

**Universidad de San Carlos de Guatemala**  
**Facultad de Ingeniería**  
**Segundo Semestre 2024**  
**Laboratorio Sistemas de Bases de Datos 2**

## **Práctica 2**

<b>NOMBRE</b>	<b>CARNE</b>
Alvaro Esaú Arenas González	202102864
Jorge Antonio Perez Ordoñez	201900810
Sergio Saúl Ralda	202103216
Ruben Alejandro Ralda	202111835

# DIA 1

**SELECT \* FROM cada tabla**

→ Tabla PACIENTE

The screenshot shows the MySQL Workbench interface. In the Database Explorer, the database 'practica2' is selected, and the 'PACIENTE' table is chosen. In the main pane, a query is being run:

```
1 ✓ SELECT * FROM PACIENTE;
2
3 SELECT * FROM HABITACION;
4
5 SELECT * FROM LOG_ACTIVIDAD;
6
7 SELECT * FROM LOG_HABITACION;
8
9 SELECT COUNT(*) FROM PACIENTE;
```

The output window displays the results of the query:

	idPaciente	edad	genero
1	100000	95	Otro
2	100001	40	Femenino
3	100002	42	Masculino
4	100003	8	Femenino
5	100004	88	Masculino
6	100005	23	Masculino
7	100006	60	Femenino
8	100007	49	Femenino
9	100008	44	Femenino
10	100009	55	Femenino
11	100010	28	Femenino
12	100011	13	Femenino
13	100012	62	Masculino
14	100013	32	Masculino
15	100014	11	Masculino
16	100015	50	Femenino
17	100016	39	Masculino
18	100017	55	Femenino
19	100018	61	Masculino
20	100019	58	Masculino

→ Tabla HABITACION

The screenshot shows the MySQL Workbench interface. In the Database Explorer, the database 'practica2' is selected, and the 'HABITACION' table is chosen. In the main pane, a query is being run:

```
1 ✓ SELECT * FROM PACIENTE;
2
3 ✓ SELECT * FROM HABITACION;
4
5 SELECT * FROM LOG_ACTIVIDAD;
6
7 SELECT * FROM LOG_HABITACION;
```

The output window displays the results of the query:

	idHabitacion	habitacion
1		

## → Tabla LOG\_ACTIVIDAD

The screenshot shows the MySQL Workbench interface. On the left, the Database Explorer pane displays the database structure under the 'localhost' connection, including the 'practica2' schema which contains tables like HABITACION, LOG\_ACTIVIDAD, LOG\_HABITACION, and PACIENTE. In the center, the 'PACIENTE' tab of the 'console\_2' editor window shows the following SQL code:

```
1 SELECT * FROM PACIENTE;
2
3 SELECT * FROM HABITACION;
4
5 ✓ SELECT * FROM LOG_ACTIVIDAD;
6
7 SELECT * FROM LOG_HABITACION;
```

The status bar at the bottom indicates the session is at row 5 of 5, with a timestamp of 22:06 on 24/04/2024.

## → Tabla LOG\_HABITACION

The screenshot shows the MySQL Workbench interface. The Database Explorer pane on the left shows the same database structure as the previous screenshot. In the center, the 'PACIENTE' tab of the 'console\_2' editor window shows the following SQL code:

```
1 SELECT * FROM PACIENTE;
2
3 SELECT * FROM HABITACION;
4
5 ✓ SELECT * FROM LOG_ACTIVIDAD;
6
7 ✓ SELECT * FROM LOG_HABITACION;
```

The status bar at the bottom indicates the session is at row 7 of 7, with a timestamp of 22:07 on 24/04/2024.

## SELECT COUNT(\*) FROM cada tabla

→ Tabla PACIENTE

The screenshot shows the MySQL Workbench interface. In the Database Explorer, there is a connection to 'localhost' with database 'practica2'. Under 'tables', the 'PACIENTE' table is selected. In the main pane, a query window titled 'console\_2' contains the following SQL code:

```
7 SELECT * FROM LOG_HABITACION;
8
9 ✓ SELECT COUNT(*) FROM PACIENTE;
10
11 SELECT COUNT(*) FROM HABITACION;
12
13 SELECT COUNT(*) FROM LOG_HABITACION;
14
15 SELECT COUNT(*) FROM LOG_ACTIVIDAD;
```

The output window shows the result of the second query:

COUNT(*)
154184

→ Tabla HABITACION

The screenshot shows the MySQL Workbench interface. In the Database Explorer, there is a connection to 'localhost' with database 'practica2'. Under 'tables', the 'HABITACION' table is selected. In the main pane, a query window titled 'console\_2' contains the following SQL code:

```
7 SELECT * FROM LOG_HABITACION;
8
9 ✓ SELECT COUNT(*) FROM PACIENTE;
10
11 ✓ SELECT COUNT(*) FROM HABITACION;
12
13 SELECT COUNT(*) FROM LOG_HABITACION;
14
15 SELECT COUNT(*) FROM LOG_ACTIVIDAD;
```

The output window shows the result of the second query:

COUNT(*)
0

## → Tabla LOG\_HABITACION

The screenshot shows the MySQL Workbench interface. In the Database Explorer, under the database 'practica2', there are four tables: HABITACION, LOG\_ACTIVIDAD, LOG\_HABITACION, and PACIENTE. The 'LOG\_HABITACION' table is selected. In the SQL Editor, the following SQL query is run:

```
SELECT COUNT(*) FROM PACIENTE;
SELECT COUNT(*) FROM HABITACION;
SELECT COUNT(*) FROM LOG_HABITACION;
SELECT COUNT(*) FROM LOG_ACTIVIDAD;
```

The output window shows the results of the first query:

Output	COUNT(*) int
1	'COUNT(*)' ▾ 6

## → Tabla LOG\_ACTIVIDAD

The screenshot shows the MySQL Workbench interface. In the Database Explorer, under the database 'practica2', there are four tables: HABITACION, LOG\_ACTIVIDAD, LOG\_HABITACION, and PACIENTE. The 'LOG\_ACTIVIDAD' table is selected. In the SQL Editor, the following SQL query is run:

```
SELECT COUNT(*) FROM PACIENTE;
SELECT COUNT(*) FROM HABITACION;
SELECT COUNT(*) FROM LOG_HABITACION;
SELECT COUNT(*) FROM LOG_ACTIVIDAD;
```

The output window shows the results of the first query:

Output	COUNT(*) int
1	'COUNT(*)' ▾ 6

## DIA 2

**SELECT \* FROM cada tabla**

→ Tabla PACIENTE

The screenshot shows the MySQL Workbench interface. In the Database Explorer, the 'practica2' database is selected, and the 'PACIENTE' table is chosen. In the SQL editor, the following query is written:

```
1 ✓ SELECT * FROM PACIENTE;
2
3 SELECT * FROM HABITACION;
4
5 SELECT * FROM LOG_ACTIVIDAD;
6
7 SELECT * FROM LOG_HABITACION;
8
9 SELECT COUNT(*) FROM PACIENTE;
10
11 SELECT COUNT(*) FROM HABITACION;
12
13 SELECT COUNT(*) FROM LOG_HABITACION;
14
15 SELECT COUNT(*) FROM LOG_ACTIVIDAD;
```

The output window displays the results of the 'PACIENTE' query:

idPaciente	t	edad	genero
1	100000	99 Otro	
2	100001	40 Femenino	
3	100002	42 Masculino	
4	100003	8 Femenino	
5	100004	88 Masculino	
6	100005	23 Masculino	
7	100006	60 Femenino	
8	100007	49 Femenino	
9	100008	44 Femenino	
10	100009	55 Femenino	
11	100010	28 Femenino	
12	100011	13 Femenino	
13	100012	62 Masculino	
14	100013	32 Masculino	
15	100014	11 Masculino	

At the bottom right of the interface, the date and time are shown as 25/08/2024 16:30.

→ Tabla HABITACION

The screenshot shows the MySQL Workbench interface. In the Database Explorer, the 'practica2' database is selected, and the 'HABITACION' table is chosen. In the SQL editor, the following query is written:

```
1 ✓ SELECT * FROM PACIENTE;
2
3 ✓ SELECT * FROM HABITACION;
4
5 SELECT * FROM LOG_ACTIVIDAD;
6
7 SELECT * FROM LOG_HABITACION;
8
9 SELECT COUNT(*) FROM PACIENTE;
10
11 SELECT COUNT(*) FROM HABITACION;
12
13 SELECT COUNT(*) FROM LOG_HABITACION;
14
15 SELECT COUNT(*) FROM LOG_ACTIVIDAD;
```

The output window displays the results of the 'HABITACION' query:

idHabitacion	habitacion
1	1 Sala de exámenes 1
2	2 Sala de exámenes 2
3	3 Sala de exámenes 3
4	4 Sala de exámenes 4
5	5 Sala de imágenes 1
6	6 Sala de procedimientos 1
7	7 Sala de procedimientos 2
8	8 Sala de procedimientos 3
9	9 Sala de procedimientos 4
10	10 Recepción
11	11 Laboratorio
12	12 Estación de revisión 1
13	13 Estación de revisión 2
14	14 Estación de revisión 3
15	15 Estación de revisión 4

At the bottom right of the interface, the date and time are shown as 25/08/2024 16:31.

## → Tabla LOG\_ACTIVIDAD

The screenshot shows the MySQL Workbench interface. In the Database Explorer, the schema 'practica2' is selected, and the table 'LOG\_ACTIVIDAD' is highlighted. The SQL editor contains the following code:

```
1 SELECT * FROM PACIENTE;
2
3 SELECT * FROM HABITACION;
4
5 ✓ SELECT * FROM LOG_ACTIVIDAD;
6
7 SELECT * FROM LOG_HABITACION;
8
9 SELECT COUNT(*) FROM PACIENTE;
10
11 SELECT COUNT(*) FROM HABITACION;
12
13 SELECT COUNT(*) FROM LOG_HABITACION;
14
15 SELECT COUNT(*) FROM LOG_ACTIVIDAD;
```

The output pane shows the results of the last query:

id_log_actividad	timestamp	actividad	idPaciente	idHabitacion
1	2024-08-25 10:30:00	Entrada	1	1

The status bar at the bottom indicates the session started at 16:32 on 25/08/2024.

## → Tabla LOG\_HABITACION

The screenshot shows the MySQL Workbench interface. In the Database Explorer, the schema 'practica2' is selected, and the table 'LOG\_HABITACION' is highlighted. The SQL editor contains the following code:

```
1 SELECT * FROM PACIENTE;
2
3 SELECT * FROM HABITACION;
4
5 ✓ SELECT * FROM LOG_ACTIVIDAD;
6
7 SELECT * FROM LOG_HABITACION;
8
9 SELECT COUNT(*) FROM PACIENTE;
10
11 SELECT COUNT(*) FROM HABITACION;
12
13 SELECT COUNT(*) FROM LOG_HABITACION;
14
15 SELECT COUNT(*) FROM LOG_ACTIVIDAD;
```

The output pane shows the results of the last query:

id_log_habitacion	timestamp	status	idHabitacion
1	2024-08-25 10:30:00	Entrada	1

The status bar at the bottom indicates the session started at 16:32 on 25/08/2024.

## SELECT COUNT(\*) FROM cada tabla

→ Tabla PACIENTE

The screenshot shows the MySQL Workbench interface. In the Database Explorer, the 'practica2' database is selected, and the 'tables' node is expanded to show 'HABITACION', 'LOG\_ACTIVIDAD', 'LOG\_HABITACION', and 'PACIENTE'. The 'PACIENTE' table is currently selected. In the main workspace, several SQL statements are listed:

```
1 SELECT * FROM PACIENTE;
2
3 SELECT * FROM HABITACION;
4
5 SELECT * FROM LOG_ACTIVIDAD;
6
7 SELECT * FROM LOG_HABITACION;
8
9 SELECT COUNT(*) FROM PACIENTE;
10
11 SELECT COUNT(*) FROM HABITACION;
12
13 SELECT COUNT(*) FROM LOG_HABITACION;
14
15 SELECT COUNT(*) FROM LOG_ACTIVIDAD;
```

The output pane shows the result of the last query:

COUNT(*)
154184

Below the workspace, the Services panel shows the status of various MySQL instances. The bottom status bar indicates the session is at row 9, column 1, with 30 characters, using CRLF line endings, and has 4 spaces.

→ Tabla HABITACION

The screenshot shows the MySQL Workbench interface. The 'practica2' database and 'HABITACION' table are selected in the Database Explorer. The main workspace contains the following SQL statements:

```
1 SELECT * FROM PACIENTE;
2
3 SELECT * FROM HABITACION;
4
5 SELECT * FROM LOG_ACTIVIDAD;
6
7 SELECT * FROM LOG_HABITACION;
8
9 SELECT COUNT(*) FROM PACIENTE;
10
11 SELECT COUNT(*) FROM HABITACION;
12
13 SELECT COUNT(*) FROM LOG_HABITACION;
14
15 SELECT COUNT(*) FROM LOG_ACTIVIDAD;
```

The output pane shows the result of the second query:

COUNT(*)
15

The bottom status bar indicates the session is at row 11, column 1, with 32 characters, using CRLF line endings, and has 4 spaces.

## → Tabla LOG\_HABITACION

The screenshot shows the MySQL Workbench interface. In the top-left pane, the Database Explorer displays the schema of the 'practica2' database, including tables like HABITACION, LOG\_ACTIVIDAD, LOG\_HABITACION, and PACIENTE. The top-right pane contains two SQL consoles: 'console\_1' and 'console\_2'. Console 2 is active and shows the following query:

```
1 SELECT * FROM PACIENTE;
2
3 SELECT * FROM HABITACION;
4
5 SELECT * FROM LOG_ACTIVIDAD;
6
7 SELECT * FROM LOG_HABITACION;
8
9 SELECT COUNT(*) FROM PACIENTE;
10
11 SELECT COUNT(*) FROM HABITACION;
12
13 SELECT COUNT(*) FROM LOG_HABITACION;
14
15 SELECT COUNT(*) FROM LOG_ACTIVIDAD;
```

The output pane below the consoles shows the results of the last query, which is 'SELECT COUNT(\*) FROM LOG\_HABITACION;'. The result is a single row with a value of 0.

## → Tabla LOG\_ACTIVIDAD

The screenshot shows the MySQL Workbench interface, similar to the previous one. The Database Explorer and top panes are identical. The active SQL console (console\_2) contains the following query:

```
1 SELECT * FROM PACIENTE;
2
3 SELECT * FROM HABITACION;
4
5 SELECT * FROM LOG_ACTIVIDAD;
6
7 SELECT * FROM LOG_HABITACION;
8
9 SELECT COUNT(*) FROM PACIENTE;
10
11 SELECT COUNT(*) FROM HABITACION;
12
13 SELECT COUNT(*) FROM LOG_HABITACION;
14
15 SELECT COUNT(*) FROM LOG_ACTIVIDAD;
```

The output pane shows the results of the last query, 'SELECT COUNT(\*) FROM LOG\_ACTIVIDAD;'. The result is a single row with a value of 0.

## DIA 3

**SELECT \* FROM cada tabla**

→ Tabla PACIENTE

The screenshot shows the MySQL Workbench interface with two consoles. The left console (console\_1) contains the following SQL code:

```
1 ✓ SELECT * FROM PACIENTE;
2
3 SELECT * FROM HABITACION;
4
5 SELECT * FROM LOG_ACTIVIDAD;
6
7 SELECT * FROM LOG_HABITACION;
8
9 SELECT COUNT(*) FROM PACIENTE;
10
11 SELECT COUNT(*) FROM HABITACION;
12
13 SELECT COUNT(*) FROM LOG_HABITACION;
14
15 SELECT COUNT(*) FROM LOG_ACTIVIDAD;
```

The right console (console\_2) displays the results of the first query:

	idPaciente	edad	genero
1	100000	95	Otro
2	100001	40	Femenino
3	100002	42	Masculino
4	100003	8	Femenino
5	100004	88	Masculino
6	100005	23	Masculino
7	100006	66	Femenino
8	100007	49	Femenino
9	100008	44	Femenino
10	100009	55	Femenino
11	100010	28	Femenino
12	100011	13	Femenino
13	100012	62	Masculino
14	100013	32	Masculino
15	100014	11	Masculino

→ Tabla HABITACION

The screenshot shows the MySQL Workbench interface with two consoles. The left console (console\_1) contains the following SQL code:

```
1 ✓ SELECT * FROM PACIENTE;
2
3 ✓ SELECT * FROM HABITACION;
4
5 SELECT * FROM LOG_ACTIVIDAD;
6
7 SELECT * FROM LOG_HABITACION;
8
9 SELECT COUNT(*) FROM PACIENTE;
10
11 SELECT COUNT(*) FROM HABITACION;
12
13 SELECT COUNT(*) FROM LOG_HABITACION;
14
15 SELECT COUNT(*) FROM LOG_ACTIVIDAD;
```

The right console (console\_2) displays the results of the second query:

	idHabitacion	habitacion
1	1	Sala de exámenes 1
2	2	Sala de exámenes 2
3	3	Sala de exámenes 3
4	4	Sala de exámenes 4
5	5	Sala de imágenes 1
6	6	Sala de procedimientos 1
7	7	Sala de procedimientos 2
8	8	Sala de procedimientos 3
9	9	Sala de procedimientos 4
10	10	Recepción
11	11	Laboratorio
12	12	Estación de revisión 1
13	13	Estación de revisión 2
14	14	Estación de revisión 3
15	15	Estación de revisión 4

## → Tabla LOG\_ACTIVIDAD

The screenshot shows the MySQL Workbench interface. In the Database Explorer, the schema 'practica2' is selected. A new table 'LOG\_ACTIVIDAD' is being created with the following structure:

```
CREATE TABLE LOG_ACTIVIDAD (
    id_log_actividad INT AUTO_INCREMENT,
    timestamp DATETIME,
    actividad VARCHAR(255),
    idPaciente INT,
    idHabitacion INT
)
```

The table has four columns: 'id\_log\_actividad' (primary key, auto-increment), 'timestamp' (DATETIME), 'actividad' (VARCHAR(255)), and 'idPaciente' (INT). The 'LOG\_ACTIVIDAD' table is highlighted in the list of tables.

## → Tabla LOG\_HABITACION

The screenshot shows the MySQL Workbench interface. In the Database Explorer, the schema 'practica2' is selected. A new table 'LOG\_HABITACION' is being created with the following structure:

```
CREATE TABLE LOG_HABITACION (
    id_log_habitacion INT AUTO_INCREMENT,
    timestamp DATETIME,
    statusx VARCHAR(255),
    idHabitacion INT
)
```

The table has three columns: 'id\_log\_habitacion' (primary key, auto-increment), 'timestamp' (DATETIME), and 'statusx' (VARCHAR(255)). The 'LOG\_HABITACION' table is highlighted in the list of tables.

