

**Consultas realizadas:**

**Alta de jugadores:**

```
INSERT INTO jugadores (nombre, email, puntosTotales) VALUES (?, ?, ?);
```

Inserta un nuevo jugador en la base de datos con puntuación inicial a 0 y guarda el ID generado.

**Contar partidas:**

```
SELECT COUNT(*) FROM partidas;
```

Se utiliza para comprobar que no se superen las 5 partidas máximas permitidas.

**Alta de partidas:**

```
INSERT INTO partidas (narrador_id, fecha, resultado) VALUES (?, ?, ?);
```

Guarda una nueva partida.

**Sumar puntos a un jugador:**

```
UPDATE jugadores SET puntosTotales = puntosTotales + ? WHERE id = ?;
```

Actualiza la puntuación de un jugador según el resultado de la partida.

**Jugador con más puntos:**

```
SELECT nombre, puntosTotales FROM jugadores ORDER BY puntosTotales DESC LIMIT 1;
```

Muestra el jugador que tiene más puntos.

**Ranking de jugadores:**

```
SELECT nombre, puntosTotales FROM jugadores ORDER BY puntosTotales DESC;
```

Lista todos los jugadores ordenados por puntos.

**Listar partidas por fecha:**

```
SELECT * FROM partidas ORDER BY fecha;
```

Muestra las partidas ordenadas.

**Capturas:**

The screenshot shows the MySQL Workbench interface with the following details:

- Navigator:** Shows the schema `jdbcmysqlacceso` with tables `jugadores`, `partidas`, `Views`, `Stored Procedures`, and `Functions`.
- SQL Editor:** Displays the SQL code for creating the database and tables.
- Output:** Shows the execution log with four entries:

#	Time	Action	Message	Duration / Fetch
1	18:48:18	CREATE DATABASE <code>jdbcMySQLAcceso</code>	1 row(s) affected	0.016 sec
2	18:48:18	CREATE TABLE <code>jdbcMySQLAcceso.jugadores</code> ( ... )	0 row(s) affected	0.031 sec
3	18:48:18	CREATE TABLE <code>jdbcMySQLAcceso.partidas</code> ( ... )	0 row(s) affected	0.031 sec
4	18:54:41	SELECT * FROM <code>jdbcmysqlacceso.jugadores</code> LIMIT ...	0 row(s) returned	0.000 sec / 0.000 sec

- Information:** Shows the schema `jdbcmysqlacceso`.
- System Tray:** Shows weather (10°C Nublado), time (19:24), date (16/12/2025), and a notification icon.

**Esquema y base de datos creada.**

## Adrián Solís León 2ºDAM

The screenshot shows the Spring Tools for Eclipse interface. The left pane displays the Package Explorer with project structures for 'AccesoADatosMaven' and 'AdrianProyectoJDBC'. The right pane shows the code editor for 'GestionadorTorneo.java'.

```
1 package Controller;
2
3 import java.time.LocalDate;
4
5 import Exceptions.MiExcepcion;
6 import Models.Jugador;
7 import Models.Partida;
8 import Models.TipoResultado;
9 import Repository.RepositorioJugador;
10 import Repository.RepositorioPartida;
11 import Service.TorneoService;
12
13 public class GestionadorTorneo {
14
15     public static void main(String[] args) {
16
17         try {
18             RepositorioJugador repoJugador = new RepositorioJugador();
19             RepositorioPartida repoPartida = new RepositorioPartida();
20             TorneoService servicio = new TorneoService(repoJugador, repoPartida);
21
22             Jugador j1 = new Jugador("JCViejo", "JCViejo@mail.com", "jici");
23             Jugador j2 = new Jugador("Rendon", "Rendon@mail.com", "alcala");
24             Jugador j3 = new Jugador("Fernando", "Fernando@mail.com", "loco");
25             Jugador j4 = new Jugador("Maste", "Maste@mail.com", "noviene");
26
27             servicio.darAltaJugador(j1);
28             servicio.darAltaJugador(j2);
29             servicio.darAltaJugador(j3);
30             servicio.darAltaJugador(j4);
31
32             servicio.crearPartida(new Partida(j1, LocalDate.now(), TipoResultado.ALGUNO));
33             servicio.crearPartida(new Partida(j2, LocalDate.now(), TipoResultado.TODOS));
34             servicio.crearPartida(new Partida(j3, LocalDate.now(), TipoResultado.NADIE));
35             servicio.crearPartida(new Partida(j4, LocalDate.now(), TipoResultado.ALGUNO));
36             servicio.crearPartida(new Partida(j1, LocalDate.now(), TipoResultado.TODOS));
37
38             // Sexta partida + excepción
39             servicio.crearPartida(new Partida(j2, LocalDate.now(), TipoResultado.ALGUNO));
40
41         } catch (MiExcepcion e) {
42             System.err.println(e.getMessage());
43         }
44     }
45 }
46
```

