

Universidad de San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ciencias y Sistemas

PROYECTO #1

GRUPO # 2
23 de septiembre de 2022

CAPTURAS ANALIZADAS

Select de la tabla pacientes

The screenshot shows the MySQL Workbench interface. The query editor window titled "Query 1" contains the following SQL code:

```
1  SELECT * FROM bd2_proyecto1.habitacion LIMIT 20;
2  •  SELECT * FROM bd2_proyecto1.log_actividad LIMIT 20;
3  •  SELECT * FROM bd2_proyecto1.log_habitacion LIMIT 20;
4  •  SELECT * FROM bd2_proyecto1.paciente LIMIT 20;
5
6  •  SELECT * FROM Tabla ORDER BY primaryKey DESC LIMIT 20;
7
```

The results grid displays the following data:

idHabitacion	habitacion
1	Sala de examenes 1
2	Sala de examenes 2
3	Sala de examenes 3
4	Sala de examenes 4
5	Sala de imagenes 1
6	Sala de procedimientos 1
7	Sala de procedimientos 2
8	Sala de procedimientos 3
9	Sala de procedimientos 4
10	Repcion
11	Laboratorio
12	EstaciÃ³n de revisiÃ³n 1
13	EstaciÃ³n de revisiÃ³n 2
14	EstaciÃ³n de revisiÃ³n 3
15	EstaciÃ³n de revisiÃ³n 4
...	...
NULL	NULL

Log actividad

The screenshot shows the MySQL Workbench interface with a query editor titled "Query 1". The editor contains the following SQL code:

```

1 •   SELECT * FROM bd2_proyecto1.habitacion LIMIT 20;
2 •   SELECT * FROM bd2_proyecto1.log_actividad LIMIT 20;
3 •   SELECT * FROM bd2_proyecto1.log_habitacion LIMIT 20;
4 •   SELECT * FROM bd2_proyecto1.paciente LIMIT 20;
5
6 •   SELECT * FROM Tabla ORDER BY primaryKey DESC LIMIT 20;
7

```

Below the editor is a "Result Grid" pane with the following schema and data:

id_log_actividad	timestapx	actividad	HABITACION_IdHabitacion	PACIENTE_IdPaciente
NULL	NULL	NULL	NULL	NULL

Order by log actividad

The screenshot shows the MySQL Workbench interface with a query editor containing the following SQL code:

```

3 •   SELECT * FROM bd2_proyecto1.log_habitacion LIMIT 20;
4 •   SELECT * FROM bd2_proyecto1.paciente LIMIT 20;
5
6 •   SELECT * FROM bd2_proyecto1.habitacion ORDER BY idHabitacion DESC LIMIT 20;
7 •   SELECT * FROM bd2_proyecto1.log_actividad ORDER BY id_log_actividad DESC LIMIT 20;
8 •   SELECT * FROM bd2_proyecto1.log_habitacion ORDER BY idHabitacion, timestapx, statusx DESC LIMIT 20;
9 •   SELECT * FROM bd2_proyecto1.paciente ORDER BY idPaciente DESC LIMIT 20;
10
11 •  LOAD DATA INFILE 'C:/ProgramData/MySQL/MySQL Server 8.0/Habitaciones.csv'

```

Below the editor is a "Result Grid" pane with the following schema and data:

Id_log_actividad	timestapx	actividad	HABITACION_IdHabitacion	PACIENTE_IdPaciente
NULL	NULL	NULL	NULL	NULL

Order by log habitación

```
4 •   SELECT * FROM bd2_proyecto1.paciente LIMIT 20;
5
6 •   SELECT * FROM bd2_proyecto1.habitacion ORDER BY idHabitacion DESC LIMIT 20;
7 •   SELECT * FROM bd2_proyecto1.log_actividad ORDER BY id_log_actividad DESC LIMIT 20;
8 •   SELECT * FROM bd2_proyecto1.log_habitacion ORDER BY idHabitacion, timestampx, statusx DESC LIMIT 20;
9 •   SELECT * FROM bd2_proyecto1.paciente ORDER BY idPaciente DESC LIMIT 20;
10
11 •  LOAD DATA INFILE 'C:/ProgramData/MySQL/MySQL Server 8.0/Habitaciones.csv'
```

Paciente order by

```
8 •    SELECT * FROM bd2_proyecto1.log_habitacion ORDER BY idHabitacion, timestampx, statusx DESC LIMIT 20;
9 •    SELECT * FROM bd2_proyecto1.paciente ORDER BY idPaciente DESC LIMIT 20;
10
11 •   LOAD DATA INFILE'C:/ProgramData/MySQL/MySQL Server 8.0/Habitaciones.csv'
<
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

idPaciente	edad	genero
NULL	NULL	NULL

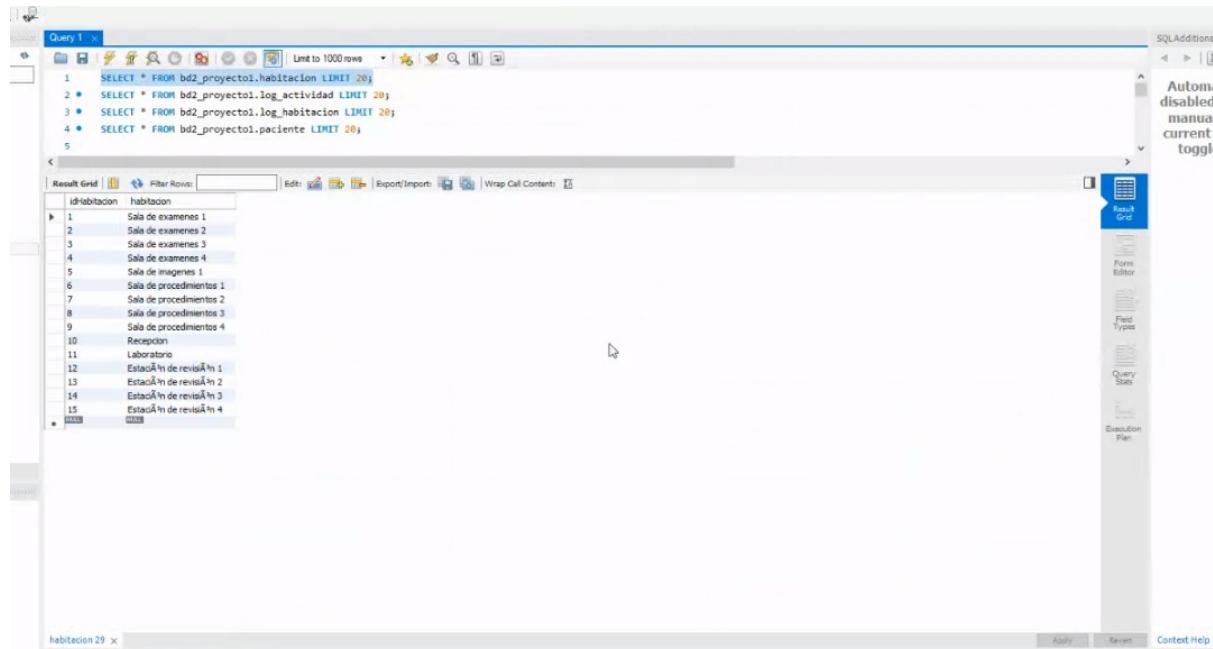
Count incremental 1

