



Datos 2.0

▼ Retos

▼ 03/10

The screenshot shows a MySQL Workbench window. The SQL editor tab contains the following code:

```
286     Left Join
287     Right Join
288     Full Join - JOIN UNION, JOIN
289 */
290
291 /* Modificaciones update sintaxis:
292 UPDATE nombreTabla SET campovalor, campo1 = valor1 WHERE condicion */
293 •   SELECT * FROM Mascota;
294 •   UPDATE Mascota SET nombreMascota = "Melu" WHERE idMascota = 7;
295 •   UPDATE Mascota SET nombreMascota = "Maximo" WHERE idMascota = 9;
296 •   UPDATE Mascota SET nombreMascota = "Melody" WHERE idMascota = 3;
```

The results grid shows the following data:

	idMascota	nombreMascota	raza	generoMascota	tipoMascota	fechaCreacion	documentoClienteFK
▶	1	Firulais	Labrador	Macho	Perro	2025-09-24 07:35:55	C001
	2	Firulais	Labrador	Macho	Perro	2025-09-24 07:35:55	C001
	3	Melody	Siames	Hembra	Gato	2025-09-24 07:35:56	C002
	4	Misu	Siames	Hembra	Gato	2025-09-24 07:35:57	C002
	5	Rocky	Bulldog	Macho	Perro	2025-09-24 07:35:58	C003
	6	Rocky	Bulldog	Macho	Perro	2025-09-24 07:35:59	C003
	7	Melu	Persa	Hembra	Gato	2025-09-24 07:36:00	C004
	8	Luna	Persa	Hembra	Gato	2025-09-24 07:36:00	C004
	9	Maximo	Golden Retriever	Macho	Perro	2025-09-24 07:36:02	C005
	10	Max	Golden Retriever	Macho	Perro	2025-09-24 07:36:03	C005
	11	Max	Golden Retriever	Macho	Perro	2025-09-24 07:36:03	C005

```
/* Modificaciones update sintaxis:
UPDATE nombreTabla SET campovalor, campo1 = valor1 WHERE condic
```

```

ion */
SELECT * FROM Mascota;
UPDATE Mascota SET nombreMascota = "Milu" WHERE idMascota = 7;
UPDATE Mascota SET nombreMascota = "Maximo" WHERE idMascota
= 9;
UPDATE Mascota SET nombreMascota = "Melody" WHERE idMascota
= 3;

```

Screenshot of MySQL Workbench showing the results of the previous SQL statements. The table 'Mascotas' contains the following data:

	idMascota	nombreMascota	raza	generoMascota	tipoMascota	fechaCreacion	documentoClienteFK
4	Misu	Siames	Hembra	Gato		2025-09-24 07:35:57	C002
5	Rocky	Bulldog	Macho	Perro		2025-09-24 07:35:58	C003
7	Milu	Persa	Hembra	Gato		2025-09-24 07:36:00	C004
8	Luna	Persa	Hembra	Gato		2025-09-24 07:36:00	C004
9	Maximo	Golden Retriever	Macho	Perro		2025-09-24 07:36:02	C005

```

/* DELETE eliminar
sintaxis: DELETE FROM nombreTabla WHERE condicion BEGIN ROLLBA
CK COMMIT*/
DELETE FROM Mascota WHERE idMascota = 6;
SELECT * FROM Mascota;

```

Screenshot of MySQL Workbench showing the results of the previous SQL statements. The table 'Mascotas' contains the following data:

	idMascota	nombreMascota	raza	generoMascota	tipoMascota	fechaCreacion	documentoClienteFK
4	Misu	Siames	Hembra	Gato		2025-09-24 07:35:57	C002
5	Rocky	Bulldog	Macho	Perro		2025-09-24 07:35:58	C003
8	Luna	Persa	Hembra	Gato		2025-09-24 07:36:00	C004
9	Maximo	Golden Retriever	Macho	Perro		2025-09-24 07:36:02	C005
10	Max	Golden Retriever	Macho	Perro		2025-09-24 07:36:03	C005

```

START TRANSACTION;
DELETE FROM Mascota WHERE idMascota = 7;
COMMIT;
SELECT * FROM Mascota;

```

```

START TRANSACTION;
DELETE FROM Mascota WHERE idMascota = 7;
ROLLBACK;
SELECT * FROM Mascota;

```

Screenshot of MySQL Workbench showing the results of the transactions. The left pane shows the SQL code, and the right pane shows the 'Result Grid' with data from the 'VistaDatosCliente' view.