Below the table creation, a table named 'log\_habitacion' is shown with 15 rows of data:

id_log_habitacion	timestamp	statusx	idHabitacion
1	5/31/2021 7:19:33 a.m.	Sala no disponible.	12
2	5/31/2021 7:20:00 a.m.	Inicia limpieza.	12
3	5/31/2021 7:20:35 a.m.	Sala disponible.	12
4	5/31/2021 7:29:15 a.m.	Sala no disponible.	12
5	5/31/2021 7:29:42 a.m.	Inicia limpieza.	12
6	5/31/2021 7:30:31 a.m.	Sala disponible.	12
7	5/31/2021 7:36:38 a.m.	Sala no disponible.	13
8	5/31/2021 7:37:05 a.m.	Inicia limpieza.	13
9	5/31/2021 7:38:53 a.m.	Sala disponible.	13
10	5/31/2021 7:42:34 a.m.	Sala no disponible.	1
11	5/31/2021 7:43:14 a.m.	Inicia limpieza.	1
12	5/31/2021 7:46:07 a.m.	Sala disponible.	1
13	5/31/2021 7:54:19 a.m.	Sala no disponible.	2
14	5/31/2021 7:54:53 a.m.	Inicia limpieza.	2
15	5/31/2021 7:55:43 a.m.	Sala no disponible.	12

**SELECT COUNT(\*) FROM cada tabla**

→ Tabla PACIENTE

The screenshot shows the MySQL Workbench interface with two consoles: console\_1 and console\_2. The Database Explorer sidebar shows a database named 'practica2' with tables: HABITACION, LOG\_ACTIVIDAD, LOG\_HABITACION, and PACIENTE. The consoles contain the following SQL code:

```
1 SELECT * FROM PACIENTE;
2
3 SELECT * FROM HABITACION;
4
5 SELECT * FROM LOG_ACTIVIDAD;
6
7 SELECT * FROM LOG_HABITACION;
8
9 SELECT COUNT(*) FROM PACIENTE;
10
11 SELECT COUNT(*) FROM HABITACION;
12
13 SELECT COUNT(*) FROM LOG_HABITACION;
14
15 SELECT COUNT(*) FROM LOG_ACTIVIDAD;
```

The output pane for console\_2 shows the results of the COUNT(\*) query for the PACIENTE table:

Output	COUNT(*):int
1 COUNT(*)`	154184

→ Tabla HABITACION

The screenshot shows the MySQL Workbench interface with two consoles: console\_1 and console\_2. The Database Explorer sidebar shows a database named 'practica2' with tables: HABITACION, LOG\_ACTIVIDAD, LOG\_HABITACION, and PACIENTE. The consoles contain the following SQL code:

```
1 SELECT * FROM PACIENTE;
2
3 SELECT * FROM HABITACION;
4
5 SELECT * FROM LOG_ACTIVIDAD;
6
7 SELECT * FROM LOG_HABITACION;
8
9 SELECT COUNT(*) FROM PACIENTE;
10
11 SELECT COUNT(*) FROM HABITACION;
12
13 SELECT COUNT(*) FROM LOG_HABITACION;
14
15 SELECT COUNT(*) FROM LOG_ACTIVIDAD;
```

The output pane for console\_2 shows the results of the COUNT(\*) query for the HABITACION table:

Output	COUNT(*):int
1 COUNT(*)`	15

## → Tabla LOG\_HABITACION

The screenshot shows the MySQL Workbench interface. In the Database Explorer, the schema 'practica2' is selected, and the table 'LOG\_HABITACION' is highlighted. In the top-right query editor, the following SQL code is run:

```
1 SELECT * FROM PACIENTE;
2
3 SELECT * FROM HABITACION;
4
5 SELECT * FROM LOG_ACTIVIDAD;
6
7 SELECT * FROM LOG_HABITACION;
8
9 SELECT COUNT(*) FROM PACIENTE;
10
11 SELECT COUNT(*) FROM HABITACION;
12
13 SELECT COUNT(*) FROM LOG_HABITACION;
14
15 SELECT COUNT(*) FROM LOG_ACTIVIDAD;
```

The output window shows the result of the last query:

Count(*)
34617

At the bottom right of the interface, the timestamp is 19:19 and the date is 26/08/2024.

## → Tabla LOG\_ACTIVIDAD

The screenshot shows the MySQL Workbench interface. In the Database Explorer, the schema 'practica2' is selected, and the table 'LOG\_ACTIVIDAD' is highlighted. In the top-right query editor, the following SQL code is run:

```
1 SELECT * FROM PACIENTE;
2
3 SELECT * FROM HABITACION;
4
5 SELECT * FROM LOG_ACTIVIDAD;
6
7 SELECT * FROM LOG_HABITACION;
8
9 SELECT COUNT(*) FROM PACIENTE;
10
11 SELECT COUNT(*) FROM HABITACION;
12
13 SELECT COUNT(*) FROM LOG_HABITACION;
14
15 SELECT COUNT(*) FROM LOG_ACTIVIDAD;
```

The output window shows the result of the last query:

Count(*)
0

At the bottom right of the interface, the timestamp is 19:19 and the date is 26/08/2024.

## DIA 4

**SELECT \* FROM cada tabla**

→ Tabla PACIENTE

The screenshot shows the MySQL Workbench interface. In the Database Explorer, the database 'practica2' is selected, and the table 'PACIENTE' is highlighted. In the top-right console window, the following SQL query is run:

```
1 ✓ SELECT * FROM PACIENTE;
```

The output window displays the results of the query:

	idPaciente	edad	genero
1	100000	95	Otro
2	100001	40	Femenino
3	100002	42	Masculino
4	100003	8	Femenino
5	100004	88	Masculino
6	100005	23	Masculino
7	100006	66	Femenino
8	100007	49	Femenino
9	100008	44	Femenino
10	100009	55	Femenino
11	100010	28	Femenino
12	100011	13	Femenino
13	100012	62	Masculino
14	100013	32	Masculino
15	100014	11	Masculino

→ Tabla HABITACION

The screenshot shows the MySQL Workbench interface. In the Database Explorer, the database 'practica2' is selected, and the table 'HABITACION' is highlighted. In the top-right console window, the following SQL query is run:

```
1 ✓ SELECT * FROM PACIENTE;
```

```
3 ✓ SELECT * FROM HABITACION;
```

The output window displays the results of the query:

	idHabitacion	habitacion
1	1	Sala de exámenes 1
2	2	Sala de exámenes 2
3	3	Sala de exámenes 3
4	4	Sala de exámenes 4
5	5	Sala de imágenes 1
6	6	Sala de procedimientos 1
7	7	Sala de procedimientos 2
8	8	Sala de procedimientos 3
9	9	Sala de procedimientos 4
10	10	Recepción
11	11	Laboratorio
12	12	Estación de revisión 1
13	13	Estación de revisión 2
14	14	Estación de revisión 3
15	15	Estación de revisión 4

## → Tabla LOG\_ACTIVIDAD

The screenshot shows the MySQL Workbench interface with the Database Explorer on the left and two consoles at the top. Console 2 is active and displays the following SQL query and its results:

```

SELECT * FROM PACIENTE;
SELECT * FROM HABITACION;
SELECT * FROM LOG_ACTIVIDAD;
SELECT * FROM LOG_HABITACION;
SELECT COUNT(*) FROM PACIENTE;
SELECT COUNT(*) FROM HABITACION;
SELECT COUNT(*) FROM LOG_HABITACION;
SELECT COUNT(*) FROM LOG_ACTIVIDAD;

```

The results table for the LOG\_ACTIVIDAD query contains 15 rows of log entries:

	timestamp	actividad	idPaciente	idHabitacion
1	5/31/2021 7:12:44 a.m.	Paciente inicia el registro.	134247	10
2	5/31/2021 7:13:16 a.m.	Paciente recibe papeleria en recepcion.	134247	10
3	5/31/2021 7:15:54 a.m.	Paciente entrega papeleria.	134247	10
4	5/31/2021 7:16:33 a.m.	Recepcionista establece la condicion del paciente como 'Urgencia'.	134247	10
5	5/31/2021 7:17:47 a.m.	Enfermera comienza la revision del paciente.	134247	10
6	5/31/2021 7:19:33 a.m.	Revision determino que el paciente es tipo 2 y su condicion es 'Estable'.	134247	10
7	5/31/2021 7:20:34 a.m.	Medico inicia con el tratamiento del paciente.	134247	10
8	5/31/2021 7:21:50 a.m.	Paciente inicia el registro.	135641	10
9	5/31/2021 7:23:01 a.m.	Paciente recibe papeleria en recepcion.	135641	10
10	5/31/2021 7:23:39 a.m.	Paciente inicia el registro.	180487	10
11	5/31/2021 7:25:44 a.m.	Paciente entrega papeleria.	135641	12
12	5/31/2021 7:26:18 a.m.	Recepcionista establece la condicion del paciente como 'Estable'.	135641	12
13	5/31/2021 7:26:43 a.m.	Paciente recibe papeleria en recepcion.	180487	5
14	5/31/2021 7:27:33 a.m.	Enfermera comienza la revision del paciente.	135641	5
15	5/31/2021 7:28:12 a.m.	Paciente entrega papeleria.	180487	7

## → Tabla LOG\_HABITACION

The screenshot shows the MySQL Workbench interface with the Database Explorer on the left and two consoles at the top. Console 2 is active and displays the following SQL query and its results:

```

SELECT * FROM PACIENTE;
SELECT * FROM HABITACION;
SELECT * FROM LOG_ACTIVIDAD;
SELECT * FROM LOG_HABITACION;
SELECT COUNT(*) FROM PACIENTE;
SELECT COUNT(*) FROM HABITACION;
SELECT COUNT(*) FROM LOG_HABITACION;
SELECT COUNT(*) FROM LOG_ACTIVIDAD;

```

The results table for the LOG\_HABITACION query contains 15 rows of log entries:

	timestamp	status	idHabitacion
1	5/31/2021 7:19:33 a.m.	Sala no disponible.	12
2	5/31/2021 7:20:35 a.m.	Inicia limpieza.	12
3	5/31/2021 7:29:15 a.m.	Sala no disponible.	12
4	5/31/2021 7:29:42 a.m.	Inicia limpieza.	12
5	5/31/2021 7:30:31 a.m.	Sala disponible.	12
6	5/31/2021 7:36:38 a.m.	Sala no disponible.	13
7	5/31/2021 7:37:05 a.m.	Inicia limpieza.	13
8	5/31/2021 7:38:53 a.m.	Sala disponible.	13
9	5/31/2021 7:42:34 a.m.	Sala no disponible.	1
10	5/31/2021 7:43:14 a.m.	Inicia limpieza.	1
11	5/31/2021 7:46:07 a.m.	Sala disponible.	1
12	5/31/2021 7:54:19 a.m.	Sala no disponible.	2
13	5/31/2021 7:54:53 a.m.	Inicia limpieza.	2
14	5/31/2021 7:55:43 a.m.	Sala no disponible.	12

**SELECT COUNT(\*) FROM cada tabla**

→ Tabla PACIENTE

The screenshot shows the MySQL Workbench interface with two consoles. Console 2 is active and displays the following SQL query and its result:

```
1 SELECT * FROM PACIENTE;
2
3 SELECT * FROM HABITACION;
4
5 SELECT * FROM LOG_ACTIVIDAD;
6
7 SELECT * FROM LOG_HABITACION;
8
9 SELECT COUNT(*) FROM PACIENTE;
10
11 SELECT COUNT(*) FROM HABITACION;
12
13 SELECT COUNT(*) FROM LOG_HABITACION;
14
15 SELECT COUNT(*) FROM LOG_ACTIVIDAD;
```

The output window shows the result of the last query:

Count(*)
154184

→ Tabla HABITACION

The screenshot shows the MySQL Workbench interface with two consoles. Console 2 is active and displays the following SQL query and its result:

```
1 SELECT * FROM PACIENTE;
2
3 SELECT * FROM HABITACION;
4
5 SELECT * FROM LOG_ACTIVIDAD;
6
7 SELECT * FROM LOG_HABITACION;
8
9 SELECT COUNT(*) FROM PACIENTE;
10
11 SELECT COUNT(*) FROM HABITACION;
12
13 SELECT COUNT(*) FROM LOG_HABITACION;
14
15 SELECT COUNT(*) FROM LOG_ACTIVIDAD;
```

The output window shows the result of the last query:

Count(*)
15

## → Tabla LOG\_HABITACION

The screenshot shows the MySQL Workbench interface. In the top-left pane, the Database Explorer displays the schema of the 'practica2' database, including tables like PACIENTE, HABITACION, LOG\_ACTIVIDAD, LOG\_HABITACION, and LOG\_ACTIVIDAD. In the top-right pane, the 'LOG\_HABITACION' tab of the 'console\_2' connection shows the following SQL code:

```
1 SELECT * FROM PACIENTE;
2
3 SELECT * FROM HABITACION;
4
5 SELECT * FROM LOG_ACTIVIDAD;
6
7 SELECT * FROM LOG_HABITACION;
8
9 SELECT COUNT(*) FROM PACIENTE;
10
11 SELECT COUNT(*) FROM HABITACION;
12
13 SELECT COUNT(*) FROM LOG_HABITACION;
14
15 SELECT COUNT(*) FROM LOG_ACTIVIDAD;
```

Below the code, the 'Output' tab shows the result of the last query:

Count(*)
34617

The bottom status bar indicates the session is at 13:1 (36 chars) CRLF UTF-8 4 spaces, and the date and time are 27/08/2024 21:39.

## → Tabla LOG\_ACTIVIDAD

The screenshot shows the MySQL Workbench interface. In the top-left pane, the Database Explorer displays the schema of the 'practica2' database. In the top-right pane, the 'LOG\_ACTIVIDAD' tab of the 'console\_2' connection shows the following SQL code:

```
1 SELECT * FROM PACIENTE;
2
3 SELECT * FROM HABITACION;
4
5 SELECT * FROM LOG_ACTIVIDAD;
6
7 SELECT * FROM LOG_HABITACION;
8
9 SELECT COUNT(*) FROM PACIENTE;
10
11 SELECT COUNT(*) FROM HABITACION;
12
13 SELECT COUNT(*) FROM LOG_HABITACION;
14
15 SELECT COUNT(*) FROM LOG_ACTIVIDAD;
```

Below the code, the 'Output' tab shows the result of the last query:

Count(*)
33841

The bottom status bar indicates the session is at 15:1 (35 chars) CRLF UTF-8 4 spaces, and the date and time are 27/08/2024 21:40.

## DIA 5

**SELECT \* FROM cada tabla**

→ Tabla PACIENTE

The screenshot shows the MySQL Workbench interface. In the Database Explorer, the database 'practica2' is selected, and the table 'PACIENTE' is highlighted under the 'HABITACION' schema. In the top-right console window, the following SQL query is run:

```
1 ✓  SELECT * FROM PACIENTE;
```

The output window displays the results of the query:

idPaciente	edad	genero
100000	95	Otro
100001	40	Femenino
100002	42	Masculino
100003	8	Femenino
100004	88	Masculino
100005	23	Masculino
100006	66	Femenino
100007	49	Femenino
100008	44	Femenino
100009	55	Femenino
100010	28	Femenino
100011	13	Femenino
100012	62	Masculino
100013	32	Masculino
100014	11	Masculino

→ Tabla HABITACION

The screenshot shows the MySQL Workbench interface. In the Database Explorer, the database 'practica2' is selected, and the table 'HABITACION' is highlighted under the 'LOG\_ACTIVIDAD' schema. In the top-right console window, the following SQL query is run:

```
1 ✓  SELECT * FROM HABITACION;
```

The output window displays the results of the query:

idHabitacion	habitacion
1	1 Sala de exámenes 1
2	2 Sala de exámenes 2
3	3 Sala de exámenes 3
4	4 Sala de exámenes 4
5	5 Sala de imágenes 1
6	6 Sala de procedimientos 1
7	7 Sala de procedimientos 2
8	8 Sala de procedimientos 3
9	9 Sala de procedimientos 4
10	10 Recepción
11	11 Laboratorio
12	12 Estación de revisión 1
13	13 Estación de revisión 2
14	14 Estación de revisión 3
15	15 Estación de revisión 4

## → Tabla LOG\_ACTIVIDAD

The screenshot shows the MySQL Workbench interface with two consoles. Console 1 is connected to the database 'practica2' and displays the following SQL query:

```

SELECT * FROM PACIENTE;
SELECT * FROM HABITACION;
SELECT * FROM LOG_ACTIVIDAD;
SELECT * FROM LOG_HABITACION;
SELECT COUNT(*) FROM PACIENTE;
SELECT COUNT(*) FROM HABITACION;
SELECT COUNT(*) FROM LOG_HABITACION;
SELECT COUNT(*) FROM LOG_ACTIVIDAD;

```

Console 2 shows the results of the last query, which is a table named 'practica2.LOG\_ACTIVIDAD'. The table has columns: id\_log\_actividad, timestamp, Actividad, idPaciente, and idHabitacion. The data is as follows:

id_log_actividad	timestamp	Actividad	idPaciente	idHabitacion
1	5/31/2021 7:22:14 a.m.	Paciente inicia el registro.	134247	10
2	5/31/2021 7:23:16 a.m.	Paciente recibe papelera en recepcion.	134247	10
3	5/31/2021 7:25:54 a.m.	Paciente entrega papelera.	134247	10
4	5/31/2021 7:26:33 a.m.	Repcionista establece la condicion del paciente como 'Urgencia'.	134247	10
5	5/31/2021 7:27:47 a.m.	Enfermera comienza la revision del paciente.	134247	10
6	5/31/2021 7:29:33 a.m.	Revision determino que el paciente es tipo 2 y su condicion es 'Estable'.	134247	10
7	5/31/2021 7:29:34 a.m.	Medico inicia con el tratamiento del paciente.	134247	10
8	5/31/2021 7:21:50 a.m.	Paciente inicia el registro.	135641	10
9	5/31/2021 7:23:01 a.m.	Paciente recibe papelera en recepcion.	135641	10
10	5/31/2021 7:23:39 a.m.	Paciente inicia el registro.	188087	10
11	5/31/2021 7:25:44 a.m.	Paciente entrega papelera.	135641	12
12	5/31/2021 7:26:18 a.m.	Repcionista establece la condicion del paciente como 'Estable'.	135641	12
13	5/31/2021 7:26:43 a.m.	Paciente recibe papelera en recepcion.	188087	5
14	5/31/2021 7:27:33 a.m.	Enfermera comienza la revision del paciente.	135641	5
15	5/31/2021 7:29:12 a.m.	Paciente entrega papelera.	188087	7

## → Tabla LOG\_HABITACION

The screenshot shows the MySQL Workbench interface with two consoles. Console 1 is connected to the database 'practica2' and displays the following SQL query:

```

SELECT * FROM PACIENTE;
SELECT * FROM HABITACION;
SELECT * FROM LOG_ACTIVIDAD;
SELECT * FROM LOG_HABITACION;
SELECT COUNT(*) FROM PACIENTE;
SELECT COUNT(*) FROM HABITACION;
SELECT COUNT(*) FROM LOG_HABITACION;
SELECT COUNT(*) FROM LOG_ACTIVIDAD;