```

8 •   SELECT * FROM bd2_proyecto1.log_habitacion ORDER BY idHabitacion, timestampx, statusx DESC LIMIT 20;
9 •   SELECT * FROM bd2_proyecto1.paciente ORDER BY idPaciente DESC LIMIT 20;
10
11 •  SELECT * FROM
12      (SELECT COUNT(*) conteoHabitaciones FROM bd2_proyecto1.habitacion) AS conteoHabitaciones,
13      (SELECT COUNT(*) conteoLogActividad FROM bd2_proyecto1.log_actividad) AS conteoLogActividad,
14      (SELECT COUNT(*) conteoLogHabitacion FROM bd2_proyecto1.log_habitacion) AS conteoLogHabitacion,
15      (SELECT COUNT(*) conteoPaciente FROM bd2_proyecto1.paciente) AS conteoPaciente;

```

Select from log actividad



The screenshot shows the MySQL Workbench interface with a query editor titled "Query 1". The query is:

```

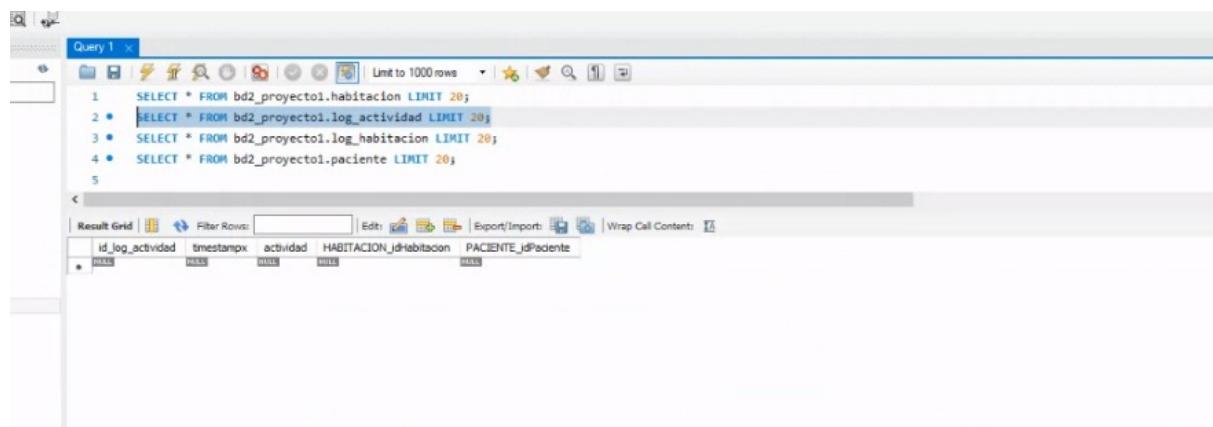
1 SELECT * FROM bd2_proyecto1.log_actividad LIMIT 20;
2 • SELECT * FROM bd2_proyecto1.habitacion LIMIT 20;
3 • SELECT * FROM bd2_proyecto1.log_habitacion LIMIT 20;
4 • SELECT * FROM bd2_proyecto1.paciente LIMIT 20;
5

```

The results grid displays the following data:

	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
...	...
20	...

Select from log habitación



The screenshot shows the MySQL Workbench interface with a query editor titled "Query 1". The query is:

```

1 SELECT * FROM bd2_proyecto1.habitacion LIMIT 20;
2 • SELECT * FROM bd2_proyecto1.log_actividad LIMIT 20;
3 • SELECT * FROM bd2_proyecto1.log_habitacion LIMIT 20;
4 • SELECT * FROM bd2_proyecto1.paciente LIMIT 20;
5

```

The results grid displays the following data:

	id_log_actividad	timestamppx	actividad	HABITACION_idHabitacion	PACIENTE_idPaciente
1	1	2023-09-18 10:00:00	Entrada	1	1
2	2	2023-09-18 10:00:00	Salida	1	1
3	3	2023-09-18 10:00:00	Entrada	2	2
4	4	2023-09-18 10:00:00	Salida	2	2
5	5	2023-09-18 10:00:00	Entrada	3	3
6	6	2023-09-18 10:00:00	Salida	3	3
7	7	2023-09-18 10:00:00	Entrada	4	4
8	8	2023-09-18 10:00:00	Salida	4	4
9	9	2023-09-18 10:00:00	Entrada	5	5
10	10	2023-09-18 10:00:00	Salida	5	5
11	11	2023-09-18 10:00:00	Entrada	6	6
12	12	2023-09-18 10:00:00	Salida	6	6
13	13	2023-09-18 10:00:00	Entrada	7	7
14	14	2023-09-18 10:00:00	Salida	7	7
15	15	2023-09-18 10:00:00	Entrada	8	8
16	16	2023-09-18 10:00:00	Salida	8	8
17	17	2023-09-18 10:00:00	Entrada	9	9
18	18	2023-09-18 10:00:00	Salida	9	9
19	19	2023-09-18 10:00:00	Entrada	10	10
20	20	2023-09-18 10:00:00	Salida	10	10

Select from paciente

```

Query 1
1   SELECT * FROM bd2_proyecto1.habitacion LIMIT 20;
2 •  SELECT * FROM bd2_proyecto1.log_actividad LIMIT 20;
3 •  SELECT * FROM bd2_proyecto1.log_habitacion LIMIT 20;
4 •  SELECT * FROM bd2_proyecto1.paciente LIMIT 20;
5

```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | timestampx | statusx | idHabitacion

Order by habitación incremental 2

```

5
6 •  SELECT * FROM bd2_proyecto1.habitacion ORDER BY idHabitacion DESC LIMIT 20;
7 •  SELECT * FROM bd2_proyecto1.log_actividad ORDER BY id_log_actividad DESC LIMIT 20;
8 •  SELECT * FROM bd2_proyecto1.log_habitacion ORDER BY idHabitacion, timestampx, statusx DESC LIMIT 20;
9 •  SELECT * FROM bd2_proyecto1.paciente ORDER BY idPaciente DESC LIMIT 20;

```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: | idHabitacion | habitacion

15	EstadoÃ³n de revisÃ³n 4
14	EstadoÃ³n de revisÃ³n 3
13	EstadoÃ³n de revisÃ³n 2
12	EstadoÃ³n de revisÃ³n 1
11	Laboratorio
10	Recepcion
9	Sala de procedimientos 4
8	Sala de procedimientos 3
7	Sala de procedimientos 2
6	Sala de procedimientos 1
5	Sala de imagenes 1
4	Sala de examenes 4
3	Sala de examenes 3
2	Sala de examenes 2
1	Sala de examenes 1
*	NULL

Order by log actividad incremental 2

The screenshot shows the MySQL Workbench interface with a query editor titled "Query 1". The query is as follows:

```

6 • SELECT * FROM bd2_proyecto1.habitacion ORDER BY idHabitacion DESC LIMIT 20;
7 • SELECT * FROM bd2_proyecto1.log_actividad ORDER BY id_log_actividad DESC LIMIT 20;
8 • SELECT * FROM bd2_proyecto1.log_habitacion ORDER BY idHabitacion, timestampx, statusx DESC LIMIT 20;
9 • SELECT * FROM bd2_proyecto1.paciente ORDER BY idPaciente DESC LIMIT 20;
10

```

The results grid displays columns: id_log_actived, timestampx, actividad, HABITACION_idHabitacion, and PACIENTE_idPadente. The results are currently empty.

Order by log habitación incremental 2

The screenshot shows the MySQL Workbench interface with a query editor titled "Query 1". The query is as follows:

```

6 • SELECT * FROM bd2_proyecto1.habitacion ORDER BY idHabitacion DESC LIMIT 20;
7 • SELECT * FROM bd2_proyecto1.log_actividad ORDER BY id_log_actividad DESC LIMIT 20;
8 • SELECT * FROM bd2_proyecto1.log_habitacion ORDER BY idHabitacion, timestampx, statusx DESC LIMIT 20;
9 • SELECT * FROM bd2_proyecto1.paciente ORDER BY idPaciente DESC LIMIT 20;
10

```

The results grid displays columns: timestampx, statusx, and idHabitacion. The results are currently empty.

Order by habitación incremental 2

```

7 • SELECT * FROM bd2_proyecto1.log_actividad ORDER BY id_log_actividad DESC LIMIT 20;
8 • SELECT * FROM bd2_proyecto1.log_habitacion ORDER BY idhabitacion, timestampx, statusx DESC LIMIT 20;
9 • SELECT * FROM bd2_proyecto1.paciente ORDER BY idPaciente DESC LIMIT 20;
10

```

Result Grid | Filter Rows: [] | Edit: [] | Export/Import: [] | Wrap Cell Contents: [] | Fetch rows: []

	idPaciente	edad	genero
▶	254183	13	Masculino
	254182	38	Femenino
	254181	14	Masculino
	254180	43	Femenino
	254179	56	Masculino
	254178	5	Masculino
	254177	41	Femenino
	254176	80	Femenino
	254175	13	Masculino
	254174	10	Masculino
	254173	37	Masculino
	254172	51	Femenino
	254171	18	Masculino
	254170	7	Masculino
	254169	57	Femenino
	254168	61	Masculino
	254167	28	Femenino
	254166	56	Femenino
	254165	83	Masculino
	254164	59	Masculino
▶	254163	22	Masculino
▶	254162	22	Masculino

Count general incremental 2

```

11 • SELECT * FROM
12     (SELECT COUNT(*) conteoHabitaciones FROM bd2_proyecto1.habitacion) AS conteoHabitaciones,
13     (SELECT COUNT(*) conteoLogActividad FROM bd2_proyecto1.log_actividad) AS conteoLogActividad,
14     (SELECT COUNT(*) conteoLogHabitacion FROM bd2_proyecto1.log_habitacion) AS conteoLogHabitacion,
15     (SELECT COUNT(*) conteoPaciente FROM bd2_proyecto1.paciente) AS conteoPaciente;
16

```

Result Grid | Filter Rows: [] | Export: [] | Wrap Cell Contents: []

	conteoHabitaciones	conteoLogActividad	conteoLogHabitacion	conteoPaciente
▶	15	0	0	154184

Select from habitación incremental 3

```

Query 1
1 SELECT * FROM bd2_proyecto1.habitacion LIMIT 20;
2 • SELECT * FROM bd2_proyecto1.log_actividad LIMIT 20;
3 • SELECT * FROM bd2_proyecto1.log_habitacion LIMIT 20;
4 • SELECT * FROM bd2_proyecto1.paciente LIMIT 20;
5

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            | Recepcción |
| 11            | Laboratorio |
| 12            | Estación de revisión 1 |
| 13            | Estación de revisión 2 |
| 14            | Estación de revisión 3 |
| 15            | Estación de revisión 4 |
| null          | null           |
+-----+-----+

```

Select from log actividad incremental 3

```

Query 1
1 SELECT * FROM bd2_proyecto1.habitacion LIMIT 20;
2 • SELECT * FROM bd2_proyecto1.log_actividad LIMIT 20;
3 • SELECT * FROM bd2_proyecto1.log_habitacion LIMIT 20;
4 • SELECT * FROM bd2_proyecto1.paciente LIMIT 20;
5

log_actividad
+-----+-----+-----+-----+-----+-----+
| id_log_actividad | timestamp | actividad | HABITACION_idhabitacion | PACIENTE_idpaciente |
+-----+-----+-----+-----+-----+-----+
| 2               | 5/31/2022 | Paciente hace el registro. | 10                | 134247              |
| 3               | 5/31/2022 | Paciente recibe papelera en recepción. | 10                | 134247              |
| 4               | 5/31/2022 | Paciente entrega papelera. | 10                | 134247              |
| 5               | 5/31/2022 | Recepcionista establece la condición del paciente. | 10                | 134247              |
| 6               | 5/31/2022 | Enfermera comienza la revisión del paciente. | 10                | 134247              |
| 7               | 5/31/2022 | Revision determina que el paciente es tipo 2 y s... | 10                | 134247              |
| 8               | 5/31/2022 | Médico inicia con el tratamiento del paciente. | 10                | 134247              |
| 9               | 5/31/2022 | Paciente hace el registro. | 10                | 135641              |
| 10              | 5/31/2022 | Paciente recibe papelera en recepción. | 10                | 135641              |
| 11              | 5/31/2022 | Paciente hace el registro. | 10                | 180487              |
| 12              | 5/31/2022 | Paciente entrega papelera. | 12                | 135641              |
| 13              | 5/31/2022 | Recepcionista establece la condición del paciente. | 12                | 135641              |
| 14              | 5/31/2022 | Paciente recibe papelera en recepción. | 5                 | 180487              |
| 15              | 5/31/2022 | Enfermera comienza la revisión del paciente. | 5                 | 135641              |
| 16              | 5/31/2022 | Paciente entrega papelera. | 7                 | 180487              |
| 17              | 5/31/2022 | Revision determina que el paciente es tipo 2 y s... | 7                 | 135641              |
| 18              | 5/31/2022 | Recepcionista establece la condición del paciente. | 7                 | 135641              |
| 19              | 5/31/2022 | Médico inicia con el tratamiento del paciente. | 7                 | 135641              |
| 20              | 5/31/2022 | Enfermera comienza la revisión del paciente. | 10                | 180487              |
| 21              | 5/31/2022 | Revision determina que el paciente es tipo 5 y s... | 10                | 180487              |
| null            | null        | null        | null              | null              |
+-----+-----+-----+-----+-----+-----+

```

Select from log habitación incremental 3

```

Query 1
1 • SELECT * FROM bd2_proyecto1.habitacion LIMIT 20;
2 • SELECT * FROM bd2_proyecto1.log_actividad LIMIT 20;
3 • SELECT * FROM bd2_proyecto1.log_habitacion LIMIT 20;
4 • SELECT * FROM bd2_proyecto1.paciente LIMIT 20;
5