The terminal window at the bottom shows the application's log output:

```
<terminated> GestionadorTorneo [Java Application] C:\Users\Chatto\Downloads\spring-tools-for-eclipse-4.32.0.RELEASE-e4.37.0-wir
19:25:10.557 [main] INFO Repository.RepositorioJugador - Jugador dado de alta: JCViejo
19:25:10.560 [main] INFO Repository.RepositorioJugador - Jugador dado de alta: Rendon
19:25:10.563 [main] INFO Repository.RepositorioJugador - Jugador dado de alta: Fernando
19:25:10.565 [main] INFO Repository.RepositorioJugador - Jugador dado de alta: Maste
```

Damos de alta a 4 jugadores.

## Adrián Solís León 2ºDAM

The screenshot shows the MySQL Workbench interface. The top navigation bar includes File, Edit, View, Query, Database, Server, Tools, Scripting, and Help. The left sidebar displays the 'Navigator' and 'Schemas' sections, with the 'jdbcmysqlacceso' schema selected. Under this schema, tables like 'jugadores' and 'partidas' are listed. The main workspace contains a query editor with the following SQL code:

```
1 • use jdbcMySQLAcceso;
2
3 • select * from jugadores;
4
```

A tooltip message on the right states: "Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help."

The 'Result Grid' tab is active, showing the following data for the 'jugadores' table:

	id	nombre	email	puntosTotales
▶	1	JCViejo	JCViejo@mail.com	0
▶	2	Rendon	Rendon@mail.com	0
▶	3	Fernando	Fernando@mail.com	0
▶	4	Maste	Maste@mail.com	0
*	NULL	NULL	NULL	NULL

The 'Output' tab at the bottom shows the execution history:

#	Time	Action	Message	Duration / Fetch
1	18:48:18	CREATE DATABASE jdbcMySQLAcceso	1 row(s) affected	0.016 sec
2	18:48:18	CREATE TABLE jdbcMySQLAcceso.jugadores ( id IN... )	0 row(s) affected	0.031 sec
3	18:48:18	CREATE TABLE jdbcMySQLAcceso.partidas ( id IN... )	0 row(s) affected	0.031 sec
4	18:54:41	SELECT * FROM jdbcmysqlacceso.jugadores LIMIT ...	0 row(s) returned	0.000 sec / 0.000 sec
5	19:26:52	use jdbcMySQLAcceso	0 row(s) affected	0.000 sec
6	19:26:52	select * from jugadores LIMIT 0, 1000	4 row(s) returned	0.000 sec / 0.000 sec

The status bar at the bottom right shows the weather (10°C, Nublado), time (19:26), and date (16/12/2025).