```

Console 2 shows the results of the last query, which is a table named 'practica2.LOG\_HABITACION'. The table has columns: id\_log\_habitacion, timestamp, status, and idHabitacion. The data is as follows:

id_log_habitacion	timestamp	status	idHabitacion
1	5/31/2021 7:19:33 a.m.	Sala no disponible.	12
2	5/31/2021 7:20:00 a.m.	Inicia limpieza.	12
3	5/31/2021 7:20:35 a.m.	Sala disponible.	12
4	5/31/2021 7:29:15 a.m.	Sala no disponible.	12
5	5/31/2021 7:29:42 a.m.	Inicia limpieza.	12
6	5/31/2021 7:30:31 a.m.	Sala disponible.	12
7	5/31/2021 7:36:38 a.m.	Sala no disponible.	13
8	5/31/2021 7:37:05 a.m.	Inicia limpieza.	13
9	5/31/2021 7:38:53 a.m.	Sala disponible.	13
10	5/31/2021 7:42:34 a.m.	Sala no disponible.	1
11	5/31/2021 7:43:14 a.m.	Inicia limpieza.	1
12	5/31/2021 7:46:07 a.m.	Sala disponible.	1
13	5/31/2021 7:54:19 a.m.	Sala no disponible.	2
14	5/31/2021 7:54:53 a.m.	Inicia limpieza.	2
15	5/31/2021 7:55:43 a.m.	Sala no disponible.	12

**SELECT COUNT(\*) FROM cada tabla**

→ Tabla PACIENTE

The screenshot shows the MySQL Workbench interface with two consoles open: 'console\_1' and 'console\_2'. The 'Database Explorer' sidebar shows a database named 'practica2' with four tables: HABITACION, LOG\_ACTIVIDAD, LOG\_HABITACION, and PACIENTE. The 'Services' sidebar shows a connection to 'localhost' with a default schema. The 'Output' pane for 'console\_2' displays the result of the query 'SELECT COUNT(\*) FROM PACIENTE;' which returns a single row with the value 154184.

```
1 SELECT * FROM PACIENTE;
2
3 SELECT * FROM HABITACION;
4
5 SELECT * FROM LOG_ACTIVIDAD;
6
7 SELECT * FROM LOG_HABITACION;
8
9 ✓ SELECT COUNT(*) FROM PACIENTE;
10
11 SELECT COUNT(*) FROM HABITACION;
12
13 SELECT COUNT(*) FROM LOG_HABITACION;
14
15 SELECT COUNT(*) FROM LOG_ACTIVIDAD;
```

Output COUNT(\*):int  
1 COUNT(\*)  
1 154184

→ Tabla HABITACION

The screenshot shows the MySQL Workbench interface with two consoles open: 'console\_1' and 'console\_2'. The 'Database Explorer' sidebar shows a database named 'practica2' with four tables: HABITACION, LOG\_ACTIVIDAD, LOG\_HABITACION, and PACIENTE. The 'Services' sidebar shows a connection to 'localhost' with a default schema. The 'Output' pane for 'console\_2' displays the result of the query 'SELECT COUNT(\*) FROM HABITACION;' which returns a single row with the value 15.

```
1 SELECT * FROM PACIENTE;
2
3 SELECT * FROM HABITACION;
4
5 SELECT * FROM LOG_ACTIVIDAD;
6
7 SELECT * FROM LOG_HABITACION;
8
9 ✓ SELECT COUNT(*) FROM PACIENTE;
10
11 ✓ SELECT COUNT(*) FROM HABITACION;
12
13 SELECT COUNT(*) FROM LOG_HABITACION;
14
15 SELECT COUNT(*) FROM LOG_ACTIVIDAD;
```

Output COUNT(\*):int  
1 COUNT(\*)  
1 15

## → Tabla LOG\_HABITACION

The screenshot shows the MySQL Workbench interface. In the Database Explorer, the schema 'practica2' is selected, and the table 'LOG\_HABITACION' is highlighted. In the SQL editor (console\_2), the following query is run:

```
1 SELECT * FROM PACIENTE;
2
3 SELECT * FROM HABITACION;
4
5 SELECT * FROM LOG_ACTIVIDAD;
6
7 SELECT * FROM LOG_HABITACION;
8
9 SELECT COUNT(*) FROM PACIENTE;
10
11 SELECT COUNT(*) FROM HABITACION;
12
13 SELECT COUNT(*) FROM LOG_HABITACION;
14
15 SELECT COUNT(*) FROM LOG_ACTIVIDAD;
```

The output window shows the result of the last query:

Count(*)
34617

## → Tabla LOG\_ACTIVIDAD

The screenshot shows the MySQL Workbench interface. In the Database Explorer, the schema 'practica2' is selected, and the table 'LOG\_ACTIVIDAD' is highlighted. In the SQL editor (console\_2), the following query is run:

```
1 SELECT * FROM PACIENTE;
2
3 SELECT * FROM HABITACION;
4
5 SELECT * FROM LOG_ACTIVIDAD;
6
7 SELECT * FROM LOG_HABITACION;
8
9 SELECT COUNT(*) FROM PACIENTE;
10
11 SELECT COUNT(*) FROM HABITACION;
12
13 SELECT COUNT(*) FROM LOG_HABITACION;
14
15 SELECT COUNT(*) FROM LOG_ACTIVIDAD;
```

The output window shows the result of the last query:

Count(*)
67684

## DIA 6

### Tiempo que duró el el full backup.

```
C:\Users\Jorge\Documents\usac\segundo-semestre-2024\bases\lab\repo\practica2\scripts\backup_completos (main -> origin)
\ echo %time%
22:24:52.78

C:\Users\Jorge\Documents\usac\segundo-semestre-2024\bases\lab\repo\practica2\scripts\backup_completos (main -> origin)
\ mysql -u root -padmin practica2 < day1_backup_completo.sql
mysql: [Warning] Using a password on the command line interface can be insecure.

C:\Users\Jorge\Documents\usac\segundo-semestre-2024\bases\lab\repo\practica2\scripts\backup_completos (main -> origin)
\ echo %time%
22:24:58.88

C:\Users\Jorge\Documents\usac\segundo-semestre-2024\bases\lab\repo\practica2\scripts\backup_completos (main -> origin)
\ |
```

### SELECT \* FROM cada tabla

#### → Tabla HABITACION

The screenshot shows the MySQL Workbench interface. In the top-left pane, there is a query editor window containing the following SQL code:

```
use practica2;
select * from habitacion;
select * from paciente;
select * from log_actividad;
select * from log_habitacion;

select count(*) from habitacion;
select count(*) from paciente;
select count(*) from log_actividad;
select count(*) from log_habitacion;
```

In the bottom-right pane, there is a results grid titled "habitacion 1". The grid has three columns: "idHabitacion" (with value 1), "habitacion" (with value 1), and "paciente" (with value 1). The grid also includes standard database navigation buttons like Refresh, Save, Cancel, and Export data.

#### → Tabla PACIENTE

The screenshot shows the MySQL Workbench interface. In the top-left pane, there is a query editor window containing the following SQL code:

```
use practica2;
select * from habitacion;
select * from paciente;
select * from log_actividad;
select * from log_habitacion;

select count(*) from habitacion;
select count(*) from paciente;
select count(*) from log_actividad;
select count(*) from log_habitacion;
```

In the bottom-right pane, there is a results grid titled "paciente 1". The grid has four columns: "idPaciente" (values 1 through 15), "edad" (values 95 down to 11), "genero" (values Otro, Femenino, Masculino, and various numerical values), and "habitacion" (values 1 through 15). The grid also includes standard database navigation buttons like Refresh, Save, Cancel, and Export data.

## → Tabla LOG\_ACTIVIDAD

The screenshot shows the MySQL Workbench interface with the 'log\_actividad' table selected. The table has four columns: id\_log\_actividad, acc\_timestampx, acc\_actividad, and idPaciente. The results grid is empty, indicating no data is present.

```
use practica2;
select * from habitacion;
select * from paciente;
select * from log_actividad;
select * from log_habitacion;

select count(*) from habitacion;
select count(*) from paciente;
select count(*) from log_actividad;
select count(*) from log_habitacion;
```

## → Tabla LOG\_HABITACION

The screenshot shows the MySQL Workbench interface with the 'log\_habitacion' table selected. The table has four columns: id\_log\_habitacion, acc\_timestampx, acc\_statusx, and idHabitacion. The results grid is empty, indicating no data is present.

```
use practica2;
select * from habitacion;
select * from paciente;
select * from log_actividad;
select * from log_habitacion;

select count(*) from habitacion;
select count(*) from paciente;
select count(*) from log_actividad;
select count(*) from log_habitacion;
```

**SELECT COUNT(\*) FROM cada tabla;**

## → Tabla HABITACION

The screenshot shows the MySQL Workbench interface with the results of a COUNT(\*) query for the 'habitacion' table. The results grid shows one row with a value of 0.

```
select * from log_habitacion;
select count(*) from habitacion;
select count(*) from paciente;
select count(*) from log_actividad;
select count(*) from log_habitacion;
```

count(*)
0

## → Tabla PACIENTE

```
use practicas;
select * from habitacion;
select * from paciente;
select * from log_actividad;
select * from log_habitacion;
...
select count(*) from habitacion;
select count(*) from paciente;
select count(*) from log_actividad;
select count(*) from log_habitacion;
```

Results 1 ×  
Enter a SQL expression to filter results (use Ctrl+Space)

Grid	Text
1	154,184

Refresh Save Cancel Export data 200 CST en Writable Smart Insert Set: 32 | 2 Desktop 22:26 29/08/2024

## → Tabla LOG\_ACTIVIDAD

```
use practicas;
select * from habitacion;
select count(*) from paciente;
select count(*) from log_actividad;
select count(*) from log_habitacion;
```

Results 1 ×  
Enter a SQL expression to filter results (use Ctrl+Space)

Grid	Text
1	0

Refresh Save Cancel Export data 200 CST en Writable Smart Insert Set: 37 | 2 Desktop 22:27 29/08/2024

## → Tabla LOG\_HABITACION

```
use practicas;
select * from paciente;
select * from log_actividad;
select * from log_habitacion;
...
select count(*) from habitacion;
select count(*) from paciente;
select count(*) from log_actividad;
select count(*) from log_habitacion;
```

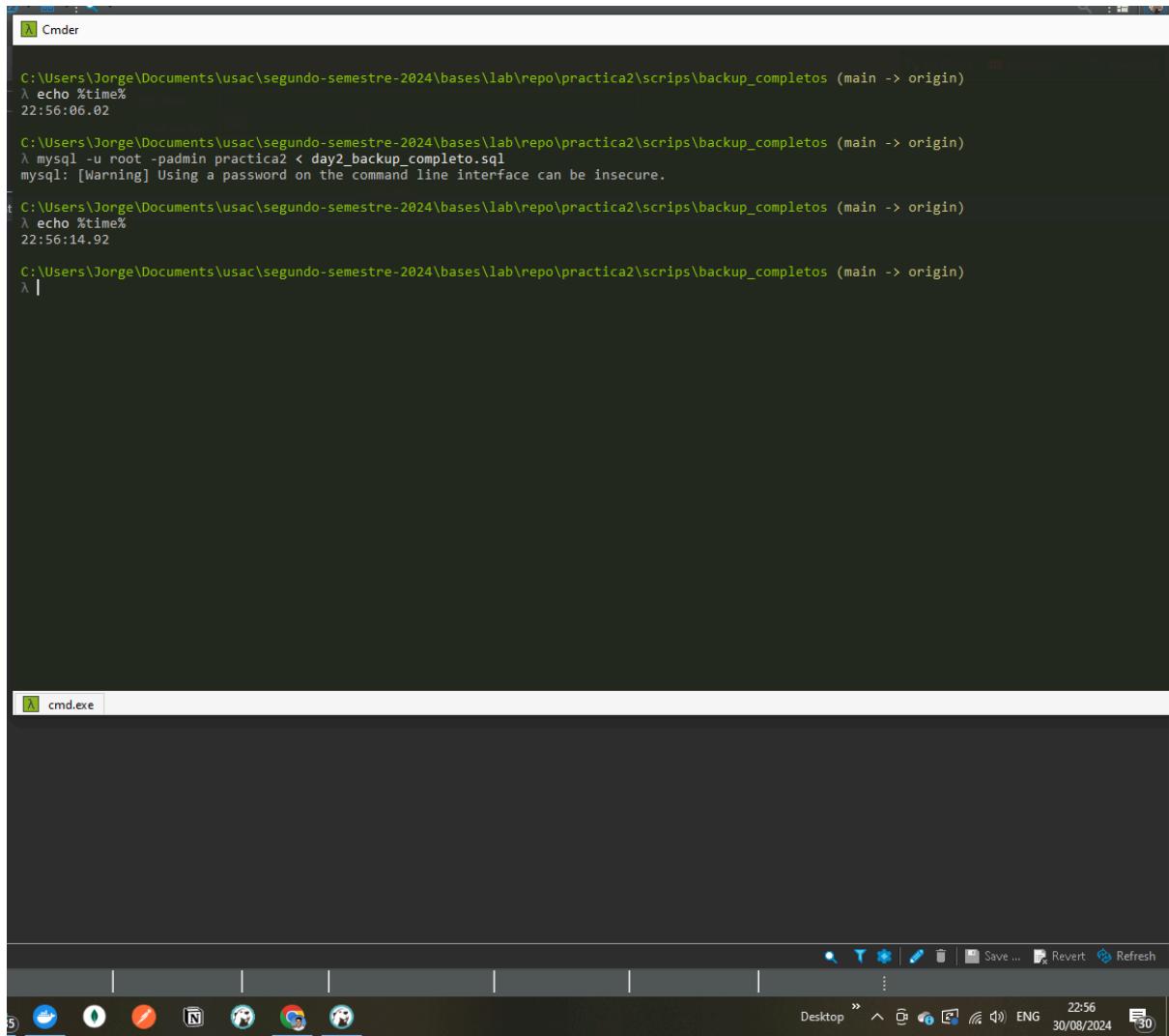
Results 1 ×  
Enter a SQL expression to filter results (use Ctrl+Space)

Grid	Text
1	0

Refresh Save Cancel Export data 200 CST en Writable Smart Insert Set: 38 | 2 Desktop 22:27 29/08/2024

## DIA 7

### Tiempo que duró el full backup



```
C:\Users\Jorge\Documents\usac\segundo-semestre-2024\bases\lab\repo\practica2\scripts\backup_completos (main -> origin)
λ echo %time%
22:56:06.02

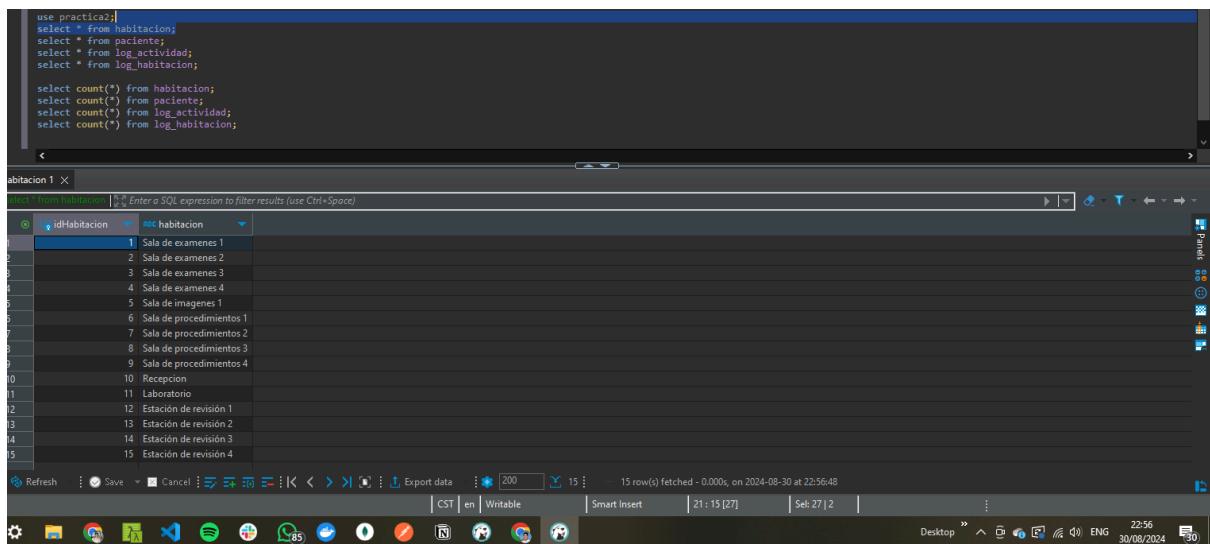
C:\Users\Jorge\Documents\usac\segundo-semestre-2024\bases\lab\repo\practica2\scripts\backup_completos (main -> origin)
λ mysql -u root -padmin practica2 < day2_backup_completo.sql
mysql: [Warning] Using a password on the command line interface can be insecure.

t C:\Users\Jorge\Documents\usac\segundo-semestre-2024\bases\lab\repo\practica2\scripts\backup_completos (main -> origin)
λ echo %time%
22:56:14.92

C:\Users\Jorge\Documents\usac\segundo-semestre-2024\bases\lab\repo\practica2\scripts\backup_completos (main -> origin)
λ |
```

**SELECT \* FROM cada tabla**

→ Tabla HABITACION



```
use practica2;
select * from habitacion;
select * from paciente;
select * from log_actividad;
select * from log_habitacion;

select count(*) from habitacion;
select count(*) from paciente;
select count(*) from log_actividad;
select count(*) from log_habitacion;
```

idHabitacion	nombre
1	Sala de exámenes 1
2	Sala de exámenes 2
3	Sala de exámenes 3
4	Sala de exámenes 4
5	Sala de imágenes 1
6	Sala de procedimientos 1
7	Sala de procedimientos 2
8	Sala de procedimientos 3
9	Sala de procedimientos 4
10	Recepción
11	Laboratorio
12	Estación de revisión 1
13	Estación de revisión 2
14	Estación de revisión 3
15	Estación de revisión 4

## → Tabla PACIENTE

```
use practica2;
select * from habitacion;
select * from paciente;
select * from log_actividad;
select * from log_habitacion;

select count(*) from habitacion;
select count(*) from paciente;
select count(*) from log_actividad;
select count(*) from log_habitacion;
```

Paciente 1 × Enter a SQL expression to filter results (use Ctrl+Space)

idPaciente	edad	genero
100,000	95	Otro
100,001	40	Femenino
100,002	42	Masculino
100,003	8	Femenino
100,004	88	Masculino
100,005	23	Masculino
100,006	60	Femenino
100,007	49	Femenino
100,008	44	Femenino
100,009	55	Femenino
100,010	28	Femenino
100,011	13	Femenino
100,012	62	Masculino
100,013	32	Masculino
100,014	11	Masculino
100,015	40	Femenino

Refresh Save Cancel Export data 200 CST en Writable Smart Insert 22:26 [25] Sel: 25 | 2 Desktop 22:57 ENG 30/08/2024

## → Tabla LOG\_ACTIVIDAD

```
use practica2;
select * from habitacion;
select * from paciente;
select * from log_actividad;
select * from log_habitacion;

select count(*) from habitacion;
select count(*) from paciente;
select count(*) from log_actividad;
select count(*) from log_habitacion;
```

actividad 1 × Enter a SQL expression to filter results (use Ctrl+Space)

id_log_actividad	timestampx	actividad	idPaciente	idHabitacion
------------------	------------	-----------	------------	--------------

Refresh Save Cancel Export data 200 CST en Writable Smart Insert 23:24 [30] Sel: 30 | 2 Desktop 22:57 ENG 30/08/2024

## → Tabla LOG\_HABITACION

```
use practica2;
select * from habitacion;
select * from paciente;
select * from log_actividad;
select * from log_habitacion;

select count(*) from habitacion;
select count(*) from paciente;
select count(*) from log_actividad;
select count(*) from log_habitacion;
```

g\_habitacion 1 × Enter a SQL expression to filter results (use Ctrl+Space)

id_Log_habitacion	timestampx	status	idHabitacion
-------------------	------------	--------	--------------

Refresh Save Cancel Export data 200 CST en Writable Smart Insert 24:29 [31] Sel: 31 | 2 Desktop 22:57 ENG 30/08/2024

**SELECT COUNT(\*) FROM cada tabla**

→ Tabla HABITACION

The screenshot shows the MySQL Workbench interface. In the top-left pane, there is a SQL editor window containing the following code:

```
use practica2;
select * from habitacion;
select * from paciente;
select * from log_actividad;
select * from log_habitacion;
|  
select count(*) from habitacion;
```

The code is highlighted with a blue selection bar. In the bottom-right pane, the results of the query are displayed in a table:

count(*)
15

Below the table, the status bar shows "1 row(s) fetched - 0.001s, on 2024-08-30 at 22:57:58". The system tray at the bottom right indicates the date as 30/08/2024.