Result Grid | Filter Rows | Report | Wrap Cell Content | 
timestampx | statusx | idhabitacion |

```

select from paciente incremental 3

	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

Order by habitación incremental 3

```

5
6 •   SELECT * FROM bd2_proyecto1.habitacion ORDER BY idHabitacion DESC LIMIT 20;
7 •   SELECT * FROM bd2_proyecto1.log_actividad ORDER BY id_log_actividad DESC LIMIT 20;
8 •   SELECT * FROM bd2_proyecto1.log_habitacion ORDER BY idHabitacion, timestampx, statusx DESC LIMIT 20;
9 •   SELECT * FROM bd2_proyecto1.paciente ORDER BY idPaciente DESC LIMIT 20;

```

Result Grid | Filter Rows: [] | Edit: [] | Export/Import: [] | Wrap Cell Contents: [] | Fetch Rows: []

	idHabitacion	habitacion
▶	15	Estaci�n de revisi�n 4
14		Estaci�n de revisi�n 3
13		Estaci�n de revisi�n 2
12		Estaci�n de revisi�n 1
11		Laboratorio
10		Recepcion
9		Sala de procedimientos 4
8		Sala de procedimientos 3
7		Sala de procedimientos 2
6		Sala de procedimientos 1
5		Sala de imagenes 1
4		Sala de exámenes 4
3		Sala de exámenes 3
2		Sala de exámenes 2
1		Sala de exámenes 1
...		...

Order by log actividad incremental 3

```

5
6 •   SELECT * FROM bd2_proyecto1.habitacion ORDER BY idHabitacion DESC LIMIT 20;
7 •   SELECT * FROM bd2_proyecto1.log_actividad ORDER BY id_log_actividad DESC LIMIT 20;
8 •   SELECT * FROM bd2_proyecto1.log_habitacion ORDER BY idHabitacion, timestampx, statusx DESC LIMIT 20;
9 •   SELECT * FROM bd2_proyecto1.paciente ORDER BY idPaciente DESC LIMIT 20;

```

Result Grid | Filter Rows: [] | Edit: [] | Export/Import: [] | Wrap Cell Contents: [] | Fetch Rows: []

	id_log_actividad	timestampx	actividad	HABITACION_idHabitacion	PACIENTE_idPaciente
▶	67685	7/25/2022	Paciente completa su registro.	10	160508
67684	7/25/2022	Paciente inicia su egreso.	10	160508	
67683	7/25/2022	Muestra entregada al laboratorio.	7	160508	
67682	7/25/2022	Tecnico de laboratorio comienza la toma de mue... 7	7	160508	
67681	7/25/2022	Paciente completa su tratamiento. Se solicita to... 13	13	160508	
67680	7/25/2022	Paciente completa su registro.	13	156310	
67679	7/25/2022	Paciente inicia su egreso.	10	156310	
67678	7/25/2022	Paciente completa su tratamiento.	10	156310	
67677	7/25/2022	Medico inicia con el tratamiento del paciente.	10	160508	
67676	7/25/2022	Revision determino que el paciente es tipo 2 y s... 10	10	160508	
67675	7/25/2022	Enfermera comienza la revision del paciente.	13	160508	
67674	7/25/2022	Repcionista establece la condicion del pacient... 13	160508		
67673	7/25/2022	Paciente entrega papelera.	10	160508	
67672	7/25/2022	Paciente recibe papelera en recepcion.	10	160508	
67671	7/25/2022	Paciente inicia el registro.	10	160508	
67670	7/25/2022	Medico inicia con el tratamiento del paciente.	10	156310	
67669	7/25/2022	Paciente completa su registro.	10	161628	
67668	7/25/2022	Revision determino que el paciente es tipo 1 y s... 10	10	156310	
67667	7/25/2022	Enfermera comienza la revision del paciente.	6	156310	
67666	7/25/2022	Revision determino que el paciente es tipo 0 y ... 6	6	148028	
...

Order by log habitación incremental 3

```

1   SELECT * FROM bd2_proyecto1.habitacion LIMIT 20;
2 •  SELECT * FROM bd2_proyecto1.log_actividad LIMIT 20;
3 •  SELECT * FROM bd2_proyecto1.log_habitacion LIMIT 20;
4 •  SELECT * FROM bd2_proyecto1.paciente LIMIT 20;
5
6 •  SELECT * FROM bd2_proyecto1.habitacion ORDER BY idHabitacion DESC LIMIT 20;
7 •  SELECT * FROM bd2_proyecto1.log_actividad ORDER BY id_log_actividad DESC LIMIT 20;
8 •  SELECT * FROM bd2_proyecto1.log_habitacion ORDER BY idHabitacion, timestampx, statusx DESC LIMIT 20;
9 •  SELECT * FROM bd2_proyecto1.paciente ORDER BY idPaciente DESC LIMIT 20;

```

The screenshot shows the MySQL Workbench interface with a query editor window titled "Query 1". The query is a multi-table incremental select statement. Below the query, there is a "Result Grid" table with columns: timestampx, statusx, and idHabitacion. The table contains 20 rows of data from the log_habitacion table.

Order by paciente incremental 3

```

1   SELECT * FROM bd2_proyecto1.habitacion LIMIT 20;
2 •  SELECT * FROM bd2_proyecto1.log_actividad LIMIT 20;
3 •  SELECT * FROM bd2_proyecto1.log_habitacion LIMIT 20;
4 •  SELECT * FROM bd2_proyecto1.paciente LIMIT 20;
5
6 •  SELECT * FROM bd2_proyecto1.habitacion ORDER BY idHabitacion DESC LIMIT 20;
7 •  SELECT * FROM bd2_proyecto1.log_actividad ORDER BY id_log_actividad DESC LIMIT 20;
8 •  SELECT * FROM bd2_proyecto1.log_habitacion ORDER BY idHabitacion, timestampx, statusx DESC LIMIT 20;
9 •  SELECT * FROM bd2_proyecto1.paciente ORDER BY idPaciente DESC LIMIT 20;

```

The screenshot shows the MySQL Workbench interface with a query editor window titled "Query 1". The query is a multi-table incremental select statement. Below the query, there is a "Result Grid" table with columns: idPaciente, edad, and genero. The table contains 20 rows of data from the paciente table, ordered by idPaciente in descending order.

Count incremental 3

```

9 •  SELECT * FROM bd2_proyecto1.paciente ORDER BY idPaciente DESC LIMIT 20;
10
11 •  SELECT * FROM
12      (SELECT COUNT(*) conteoHabitaciones FROM bd2_proyecto1.habitacion) AS conteoHabitaciones,
13      (SELECT COUNT(*) conteoLogActividad FROM bd2_proyecto1.log_actividad) AS conteoLogActividad,
14      (SELECT COUNT(*) conteoLogHabitacion FROM bd2_proyecto1.log_habitacion) AS conteoLogHabitacion,
15      (SELECT COUNT(*) conteoPaciente FROM bd2_proyecto1.paciente) AS conteoPaciente
16

```

	conteoHabitaciones	conteoLogActividad	conteoLogHabitacion	conteoPaciente
▶	15	67684	0	154384

Select from habitaciones incremental 4

Query 1