**Comprobamos que se han añadido correctamente.**

## Adrián Solís León 2ºDAM

The screenshot shows the Spring Tools for Eclipse interface. The left sidebar displays the Package Explorer with project structure. The main editor window shows the Java code for `GestionadorTorneo.java`. The code creates four players and attempts to create five matches, but only four are successfully created due to a limit of 5. The right side shows the Console tab with the execution output.

```
AccesoDatos - AdrianProyectoJDBC/src/main/java/Controller/GestionadorTorneo.java - Spring Tools for Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer X
GestionadorTorneo.java X
7 import Models.Partida;
8 import Models.TipoResultado;
9 import Repository.RepositorioJugador;
10 import Repository.RepositorioPartida;
11 import Service.TorneoService;
12
13 public class GestionadorTorneo {
14
15     public static void main(String[] args) {
16
17         try {
18             RepositorioJugador repoJugador = new RepositorioJugador();
19             RepositorioPartida repoPartida = new RepositorioPartida();
20             TorneoService servicio = new TorneoService(repoJugador, repoPartida);
21
22             Jugador j1 = new Jugador("JCViejo", "JCViejo@mail.com", "jici");
23             Jugador j2 = new Jugador("Rendon", "Rendon@mail.com", "alcala");
24             Jugador j3 = new Jugador("Fernando", "Fernando@mail.com", "loco");
25             Jugador j4 = new Jugador("Maste", "Maste@mail.com", "noviene");
26
27             servicio.darAltaJugador(j1);
28             servicio.darAltaJugador(j2);
29             servicio.darAltaJugador(j3);
29             servicio.darAltaJugador(j4);
29
30             servicio.crearPartida(new Partida(j1, LocalDate.now(), TipoResultado.ALGUNO));
31             servicio.crearPartida(new Partida(j2, LocalDate.now(), TipoResultado.TODO));
32             servicio.crearPartida(new Partida(j3, LocalDate.now(), TipoResultado.NADIE));
33             servicio.crearPartida(new Partida(j4, LocalDate.now(), TipoResultado.ALGUNO));
34             servicio.crearPartida(new Partida(j1, LocalDate.now(), TipoResultado.TODO));
35
36             // Sexta partida → excepción
37             servicio.crearPartida(new Partida(j2, LocalDate.now(), TipoResultado.ALGUNO));
38
39         } catch (MiExcepcion e) {
40             System.err.println(e.getMessage());
41         }
42     }
43 }
44
45 }
```

Console X

```
<terminated> GestionadorTorneo [Java Application] C:\Users\Chat0\Downloads\spring-tools-for-eclipse-4.32.0.RELEASE-e4.37.0-wire
19:32:14.660 [main] INFO Repository.RepositorioJugador - Jugador dado de alta: JCViejo
19:32:14.664 [main] INFO Repository.RepositorioJugador - Jugador dado de alta: Rendon
19:32:14.666 [main] INFO Repository.RepositorioJugador - Jugador dado de alta: Fernando
19:32:14.669 [main] INFO Repository.RepositorioJugador - Jugador dado de alta: Maste
19:32:14.683 [main] INFO Repository.RepositorioPartida - Partida dada de alta
19:32:14.687 [main] INFO Repository.RepositorioPartida - Partida dada de alta
19:32:14.711 [main] INFO Repository.RepositorioPartida - Partida dada de alta
19:32:14.733 [main] INFO Repository.RepositorioPartida - Partida dada de alta
19:32:14.753 [main] INFO Repository.RepositorioPartida - Partida dada de alta
No se pueden crear más de 5 partidas
```

Creamos las 5 partidas.

## Adrián Solís León 2ºDAM

The screenshot shows the MySQL Workbench interface. The top menu bar includes File, Edit, View, Query, Database, Server, Tools, Scripting, and Help. The left sidebar displays the 'Navigator' and 'Schemas' sections, showing the database structure for 'jdbcmysqlacceso'. The central workspace contains a SQL editor tab with the query:

```
1 •   SELECT * FROM jdbcmysqlacceso.partidas;
```

To the right of the query, a note states: "Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help."

Below the query, there is a 'Result Grid' tab showing the results of the query:

	id	torneo_id	narrador_id	fecha	resultado
▶	7	NULL	5	2025-12-16	ALGUNOS
8	NULL	6	2025-12-16	TODOS	
9	NULL	7	2025-12-16	NADIE	
10	NULL	8	2025-12-16	ALGUNOS	
11	NULL	5	2025-12-16	TODOS	
*	NULL	NULL	NULL	NULL	NULL

The bottom section shows the 'Action Output' log:

#	Time	Action	Message	Duration / Fetch
5	19:26:52	use jdbcMySqlAcceso	0 row(s) affected	0.000 sec
6	19:26:52	select * from jugadores LIMIT 0, 1000	4 row(s) returned	0.000 sec / 0.000 sec
7	19:31:32	SELECT * FROM jdbcmysqlacceso.jugadores LIMIT 0, 1000	4 row(s) returned	0.000 sec / 0.000 sec
8	19:31:47	Refresh Recordset	There are pending changes. Please commit or roll...	
9	19:31:52	SELECT * FROM jdbcmysqlacceso.jugadores LIMIT 0, 1000	4 row(s) returned	0.000 sec / 0.000 sec
10	19:33:01	SELECT * FROM jdbcmysqlacceso.partidas LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec

Comprobamos en la base de datos

## Adrián Solís León 2ºDAM

The screenshot shows the Spring Tools for Eclipse interface. The left side features the Package Explorer with project structure, and the right side shows the code editor and a terminal window.

**Code Editor (GestionadorTorneo.java):**

```
1 import Models.Partida;
2 import Models.TipoResultado;
3 import Repository.RepositorioJugador;
4 import Repository.RepositorioPartida;
5 import Service.TorneoService;
6
7 public class GestionadorTorneo {
8
9     public static void main(String[] args) {
10
11         try {
12             RepositorioJugador repoJugador = new RepositorioJugador();
13             RepositorioPartida repoPartida = new RepositorioPartida();
14             TorneoService servicio = new TorneoService(repoJugador, repoPartida);
15
16             Jugador j1 = new Jugador("JCViejo", "JCViejo@mail.com", "jici");
17             Jugador j2 = new Jugador("Rendon", "Rendon@mail.com", "alcala");
18             Jugador j3 = new Jugador("Fernando", "Fernando@mail.com", "loco");
19             Jugador j4 = new Jugador("Maste", "Maste@mail.com", "noviene");
20
21             servicio.darAltaJugador(j1);
22             servicio.darAltaJugador(j2);
23             servicio.darAltaJugador(j3);
24             servicio.darAltaJugador(j4);
25
26             servicio.crearPartida(new Partida(j1, LocalDate.now(), TipoResultado.ALGUNO));
27             servicio.crearPartida(new Partida(j2, LocalDate.now(), TipoResultado.TODO));
28             servicio.crearPartida(new Partida(j3, LocalDate.now(), TipoResultado.NADIE));
29             servicio.crearPartida(new Partida(j4, LocalDate.now(), TipoResultado.ALGUNO));
30             servicio.crearPartida(new Partida(j1, LocalDate.now(), TipoResultado.TODO));
31
32             // Sexta partida → excepción
33             servicio.crearPartida(new Partida(j2, LocalDate.now(), TipoResultado.ALGUNO));
34
35         } catch (MiExcepcion e) {
36             System.err.println(e.getMessage());
37         }
38     }
39 }
40
41
42
43
44
45
46
```

**Console Output:**

```
<terminated> GestionadorTorneo [Java Application] C:\Users\Chat0\Downloads\spring-tools-for-eclipse-4.32.0.RELEASE-e4.37.0-wi
19:32:14.660 [main] INFO Repository.RepositorioJugador - Jugador dado de alta: JCViejo
19:32:14.664 [main] INFO Repository.RepositorioJugador - Jugador dado de alta: Rendon
19:32:14.666 [main] INFO Repository.RepositorioJugador - Jugador dado de alta: Fernando
19:32:14.669 [main] INFO Repository.RepositorioJugador - Jugador dado de alta: Maste
19:32:14.683 [main] INFO Repository.RepositorioPartida - Partida dada de alta
19:32:14.687 [main] INFO Repository.RepositorioPartida - Partida dada de alta
19:32:14.711 [main] INFO Repository.RepositorioPartida - Partida dada de alta
19:32:14.733 [main] INFO Repository.RepositorioPartida - Partida dada de alta
19:32:14.753 [main] INFO Repository.RepositorioPartida - Partida dada de alta
No se pueden crear más de 5 partidas
```

Nos salta la excepción al crear la 6 partida.

