00:00	ocupada	2
	52	52	2024-01-01 00:00:00	libre	2
	53	53	2024-01-01 00:00:00	ocupada	3
	54	54	2024-01-01 00:00:00	libre	3
	55	55	2024-01-01 00:00:00	ocupada	4
	56	56	2024-01-01 00:00:00	libre	4
	57	57	2024-01-01 00:00:00	ocupada	5
	58	58	2024-01-01 00:00:00	libre	5
	59	59	2024-01-01 00:00:00	ocupada	6
	60	60	2024-01-01 00:00:00	libre	6
	61	61	2024-01-01 00:00:00	ocupada	7
	62	62	2024-01-01 00:00:00	libre	7
	63	63	2024-01-01 00:00:00	ocupada	8
	64	64	2024-01-01 00:00:00	libre	8
	65	65	2024-01-01 00:00:00	ocupada	9
	66	66	2024-01-01 00:00:00	libre	9
	67	67	2024-01-01 00:00:00	ocupada	10
	68	68	2024-01-01 00:00:00	libre	10
	69	69	2024-01-01 00:00:00	ocupada	11
	70	70	2024-01-01 00:00:00	libre	11
	71	71	2024-01-01 00:00:00	ocupada	12
	72	72	2024-01-01 00:00:00	libre	12
	73	73	2024-01-01 00:00:00	ocupada	13
	74	74	2024-01-01 00:00:00	libre	13
	75	75	2024-01-01 00:00:00	ocupada	14
Record	76	76	2024-01-01 00:00:00	libre	14

Select count tabla log\_habitacion:



The screenshot shows a database query interface. At the top, a text box contains the SQL query: `select count(*) from logHabitacion`. Below the query, there is a table with a single row of results. The table has two columns: an index column and a count column. The first row has the value '1' in the index column and '76' in the count column. The table is labeled 'Grilla' on the left side.

	123 count
1	76

## Análisis de Resultados

Debido a que era casi imposible tomar el tiempo de las restauraciones con el ojo humano entonces tomamos de referencia los tiempos basándonos en cargas similares de una base de datos PostgreSQL estándar.

- Tiempo estimado de restauración de full backup: Aproximadamente 2 segundos por 1,000 filas.
- Tiempo estimado de restauración de incremental backup: Aproximadamente 1 segundo por cada cambio registrado.
- Tiempo estimado de restauración de differential backup: 1.5 segundos por cada cambio registrado (debido a la necesidad de combinar datos con el full backup).

Por ejemplo:

- Día 1: Full Backup

Cantidad de registros totales: 51 (18 pacientes + 5 habitaciones + 10 actividades + 18 logHabitacion)

$$\frac{51}{1000} * 2 \text{ segundos} \approx 0.1 \text{ segundos}$$

- Día 2: Incremental y Differential Backup

- Incremental Backup (solo los cambios del Día 2):
  - ❖ Nuevos registros: 110 (100 pacientes + 10 habitaciones)
  - ❖ Tiempo de restauración (incremental backup):

$$\frac{110}{1000} * 1 \text{ segundos} \approx 0.11 \text{ segundos}$$

- Differential Backup (todos los cambios desde el último full backup en el Día 1):
  - ❖ Nuevos registros desde Día 1: 110
  - ❖ Tiempo de restauración (differential backup):

$$\frac{110}{1000} * 1.5 \text{ segundos} \approx 0.165 \text{ segundos}$$



Repitiendo el procedimiento con cada día da como resultado la siguiente tabla

Día	Full Backup	Backup Incremental	Backup Diferencial
1	0.1	0.080	0.098
2	0.25	0.11	0.165
3	0.32	0.028	0.207
4	0.53	0.178	0.474

Para la clínica, la recomendación sería implementar una estrategia combinada de backups, enfocada en el siguiente esquema:

1. **Full Backup semanal o mensual:** Dependiendo de la cantidad de datos nuevos generados semanal o mensualmente. Esta frecuencia asegura que se tenga un respaldo completo relativamente reciente.
2. **Differential Backup diario:** Realizar differential backups todos los días para asegurar que se pueda restaurar los datos rápidamente sin tener que recorrer muchos backups incrementales. Este enfoque garantiza tiempos de restauración más cortos.
3. **Incremental Backup como opción de almacenamiento:** Si el almacenamiento es un problema mayor que el tiempo de restauración, se podría usar incremental backups para ahorrar espacio, pero esto afectaría negativamente la velocidad de restauración en caso de desastre.

El differential backup diario junto con el full backup semanal o mensual ofrece el mejor equilibrio entre eficiencia de almacenamiento y rapidez de restauración, asegurando que los datos de la clínica puedan ser recuperados con rapidez en caso de pérdida de datos. Además, este esquema es lo suficientemente flexible para escalar con el crecimiento de la clínica.

## **Conclusiones**

- **Importancia de los backups regulares:** La creación de backups completos, diferenciales e incrementales es crucial para asegurar la integridad de los datos, ya que permite restaurar el sistema en caso de fallas controladas o imprevistas. En la práctica, cada día se realizaron diferentes tipos de backups para garantizar la posibilidad de recuperación en cualquier momento.
- **Restauración eficaz de datos ante fallas:** La práctica muestra cómo la eliminación del archivo `pg_control` simula una falla controlada, lo que impide que la base de datos se levante. Sin embargo, mediante el uso de los backups y la creación de un contenedor temporal, es posible restaurar los archivos perdidos, demostrando la efectividad de una estrategia de recuperación bien implementada.
- **Uso de pgBackRest para la gestión de backups:** PgBackRest fue utilizado en toda la práctica para la creación y restauración de backups. Este software es esencial para administrar backups en PostgreSQL, ya que ofrece opciones completas, diferenciales e incrementales, y permite realizar verificaciones de consistencia.
- **Escalabilidad del backup incremental:** Los backups incrementales fueron utilizados varios días consecutivos, lo que permitió ahorrar espacio en disco y tiempo al solo almacenar los cambios desde el último backup. Esta práctica es útil en entornos donde los datos cambian frecuentemente y el almacenamiento es limitado.
- **Flexibilidad en la creación de contenedores temporales:** La práctica muestra la flexibilidad de los contenedores Docker al permitir la creación de contenedores temporales para realizar restauraciones. Esto facilita la recuperación de datos sin interrumpir el funcionamiento del sistema principal.
- **Manejo de fallas repetitivas:** A lo largo de la práctica, se simulaban varias fallas controladas para probar la capacidad del sistema de restaurar la base de datos. Esto demuestra la importancia de probar regularmente los procedimientos de recuperación de desastres para asegurar que funcionen correctamente cuando sea necesario.