→ Tabla PACIENTE

The screenshot shows the MySQL Workbench interface. In the top-left pane, there is a SQL editor window containing the following code:

```
use practica2;
select * from habitacion;
select * from paciente;
select * from log_actividad;
select * from log_habitacion;
|  
select count(*) from habitacion;
select count(*) from paciente;
```

The code is highlighted with a blue selection bar. In the bottom-right pane, the results of the query are displayed in a table:

count(*)
154,184

Below the table, the status bar shows "1 row(s) fetched - 0.003s, on 2024-08-30 at 22:58:13". The system tray at the bottom right indicates the date as 30/08/2024.

→ Tabla LOG\_ACTIVIDAD

The screenshot shows the MySQL Workbench interface. In the top-left pane, there is a SQL editor window containing the following code:

```
use practica2;
select * from habitacion;
select * from paciente;
select * from log_actividad;
select * from log_habitacion;
|  
select count(*) from habitacion;
select count(*) from paciente;
select count(*) from log_actividad;
```

The code is highlighted with a blue selection bar. In the bottom-right pane, the results of the query are displayed in a table:

count(*)
0

Below the table, the status bar shows "1 row(s) fetched - 0.000s, on 2024-08-30 at 22:58:23". The system tray at the bottom right indicates the date as 30/08/2024.

## → Tabla LOG\_HABITACION

The screenshot shows a DBeaver interface with a SQL editor window at the top containing the following code:

```
use practica2;
select * from habitacion;
select * from paciente;
select * from log_actividad;
select * from log_habitacion;

select count(*) from habitacion;
select count(*) from paciente;
select count(*) from log_actividad;
select count(*) from log_habitacion;
```

Below the editor is a results grid with one row of data:

count()
0

At the bottom of the interface, there is a toolbar with various icons and a status bar showing the date and time.

## DIA 8

### Tiempo que duró el full backup

The screenshot shows a terminal window titled "cmd.exe" with the following command history:

```
C:\Users\Jorge\Documents\usac\segundo-semestre-2024\bases\lab\repo\practica2\scripts\backup_completos (main -> origin)
λ echo %time%
23:01:03.39

C:\Users\Jorge\Documents\usac\segundo-semestre-2024\bases\lab\repo\practica2\scripts\backup_completos (main -> origin)
λ mysql -u root -padmin practica2 < day3_backup_completo.sql
mysql: [Warning] Using a password on the command line interface can be insecure.

C:\Users\Jorge\Documents\usac\segundo-semestre-2024\bases\lab\repo\practica2\scripts\backup_completos (main -> origin)
λ echo %time%
23:01:08.85

C:\Users\Jorge\Documents\usac\segundo-semestre-2024\bases\lab\repo\practica2\scripts\backup_completos (main -> origin)
λ |
```

At the bottom of the terminal window, there is a toolbar with various icons and a status bar showing the date and time.

**SELECT \* FROM cada tabla**

→ Tabla HABITACION

```
use practica2;
select * from habitacion;
select * from paciente;
select * from log_actividad;
select * from log_habitacion;

select count(*) from habitacion;
select count(*) from paciente;
select count(*) from log_actividad;
select count(*) from log_habitacion;
```

Habitacion 1 | Enter a SQL expression to filter results (use Ctrl+Space)

idHabitacion	nmr_habitacion
1	Sala de exámenes 1
2	Sala de exámenes 2
3	Sala de exámenes 3
4	Sala de exámenes 4
5	Sala de imágenes 1
6	Sala de procedimientos 1
7	Sala de procedimientos 2
8	Sala de procedimientos 3
9	Sala de procedimientos 4
10	Recepción
11	Laboratorio
12	Estación de revisión 1
13	Estación de revisión 2
14	Estación de revisión 3
15	Estación de revisión 4

Refresh Save Cancel Export data 200 CST en Writable Smart Insert 21:15 [27] Sel 27 | 2 Desktop 23:01 ENG 31/08/2024

→ Tabla PACIENTE

```
use practica2;
select * from habitacion;
select * from paciente;
select * from log_actividad;
select * from log_habitacion;

select count(*) from habitacion;
select count(*) from paciente;
select count(*) from log_actividad;
select count(*) from log_habitacion;
```

Paciente 1 | Enter a SQL expression to filter results (use Ctrl+Space)

idPaciente	edad	nmr_genero
100.000	95	Otto
100.001	40	Femenino
100.002	42	Masculino
100.003	8	Femenino
100.004	88	Masculino
100.005	23	Masculino
100.006	60	Femenino
100.007	49	Femenino
100.008	44	Femenino
100.009	55	Femenino
100.010	28	Femenino
100.011	13	Femenino
100.012	62	Masculino
100.013	32	Masculino
100.014	11	Masculino
100.015	49	Femenino

Refresh Save Cancel Export data 200 CST en Writable Smart Insert 22:26 [25] Sel 25 | 2 Desktop 23:01 ENG 31/08/2024

→ Tabla LOG\_ACTIVIDAD

```
use practica2;
select * from habitacion;
select * from paciente;
select * from log_actividad;
select * from log_habitacion;

select count(*) from habitacion;
select count(*) from paciente;
select count(*) from log_actividad;
select count(*) from log_habitacion;
```

actividad 1 | Enter a SQL expression to filter results (use Ctrl+Space)

id_log_actividad	nmr_timestamp	nmr_actividad	idPaciente	idHabitacion
------------------	---------------	---------------	------------	--------------

Refresh Save Cancel Export data 200 CST en Writable Smart Insert 24:1 [28] Sel 28 | 1 Desktop 23:02 ENG 31/08/2024

## → Tabla LOG\_HABITACION

```
use practica;
select * from habitacion;
select * from paciente;
select * from log_actividad;
select * from log_habitacion;

select count(*) from habitacion;
select count(*) from paciente;
select count(*) from log_actividad;
select count(*) from log_habitacion;
```

habitacion 1 | Enter a SQL expression to filter results (use Ctrl+Space)

id_log_habitacion	ab timestamp	log status	idHabitacion
1	5/31/2021 7:19:33 a.m.	Sala no disponible.	12
2	5/31/2021 7:20:00 a.m.	Inicia limpieza.	12
3	5/31/2021 7:20:35 a.m.	Sala disponible.	12
4	5/31/2021 7:29:15 a.m.	Sala no disponible.	12
5	5/31/2021 7:29:42 a.m.	Inicia limpieza.	12
6	5/31/2021 7:30:31 a.m.	Sala disponible.	12
7	5/31/2021 7:36:38 a.m.	Sala no disponible.	13
8	5/31/2021 7:37:05 a.m.	Inicia limpieza.	13
9	5/31/2021 7:38:53 a.m.	Sala disponible.	13
10	5/31/2021 7:42:34 a.m.	Sala no disponible.	1
11	5/31/2021 7:43:14 a.m.	Inicia limpieza.	1
12	5/31/2021 7:46:07 a.m.	Sala disponible.	1
13	5/31/2021 7:54:19 a.m.	Sala no disponible.	2
14	5/31/2021 7:54:53 a.m.	Inicia limpieza.	2
15	5/31/2021 7:55:43 a.m.	Sala no disponible.	12
16	5/31/2021 7:56:00 a.m.	Inicia limpieza.	13

Refresh Save Cancel Export data 200 200+ 200 row(s) fetched - 0.001s, on 2024-08-31 at 23:02:22

CST en Writable Smart Insert 24: 29 [31] Sel: 31 | 2

Desktop ENG 23:02 31/08/2024

## SELECT COUNT(\*) FROM cada tabla

### → Tabla HABITACION

```
use practica;
select * from habitacion;
select * from paciente;
select * from log_actividad;
select * from log_habitacion;

select count(*) from habitacion;
select count(*) from paciente;
select count(*) from log_actividad;
select count(*) from log_habitacion;
```

its 1 | Enter a SQL expression to filter results (use Ctrl+Space)

count(*)
15

Refresh Save Cancel Export data 200 1 1 row(s) fetched - 0.001s, on 2024-08-31 at 23:02:36

CST en Writable Smart Insert 26: 1 [34] Sel: 34 | 2

Desktop ENG 23:02 31/08/2024

### → Tabla PACIENTE

```
use practica;
select * from log_actividad;
select * from log_habitacion;

select count(*) from habitacion;
select count(*) from paciente;
select count(*) from log_actividad;
select count(*) from log_habitacion;
```

results 1 | Enter a SQL expression to filter results (use Ctrl+Space)

count(*)
154,184

Refresh Save Cancel Export data 200 1 1 row(s) fetched - 0.004s, on 2024-08-31 at 23:02:49

CST en Writable Smart Insert 27: 33 [32] Sel: 32 | 2

Desktop ENG 23:02 31/08/2024

→ Tabla LOG\_ACTVIDAD

```
use practica2;
select * from habitacion;
select * from paciente;
select * from log_actividad;
select * from log_habitacion;

select count(*) from habitacion;
select count(*) from paciente;
select count(*) from log_actividad;
select count(*) from log_habitacion;
```

Results 1 ×

count(*)
0

Export data | 200 | CST | en | Writable | Smart Insert | 28:31 [37] | Sel: 37 | 2 | ...

1 row(s) fetched - 0.000s, on 2024-08-31 at 23:03:01

Desktop | 23:03 | ENG | 31/08/2024 | 30

→ Tabla LOG\_HABITACION

```
use practica2;
select * from habitacion;
select * from paciente;
select * from log_actividad;
select * from log_habitacion;

select count(*) from habitacion;
select count(*) from paciente;
select count(*) from log_actividad;
select count(*) from log_habitacion;
```

Results 1 ×

count(*)
34,617

Export data | 200 | CST | en | Writable | Smart Insert | 29:38 [38] | Sel: 38 | 2 | ...

1 row(s) fetched - 0.000s, on 2024-08-31 at 23:03:15

Desktop | 23:03 | ENG | 31/08/2024 | 30

## DIA 9

### Tiempo que duró el full backup

```
C:\Users\Jorge\Documents\usac\segundo-semestre-2024\bases\lab\repo\practica2\scripts\backup_completos (main -> origin)
λ echo %time%
22:50:01.17

C:\Users\Jorge\Documents\usac\segundo-semestre-2024\bases\lab\repo\practica2\scripts\backup_completos (main -> origin)
λ mysql -u root -padmin practica2 < day4_backup_completo.sql
mysql: [Warning] Using a password on the command line interface can be insecure.

C:\Users\Jorge\Documents\usac\segundo-semestre-2024\bases\lab\repo\practica2\scripts\backup_completos (main -> origin)
λ echo %time%
22:50:13.37

C:\Users\Jorge\Documents\usac\segundo-semestre-2024\bases\lab\repo\practica2\scripts\backup_completos (main -> origin)
λ |
```

```
cmd.exe | CST | en | Writable | Smart Insert | 29 : 36 [38] | Sek 38 | 2 | ... | Desktop | ⌂ | ⌂ | ⌂ | ENG | 22:50 | 1/09/2024 | 30 |
```

### SELECT \* FROM cada tabla

→ Tabla HABITACION

```
use practica2;
select * from habitacion;
select * from paciente;
select * from log_actividad;
select * from log_habitacion;

select count(*) from habitacion;
select count(*) from paciente;
select count(*) from log_actividad;
select count(*) from log_habitacion;
```

idHabitacion	habitacion
1	Sala de exámenes 1
2	Sala de exámenes 2
3	Sala de exámenes 3
4	Sala de exámenes 4
5	Sala de imágenes 1
6	Sala de procedimientos 1
7	Sala de procedimientos 2
8	Sala de procedimientos 3
9	Sala de procedimientos 4
10	Recepción
11	Laboratorio
12	Estación de revisión 1
13	Estación de revisión 2
14	Estación de revisión 3
15	Estación de revisión 4

```
Desktop | ⌂ | ⌂ | ⌂ | ENG | 22:50 | 1/09/2024 | 30 |
```

## → Tabla PACIENTE

```
use practica0;
select * from habitacion;
select * from paciente;
select * from log_actividad;
select * from log_habitacion;

select count(*) from habitacion;
select count(*) from paciente;
select count(*) from log_actividad;
select count(*) from log_habitacion;
```

paciente 1

	idPaciente	edad	genero
1	100.000	95	Otro
2	100.001	40	Femenino
3	100.002	42	Masculino
4	100.003	8	Femenino
5	100.004	88	Masculino
6	100.005	23	Masculino
7	100.006	60	Femenino
8	100.007	49	Femenino
9	100.008	44	Femenino
0	100.009	55	Femenino
1	100.010	28	Femenino
2	100.011	13	Femenino
3	100.012	62	Masculino
4	100.013	32	Masculino
5	100.014	11	Masculino
6	100.015	60	Femenino

Refresh Save Cancel Export data 200 200+ 200 row(s) fetched - 0.001s, on 2024-09-01 at 22:51:15

CST en Writable Smart Insert 22 : 26 [25] Sel: 25 | 2 Desktop ↻ ⌂ ⌂ ENG 22:51 1/09/2024 30

→ Tabla LOG\_ACTIVIDAD

```
use practica2;
select * from habitacion;
select * from paciente;
select * from log_actividad;
select * from log_habitacion;

select count(*) from habitacion;
select count(*) from paciente;
select count(*) from log_actividad;
select count(*) from log_habitacion;

actividad 1 ×
SQL Statements Available | % ⌂ Enter a SQL expression to filter results (use Ctrl+Space)


|    | log_id_log_actividad   | log_timestamp                                                                                                                    | log_actividad | log_idPaciente | log_idHabitacion |
|----|------------------------|----------------------------------------------------------------------------------------------------------------------------------|---------------|----------------|------------------|
| 1  | 5/31/2021 7:12:14 a.m. | Paciente inicia el registro.                                                                                                     |               | 134,247        | 10               |
| 2  | 5/31/2021 7:13:16 a.m. | Paciente recibe papelera en recepcion.                                                                                           |               | 134,247        | 10               |
| 3  | 5/31/2021 7:15:54 a.m. | Paciente entrega papelera.                                                                                                       |               | 134,247        | 10               |
| 4  | 5/31/2021 7:16:33 a.m. | Recepcionista establece la condicion del paciente como 'Urgente'. Paciente es enviado a la sala de espera para esperar revision. |               | 134,247        | 10               |
| 5  | 5/31/2021 7:17:47 a.m. | Enfermera comienza la revision del paciente.                                                                                     |               | 134,247        | 10               |
| 6  | 5/31/2021 7:19:33 a.m. | Revision determino que el paciente es tipo 2 y su condicion es 'Urgente'.                                                        |               | 134,247        | 10               |
| 7  | 5/31/2021 7:20:34 a.m. | Medico inicia con el tratamiento del paciente.                                                                                   |               | 134,247        | 10               |
| 8  | 5/31/2021 7:21:50 a.m. | Paciente inicia el registro.                                                                                                     |               | 135,641        | 10               |
| 9  | 5/31/2021 7:23:01 a.m. | Paciente recibe papelera en recepcion.                                                                                           |               | 135,641        | 10               |
| 10 | 5/31/2021 7:25:39 a.m. | Paciente inicia el registro.                                                                                                     |               | 180,487        | 10               |
| 11 | 5/31/2021 7:25:44 a.m. | Paciente entrega papelera.                                                                                                       |               | 135,641        | 12               |
| 12 | 5/31/2021 7:26:18 a.m. | Recepcionista establece la condicion del paciente como 'Estable'. Paciente es enviado a la sala de espera para esperar revision. |               | 135,641        | 12               |
| 13 | 5/31/2021 7:26:43 a.m. | Paciente recibe papelera en recepcion.                                                                                           |               | 180,487        | 5                |
| 14 | 5/31/2021 7:27:33 a.m. | Enfermera comienza la revision del paciente.                                                                                     |               | 135,641        | 5                |
| 15 | 5/31/2021 7:29:12 a.m. | Paciente entrega papelera.                                                                                                       |               | 180,487        | 7                |
| 16 | 4/2/2021 7:20:15 a.m.  | Revision determino que el paciente es tipo 2 y su condicion es 'Urgente'.                                                        |               | 125,841        | 7                |


```

## → Tabla LOG HABITACION

```
use practice2;
select * from habitacion;
select * from paciente;
select * from log_actividad;
select * from log_habitacion;

select count(*) from habitacion;
select count(*) from paciente;
select count(*) from log_actividad;
select count(*) from log_habitacion;
```

habitacion 1

id_log_habitacion	timestampx	status	idHabitacion
1	5/31/2021 7:19:33 a.m.	Sala no disponible.	12
2	5/31/2021 7:20:00 a.m.	Inicia limpieza.	12
3	5/31/2021 7:20:35 a.m.	Sala disponible.	12
4	5/31/2021 7:29:15 a.m.	Sala no disponible.	12
5	5/31/2021 7:29:42 a.m.	Inicia limpieza.	12
6	5/31/2021 7:30:31 a.m.	Sala disponible.	12
7	5/31/2021 7:36:38 a.m.	Sala no disponible.	13
8	5/31/2021 7:57:05 a.m.	Inicia limpieza.	13
9	5/31/2021 7:58:53 a.m.	Sala disponible.	13
10	5/31/2021 7:42:34 a.m.	Sala no disponible.	1
11	5/31/2021 7:43:14 a.m.	Inicia limpieza.	1
12	5/31/2021 7:46:07 a.m.	Sala disponible.	1
13	5/31/2021 7:54:19 a.m.	Sala no disponible.	2
14	5/31/2021 7:54:53 a.m.	Inicia limpieza.	2
15	5/31/2021 7:55:43 a.m.	Sala no disponible.	12
16	5/31/2021 7:56:40 a.m.	Inicia limpieza.	17

**SELECT COUNT(\*) FROM cada tabla**

→ Tabla HABITACION

The screenshot shows the MySQL Workbench interface with a query editor and results pane. The query editor contains the following SQL code:

```
use practica2;
select * from habitacion;
select * from paciente;
select * from log_actividad;
select * from log_habitacion;
| select count(*) from habitacion;
```

The results pane displays the output of the last query, which is '15'. The status bar at the bottom right indicates the result was fetched in 0.001s on 2024-09-01 at 22:51:57.