## Adrián Solís León 2ºDAM

The screenshot shows the Spring Tools for Eclipse interface. The central area displays the Java code for `GestionadorTorneo.java`. The code implements a `TorneoService` interface, interacting with `Repository` classes for `Jugador` and `Partida`. It creates four `Jugador` objects and six `Partida` objects, updating player scores and displaying the top player. A `MiException` is catched to handle any errors.

```
import Repository.RepositorioPartida;
import Service.TorneoService;

public class GestionadorTorneo {

    public static void main(String[] args) {
        try {
            RepositorioJugador repoJugador = new RepositorioJugador();
            RepositorioPartida repoPartida = new RepositorioPartida();
            TorneoService servicio = new TorneoService(repoJugador, repoPartida);

            Jugador j1 = new Jugador("JCViejo", "JCViejo@mail.com", "jici");
            Jugador j2 = new Jugador("Rendon", "Rendon@mail.com", "alcala");
            Jugador j3 = new Jugador("Fernando", "Fernando@mail.com", "loco");
            Jugador j4 = new Jugador("Maste", "Maste@mail.com", "noviene");

            servicio.darAltaJugador(j1);
            servicio.darAltaJugador(j2);
            servicio.darAltaJugador(j3);
            servicio.darAltaJugador(j4);

            servicio.actualizarPuntuacionAcertante(1, TipoResultado.TODO);
            /*servicio.crearPartida(new Partida(j1, LocalDate.now(), TipoResultado.ALGO));
            servicio.crearPartida(new Partida(j2, LocalDate.now(), TipoResultado.TODO));
            servicio.crearPartida(new Partida(j3, LocalDate.now(), TipoResultado.NADIE));
            servicio.crearPartida(new Partida(j4, LocalDate.now(), TipoResultado.ALGUNO));
            servicio.crearPartida(new Partida(j1, LocalDate.now(), TipoResultado.TODO));

            // Sexta partida → excepción
            //servicio.crearPartida(new Partida(j2, LocalDate.now(), TipoResultado.ALGO));
            */
            servicio.mostrarJugadorConMasPuntos();
            servicio.mostrarRanking();
            servicio.listarPartidas();
        } catch (MiException e) {
            System.err.println(e.getMessage());
        }
    }
}
```

The `Console` tab at the bottom shows the execution output:

```
<terminated> GestionadorTorneo [Java Application] C:\Users\Chat0\Downloads\spring-tools-for-eclipse-4.32.0.RELEASE-e4.37.0-wi
5.600 [main] INFO Repository.RepositorioJugador - Jugador dado de alta: JCViejo
5.605 [main] INFO Repository.RepositorioJugador - Jugador dado de alta: Rendon
5.609 [main] INFO Repository.RepositorioJugador - Jugador dado de alta: Fernando
5.612 [main] INFO Repository.RepositorioJugador - Jugador dado de alta: Maste
5.617 [main] INFO Repository.RepositorioJugador - Jugador con más puntos: JCViejo (2 puntos)
5.618 [main] INFO Repository.RepositorioJugador - Jugador: JCViejo | Puntos: 2
5.619 [main] INFO Repository.RepositorioJugador - Jugador: Rendon | Puntos: 0
5.619 [main] INFO Repository.RepositorioJugador - Jugador: Fernando | Puntos: 0
5.619 [main] INFO Repository.RepositorioJugador - Jugador: Maste | Puntos: 0
5.619 [main] INFO Repository.RepositorioJugador - Jugador: JCViejo | Puntos: 0
5.619 [main] INFO Repository.RepositorioJugador - Jugador: Rendon | Puntos: 0
5.619 [main] INFO Repository.RepositorioJugador - Jugador: Fernando | Puntos: 0
5.620 [main] INFO Repository.RepositorioJugador - Jugador: Maste | Puntos: 0
5.620 [main] INFO Repository.RepositorioJugador - Jugador: JCViejo | Puntos: 0
```

Info por puntos y por fecha de partidas:

The screenshot shows the Spring Tools for Eclipse interface. The central area displays the Java code for `GestionadorTorneo.java`. The code imports `Repository.RepositorioPartida` and `Service.TorneoService`, defines a `public static void main(String[] args)` method, and creates four `Jugador` objects (`j1` through `j4`). It then calls methods from `TorneoService` to add each player and update their scores. The `actualizarPuntuacionAcertante` method is highlighted with a blue selection bar. The code ends with a `catch` block for `MiExcepcion` exceptions. Below the code editor is the `Console` view, which shows the terminal output of the application's execution. The output logs various `INFO` messages from the `main` method of the `GestionadorTorneo` class, detailing the creation of players and matches, and their resulting points.

```
import Repository.RepositorioPartida;
import Service.TorneoService;

public class GestionadorTorneo {

    public static void main(String[] args) {
        try {
            RepositorioJugador repoJugador = new RepositorioJugador();
            RepositorioPartida repoPartida = new RepositorioPartida();
            TorneoService servicio = new TorneoService(repoJugador, repoPartida);

            Jugador j1 = new Jugador("JCViejo", "JCViejo@mail.com", "jici");
            Jugador j2 = new Jugador("Rendon", "Rendon@mail.com", "alcala");
            Jugador j3 = new Jugador("fernando", "Fernando@mail.com", "loco");
            Jugador j4 = new Jugador("Maste", "Maste@mail.com", "noviene");

            servicio.darAltaJugador(j1);
            servicio.darAltaJugador(j2);
            servicio.darAltaJugador(j3);
            servicio.darAltaJugador(j4);

            servicio.actualizarPuntuacionAcertante(j1, TipoResultado.TODO);
            /*servicio.crearPartida(new Partida(j1, LocalDate.now(), TipoResultado.ALGUNO));
            servicio.crearPartida(new Partida(j2, LocalDate.now(), TipoResultado.TODO));
            servicio.crearPartida(new Partida(j3, LocalDate.now(), TipoResultado.NADIE));
            servicio.crearPartida(new Partida(j4, LocalDate.now(), TipoResultado.ALGUNO));
            servicio.crearPartida(new Partida(j1, LocalDate.now(), TipoResultado.TODO));

            // Sexta partida > excepción
            //servicio.crearPartida(new Partida(j2, LocalDate.now(), TipoResultado.ALGUNO));
            */
            servicio.mostrarJugadorConMasPuntos();
            servicio.mostrarRanking();
            servicio.listarPartidas();
        } catch (MiExcepcion e) {
            System.err.println(e.getMessage());
        }
    }
}
```

```
<terminated> GestionadorTorneo [Java Application] C:\Users\Chato\Downloads\spring-tools-for-eclipse-4.32.0.RELEASE-e4.37.0-wi
5.621 [main] INFO Reppositoy.RepositorioJugador - Jugador: JCViejo | Puntos: 0
5.621 [main] INFO Reppositoy.RepositorioJugador - Jugador: Rendon | Puntos: 0
5.621 [main] INFO Reppositoy.RepositorioJugador - Jugador: Fernando | Puntos: 0
5.621 [main] INFO Reppositoy.RepositorioJugador - Jugador: Maste | Puntos: 0
5.621 [main] INFO Reppositoy.RepositorioJugador - Jugador: JCViejo | Puntos: 0
5.621 [main] INFO Reppositoy.RepositorioJugador - Jugador: Rendon | Puntos: 0
5.621 [main] INFO Reppositoy.RepositorioJugador - Jugador: Fernando | Puntos: 0
5.621 [main] INFO Reppositoy.RepositorioJugador - Jugador: Maste | Puntos: 0
5.624 [main] INFO Reppositoy.RepositorioPartida - Partida ID 7 | Fecha 2025-12-16T00:00:00.00
5.625 [main] INFO Reppositoy.RepositorioPartida - Partida ID 8 | Fecha 2025-12-16T00:00:00.00
5.625 [main] INFO Reppositoy.RepositorioPartida - Partida ID 9 | Fecha 2025-12-16T00:00:00.00
5.625 [main] INFO Reppositoy.RepositorioPartida - Partida ID 10 | Fecha 2025-12-16T00:00:00.00
5.625 [main] INFO Reppositoy.RepositorioPartida - Partida ID 11 | Fecha 2025-12-16T00:00:00.00
```