```

1  SELECT * FROM bd2_proyecto1.habitacion LIMIT 20;
2 •  SELECT * FROM bd2_proyecto1.log_actividad LIMIT 20;
3 •  SELECT * FROM bd2_proyecto1.log_habitacion LIMIT 20;
4 •  SELECT * FROM bd2_proyecto1.paciente LIMIT 20;
5

```

Result Grid

	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	Estandar de revisión 1
	13	Estandar de revisión 2
	14	Estandar de revisión 3
	15	Estandar de revisión 4
*	16	

Select from log actividad incremental 4

Query 1

```

1 SELECT * FROM bd2_proyecto1.habitacion LIMIT 20;
2 • SELECT * FROM bd2_proyecto1.log_actividad LIMIT 20;
3 • SELECT * FROM bd2_proyecto1.log_habitacion LIMIT 20;
4 • SELECT * FROM bd2_proyecto1.paciente LIMIT 20;
5

```

Result Grid | Filter Rows: [] Edit | Export/Import | Wrap Cell Content | Batch rows: []

id_log_actividad	timestamp	actividad	HABITACION_idHabitacion	PACIENTE_idPaciente
2	5/31/2022	Paciente inicia el registro.	10	134247
3	5/31/2022	Paciente recibe papelera en recepcion.	10	134247
4	5/31/2022	Paciente entrega papelera.	10	134247
5	5/31/2022	Recojonista establece la condicion del paciente.	10	134247
6	5/31/2022	Enfermera comienza la revision del paciente.	10	134247
7	5/31/2022	Revision determino que el paciente es tipo 2 y s...	10	134247
8	5/31/2022	Medico inicia con el tratamiento del paciente.	10	134247
9	5/31/2022	Paciente inicia el registro.	10	135641
10	5/31/2022	Paciente recibe papelera en recepcion.	10	135641
11	5/31/2022	Paciente entrega papelera.	10	180487
12	5/31/2022	Paciente entrega papelera.	12	135641
13	5/31/2022	Recojonista establece la condicion del paciente.	12	135641
14	5/31/2022	Paciente recibe papelera en recepcion.	5	180487
15	5/31/2022	Enfermera comienza la revision del paciente.	5	135641
16	5/31/2022	Paciente entrega papelera.	7	180487
17	5/31/2022	Revision determino que el paciente es tipo 2 y s...	7	135641
18	5/31/2022	Recojonista establece la condicion del paciente.	7	180487
19	5/31/2022	Medico inicia con el tratamiento del paciente.	7	135641
20	5/31/2022	Enfermera comienza la revision del paciente.	10	180487
21	5/31/2022	Revision determino que el paciente es tipo 5 y s...	10	180487

Select from paciente incremental 4

Query 1

```

1 SELECT * FROM bd2_proyecto1.habitacion LIMIT 20;
2 • SELECT * FROM bd2_proyecto1.log_actividad LIMIT 20;
3 • SELECT * FROM bd2_proyecto1.log_habitacion LIMIT 20;
4 • SELECT * FROM bd2_proyecto1.paciente LIMIT 20;
5

```

Result Grid | Filter Rows: [] Edit | Export/Import | Wrap Cell Content | Batch rows: []

idPaciente	edad	genero
100000	95	Otro
100001	40	Femenino
100002	42	Masculino
100003	8	Femenino
100004	88	Masculino
100005	23	Masculino
100006	60	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
100015	50	Femenino
100016	39	Masculino
100017	55	Femenino
100018	61	Masculino
100019	58	Masculino

Order by incremental 4 habitacion

```

4 •   SELECT * FROM bd2_proyecto1.paciente LIMIT 20;
5
6 •   SELECT * FROM bd2_proyecto1.habitacion ORDER BY idHabitacion DESC LIMIT 20;
7 •   SELECT * FROM bd2_proyecto1.log_actividad ORDER BY id_log_actividad DESC LIMIT 20;
8 •   SELECT * FROM bd2_proyecto1.log_habitacion ORDER BY idHabitacion, timestampx, statusx DESC LIMIT 20;
9 •   SELECT * FROM bd2_proyecto1.paciente ORDER BY idPaciente DESC LIMIT 20;

```

Result Grid | Filter Rows | Edit | Export/Import | Wrap Cell Contents | Fetch rows:

idHabitacion	habitacion
15	Estacionamiento 4
14	Estacionamiento 3
13	Estacionamiento 2
12	Estacionamiento 1
11	Laboratorio
10	Repcion
9	Sala de procedimientos 4
8	Sala de procedimientos 3
7	Sala de procedimientos 2
6	Sala de procedimientos 1
5	Sala de imagenes 1
4	Sala de examenes 4
3	Sala de examenes 3
2	Sala de examenes 2
1	Sala de examenes 1

Order by log actividad incremental 4

```

5
6 •   SELECT * FROM bd2_proyecto1.habitacion ORDER BY idHabitacion DESC LIMIT 20;
7 •   SELECT * FROM bd2_proyecto1.log_actividad ORDER BY id_log_actividad DESC LIMIT 20;
8 •   SELECT * FROM bd2_proyecto1.log_habitacion ORDER BY idHabitacion, timestampx, statusx DESC LIMIT 20;
9 •   SELECT * FROM bd2_proyecto1.paciente ORDER BY idPaciente DESC LIMIT 20;
10

```

Result Grid | Filter Rows | Edit | Export/Import | Wrap Cell Contents | Wrap Cell Contents | Fetch rows:

id_log_actividad	timestampx	actividad	HABITACION_idHabitacion	PACIENTE_idPaciente
164915	7/25/2021 9:24:03 PM	Paciente completa su registro.	10	160508
164914	7/25/2021 9:20:13 PM	Paciente inicia su egreso.	10	160508
164913	7/25/2021 9:19:19 PM	Muestra entregada al laboratorio.	7	160508
164912	7/25/2021 9:16:11 PM	Tecnico de laboratorio comienza la toma de mue... s.	7	160508
164911	7/25/2021 9:15:54 PM	Paciente completa su tratamiento. Se solicita to... mografía.	13	160508
164910	7/25/2021 9:02:55 PM	Paciente completa su registro.	13	156310
164909	7/25/2021 8:58:32 PM	Paciente inicia su egreso.	10	156310
164908	7/25/2021 8:57:06 PM	Paciente completa su tratamiento.	10	156310
164907	7/25/2021 8:55:50 PM	Medico inicia con el tratamiento del paciente.	10	160508
164906	7/25/2021 8:55:01 PM	Revision determino que el paciente es tipo 2 y s... e.	10	160508
164905	7/25/2021 8:53:10 PM	Enfermera comienza la revision del paciente.	13	160508
164904	7/25/2021 8:52:20 PM	Recepcionista establece la condicion del patient... e.	13	160508
164903	7/25/2021 8:51:51 PM	Paciente entrega papelera.	10	160508
164902	7/25/2021 8:49:56 PM	Paciente recibe papelera en recepcion.	10	160508
164901	7/25/2021 8:48:56 PM	Paciente inicia el registro.	10	160508
164900	7/25/2021 8:43:20 PM	Medico inicia con el tratamiento del paciente.	10	156310
164899	7/25/2021 8:42:35 PM	Paciente completa su registro.	10	161628
164898	7/25/2021 8:41:49 PM	Revision determino que el paciente es tipo 1 y s... e.	10	156310
164897	7/25/2021 8:40:08 PM	Enfermera comienza la revision del paciente.	6	156310
164896	7/25/2021 8:39:47 PM	Revision determino que el paciente es tipo 3 y ... e.	6	148028

Order by log habitación incremental 4