→ Tabla PACIENTE

The screenshot shows the MySQL Workbench interface with a query editor and results pane. The query editor contains the following SQL code:

```
use practica2;
select * from habitacion;
select * from paciente;
select * from log_actividad;
select * from log_habitacion;
| select count(*) from habitacion;
| select count(*) from paciente;
```

The results pane displays the output of the second query, which is '154,184'. The status bar at the bottom right indicates the result was fetched in 0.004s on 2024-09-01 at 22:52:08.

→ Tabla LOG\_ACTIVIDAD

The screenshot shows the MySQL Workbench interface with a query editor and results pane. The query editor contains the following SQL code:

```
use practica2;
select * from habitacion;
select * from paciente;
select * from log_actividad;
select * from log_habitacion;
| select count(*) from habitacion;
| select count(*) from paciente;
| select count(*) from log_actividad;
```

The results pane displays the output of the third query, which is '33,841'. The status bar at the bottom right indicates the result was fetched in 0.005s on 2024-09-01 at 22:52:20.

## → Tabla LOG\_HABITACION

The screenshot shows the MySQL Workbench interface. A query window contains the following SQL code:

```
use practica2;
select * from habitacion;
select * from paciente;
select * from log_actividad;
select * from log_habitacion;

select count(*) from habitacion;
select count(*) from paciente;
select count(*) from log_actividad;
select count(*) from log_habitacion;
```

The results pane shows a single row of data:

count(*)
34,617

Below the results, the status bar indicates: 1 row(s) fetched - 0.004s, on 2024-09-01 at 22:52:31.

## DIA 10

### Tiempo que duró el full backup

The screenshot shows a terminal window titled "cmd.exe". The command history shows the execution of a backup script:

```
C:\Users\Jorge\Documents\usac\segundo-semestre-2024\bases\lab\repo\practica2\scripts\backup_completos (main -> origin)
λ echo %time%
23:03:50.36

C:\Users\Jorge\Documents\usac\segundo-semestre-2024\bases\lab\repo\practica2\scripts\backup_completos (main -> origin)
λ mysql -u root -padmin practica2 < day5_backup_completo.sql
mysql: [Warning] Using a password on the command line interface can be insecure.

C:\Users\Jorge\Documents\usac\segundo-semestre-2024\bases\lab\repo\practica2\scripts\backup_completos (main -> origin)
λ echo %time%
23:04:00.60

C:\Users\Jorge\Documents\usac\segundo-semestre-2024\bases\lab\repo\practica2\scripts\backup_completos (main -> origin)
λ |
```

The status bar at the bottom of the terminal window shows: Not connected to database.

**SELECT \* FROM cada tabla**

→ Tabla HABITACION

```
use practica2;
select * from habitacion;
select * from paciente;
select * from log_actividad;
select * from log_habitacion;

select count(*) from habitacion;
select count(*) from paciente;
select count(*) from log_actividad;
select count(*) from log_habitacion;
```

cion 1 × Enter a SQL expression to filter results (use Ctrl+Space)

idHabitacion	habitacion
1	Sala de exámenes 1
2	Sala de exámenes 2
3	Sala de exámenes 3
4	Sala de exámenes 4
5	Sala de imágenes 1
6	Sala de procedimientos 1
7	Sala de procedimientos 2
8	Sala de procedimientos 3
9	Sala de procedimientos 4
10	Recepción
11	Laboratorio
12	Estación de revisión 1
13	Estación de revisión 2
14	Estación de revisión 3
15	Estación de revisión 4

Refresh Save Cancel Export data 200 CST en Writable Smart Insert 21: 15 [27] Sel: 2712 Desktop 23:04 ENG 2/09/2024

→ Tabla PACIENTE

```
use practica2;
select * from habitacion;
select * from paciente;
select * from log_actividad;
select * from log_habitacion;

select count(*) from habitacion;
select count(*) from paciente;
select count(*) from log_actividad;
select count(*) from log_habitacion;
```

paciente 1 × Enter a SQL expression to filter results (use Ctrl+Space)

idPaciente	edad	genero
100,000	95	Otro
100,001	40	Femenino
100,002	42	Masculino
100,003	8	Femenino
100,004	88	Masculino
100,005	23	Masculino
100,006	60	Femenino
100,007	49	Femenino
100,008	44	Femenino
100,009	55	Femenino
100,010	28	Femenino
100,011	13	Femenino
100,012	62	Masculino
100,013	32	Masculino
100,014	11	Masculino
100,015	50	Femenina

refresh Save Cancel Export data 200 CST en Writable Smart Insert 22: 26 [25] Sel: 25 [2] Desktop 23:05 ENG 2/09/2024

→ Tabla LOG\_ACTIVIDAD

```
use practica2;
select * from habitacion;
select * from paciente;
select * from log_actividad;
select * from log_habitacion;

select count(*) from habitacion;
select count(*) from paciente;
select count(*) from log_actividad;
select count(*) from log_habitacion;
```

actividad 1 × Enter a SQL expression to filter results (use Ctrl+Space)

id_log_actividad	actimestampx	actividad	idPaciente	idHabitacion
1	5/31/2021 7:21:14 a.m.	Paciente inicia el registro.	134,247	10
2	5/31/2021 7:18:16 a.m.	Paciente recibe papelería en recepción.	134,247	10
3	5/31/2021 7:18:16 a.m.	Paciente entrega papelería.	134,247	10
4	5/31/2021 7:16:33 a.m.	Recepcionista establece la condición del paciente como 'Urgente'. Paciente es enviado a la sala de espera para esperar revisión.	134,247	10
5	5/31/2021 7:17:47 a.m.	Enfermera comienza la revisión del paciente.	134,247	10
6	5/31/2021 7:19:33 a.m.	Revisión determina que el paciente es tipo 2 y su condición es 'Urgente'.	134,247	10
7	5/31/2021 7:20:34 a.m.	Médico inicia con el tratamiento del paciente.	134,247	10
8	5/31/2021 7:21:50 a.m.	Paciente inicia el registro.	135,641	10
9	5/31/2021 7:23:01 a.m.	Paciente recibe papelería en recepción.	135,641	10
10	5/31/2021 7:25:39 a.m.	Paciente inicia el registro.	180,487	10
11	5/31/2021 7:25:44 a.m.	Paciente entrega papelería.	135,641	12
12	5/31/2021 7:26:18 a.m.	Recepcionista establece la condición del paciente como 'Estable'. Paciente es enviado a la sala de espera para esperar revisión.	135,641	12
13	5/31/2021 7:26:43 a.m.	Paciente recibe papelería en recepción.	180,487	5
14	5/31/2021 7:27:33 a.m.	Enfermera comienza la revisión del paciente.	135,641	5
15	5/31/2021 7:29:12 a.m.	Paciente entrega papelería.	180,487	7
16	5/31/2021 7:30:15 a.m.	Revisión determina que el paciente es tipo 2 y su condición es 'Urgente'.	134,441	9

refresh Save Cancel Export data 200 CST en Writable Smart Insert 23: 24 [30] Sel: 30 [2] Desktop 23:05 ENG 2/09/2024

## → Tabla LOG\_HABITACION

```
use practica2;
select * from habitacion;
select * from paciente;
select * from log_actividad;
select * from log_habitacion;

select count(*) from habitacion;
select count(*) from paciente;
select count(*) from log_actividad;
select count(*) from log_habitacion;
```

log\_habitacion 1 × Enter a SQL expression to filter results (use Ctrl+Space)

id_log_habitacion	rec_timestamp	rec_status	idHabitacion
1	5/31/2021 7:19:33 a.m.	Sala no disponible.	12
2	5/31/2021 7:20:00 a.m.	Inicia Limpieza.	12
3	5/31/2021 7:20:35 a.m.	Sala disponible.	12
4	5/31/2021 7:29:15 a.m.	Sala no disponible.	12
5	5/31/2021 7:29:42 a.m.	Inicia Limpieza.	12
6	5/31/2021 7:30:31 a.m.	Sala disponible.	12
7	5/31/2021 7:36:38 a.m.	Sala no disponible.	13
8	5/31/2021 7:37:00 a.m.	Inicia Limpieza.	13
9	5/31/2021 7:38:53 a.m.	Sala disponible.	13
10	5/31/2021 7:42:34 a.m.	Sala no disponible.	1
11	5/31/2021 7:43:14 a.m.	Inicia Limpieza.	1
12	5/31/2021 7:46:07 a.m.	Sala disponible.	1
13	5/31/2021 7:54:19 a.m.	Sala no disponible.	2
14	5/31/2021 7:54:53 a.m.	Inicia Limpieza.	2
15	5/31/2021 7:55:43 a.m.	Sala no disponible.	12
16	5/31/2021 7:46:00 a.m.	Inicia Limpieza.	13

Refresh Save Cancel Export data CST en Writable Smart Insert 200 200+ 200 row(s) fetched - 0.000s, on 2024-09-02 at 23:05:29 Sel 31 | 2 Desktop ENG 23:05 2/09/2024

## SELECT COUNT(\*) FROM cada tabla

### → Tabla HABITACION

```
use practica2;
select * from habitacion;
select * from paciente;
select * from log_actividad;
select * from log_habitacion;

select count(*) from habitacion;
select count(*) from paciente;
select count(*) from log_actividad;
select count(*) from log_habitacion;
```

Results 1 × Enter a SQL expression to filter results (use Ctrl+Space)

count(*)
15

Refresh Save Cancel Export data CST en Writable Smart Insert 26:1 [34] Sel 34 | 2 1 row(s) fetched - 0.002s (0.001s fetch), on 2024-09-02 at 23:05:48 Desktop ENG 23:05 2/09/2024

### → Tabla PACIENTE

```
use practica2;
select * from habitacion;
select * from paciente;
select * from log_actividad;
select * from log_habitacion;

select count(*) from habitacion;
select count(*) from paciente;
select count(*) from log_actividad;
select count(*) from log_habitacion;
```

Results 1 × Enter a SQL expression to filter results (use Ctrl+Space)

count(*)
154,184

Refresh Save Cancel Export data CST en Writable Smart Insert 27:33 [32] Sel 32 | 2 1 row(s) fetched - 0.003s, on 2024-09-02 at 23:05:59 Desktop ENG 23:06 2/09/2024

## → Tabla LOG\_ACTIVIDAD

The screenshot shows a MySQL Workbench interface. In the SQL tab, a query is run:

```
use practica2;
select * from habitacion;
select * from paciente;
select * from log_actividad;
select * from log_habitacion;

select count(*) from habitacion;
select count(*) from paciente;
select count(*) from log_actividad;
select count(*) from log_habitacion;
```

The results tab displays the output of the last query:

count(*)
67,684

Below the results, the status bar indicates "1 row(s) fetched - 0.008s, on 2024-09-02 at 23:06:10".

## → Tabla LOG\_HABITACION

The screenshot shows a MySQL Workbench interface. In the SQL tab, a query is run:

```
use practica2;
select * from habitacion;
select * from paciente;
select * from log_actividad;
select * from log_habitacion;

select count(*) from habitacion;
select count(*) from paciente;
select count(*) from log_actividad;
select count(*) from log_habitacion;
```

The results tab displays the output of the last query:

count(*)
34,617

Below the results, the status bar indicates "1 row(s) fetched - 0.004s, on 2024-09-02 at 23:06:20".

DIA 11

### Tiempo que duró el backup Incremental 1

```
sh-5.1# time mysql -uroot -p1234 practica2 < ./day1_backup_incremental.sql
mysql: [Warning] Using a password on the command line interface can be insecure.

real 0m1.793s
user 0m0.043s
sys 0m0.008s
sh-5.1#
```

**SELECT \* FROM cada tabla**

## → Tabla PACIENTE

The screenshot shows the DBVisualizer Free interface version 24.2.1. The top menu includes File, Edit, View, Database, SQL Commander, Tools, Window, Help, and a toolbar with icons for file operations. The left sidebar lists databases: Cassandra, MySQL 8, Databases, and DBA Views. The main area has tabs for '1: p2\_bases\_2.sql' and 'MySQL 8'. A 'Database Connection' dropdown is set to 'MySQL 8'. The 'Sticky Database' dropdown is also set to 'MySQL 8'. The 'Schema' dropdown shows 'p2\_bases\_2'. On the right, there are filters for 'Max Rows' (1000) and 'Max Chars' (-1). The SQL editor contains a query:

```
37
38 SELECT * FROM PACIENTE;
39 SELECT COUNT(*) AS "Cantidad de Registros" FROM PACIENTE;
40
41 SELECT * FROM HABITACION;
42 SELECT COUNT(*) AS "Cantidad de Registros" FROM HABITACION;
43
44 SELECT * FROM LOG_HABITACION;
45 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_HABITACION;
46
47 SELECT * FROM LOG_ACTIVIDAD;
48 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_ACTIVIDAD;
49
50
51
```

The status bar at the bottom shows 'Format: <Select a Cell>', '0.0000.000 sec', '1000/3 1-22', '14M of 252M', '09:08 p.m.', and the date '3/09/2024'.

## → Tabla HABITACION

→ Tabla LOG\_HABITACION

The screenshot shows the DbVisualizer Free 24.2.1 interface. The top menu bar includes File, Edit, View, Database, SQL Commander, Tools, Window, Help, and a status bar at the bottom right showing performance metrics (0.0060/0.002 sec, 0/4, 0/0, 09:19 p.m., 3/09/2024). The left sidebar lists databases: Cassandra, MySQL 8, Databases, and DBA Views. The main area has tabs for 'Database Connection' (MySQL 8) and 'Sticky Database'. A query editor window titled '1: p2\_bases\_2.sql x' contains the following SQL code:

```
SELECT * FROM PACIENTE;
SELECT COUNT(*) AS "Cantidad de Registros" FROM PACIENTE;
SELECT * FROM HABITACION;
SELECT COUNT(*) AS "Cantidad de Registros" FROM HABITACION;
SELECT * FROM LOG_HABITACION;
SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_HABITACION;
SELECT * FROM LOG_ACTIVIDAD;
SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_ACTIVIDAD;
```

The results pane shows a table named 'Log' with one row: '1: LOG\_HABITACION [0]'. The table has columns: id\_log\_habitacion, timestampx, statusx, and idhabitacion. The status column value is 'INS'. Below the table, a message says 'Result set is empty.' The bottom status bar also displays the current date and time: 3/09/2024.

→ Tabla LOG\_ACTIVIDAD

The screenshot shows the DbVisualizer Free 24.2.1 interface. The top menu bar includes File, Edit, View, Database, SQL Commander, Tools, Window, Help, and a Database Connection dropdown set to MySQL 8. The left sidebar lists databases: Cassandra, MySQL 8, Databases, and DBA Views. The main area has tabs for Scripts and Favorites, with the MySQL 8 tab selected. A query editor window titled '1: p2\_bases\_2.sql' is open, displaying the following SQL code:

```
40 SELECT * FROM PACIENTE;
41 SELECT COUNT(*) AS "Cantidad de Registros" FROM PACIENTE;
42
43 SELECT * FROM HABITACION;
44 SELECT COUNT(*) AS "Cantidad de Registros" FROM HABITACION;
45
46 SELECT * FROM LOG_HABITACION;
47 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_HABITACION;
48
49 SELECT * FROM LOG_ACTIVIDAD;
50 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_ACTIVIDAD;
51
52
```

Below the code, it says '49/51 [1,160] INS'. The bottom section shows a table named '1: LOG\_ACTIVIDAD [0]' with columns: id\_log\_actividad, timestampx, actividad, idPaciente, and idHabitacion. A message box says 'Result set is empty'. The status bar at the bottom right shows performance metrics: 0.007/0.002 sec, 0/5, 0-0, and system information: ENG, 09:20 p.m., 3/09/2024.

## **SELECT COUNT(\*) FROM cada tabla**

→ Tabla PACIENTE

The screenshot shows the DbVisualizer Free 24.2.1 interface. The database connection is set to MySQL 8. A query window titled 't:p2\_bases\_2.sql' is open, displaying the following SQL code:

```
40 SELECT * FROM PACIENTE;
41 SELECT COUNT(*) AS "Cantidad de Registros" FROM PACIENTE;
42
43 SELECT * FROM HABITACION;
44 SELECT COUNT(*) AS "Cantidad de Registros" FROM HABITACION;
45
46 SELECT * FROM LOG_HABITACION;
47 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_HABITACION;
48
49 SELECT * FROM LOG_ACTIVIDAD;
50 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_ACTIVIDAD;
51
52
```

The results pane shows a single row of data from the 'PACIENTE' table:

Cantidad de Registros
154184

At the bottom of the interface, the status bar indicates: 0.018/0.002 sec 1/1 1-1 0.47M of 2048M 09:22 p.m. 3/09/2024.

→ Tabla HABITACION

The screenshot shows the DbVisualizer Free 24.2.1 interface. The database connection is set to MySQL 8. A query window titled 't:p2\_bases\_2.sql' is open, displaying the same SQL code as the previous screenshot:

```
40 SELECT * FROM PACIENTE;
41 SELECT COUNT(*) AS "Cantidad de Registros" FROM PACIENTE;
42
43 SELECT * FROM HABITACION;
44 SELECT COUNT(*) AS "Cantidad de Registros" FROM HABITACION;
45
46 SELECT * FROM LOG_HABITACION;
47 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_HABITACION;
48
49 SELECT * FROM LOG_ACTIVIDAD;
50 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_ACTIVIDAD;
51
52
```

The results pane shows a single row of data from the 'HABITACION' table:

Cantidad de Registros
0

At the bottom of the interface, the status bar indicates: 0.005/0.001 sec 1/1 1-1 0.47M of 2048M 09:22 p.m. 3/09/2024.