SQL History:

```

314 •   SELECT * FROM information_schema.innodb_trx;
315 •   SHOW BINARY LOGS;
316
317 /* Views - triggers - procedimientos almacenados
318 CREATE VIEW nombreVista AS campos FROM tabla WHERE condicion;
319 SELECT * FROM nombreVista;
320 */
321 •   CREATE VIEW VistaMascota AS SELECT M.generoMascota AS 'Raza Mascota', M.tipoMascota AS 'Tipo Mascota'
322 •   SELECT * FROM VistaMascota;
323 •   CREATE VIEW VistaDatosCliente AS SELECT * FROM Cliente C INNER JOIN Mascota M ON C.documentoCliente =
324 •   SELECT * FROM VistaDatosCliente;

```

Result Grid:

documentoCliente	nombreCliente1	nombreCliente2	apellidoCliente1	apellidoCliente2	direccionCliente	idMascota	nombreMascota
C001	Ana	NULL	Ramírez	Gómez	Calle 10 #5-23	1	Firulais
C001	Ana	NULL	Ramírez	Gómez	Calle 10 #5-23	2	Firulais
C002	Luis	Fernando	Pérez	NULL	Carrera 45 #12-09	3	Melody
C002	Luis	Fernando	Pérez	NULL	Carrera 45 #12-09	4	Misu
C003	Maria	Isabel	Rodríguez	Díaz	Av. Siempre Viva 123	5	Rocky
C004	Andrés	NULL	López	Martínez	Calle 8 #20-15	8	Luna
C005	Carolina	Sofia	García	Suárez	Carrera 7 #33-45	9	Maximo
C005	Carolina	Sofia	García	Suárez	Carrera 7 #33-45	10	Max
C005	Carolina	Sofia	García	Suárez	Carrera 7 #33-45	11	Max

```

/* Views - triggers - procedimientos almacenados
CREATE VIEW nombreVista AS campos FROM tabla WHERE condicion;
SELECT * FROM nombreVista;
*/
CREATE VIEW VistaMascota AS SELECT M.generoMascota AS 'Raza M
ascota', M.tipoMascota AS 'Tipo Mascota' FROM Mascota M;
SELECT * FROM VistaMascota;
CREATE VIEW VistaDatosCliente AS SELECT * FROM Cliente C INNER J

```

```
OIN Mascota M ON C.documentoCliente = M.documentoClienteFK;  
SELECT * FROM VistaDatosCliente;
```

▼ 14/11

Tarea 1: Obtener todas las tareas

"obtenerTodas": "GET /api/tareas"

The screenshot shows the Postman interface with a request to `http://localhost:3005/api/tareas`. The response body is a JSON object:

```
1 {  
2   "success": true,  
3   "count": 2,  
4   "total": 2,  
5   "totalPages": 1,  
6   "currentPage": 1,  
7   "data": [  
8     {  
9       "_id": "691918f31491dd56a3043ca2",  
10      "titulo": "Tarea de prueba",  
11      "descripcion": "Esta es una prueba",  
12      "prioridad": "alta",  
13      "categoria": "trabajo",  
14      "estado": "pendiente",  
15      "fetchedAt": "2025-11-16T09:21:07.385Z",  
16      "__v": 0  
17    },  
18    {  
19      "_id": "69191aa3bade1269ca847359",  
20      "tit
```

Tarea 2: Obtener todas las tareas por ID

"obtenerPorId": "GET /api/tareas/:id"

The screenshot shows the Postman interface with a dark theme. In the left sidebar, there's a 'Collections' section with 'Ingeniería de Datos' expanded, showing a 'New Request' button. The main area has a 'Default workspace' header. A search bar at the top right says 'Search Postman'. Below it, a toolbar includes 'Invite', 'Upgrade', 'Save', 'Share', and 'Send'. A list of requests shows 'Ingeniería de Datos' with three items: 'GET http://localhost:3005/api/tareas/:id' (selected), 'GET http://localhost:3005/api/tareas', and 'GET http://localhost:3005/api/tareas'. The selected request details show a 'GET' method and the URL 'http://localhost:3005/api/tareas/:id'. The 'Params' tab is active, showing a table with one row: 'Key' (Value) and 'Description'. The 'Path Variables' section also has a table with one row: 'id' (Value) and 'Description'. The 'Body' tab shows a JSON response with the following content:

```
{ "success": false, "message": "ID no válido", "error": "\"":id\" no es un ObjectId válido de MongoDB" }
```