```

2 •  SELECT * FROM bd2_proyecto1.log_actividad LIMIT 20;
3 •  SELECT * FROM bd2_proyecto1.log_habitacion ORDER BY idHabitacion DESC LIMIT 20;
4 •  SELECT * FROM bd2_proyecto1.paciente LIMIT 20;
5
6 •  SELECT * FROM bd2_proyecto1.habitacion ORDER BY idHabitacion DESC LIMIT 20;
7 •  SELECT * FROM bd2_proyecto1.log_actividad ORDER BY id_log_actividad DESC LIMIT 20;
8 •  SELECT * FROM bd2_proyecto1.log_habitacion ORDER BY idHabitacion, timestampx, statusx DESC LIMIT 20;
9 •  SELECT * FROM bd2_proyecto1.paciente ORDER BY idPaciente DESC LIMIT 20;
10

```

The screenshot shows the MySQL Workbench interface with a query editor window. The query is displayed in the editor, and the results are shown in a grid below. The grid has columns labeled 'timestampx', 'statusx', and 'idHabitacion'. The results are ordered by 'idHabitacion' in descending order.

Order by paciente incremental 4

```

2 •  SELECT * FROM bd2_proyecto1.log_actividad LIMIT 20;
3 •  SELECT * FROM bd2_proyecto1.log_habitacion LIMIT 20;
4 •  SELECT * FROM bd2_proyecto1.paciente LIMIT 20;
5
6 •  SELECT * FROM bd2_proyecto1.habitacion ORDER BY idHabitacion DESC LIMIT 20;
7 •  SELECT * FROM bd2_proyecto1.log_actividad ORDER BY id_log_actividad DESC LIMIT 20;
8 •  SELECT * FROM bd2_proyecto1.log_habitacion ORDER BY idHabitacion, timestampx, statusx DESC LIMIT 20;
9 •  SELECT * FROM bd2_proyecto1.paciente ORDER BY idPaciente DESC LIMIT 20;
10

```

The screenshot shows the MySQL Workbench interface with a query editor window. The query is displayed in the editor, and the results are shown in a grid below. The grid has columns labeled 'idPaciente', 'edad', and 'genero'. The results are ordered by 'idPaciente' in descending order.

idPaciente	edad	genero
254383	13	Masculino
254382	38	Femenino
254381	14	Masculino
254380	43	Femenino
254179	56	Masculino
254178	5	Masculino
254177	41	Femenino
254176	80	Femenino
254175	13	Masculino
254174	10	Masculino
254173	37	Masculino
254172	51	Femenino
254171	18	Masculino
254170	7	Masculino
254169	57	Femenino
254168	61	Masculino
254167	28	Femenino
254166	56	Femenino
254165	83	Masculino
254164	59	Masculino

Count incremental 4

```
8 • SELECT * FROM bd2_proyecto1.log_habitacion ORDER BY idHabitacion, timestampx, statusx DESC LIMIT 20;
9 • SELECT * FROM bd2_proyecto1.paciente ORDER BY idPaciente DESC LIMIT 20;
10
11 • SELECT * FROM
12     (SELECT COUNT(*) conteoHabitaciones FROM bd2_proyecto1.habitacion) AS conteoHabitaciones,
13     (SELECT COUNT(*) conteoLogActividad FROM bd2_proyecto1.log_actividad) AS conteoLogActividad,
14     (SELECT COUNT(*) conteoLogHabitacion FROM bd2_proyecto1.log_habitacion) AS conteoLogHabitacion,
15     (SELECT COUNT(*) conteoPaciente FROM bd2_proyecto1.paciente) AS conteoPaciente;
16
```

Select from habitación incremental 5

Query 1

```
1 SELECT * FROM bd2_proyecto1.habitacion LIMIT 20;
2 • SELECT * FROM bd2_proyecto1.log_actividad LIMIT 20;
3 • SELECT * FROM bd2_proyecto1.log_habitacion LIMIT 20;
4 • SELECT * FROM bd2_proyecto1.paciente LIMIT 20;
5
6 • SELECT * FROM bd2_proyecto1.habitacion ORDER BY idhabitacion DESC LIMIT 20;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

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	Estado de revisión 1
13	Estado de revisión 2
14	Estado de revisión 3
15	Estado de revisión 4
• 16	• 16

Select from log actividad incremental 5

The screenshot shows the MySQL Workbench interface with a query editor window titled "Query 1". The SQL code is as follows:

```

1  SELECT * FROM bd2_proyecto1.habitacion LIMIT 20;
2  • SELECT * FROM bd2_proyecto1.log_actividad LIMIT 20;
3  • SELECT * FROM bd2_proyecto1.log_habitacion LIMIT 20;
4  • SELECT * FROM bd2_proyecto1.paciente LIMIT 20;
5
6  • SELECT * FROM bd2_proyecto1.habitacion ORDER BY idhabitacion DESC LIMIT 20;

```

The results grid displays 21 rows of data from the log_actividad table, showing timestamp, actividad, HABITACION_idHabitacion, and PACIENTE_idPaciente. The data includes various patient interactions like check-ins and document delivery.

id_log_actividad	timestampx	actividad	HABITACION_idHabitacion	PACIENTE_idPaciente
2	5/31/2022	Paciente inicia el registro.	10	134247
3	5/31/2022	Paciente recibe papelera en recepción.	10	134247
4	5/31/2022	Recepcionista entrega papelera.	10	134247
5	5/31/2022	Recepcionista establece la condición del paciente.	10	134247
6	5/31/2022	Enfermera comienza la revisión del paciente.	10	134247
7	5/31/2022	Revisión determinó que el paciente es tipo 2 y se...	10	134247
8	5/31/2022	Médico inicia con el tratamiento del paciente.	10	134247
9	5/31/2022	Paciente inicia el registro.	10	135641
10	5/31/2022	Paciente recibe papelera en recepción.	10	135641
11	5/31/2022	Paciente inicia el registro.	10	180487
12	5/31/2022	Paciente entrega papelera.	12	135641
13	5/31/2022	Recepcionista establece la condición del paciente.	12	135641
14	5/31/2022	Paciente recibe papelera en recepción.	5	180487
15	5/31/2022	Enfermera comienza la revisión del paciente.	5	135641
16	5/31/2022	Paciente entrega papelera.	7	180487
17	5/31/2022	Revisión determinó que el paciente es tipo 2 y se...	7	135641
18	5/31/2022	Recepcionista establece la condición del paciente.	7	180487
19	5/31/2022	Médico inicia con el tratamiento del paciente.	7	135641
20	5/31/2022	Enfermera comienza la revisión del paciente.	10	180487
21	5/31/2022	Revisión determinó que el paciente es tipo 5 y se...	10	180487