→ Tabla LOG\_HABITACION

The screenshot shows the DbVisualizer Free 24.2.1 interface. At the top, the menu bar includes File, Edit, View, Database, SQL Commander, Tools, Window, Help, and a tab for DbVisualizer Free 24.2.1 - C:\Users\sergi\dbvis\Bookmarks\p2\_bases\_2.sql. The left sidebar lists databases: Cassandra, MySQL 8, Databases, and DBA Views. The main area has tabs for '1: p2\_bases\_2.sql' and 'MySQL 8 x'. A 'Database Connection' dropdown is set to MySQL 8. Below it, a 'Sticky Database' dropdown is also set to MySQL 8. The 'Schema' dropdown shows 'p2\_bases\_2'. On the right, 'Max Rows' is set to 1000 and 'Max Chars' to -1. The central workspace contains a SQL query:

```
1 SELECT * FROM PACIENTE;
2 SELECT COUNT(*) AS "Cantidad de Registros" FROM PACIENTE;
3
4 SELECT * FROM HABITACION;
5 SELECT COUNT(*) AS "Cantidad de Registros" FROM HABITACION;
6
7 SELECT * FROM LOG_HABITACION;
8 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_HABITACION;
9
10 SELECT * FROM LOG_ACTIVIDAD;
11 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_ACTIVIDAD;
12
13
```

Below the query, the status bar indicates 47 / 1 [1,098] INS. The bottom right corner shows performance metrics: 0.0070/0.001 sec, 1/1, 0.47M of 2048MB, and a timestamp of 09:23 p.m. on 3/09/2024.

→ Tabla LOG\_ACTIVIDAD

The screenshot shows the DBVisualizer Free 24.2.1 interface. The top menu bar includes File, Edit, View, Database, SQL Commander, Tools, Window, Help, and a toolbar with various icons. The left sidebar lists databases: Cassandra, MySQL\_8, Databases, and DBA Views. The main area has tabs for '1: p2\_bases\_2.sql' and 'MySQL 8'. Under 'MySQL 8', the 'Database Connection' dropdown is set to 'MySQL 8'. The 'Sticky Database' dropdown is also set to 'MySQL 8'. The 'Schema' dropdown shows 'p2\_bases\_2'. On the right, there are 'Max Rows' (set to 3000) and 'Max Chars' (-1) settings. The central pane displays a SQL query:

```
40 SELECT * FROM PACIENTE;
41 SELECT COUNT(*) AS "Cantidad de Registros" FROM PACIENTE;
42
43 SELECT * FROM HABITACION;
44 SELECT COUNT(*) AS "Cantidad de Registros" FROM HABITACION;
45
46 SELECT * FROM LOG_HABITACION;
47 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_HABITACION;
48
49 SELECT * FROM LOG_ACTIVIDAD;
50 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_ACTIVIDAD;
51
52
```

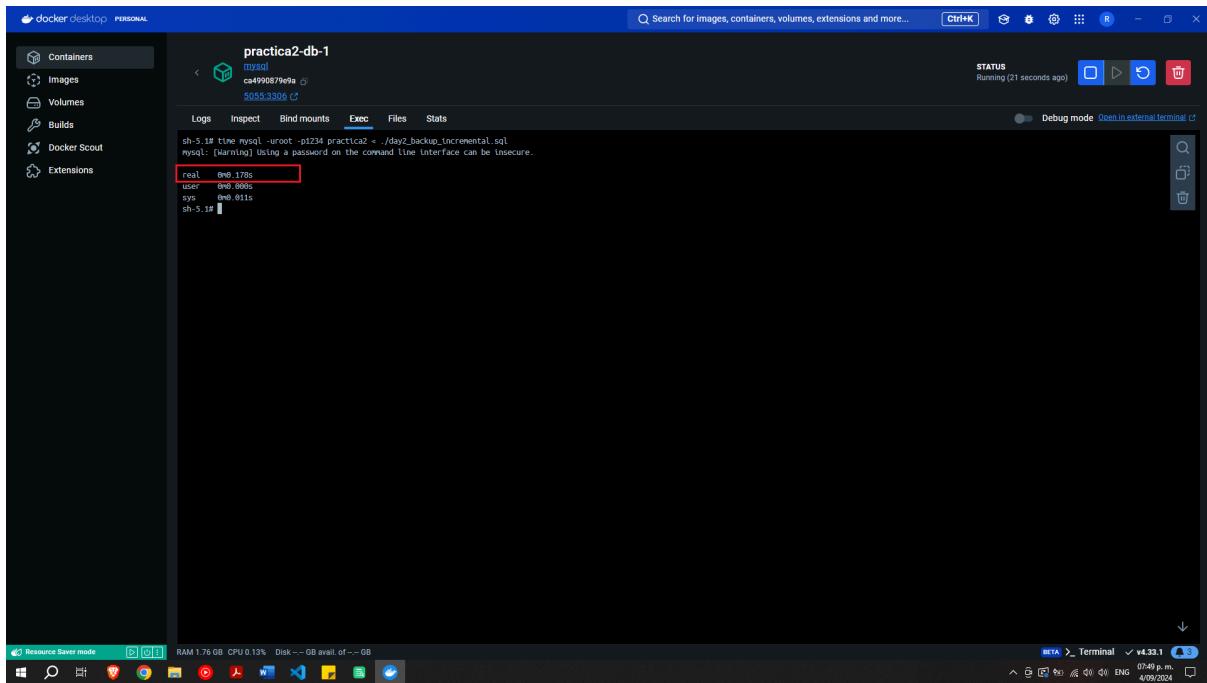
Below the query, it says '50/51 [1,188] INS'. The bottom pane shows a table titled 'Log' with one row:

	1: LOG_ACTIVIDAD [1]
• Cantidad de Registros	1 0

At the bottom, the status bar shows 'Format: <Select a Cell>', performance metrics (0.0050/0.002 sec, 1/1-1), system information (0923 g, ENG, 3/09/2024), and a taskbar with various application icons.

## DIA 12

### Tiempo que duró el backup Incremental 2

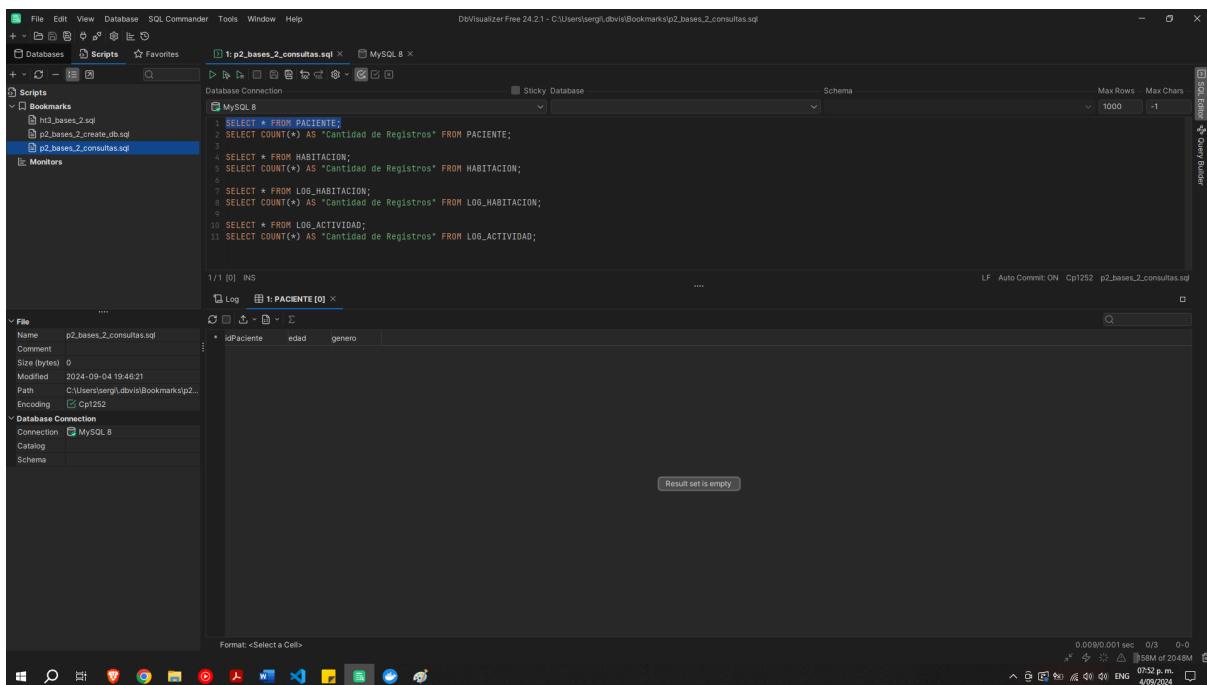


```
sh-5.1$ time mysql -uroot -p1234 practica2 < ./day2_backup_incremental.sql
mysql: [Warning] Using a password on the command line interface can be insecure.

real    0m0.178s
user    0m0.000s
sys     0m0.011s
sh-5.1$
```

**SELECT \* FROM cada tabla**

→ Tabla PACIENTE



```
1 SELECT * FROM PACIENTE;
2 SELECT COUNT(*) AS "Cantidad de Registros" FROM PACIENTE;
3
4 SELECT * FROM HABITACION;
5 SELECT COUNT(*) AS "Cantidad de Registros" FROM HABITACION;
6
7 SELECT * FROM LOG_HABITACION;
8 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_HABITACION;
9
10 SELECT * FROM LOG_ACTIVIDAD;
11 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_ACTIVIDAD;
```

## → Tabla HABITACION

The screenshot shows the DBVisualizer Free interface with the 'Scripts' tab selected. A database connection to MySQL 8 is established. The main pane displays the results of a query against the 'HABITACION' table. The table has two columns: 'idHabitacion' and 'Habitacion'. The data consists of 15 rows, each containing a unique identifier and a room name. The bottom pane shows the table structure with columns 'idHabitacion' and 'Habitacion'.

```
1 SELECT * FROM PACIENTE;
2 SELECT COUNT(*) AS "Cantidad de Registros" FROM PACIENTE;
3
4 SELECT * FROM HABITACION;
5 SELECT COUNT(*) AS "Cantidad de Registros" FROM HABITACION;
6
7 SELECT * FROM LOG_HABITACION;
8 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_HABITACION;
9
10 SELECT * FROM LOG_ACTIVIDAD;
11 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_ACTIVIDAD;
```

	Habitacion
1	Sala de exámenes 1
2	Sala de exámenes 2
3	Sala de exámenes 3
4	Sala de exámenes 4
5	Sala de imágenes 1
6	Sala de procedimientos 1
7	Sala de procedimientos 2
8	Sala de procedimientos 3
9	Sala de procedimientos 4
10	Recepción
11	Laboratorio
12	Estación de revisión 1
13	Estación de revisión 2
14	Estación de revisión 3
15	Estación de revisión 4

## → Tabla LOG\_HABITACION

The screenshot shows the DBVisualizer Free interface with the 'Scripts' tab selected. A database connection to MySQL 8 is established. The main pane displays the results of a query against the 'LOG\_HABITACION' table. The table has three columns: 'id\_log\_habitacion', 'timestampx', and 'statusx'. The status column contains the message 'Result set is empty.' The bottom pane shows the table structure with columns 'id\_log\_habitacion', 'timestampx', and 'statusx'.

```
1 SELECT * FROM PACIENTE;
2 SELECT COUNT(*) AS "Cantidad de Registros" FROM PACIENTE;
3
4 SELECT * FROM HABITACION;
5 SELECT COUNT(*) AS "Cantidad de Registros" FROM HABITACION;
6
7 SELECT * FROM LOG_HABITACION;
8 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_HABITACION;
9
10 SELECT * FROM LOG_ACTIVIDAD;
11 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_ACTIVIDAD;
```

	timestampx	statusx
		Result set is empty.

## → Tabla LOG\_ACTIVIDAD

The screenshot shows the DBVisualizer Free 24.2.1 interface. The main window displays a SQL script named 'p2\_bases\_2\_consultas.sql' with the following content:

```
1 SELECT * FROM PACIENTE;
2 SELECT COUNT(*) AS "Cantidad de Registros" FROM PACIENTE;
3
4 SELECT * FROM HABITACION;
5 SELECT COUNT(*) AS "Cantidad de Registros" FROM HABITACION;
6
7 SELECT * FROM LOG_HABITACION;
8 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_HABITACION;
9
10 SELECT * FROM LOG_ACTIVIDAD;
11 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_ACTIVIDAD;
```

The status bar at the bottom indicates the execution time: 208ms 1 of 1 (4.8s) ✓ 1 0 3/8 1-1.

## SELECT COUNT(\*) FROM cada tabla

### → Tabla PACIENTE

The screenshot shows the DBVisualizer Free 24.2.1 interface. The main window displays a SQL script named 'p2\_bases\_2\_consultas.sql' with the following content:

```
1 SELECT * FROM PACIENTE;
2 SELECT COUNT(*) AS "Cantidad de Registros" FROM PACIENTE;
3
4 SELECT * FROM HABITACION;
5 SELECT COUNT(*) AS "Cantidad de Registros" FROM HABITACION;
6
7 SELECT * FROM LOG_HABITACION;
8 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_HABITACION;
9
10 SELECT * FROM LOG_ACTIVIDAD;
11 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_ACTIVIDAD;
```

The results pane shows the output for the first query:

Cantidad de Registros
0

The status bar at the bottom indicates the execution time: 0.006/0.001 sec 1/1 1-1.

## → Tabla HABITACION

The screenshot shows the DbVisualizer Free 24.2.1 interface. The main window displays the results of a SQL query against the 'HABITACION' table. The query counts the number of records in various tables: PACIENTE, HABITACION, LOG\_HABITACION, and LOG\_ACTIVIDAD. The results are shown in a table with one row and four columns, labeled 'Cantidad de Registros'. The table has two rows: one for 'PACIENTE' with a value of 15, and one for 'LOG\_HABITACION' with a value of 0.

Cantidad de Registros	PACIENTE	HABITACION	LOG_HABITACION
15	1	0	0

Below the results, the file 'p2\_bases\_2\_consultas.sql' is open in the editor, showing the full SQL script used to generate the results. The connection is set to MySQL 8.

```
1 SELECT * FROM PACIENTE;
2 SELECT COUNT(*) AS "Cantidad de Registros" FROM PACIENTE;
3
4 SELECT * FROM HABITACION;
5 SELECT COUNT(*) AS "Cantidad de Registros" FROM HABITACION;
6
7 SELECT * FROM LOG_HABITACION;
8 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_HABITACION;
9
10 SELECT * FROM LOG_ACTIVIDAD;
11 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_ACTIVIDAD;
```

## → Tabla LOG\_HABITACION

The screenshot shows the DbVisualizer Free 24.2.1 interface. The main window displays the results of a SQL query against the 'LOG\_HABITACION' table. The query counts the number of records in the 'LOG\_HABITACION' table. The results are shown in a table with one row and two columns, labeled 'Cantidad de Registros'. The table has one row with a value of 0.

Cantidad de Registros	LOG_HABITACION
0	0

Below the results, the file 'p2\_bases\_2\_consultas.sql' is open in the editor, showing the full SQL script used to generate the results. The connection is set to MySQL 8.

```
1 SELECT * FROM PACIENTE;
2 SELECT COUNT(*) AS "Cantidad de Registros" FROM PACIENTE;
3
4 SELECT * FROM HABITACION;
5 SELECT COUNT(*) AS "Cantidad de Registros" FROM HABITACION;
6
7 SELECT * FROM LOG_HABITACION;
8 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_HABITACION;
9
10 SELECT * FROM LOG_ACTIVIDAD;
11 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_ACTIVIDAD;
```

## → Tabla LOG\_ACTIVIDAD

The screenshot shows the DeVisualizer Free 24.2.1 interface. In the top navigation bar, 'Database' is selected. The main area displays a script named 'p2\_bases\_2\_consultas.sql' containing the following SQL code:

```
1 SELECT * FROM PACIENTE;
2 SELECT COUNT(*) AS "Cantidad de Registros" FROM PACIENTE;
3
4 SELECT * FROM HABITACION;
5 SELECT COUNT(*) AS "Cantidad de Registros" FROM HABITACION;
6
7 SELECT * FROM LOG_HABITACION;
8 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_HABITACION;
9
10 SELECT * FROM LOG_ACTIVIDAD;
11 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_ACTIVIDAD;
```

Below the code, the results of the last query are shown in a table:

Cantidad de Registros
0

At the bottom of the interface, there is a status bar with various icons and information.

## DIA 13

### Tiempo que duró el backup Incremental 3

The screenshot shows the Docker Desktop interface. On the left sidebar, 'Containers' is selected, showing a single container named 'practica2-db-1'. The container details show it is 'Running (23 seconds ago)'. The 'Exec' tab is active, displaying a terminal session:

```
sh-5.1# time mysql -uroot -p1234 practica2 < ./day3_backup_incremental.sql
mysql: [Warning] Using a password on the command line interface can be insecure.

real 0m1.179s
user 0m0.022s
sys 0m0.001s
sh-5.1#
```

At the bottom of the screen, the Windows taskbar is visible with various icons and system status.

## SELECT \* FROM cada tabla

→ Tabla PACIENTE

The screenshot shows the DBVisualizer Free 24.2.1 interface. The main window displays a SQL script named 'p2\_bases\_2\_consultas.sql' with the following content:

```
1 SELECT * FROM PACIENTE;
2 SELECT COUNT(*) AS "Cantidad de Registros" FROM PACIENTE;
3
4 SELECT * FROM HABITACION;
5 SELECT COUNT(*) AS "Cantidad de Registros" FROM HABITACION;
6
7 SELECT * FROM LOG_HABITACION;
8 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_HABITACION;
9
10 SELECT * FROM LOG_ACTIVIDAD;
11 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_ACTIVIDAD;
```

The results pane shows the output of the first query:

idPaciente	edad	genero

A message box indicates: "Result set is empty."

At the bottom right, the status bar shows: 0.000/0.002 sec 0/3 0-0 67M of 2048M 10:49 a.m. ENG 5/09/2024.

→ Tabla HABITACION

The screenshot shows the DBVisualizer Free 24.2.1 interface. The main window displays a SQL script named 'p2\_bases\_2\_consultas.sql' with the following content:

```
1 SELECT * FROM PACIENTE;
2 SELECT COUNT(*) AS "Cantidad de Registros" FROM PACIENTE;
3
4 SELECT * FROM HABITACION;
5 SELECT COUNT(*) AS "Cantidad de Registros" FROM HABITACION;
6
7 SELECT * FROM LOG_HABITACION;
8 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_HABITACION;
9
10 SELECT * FROM LOG_ACTIVIDAD;
11 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_ACTIVIDAD;
```

The results pane shows the output of the first query:

idHabitacion	habitacion

A message box indicates: "Result set is empty."