Tarea 3: Crear tareas

"crear": "POST /api/tareas"

This screenshot is similar to the previous one but shows a different request. The 'Default workspace' header is present, along with the 'Collections' sidebar and the 'Ingeniería de Datos' collection. The main area shows a 'POST' method and the URL 'http://localhost:3005/api/tareas'. The 'Params' tab is active, showing an empty table. The 'Body' tab shows a JSON response with the following content:

```
{ "success": false, "message": "Error de validación", "errors": [ { "Path": "descripcion" }, { "Path": "titulo" } ] }
```

Tarea 4: Actualizar tareas

"actualizar": "PUT /api/tareas/:id"

The screenshot shows the Postman interface with a request to `http://localhost:3005/api/tareas/:id`. The request method is `PUT`. The response status is `400 Bad Request`, with a duration of `10 ms` and a size of `374 B`. The response body is:

```
{  
  "success": false,  
  "message": "ID no válido",  
  "error": "\"':id\\\" no es un ObjectId válido de MongoDB"  
}
```

Tarea 5: Eliminar tareas

"eliminar": "DELETE /api/tareas/:id"

The screenshot shows the Postman interface with a DELETE request to `http://localhost:3005/api/tareas/:id`. The URL field contains `http://localhost:3005/api/tareas/5d34334fb10ff4001c000001`. The response status is 400 Bad Request, with a message: "success": false, "message": "ID no válido", "error": "\":id\" no es un ObjectId válido de MongoDB".

Tarea 6: Completar tareas

"completar": "PATCH /api/tareas/:id/completar"

The screenshot shows the Postman interface with a PATCH request to `http://localhost:3005/api/tareas/:id/completar`. The URL field contains `http://localhost:3005/api/tareas/5d34334fb10ff4001c000001/completar`. The response status is 400 Bad Request, with a message: "success": false, "message": "ID no válido", "error": "\":id\" no es un ObjectId válido de MongoDB".

Tarea 7: Reabrir tareas

"reabrir": "PATCH /api/tareas/:id/reabrir"

The screenshot shows the Postman interface with a request to `http://localhost:3005/api/tareas/:id/reabrir`. The request method is set to PATCH. The response status is 400 Bad Request, with a response time of 55 ms and a body size of 374 B. The response JSON is:

```
{  
  "success": false,  
  "message": "ID no válido",  
  "error": "\"':id\\\" no es un ObjectId válido de MongoDB"  
}
```

Tarea 8: Obtener estadísticas de las tareas

"estadisticas": "GET /api/tareas/stats"

The screenshot shows the Postman application interface. In the top navigation bar, 'Default workspace' is selected. The main area displays a GET request to 'http://localhost:3005/api/tareas/stats'. The 'Params' tab is active, showing a single parameter 'Key' with the value 'Value'. The 'Body' tab shows a JSON response with an error message:

```
1 {  
2   "success": false,  
3   "message": "Error al obtener estadisticas",  
4   "error": "Tarea.obtenerEstadisticas is not a function"  
5 }
```

The status bar at the bottom indicates a '500 Internal Server Error' with a duration of 12 ms and a size of 401 B.

Tarea 9: Obtener tareas por prioridad

"porPrioridad": "GET /api/tareas/prioridad/:prioridad"

The screenshot shows the Postman application interface. In the top navigation bar, 'Default workspace' is selected. The main area displays a GET request to 'http://localhost:3005/api/tareas/prioridad/:prioridad'. The 'Params' tab is active, showing a path variable 'prioridad' with the value 'Value'. The 'Body' tab shows a JSON response with an error message:

```
1 {  
2   "success": false,  
3   "message": "Prioridad no válida. Use: baja, media o alta"  
4 }
```

The status bar at the bottom indicates a '400 Bad Request' with a duration of 11 ms and a size of 351 B.