Select from log habitación incremental 5

The screenshot shows the MySQL Workbench interface with a query editor window titled "Query 1". The SQL code is as follows:

```

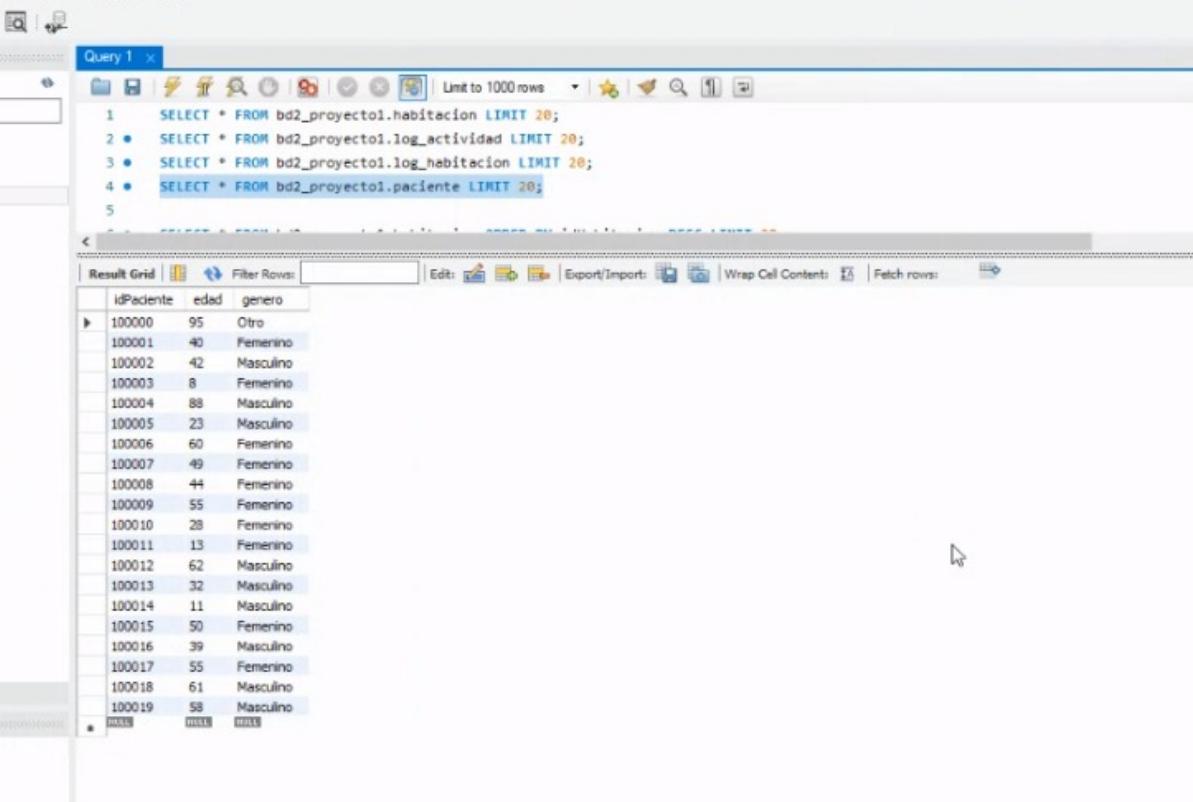
1  SELECT * FROM bd2_proyecto1.habitacion LIMIT 20;
2  • SELECT * FROM bd2_proyecto1.log_actividad LIMIT 20;
3  • SELECT * FROM bd2_proyecto1.log_habitacion LIMIT 20;
4  • SELECT * FROM bd2_proyecto1.paciente LIMIT 20;
5
6  • SELECT * FROM bd2_proyecto1.habitacion ORDER BY idhabitacion DESC LIMIT 20;

```

The results grid displays 20 rows of data from the log_habitacion table, showing timestamp, statusx, and idHabitacion. The data tracks room availability and cleaning status.

timestampx	statusx	idHabitacion
5/31/2022	Sala no disponible.	12
5/31/2022	Inicia limpieza.	12
5/31/2022	Sala disponible.	12
5/31/2022	Sala no disponible.	12
5/31/2022	Inicia limpieza.	12
5/31/2022	Sala disponible.	12
5/31/2022	Sala no disponible.	13
5/31/2022	Inicia limpieza.	13
5/31/2022	Sala disponible.	13
5/31/2022	Sala no disponible.	1
5/31/2022	Inicia limpieza.	1
5/31/2022	Sala disponible.	1
5/31/2022	Sala no disponible.	2
5/31/2022	Inicia limpieza.	2
5/31/2022	Sala no disponible.	12
5/31/2022	Inicia limpieza.	12
5/31/2022	Sala disponible.	12
5/31/2022	Sala disponible.	2
5/31/2022	Sala no disponible.	6
5/31/2022	Inicia limpieza.	6

Select from paciente incremental 5



The screenshot shows a MySQL Workbench interface with a query editor titled "Query 1" containing the following SQL code:

```

1   SELECT * FROM bd2_proyecto1.habitacion LIMIT 20;
2 •  SELECT * FROM bd2_proyecto1.log_actividad LIMIT 20;
3 •  SELECT * FROM bd2_proyecto1.log_habitacion LIMIT 20;
4 •  SELECT * FROM bd2_proyecto1.paciente LIMIT 20;
5

```

The result grid displays 20 rows of patient data:

	idPaciente	edad	genero
▶	100000	95	Otro
▶	100001	40	Femenino
▶	100002	42	Masculino
▶	100003	8	Femenino
▶	100004	88	Masculino
▶	100005	23	Masculino
▶	100006	60	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
▶	100015	50	Femenino
▶	100016	39	Masculino
▶	100017	55	Femenino
▶	100018	61	Masculino
▶	100019	58	Masculino

Order by habitación incremental 5



The screenshot shows a MySQL Workbench interface with a query editor titled "Query 1" containing the following SQL code:

```

5
6 •  SELECT * FROM bd2_proyecto1.habitacion ORDER BY idHabitacion DESC LIMIT 20;
7 •  SELECT * FROM bd2_proyecto1.log_actividad ORDER BY id_log_actividad DESC LIMIT 20;
8 •  SELECT * FROM bd2_proyecto1.log_habitacion ORDER BY idHabitacion, timestamppx, statusx DESC LIMIT 20;
9 •  SELECT * FROM bd2_proyecto1.paciente ORDER BY idPaciente DESC LIMIT 20;

```

The result grid displays 20 rows of room data:

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

Order by log actividad incremental 5

Query 1

```
5
6 • SELECT * FROM bd2_proyecto1.habitacion ORDER BY idHabitacion DESC LIMIT 20;
7 • SELECT * FROM bd2_proyecto1.log_actividad ORDER BY id_log_actividad DESC LIMIT 20;
8 • SELECT * FROM bd2_proyecto1.log_habitacion ORDER BY idHabitacion, timestampx, statusx DESC LIMIT 20;
9 • SELECT * FROM bd2_proyecto1.paciente ORDER BY idPaciente DESC LIMIT 20;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Contents: | Fetch rows:

	id_log_actividad	timestampx	actividad	HABITACION_idhabitacion	PACIENTE_idPaciente
▶	164915	7/25/2021 9:24:03 PM	Paciente completa su registro.	10	160508
	164914	7/25/2021 9:20:13 PM	Paciente inicia su egreso.	10	160508
	164913	7/25/2021 9:19:19 PM	Muestra entregada al laboratorio.	7	160508
	164912	7/25/2021 9:16:11 PM	Tecnico de laboratorio comienza la toma de mue...	7	160508
	164911	7/25/2021 9:15:54 PM	Paciente completa su tratamiento. Se solicita to...	13	160508
	164910	7/25/2021 9:02:55 PM	Paciente completa su registro.	13	156310
	164909	7/25/2021 8:58:32 PM	Paciente inicia su egreso.	10	156310
	164908	7/25/2021 8:57:06 PM	Paciente completa su tratamiento.	10	156310
	164907	7/25/2021 8:55:50 PM	Medico inicia con el tratamiento del paciente.	10	160508
	164906	7/25/2021 8:55:01 PM	Revision determino que el paciente es tipo 2 y s...	10	160508
	164905	7/25/2021 8:53:10 PM	Enfermera comienza la revision del paciente.	13	160508
	164904	7/25/2021 8:52:20 PM	Receptor/operador establece la condicion del pacient...	13	160508
	164903	7/25/2021 8:51:51 PM	Paciente entrega papelera.	10	160508
	164902	7/25/2021 8:49:56 PM	Paciente recibe papelera en recepcion.	10	160508
	164901	7/25/2021 8:48:56 PM	Paciente inicia el registro.	10	160508
	164900	7/25/2021 8:43:10 PM	Medico inicia con el tratamiento del paciente.	10	156310
	164899	7/25/2021 8:42:35 PM	Paciente completa su registro.	10	161628
	164898	7/25/2021 8:41:49 PM	Revision determino que el paciente es tipo 1 y s...	10	156310
	164897	7/25/2021 8:40:08 PM	Enfermera comienza la revision del paciente.	6	156310
	164896	7/25/2021 8:39:47 PM	Revision determino que el paciente es tipo B y ...	6	148028

Order by log habitación incremental 5

Order by paciente incremental 5

```
Query 1 <| SELECT * FROM bd2_proyecto1.habitacion ORDER BY idHabitacion DESC LIMIT 20;
SELECT * FROM bd2_proyecto1.log_actividad ORDER BY id_log_actividad DESC LIMIT 20;
SELECT * FROM bd2_proyecto1.log_habitacion ORDER BY idHabitacion, timestamppx, statusx DESC LIMIT 20;
SELECT * FROM bd2_proyecto1.paciente ORDER BY idPaciente DESC LIMIT 20;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Contents: | Fetch rows: |

ID Paciente	Edad	Género
254183	13	Masculino
254182	38	Femenino
254181	14	Masculino
254180	43	Femenino
254179	56	Masculino
254178	5	Masculino
254177	41	Femenino
254176	80	Femenino
254175	13	Masculino
254174	10	Masculino
254173	37	Masculino
254172	51	Femenino
254171	18	Masculino
254170	7	Masculino
254169	57	Femenino
254168	61	Masculino
254167	28	Femenino
254166	56	Femenino
254165	83	Masculino
254164	59	Masculino
NULL	NULL	NULL

```
9 •     SELECT * FROM bd2_proyecto1.paciente ORDER BY idPaciente DESC LIMIT 20;
10
11 •    SELECT * FROM
12      (SELECT COUNT(*) conteoHabitaciones FROM bd2_proyecto1.habitacion) AS conteoHabitaciones,
13      (SELECT COUNT(*) conteoLogActividad FROM bd2_proyecto1.log_actividad) AS conteoLogActividad,
14      (SELECT COUNT(*) conteoLogHabitacion FROM bd2_proyecto1.log_habitacion) AS conteoLogHabitacion,
15      (SELECT COUNT(*) conteoPaciente FROM bd2_proyecto1.paciente) AS conteoPaciente;
```

Result Grid			
Filter Rows:		Exports:	Wrap Cell Contents:
conteoHabitaciones	conteoLogActividad	conteoLogHabitacion	conteoPaciente
15	101527	34617	154184

Análisis de los resultados obtenidos

Como se observa en las capturas de pantalla se puede observar el desempeño de cada uno de los tipo de backups a lo largo del tiempo

¿Qué tipo de backup se recomienda para el volumen de datos proporcionado?

Como administrador de sistemas de bases de datos, es realmente importante velar por la seguridad e integridad de la base de datos de la mayor cantidad de maneras posible.

El tipo de backup que es recomendado tener siempre en caso de que en el servidor de la base de datos exista algún percance es **el backup completo** ya que como daños físicos al servidor así como posibles ataques por parte de hackers. Es necesario programar de forma automatizada este tipo de backups desde servidores remotos cada cierta cantidad de tiempo, para que se asegure que los datos estén almacenados en otro sitio.

También es recomendable utilizar **backup incremental** en casos de que solamente se necesite hacer respaldo de los datos que corresponden a cierta cantidad de tiempo, por ejemplo (un mes, un trimestre, un semestre, etc), lo cual puede ser útil en casos de que la organización pueda ser sometida a procesos de auditoría interna o externa.

En cuanto al volumen de datos es mucho más recomendable utilizar backup incremental cuando este volumen tiene a ser muy robusto, ya que al momento de realizar un backup completo de ese volumen de datos el tiempo en generarse puede ser mucho mayor, lo cual puede reducir el rendimiento del servidor de la base de datos. Por lo regular los datos que quieren ser almacenados (en caso de procesos internos) forman parte de periodos de tiempo en específico.

En conclusión como empresa siempre se recomienda implementar ambos tipos de backups para estar preparados ante cualquier eventualidad que pueda existir en la organización, como ataques, daños físicos tanto como de hardware y también es recomendable automatizar cada uno de los backups para que los datos de forma uniforme.

Conclusión final

Para asegurar de que los datos estén de forma completamente segura es muy importante hacer backups completos de forma periódica, para que en cualquier caso ya sea en un ataque, que el servidor presente fallas físicas o de software se pueda contar con un backup de forma completa y tenerlo disponible en cualquier momento.

Debido a que un backup completo con el tiempo puede llegar a ser muy grande puede llegar a ser muy complicado de ejecutar debido a que puede que en el servidor se hayan aplicado cambios (actualizaciones, migraciones de infraestructura) lo cual puede complicar la seguridad de la información, pero en caso de que se quiera recuperar datos correspondientes a cierto lapso de tiempo es mejor inclinarse por un backup incremental, ya que solo contiene una cantidad de datos específica y necesaria de acuerdo a los datos de la empresa.

En conclusión es recomendable manejar ambos tipos de backups, debido a que como organización siempre se debe estar preparado para los mejores y peores casos, ya que pueden presentarse fallas técnicas o que durante procesos administrativos o legales sea necesario realizar respaldos correspondientes a una cantidad de tiempo en específico (semestral, anual, mensual e incluso semanal).