At the bottom right, the status bar shows: 0.005/0.001 sec 0/2 0-0 67M of 2048M 10:50 a.m. ENG 5/09/2024.

## → Tabla LOG\_HABITACION

File Edit View Database SQL Commander Tools Window Help

Databases Scripts Favorites t:p2\_bases\_2\_consultas.sql MySQL 8

Database Connection Sticky Database Schema Max Rows Max Chars

1 SELECT \* FROM PACIENTE;

2 SELECT COUNT(\*) AS "Cantidad de Registros" FROM PACIENTE;

3

4 SELECT \* FROM HABITACION;

5 SELECT COUNT(\*) AS "Cantidad de Registros" FROM HABITACION;

6

7 SELECT \* FROM LOG\_HABITACION;

8 SELECT COUNT(\*) AS "Cantidad de Registros" FROM LOG\_HABITACION;

9

10 SELECT \* FROM LOG\_ACTIVIDAD;

11 SELECT COUNT(\*) AS "Cantidad de Registros" FROM LOG\_ACTIVIDAD;

12

7 / 1 [170] INS

Log t:1: LOG\_HABITACION [1000] x

	id_log_habitacion	timestampx	statusx	idHabitacion
1	1	1/3/2021 7:19:33 a.m.	Sala no disponible.	12
2	2	2/3/2021 7:20:50 a.m.	Inicia limpieza.	12
3	3	3/3/2021 7:20:55 a.m.	Sala disponible.	12
4	4	4/3/2021 7:28:15 a.m.	Sala no disponible.	12
5	5	5/3/2021 7:29:42 a.m.	Inicia limpieza.	12
6	6	6/3/2021 7:30:31 a.m.	Sala disponible.	12
7	7	7/3/2021 7:38:38 a.m.	Sala no disponible.	13
8	8	8/3/2021 7:39:59 a.m.	Inicia limpieza.	13
9	9	9/3/2021 7:40:10 a.m.	Sala disponible.	13
10	10	10/3/2021 7:42:34 a.m.	Sala no disponible.	1
11	11	11/3/2021 7:43:14 a.m.	Inicia limpieza.	1
12	12	12/3/2021 7:46:07 a.m.	Sala disponible.	1
13	13	13/3/2021 7:54:19 a.m.	Sala disponible.	2
14	14	14/3/2021 7:54:33 a.m.	Inicia limpieza.	2
15	15	15/3/2021 7:55:43 a.m.	Sala no disponible.	12
16	16	16/3/2021 7:56:09 a.m.	Inicia limpieza.	12
17	17	17/3/2021 7:56:58 a.m.	Sala disponible.	12
18	18	18/3/2021 7:57:19 a.m.	Sala disponible.	2
19	19	19/3/2021 7:59:14 a.m.	Sala no disponible.	6
20	20	20/3/2021 7:59:51 a.m.	Inicia limpieza.	6
21	21	21/3/2021 8:00:18 a.m.	Sala no disponible.	13

Format: <Select a Cell>

0.011/0.017 sec 1-22 1000/4 1/67M of 2048M 1050 k. m. ENG 5/09/2024

## → Tabla LOG\_ACTIVIDAD

File Edit View Database SQL Commander Tools Window Help

Databases Scripts Favorites t:p2\_bases\_2\_consultas.sql MySQL 8

Database Connection Sticky Database Schema Max Rows Max Chars

1 SELECT \* FROM PACIENTE;

2 SELECT COUNT(\*) AS "Cantidad de Registros" FROM PACIENTE;

3

4 SELECT \* FROM HABITACION;

5 SELECT COUNT(\*) AS "Cantidad de Registros" FROM HABITACION;

6

7 SELECT \* FROM LOG\_HABITACION;

8 SELECT COUNT(\*) AS "Cantidad de Registros" FROM LOG\_HABITACION;

9

10 SELECT \* FROM LOG\_ACTIVIDAD;

11 SELECT COUNT(\*) AS "Cantidad de Registros" FROM LOG\_ACTIVIDAD;

12

10 / 1 [265] INS

Log t:1: LOG\_ACTIVIDAD [0] x

	id_log_actividad	timestampx	actividad	idpaciente	idHabitacion

Result set is empty.

Format: <Select a Cell>

0.006/0.002 sec 0/5 0-0 1/67M of 2048M 1050 k. m. ENG 5/09/2024

## SELECT COUNT(\*) FROM cada tabla

→ Tabla PACIENTE

The screenshot shows the Delveilizer Free 24.2.1 interface. The main window displays a SQL script named 'p2\_bases\_2\_consultas.sql' with the following content:

```
1 SELECT * FROM PACIENTE;
2 SELECT COUNT(*) AS "Cantidad de Registros" FROM PACIENTE;
3
4 SELECT * FROM HABITACION;
5 SELECT COUNT(*) AS "Cantidad de Registros" FROM HABITACION;
6
7 SELECT * FROM LOG_HABITACION;
8 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_HABITACION;
9
10 SELECT * FROM LOG_ACTIVIDAD;
11 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_ACTIVIDAD;
```

The results pane shows a single row of data for the 'PACIENTE' table:

Cantidad de Registros
0

At the bottom right of the interface, there is a status bar with the text: '0.005/0.002 sec 1/1 1-1 07M of 2048M 10:52 a.m. ENG 5/09/2024'.

→ Tabla HABITACION

The screenshot shows the Delveilizer Free 24.2.1 interface. The main window displays a SQL script named 'p2\_bases\_2\_consultas.sql' with the following content:

```
1 SELECT * FROM PACIENTE;
2 SELECT COUNT(*) AS "Cantidad de Registros" FROM PACIENTE;
3
4 SELECT * FROM HABITACION;
5 SELECT COUNT(*) AS "Cantidad de Registros" FROM HABITACION;
6
7 SELECT * FROM LOG_HABITACION;
8 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_HABITACION;
9
10 SELECT * FROM LOG_ACTIVIDAD;
11 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_ACTIVIDAD;
```

The results pane shows a single row of data for the 'HABITACION' table:

Cantidad de Registros
0

At the bottom right of the interface, there is a status bar with the text: '0.004/0.002 sec 1/1 1-1 07M of 2048M 10:52 a.m. ENG 5/09/2024'.

## → Tabla LOG\_HABITACION

The screenshot shows the DBVisualizer Free interface with the following details:

- File Bar:** File, Edit, View, Database, SQL Commander, Tools, Window, Help.
- Toolbars:** Standard toolbar with icons for New, Open, Save, Print, etc.
- Left Sidebar:**
  - Databases: ht3\_bases\_2.sql, p2\_bases\_2\_create\_db.sql, p2\_bases\_2\_consultas.sql
  - Scripts: Bookmarks (ht3\_bases\_2.sql, p2\_bases\_2\_create\_db.sql, p2\_bases\_2\_consultas.sql)
  - Monitors
- Central Area:**
  - Database Connection:** MySQL 8
  - SQL Editor:** Contains the following SQL code:

```
1 SELECT * FROM PACIENTE;
2 SELECT COUNT(*) AS "Cantidad de Registros" FROM PACIENTE;
3
4 SELECT * FROM HABITACION;
5 SELECT COUNT(*) AS "Cantidad de Registros" FROM HABITACION;
6
7 SELECT * FROM LOG_HABITACION;
8 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_HABITACION;
9
10 SELECT * FROM LOG_ACTIVIDAD;
11 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_ACTIVIDAD;
```
  - Results:** A table titled "t: LOG\_HABITACION [1]" with one row:

Cantidad de Registros
34617
- Bottom Status Bar:** Shows execution time (0.011/0.002 sec), rows (1/1), and memory usage (69M of 2048M).

## → Tabla LOG\_ACTIVIDAD

The screenshot shows the DBVisualizer Free interface with the following details:

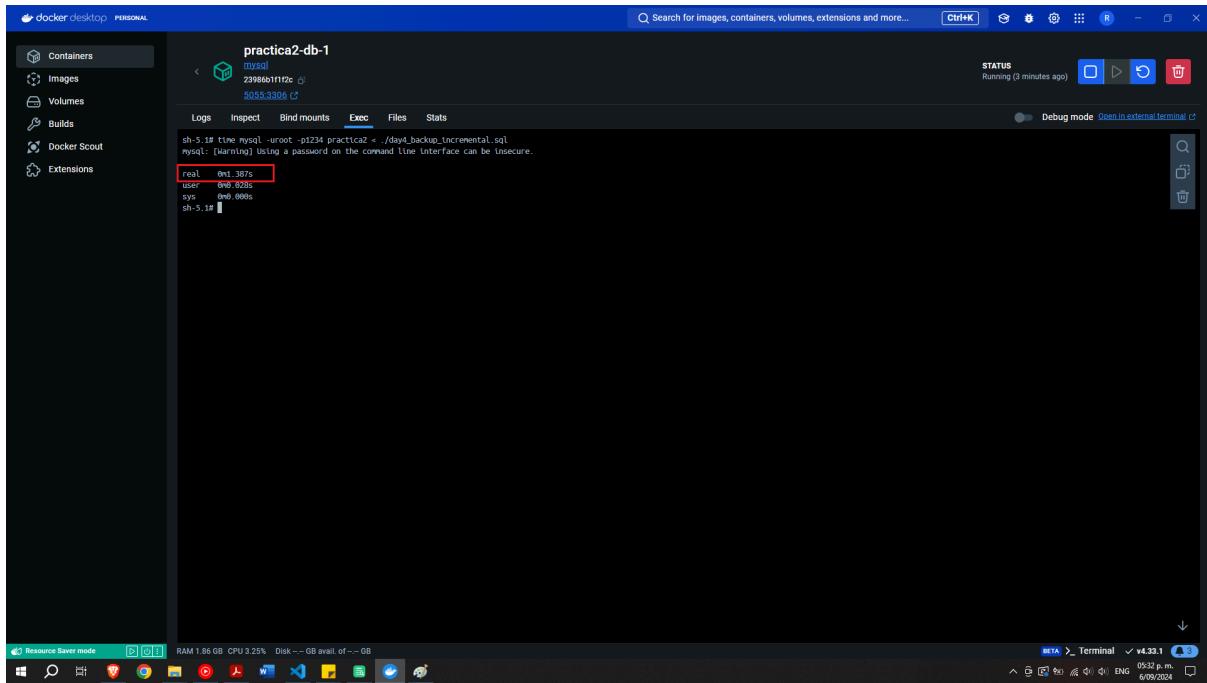
- File Bar:** File, Edit, View, Database, SQL Commander, Tools, Window, Help.
- Toolbars:** Standard toolbar with icons for New, Open, Save, Print, etc.
- Left Sidebar:**
  - Databases: ht3\_bases\_2.sql, p2\_bases\_2\_create\_db.sql, p2\_bases\_2\_consultas.sql
  - Scripts: Bookmarks (ht3\_bases\_2.sql, p2\_bases\_2\_create\_db.sql, p2\_bases\_2\_consultas.sql)
  - Monitors
- Central Area:**
  - Database Connection:** MySQL 8
  - SQL Editor:** Contains the following SQL code:

```
1 SELECT * FROM PACIENTE;
2 SELECT COUNT(*) AS "Cantidad de Registros" FROM PACIENTE;
3
4 SELECT * FROM HABITACION;
5 SELECT COUNT(*) AS "Cantidad de Registros" FROM HABITACION;
6
7 SELECT * FROM LOG_HABITACION;
8 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_HABITACION;
9
10 SELECT * FROM LOG_ACTIVIDAD;
11 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_ACTIVIDAD;
```
  - Results:** A table titled "t: LOG\_ACTIVIDAD [1]" with one row:

Cantidad de Registros
0
- Bottom Status Bar:** Shows execution time (0.005/0.001 sec), rows (1/1), and memory usage (69M of 2048M).

## DIA 14 (1)

### Tiempo que duró el backup Incremental 4

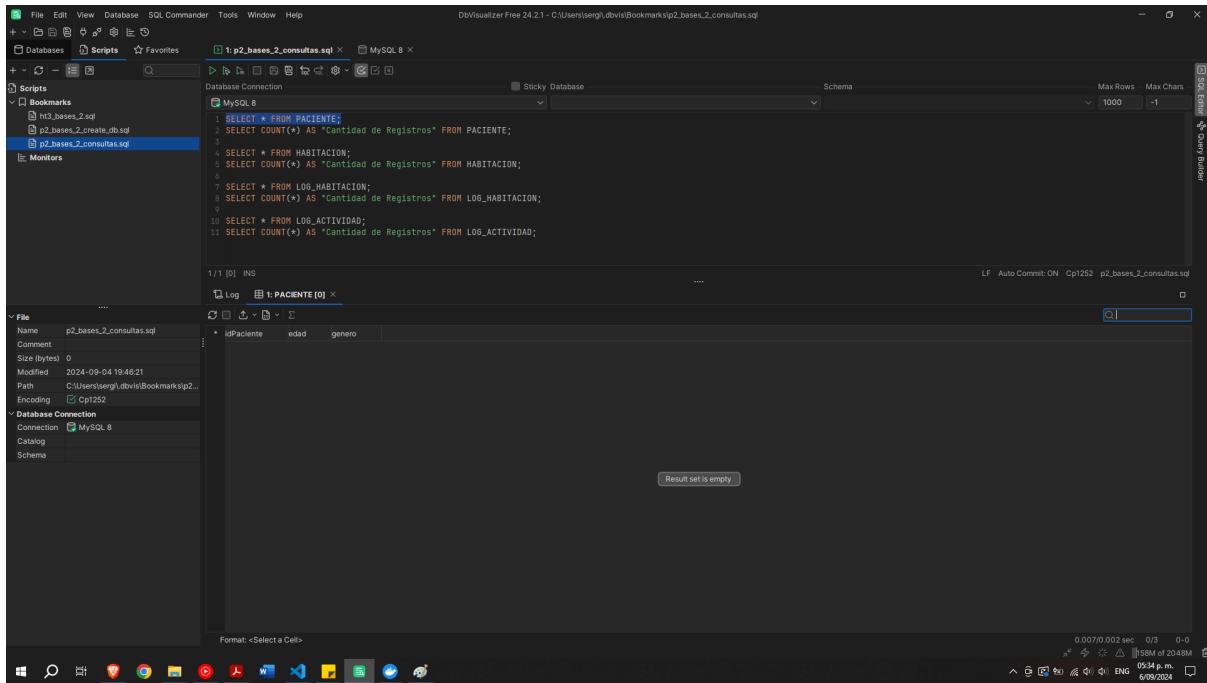


```
sh-5.1# time mysql -uroot -p1234 practica2 < ./dayd_backup_incremental.sql
mysql: [Warning] Using a password on the command line interface can be insecure.

real 0m1.367s
user 0m0.828s
sys 0m0.660s
sh-5.1#
```

### SELECT \* FROM cada tabla

→ Tabla PACIENTE



```
1 SELECT * FROM PACIENTE;
2 SELECT COUNT(*) AS "Cantidad de Registros" FROM PACIENTE;
3
4 SELECT * FROM HABITACION;
5 SELECT COUNT(*) AS "Cantidad de Registros" FROM HABITACION;
6
7 SELECT * FROM LOG_HABITACION;
8 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_HABITACION;
9
10 SELECT * FROM LOG_ACTIVIDAD;
11 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_ACTIVIDAD;
```

## → Tabla HABITACION

The screenshot shows the DbVisualizer Free 24.2.1 interface. The main window displays a SQL script titled 'p2\_bases\_2\_consultas.sql' with the following code:

```
1 SELECT * FROM PACIENTE;
2 SELECT COUNT(*) AS "Cantidad de Registros" FROM PACIENTE;
3
4 SELECT * FROM HABITACION;
5 SELECT COUNT(*) AS "Cantidad de Registros" FROM HABITACION;
6
7 SELECT * FROM LOG_HABITACION;
8 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_HABITACION;
9
10 SELECT * FROM LOG_ACTIVIDAD;
11 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_ACTIVIDAD;
```

The results pane shows a table named 'HABITACION [0]' with one row:

idHabitacion	habitacion

A message box indicates: 'Result set is empty.'

Below the results pane, the status bar shows: 0.005/0.002 sec 0/2 0-0 058M of 2048M 0534 p.m. ENG 6/09/2024

## → Tabla LOG\_HABITACION

The screenshot shows the DbVisualizer Free 24.2.1 interface. The main window displays a SQL script titled 'p2\_bases\_2\_consultas.sql' with the same code as the previous screenshot:

```
1 SELECT * FROM PACIENTE;
2 SELECT COUNT(*) AS "Cantidad de Registros" FROM PACIENTE;
3
4 SELECT * FROM HABITACION;
5 SELECT COUNT(*) AS "Cantidad de Registros" FROM HABITACION;
6
7 SELECT * FROM LOG_HABITACION;
8 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_HABITACION;
9
10 SELECT * FROM LOG_ACTIVIDAD;
11 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_ACTIVIDAD;
```

The results pane shows a table named 'LOG\_HABITACION [0]' with one row:

id_log_habitacion	timestampx	statusx	idHabitacion

A message box indicates: 'Result set is empty.'

Below the results pane, the status bar shows: 0.006/0.003 sec 0/4 0-0 058M of 2048M 0535 p.m. ENG 6/09/2024

## → Tabla LOG\_ACTIVIDAD

The screenshot shows the DBVisualizer Free interface with the 'LOG\_ACTIVIDAD' table selected. The table has columns: idLog\_Actividad, timestamp, actividad, idPaciente, and idHabitacion. The data shows various patient interactions over time, such as patients checking in at reception or being treated by medical staff.

	idLog_Actividad	timestamp	actividad	idPaciente	idHabitacion
1	1	15/9/2021 7:12:14 a.m.	Paciente inicia el registro.	134247	10
2	2	15/9/2021 7:13:16 a.m.	Paciente recibe papelera en recepcion.	134247	10
3	3	15/9/2021 7:15:54 a.m.	Paciente entrega papelera.	134247	10
4	4	15/9/2021 7:18:33 a.m.	Recepcionista establece la condición del paciente como 'Urgente'. Paciente es enviado a la sala de espera para esperar revisión.	134247	10
5	5	15/9/2021 7:19:47 a.m.	Enfermera comienza la revisión del paciente.	134247	10
6	6	15/9/2021 7:19:53 a.m.	Revisión determinó que el paciente es tipo 2 y su condición es 'Urgente'.	134247	10
7	7	15/9/2021 7:20:34 a.m.	Medico inicia con el tratamiento del paciente.	135641	10
8	8	15/9/2021 7:21:50 a.m.	Paciente inicia el registro.	135641	10
9	9	15/9/2021 7:23:00 a.m.	Paciente recibe papelera en recepción.	135641	10
10	10	15/9/2021 7:23:20 a.m.	Paciente inicia el registro.	180487	10
11	11	15/9/2021 7:23:44 a.m.	Paciente entrega papelera.	135641	12
12	12	15/9/2021 7:28:18 a.m.	Recepcionista establece la condición del paciente como 'Estable'. Paciente es enviado a la sala de espera para esperar revisión.	135641	12
13	13	15/9/2021 7:28:43 a.m.	Enfermera recibe paciente en recepción.	180487	5
14	14	15/9/2021 7:29:23 a.m.	Enfermera comienza la revisión del paciente.	135641	5
15	15	15/9/2021 7:29:37 a.m.	Paciente entrega papelera.	180487	7
16	16	15/9/2021 7:39:15 a.m.	Revisión determinó que el paciente es tipo 2 y su condición es 'Urgente'.	135641	7
17	17	15/9/2021 7:39:47 a.m.	Recepcionista establece la condición del paciente como 'Urgente'. Paciente es enviado a la sala de espera para esperar revisión.	180487	7
18	18	15/9/2021 7:39:54 a.m.	Medico inicia con el tratamiento del paciente.	135641	7
19	19	15/9/2021 7:39:55 a.m.	Enfermera comienza la revisión del paciente.	180487	10
20	20	15/9/2021 7:39:58 a.m.	Revisión determinó que el paciente es tipo 5 y su condición es 'Urgente'.	180487	10
21	21	15/9/2021 7:39:58 a.m.	Medico inicia con el tratamiento del paciente.	180487	10

## SELECT COUNT(\*) FROM cada tabla

### → Tabla PACIENTE

The screenshot shows the DBVisualizer Free interface with the 'PACIENTE' table selected. The table has a single column: Cantidad de Registros, which contains the value 0.

Cantidad de Registros
0

## → Tabla HABITACION

The screenshot shows the DBVisualizer Free 24.2.1 interface. The main window displays a SQL script named 'p2\_bases\_2\_consultas.sql' with the following code:

```
1 SELECT * FROM PACIENTE;
2 SELECT COUNT(*) AS "Cantidad de Registros" FROM PACIENTE;
3
4 SELECT * FROM HABITACION;
5 SELECT COUNT(*) AS "Cantidad de Registros" FROM HABITACION;
6
7 SELECT * FROM LOG_HABITACION;
8 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_HABITACION;
9
10 SELECT * FROM LOG_ACTIVIDAD;
11 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_ACTIVIDAD;
```

The results pane shows the output for the fifth query:

Cantidad de Registros
0

At the bottom right, the status bar indicates: 0.004/0.001 sec 1/1 1-1 0536 p.m. 6/09/2024.

## → Tabla LOG\_HABITACION

The screenshot shows the DBVisualizer Free 24.2.1 interface. The main window displays a SQL script named 'p2\_bases\_2\_consultas.sql' with the same code as the previous screenshot:

```
1 SELECT * FROM PACIENTE;
2 SELECT COUNT(*) AS "Cantidad de Registros" FROM PACIENTE;
3
4 SELECT * FROM HABITACION;
5 SELECT COUNT(*) AS "Cantidad de Registros" FROM HABITACION;
6
7 SELECT * FROM LOG_HABITACION;
8 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_HABITACION;
9
10 SELECT * FROM LOG_ACTIVIDAD;
11 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_ACTIVIDAD;
```

The results pane shows the output for the eighth query:

Cantidad de Registros
0

At the bottom right, the status bar indicates: 0.008/0.002 sec 1/1 1-1 0536 p.m. 6/09/2024.

## → Tabla LOG\_ACTIVIDAD

The screenshot shows the DBVisualizer Free 24.2.1 interface. In the top navigation bar, the 'Database' tab is selected. The main area displays a SQL script named 'p2\_bases\_2\_consultas.sql' with the following content:

```
1 SELECT * FROM PACIENTE;
2 SELECT COUNT(*) AS "Cantidad de Registros" FROM PACIENTE;
3
4 SELECT * FROM HABITACION;
5 SELECT COUNT(*) AS "Cantidad de Registros" FROM HABITACION;
6
7 SELECT * FROM LOG_HABITACION;
8 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_HABITACION;
9
10 SELECT * FROM LOG_ACTIVIDAD;
11 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_ACTIVIDAD;
```

The status bar at the bottom indicates: 0.012/0.002 sec, 1/1, 0537 p.m., ENG, 6/09/2024.

## DIA 14 (2)

### Tiempo que duró el backup Incremental 5

The screenshot shows the Docker Desktop interface. A MySQL container named 'practica2-db-1' is running. The terminal window shows the following command being run:

```
sh-5.1# time mysql -uroot -p1234 practica2 < ./day5/backup_incremental.sql
mysql: [Warning] Using a password on the command line interface can be insecure.
```

The terminal output shows the execution times:

```
real 0m2.136s
user 0m0.826s
sys 0m0.068s
sh-5.1#
```

The status bar at the bottom indicates: Resource Saver mode, RAM 1 93 GB, CPU 3.75%, Disk -- GB avail. of -- GB, BETA, Terminal v4.33.1, 0541 p.m., ENG, 6/09/2024.

## SELECT \* FROM cada tabla

→ Tabla PACIENTE

The screenshot shows the DBVisualizer Free interface. The main window displays a SQL script named 't:p2\_bases\_2\_consultas.sql' with the following code:

```
1 SELECT * FROM PACIENTE;
2 SELECT COUNT(*) AS "Cantidad de Registros" FROM PACIENTE;
3
4 SELECT * FROM HABITACION;
5 SELECT COUNT(*) AS "Cantidad de Registros" FROM HABITACION;
6
7 SELECT * FROM LOG_HABITACION;
8 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_HABITACION;
9
10 SELECT * FROM LOG_ACTIVIDAD;
11 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_ACTIVIDAD;
```

The results pane shows a table with columns 'idPaciente', 'edad', and 'genero'. A message 'Result set is empty.' is displayed at the bottom.

→ Tabla HABITACION

The screenshot shows the DBVisualizer Free interface. The main window displays a SQL script named 't:p2\_bases\_2\_consultas.sql' with the same code as the previous screenshot:

```
1 SELECT * FROM PACIENTE;
2 SELECT COUNT(*) AS "Cantidad de Registros" FROM PACIENTE;
3
4 SELECT * FROM HABITACION;
5 SELECT COUNT(*) AS "Cantidad de Registros" FROM HABITACION;
6
7 SELECT * FROM LOG_HABITACION;
8 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_HABITACION;
9
10 SELECT * FROM LOG_ACTIVIDAD;
11 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_ACTIVIDAD;
```

The results pane shows a table with columns 'idHabitacion' and 'habitacion'. A message 'Result set is empty.' is displayed at the bottom.

## → Tabla LOG\_HABITACION

The screenshot shows the DBVisualizer interface with a query window titled "1: p2\_bases\_2\_consultas.sql" running against a MySQL 8 database. The query retrieves data from four tables: PACIENTE, HABITACION, LOG\_HABITACION, and LOG\_ACTIVIDAD. The results are displayed in a table titled "1: LOG\_HABITACION [0]" with columns: id\_log\_habitacion, timestampx, statusx, and idhabitacion. A note at the bottom of the results table states "Result set is empty." The status bar at the bottom right indicates "0.005/0.002 sec" and "0/4 0-0".

```

1 SELECT * FROM PACIENTE;
2 SELECT COUNT(*) AS "Cantidad de Registros" FROM PACIENTE;
3
4 SELECT * FROM HABITACION;
5 SELECT COUNT(*) AS "Cantidad de Registros" FROM HABITACION;
6
7 SELECT * FROM LOG_HABITACION;
8 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_HABITACION;
9
10 SELECT * FROM LOG_ACTIVIDAD;
11 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_ACTIVIDAD;

```

## → Tabla LOG\_ACTIVIDAD

The screenshot shows the DBVisualizer interface with a query window titled "1: p2\_bases\_2\_consultas.sql" running against a MySQL 8 database. The query retrieves data from four tables: PACIENTE, HABITACION, LOG\_HABITACION, and LOG\_ACTIVIDAD. The results are displayed in a table titled "1: LOG\_ACTIVIDAD [1000]" with columns: id\_log\_actividad, timestampx, actividad, idpaciente, and idhabitacion. The status bar at the bottom right indicates "0.006/0.016 sec" and "1000/1000 1-22".

```

1 SELECT * FROM PACIENTE;
2 SELECT COUNT(*) AS "Cantidad de Registros" FROM PACIENTE;
3
4 SELECT * FROM HABITACION;
5 SELECT COUNT(*) AS "Cantidad de Registros" FROM HABITACION;
6
7 SELECT * FROM LOG_HABITACION;
8 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_HABITACION;
9
10 SELECT * FROM LOG_ACTIVIDAD;
11 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_ACTIVIDAD;

```

id_log_actividad	timestampx	actividad	idpaciente	idhabitacion
1	1/5/2021 7:12:14 a.m.	Paciente inicia el registro.	104247	10
2	1/5/2021 7:12:30 a.m.	Paciente recibe papelera en recepcion.	104247	10
3	1/5/2021 7:15:54 a.m.	Paciente entrega papelera.	104247	10
4	1/5/2021 7:18:23 a.m.	Recepcionista establece la condicion del paciente como 'Urgente'. Paciente es enviado a la sala de espera para esperar revision.	104247	10
5	1/5/2021 7:19:47 a.m.	Recepcionista establece la condicion del paciente como 'Urgente'. Paciente es enviado a la sala de espera para esperar revision.	104247	10
6	1/5/2021 7:19:33 a.m.	Revision determino que el paciente es tipo 2 y su condicion es 'Urgente'.	104247	10
7	1/5/2021 7:20:24 a.m.	Revision determino que el paciente es tipo 2 y su condicion es 'Urgente'.	104247	10
8	1/5/2021 7:20:50 a.m.	Medico inicia con el tratamiento del paciente.	135641	10
9	1/5/2021 7:23:07 a.m.	Paciente recibe papelera en recepcion.	104247	10
10	1/5/2021 7:23:20 a.m.	Paciente inicia el registro.	104247	10
11	1/5/2021 7:35:44 a.m.	Paciente entrega papelera.	104247	10
12	1/5/2021 7:38:18 a.m.	Recepcionista establece la condicion del paciente como 'Estable'. Paciente es enviado a la sala de espera para esperar revision.	104247	10
13	1/5/2021 7:38:43 a.m.	Paciente recibe papelera en recepcion.	104247	10
14	1/5/2021 7:37:33 a.m.	Enfermera comienza la revision del paciente.	104247	10
15	1/5/2021 7:39:17 a.m.	Paciente entrega papelera.	104247	10
16	1/5/2021 7:39:15 a.m.	Revision determino que el paciente es tipo 2 y su condicion es 'Urgente'.	104247	10
17	1/5/2021 7:39:47 a.m.	Recepcionista establece la condicion del paciente como 'Urgente'. Paciente es enviado a la sala de espera para esperar revision.	104247	10
18	1/5/2021 7:30:34 a.m.	Medico inicia con el tratamiento del paciente.	104247	10
19	1/5/2021 7:30:53 a.m.	Enfermera comienza la revision del paciente.	104247	10
20	1/5/2021 7:36:38 a.m.	Revision determino que el paciente es tipo 5 y su condicion es 'Urgente'.	104247	10
21	1/5/2021 7:37:33 a.m.	Medico inicia con el tratamiento del paciente.	104247	10

## SELECT COUNT(\*) FROM cada tabla

→ Tabla PACIENTE

The screenshot shows the DBVisualizer Free interface. The main window displays a SQL script named 'p2\_bases\_2\_consultas.sql' with the following content:

```
1 SELECT * FROM PACIENTE;
2 SELECT COUNT(*) AS "Cantidad de Registros" FROM PACIENTE;
3
4 SELECT * FROM HABITACION;
5 SELECT COUNT(*) AS "Cantidad de Registros" FROM HABITACION;
6
7 SELECT * FROM LOG_HABITACION;
8 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_HABITACION;
9
10 SELECT * FROM LOG_ACTIVIDAD;
11 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_ACTIVIDAD;
```

The results pane shows the output of the second query:

Cantidad de Registros
0

The status bar at the bottom right indicates: 0.006/0.001 sec 1/1 1-1 564 of 2048M 0544 p.m. 6/09/2024.

→ Tabla HABITACION

The screenshot shows the DBVisualizer Free interface. The main window displays a SQL script named 'p2\_bases\_2\_consultas.sql' with the following content:

```
1 SELECT * FROM PACIENTE;
2 SELECT COUNT(*) AS "Cantidad de Registros" FROM PACIENTE;
3
4 SELECT * FROM HABITACION;
5 SELECT COUNT(*) AS "Cantidad de Registros" FROM HABITACION;
6
7 SELECT * FROM LOG_HABITACION;
8 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_HABITACION;
9
10 SELECT * FROM LOG_ACTIVIDAD;
11 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_ACTIVIDAD;
```

The results pane shows the output of the fifth query:

Cantidad de Registros
0

The status bar at the bottom right indicates: 0.005/0.001 sec 1/1 1-1 564 of 2048M 0544 p.m. 6/09/2024.

→ Tabla LOG\_HABITACION

The screenshot shows the MySQL Workbench interface. The top menu bar includes File, Edit, View, Database, SQL Commander, Tools, Window, Help, and a tab for Databases, Scripts, or Favorites. The title bar indicates the application is "DbVisualizer 24.2.1 - C:\Users\sergi\Downloads\bookmarks\p2\_bases\_2\_consultas.sql". The main area features a "Database Connection" dropdown set to "MySQL 8", a "Schema" dropdown, and filters for "Max Rows" (1000) and "Max Chars" (-1). A "Sticky Database" checkbox is also present. The left sidebar contains sections for Scripts, Bookmarks, Monitors, and a detailed view for the selected file "p2\_bases\_2\_consultas.sql". This file details its name, path, size (0 bytes), modification date (2024-09-04 19:46:21), and encoding (Cp1252). It also lists the database connection ("MySQL 8") and catalog ("p2bases\_2"). The central workspace displays a query editor with the following SQL code:

```
1 SELECT * FROM PACIENTE;
2
3 SELECT COUNT(*) AS "Cantidad de Registros" FROM PACIENTE;
4
5 SELECT * FROM HABITACION;
6
7 SELECT COUNT(*) AS "Cantidad de Registros" FROM HABITACION;
8
9 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_HABITACION;
10
11 SELECT * FROM LOG_ACTIVIDAD;
12
13 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_ACTIVIDAD;
```

The status bar at the bottom shows the current time as 05:44 p.m. on 6/09/2024.

→ Tabla LOG\_ACTIVIDAD

The screenshot shows the DBVisualizer Free 24.2.1 interface. The top menu bar includes File, Edit, View, Database, SQL Commander, Tools, Window, and Help. The title bar displays "DBVisualizer Free 24.2.1 - C:\Users\sergi\dbvis\Bookmarks\p2\_bases\_2\_consultas.sql". The left sidebar contains a Databases tree (with 'Databases' selected), a Scripts tree (with 'p2\_bases\_2\_consultas.sql' selected), and a Monitors section. The main workspace has a 'Database Connection' dropdown set to 'MySQL 8'. A 'Sticky Database' checkbox is checked. The 'Schema' dropdown is empty. On the right, there are 'Max Rows' (1000) and 'Max Chars' (-1) settings. The central area shows a SQL editor with the following code:

```
1 SELECT * FROM PACIENTE;
2 SELECT COUNT(*) AS "Cantidad de Registros" FROM PACIENTE;
3
4 SELECT * FROM HABITACION;
5 SELECT COUNT(*) AS "Cantidad de Registros" FROM HABITACION;
6
7 SELECT * FROM LOG_HABITACION;
8 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_HABITACION;
9
10 SELECT * FROM LOG_ACTIVIDAD;
11 SELECT COUNT(*) AS "Cantidad de Registros" FROM LOG_ACTIVIDAD;
```

Below the editor, it says "11/1 [294] INS". The bottom status bar shows "LF Auto Commit: ON Cp1252 p2\_bases\_2\_consultas.sql".

The bottom-left pane shows file details for "p2\_bases\_2\_consultas.sql":

- Name: p2\_bases\_2\_consultas.sql
- Comment:
- Size (bytes): 0
- Modified: 2024-09-04 19:46:21
- Path: C:\Users\sergi\dbvis\Bookmarks\p2...
- Encoding: Cp1252

The bottom-right pane shows a results grid for the query "SELECT COUNT(\*) AS 'Cantidad de Registros' FROM LOG\_ACTIVIDAD;". It has one row with value 67684.

## ANÁLISIS DE RESULTADOS

Tabla de comparación de tiempos para restaurar el backup

Día	Completo	Incremental
1	6.1 s	1.793s
2	8.9 s	0.178s
3	5.46 s	1.79s
4	12.2	1.387s
5	10.24	2.136s

Como se puede observar el tiempo para restaurar un backup incremental es menor ya que es de una sola tabla

## CONCLUSIONES

### Alvaro Esaú Arenas

1. Los respaldos son esenciales para garantizar la integridad y disponibilidad de los datos. En un entorno donde la pérdida de datos puede tener consecuencias críticas, como en nuestro caso una clínica médica que puede llegar a manejar grandes volúmenes de información, los backups permiten recuperar rápidamente los datos ante fallos inesperados, ataques o errores humanos.
2. Para bases de datos grandes con un alto volumen de datos, se recomienda una combinación de respaldos completos e incrementales. Realizar un respaldo completo por ejemplo, semanalmente y respaldos incrementales con mayor frecuencia, esto puede optimizar el tiempo de restauración, asegurando que se mantenga un buen equilibrio entre eficiencia y facilidad de recuperación

### Jorge Antonio Perez

1. Los backups completos tienden a requerir más recursos, almacenamiento y tiempo para lograr ejecutarse por completo, pero aseguran la integridad de los datos ya que es una restauración más sencilla. A diferencia de los backups incrementales, que son relativamente más ligeros, veloces y eficientes para ejecutarse, sin embargo existe más riesgo a pérdidas de datos

debido a que es una restauración más compleja y depende de los backups anteriores para establecer la base de datos en un punto del tiempo en específico.

2. MySQL cuenta con una herramienta para crear backups de bases de datos llamada mysqldump. Se puede utilizar por medio de la interfaz de comandos y genera automáticamente los archivos .sql para poder ejecutarlos y restablecer la base de datos sin problemas.

### **Sergio Saúl Ralda Mejía**

1. No se detectó una diferencia significativa entre la restauración del backup incremental y el backup completo, ya que los tiempos de ambos procesos se mantuvieron en segundos con el volumen de datos manejado. Esto indica que, para el tamaño de datos actual, la restauración del backup incremental no ofrece una ventaja de tiempo relevante frente al backup completo
2. Con los tiempos de restauración obtenidos en esta prueba, optar por backups completos podría simplificar la administración de backups, eliminando la necesidad de mantener un seguimiento estricto de múltiples archivos incrementales.

### **Rubén Alejandro Ralda**

- 1.
- 2.