Parcial

Michael Stiven Castillo Castillo Id 825947

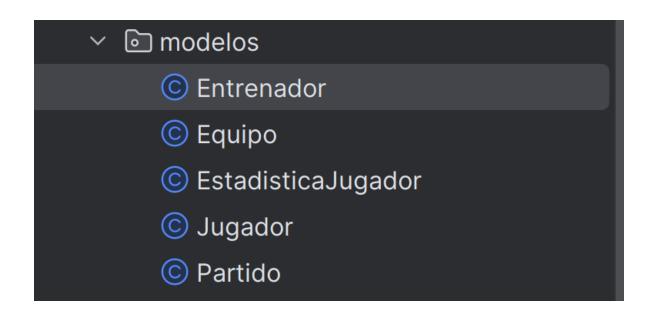
Corporación Universitaria Minuto de Dios

Ingeniería de Sistemas

Arquitectura

Alexander Matallana Porras

24 de abril del 2025

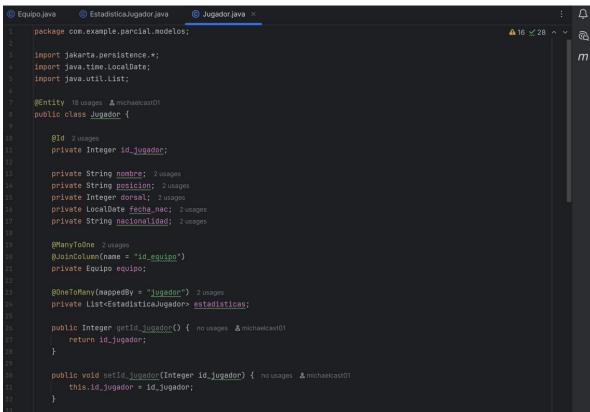


```
public class Equipo {
    private Integer id_equipo;
    private String nombre; 2 usages
    private String nombre; 2 usages
    private String cludad; 2 usages
    private Listring requipo;
    @OneToMany(mappedBy = "equipo", cascade = CascadeType.ALL) 2 usages
    private Listring nombre; 2 usages
    private Listring requipo", cascade = CascadeType.ALL) 2 usages
    private Listring requipo", cascade = CascadeType.ALL) 2 usages
    private Listring requipo", cascade = CascadeType.ALL) 2 usages
    private Listring requipo = "equipo", cascade = CascadeType.ALL) 2 usages
    private Listring requipo = "equipo", cascade = CascadeType.ALL) 2 usages
    private Listring requipo = "equipo local") 2 usages
    private Listring requipo = "equipo", no usages & michaelcast01
    return id_equipo = id_equipo;
}

public void setId_equipo { no usages & michaelcast01
    return nombre;
}

public void setMombre(String nombre) { no usages & michaelcast01
    return nombre;
}

public void setMombre(String nombre) { no usages & michaelcast01
    return nombre;
}
```



- com.example.parcial
 controladores
 modelos
 repositorios
 servicios
 - **@** ParcialApplication

```
package com.example.parcial.controladores;
import com.example.parcial.servicios.Entrenador;
import com.example.parcial.servicios.EntrenadorServicio;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.web.bind.annotation.*;

mort java.util.List;

RestController nousages new*
RequestHapping("/entrenadores")

public class IntrenadorControlador {

Rautowired 3 usages
private EntrenadorServicio servicio;

RegtUp (RestController nousages new*
public List<Entrenadors obtenerTodos() {
 return servicio.obtenerTodos();
}

RepostHapping nousages new*
public Entrenador quardar(RequestBody Entrenador entrenador) {
 return servicio.guardar(entrenador);
}

RepostHapping nousages new*
public Entrenador quardar(RequestBody Entrenador entrenador) {
 return servicio.guardar(entrenador);
}

RepostHapping nousages new*
public contrenador quardar(RequestBody Entrenador entrenador) {
 return servicio.guardar(entrenador);
}

RepostHapping nousages new*
public contrenador quardar(RequestBody Entrenador entrenador) {
 return servicio.guardar(entrenador);
}

RepostHapping nousages new*
public contrenador quardar(RequestBody Entrenador entrenador) {
 return servicio.guardar(entrenador);
}

RepostHapping nousages new*
public contrenador quardar(RequestBody Entrenador entrenador) {
 return servicio.guardar(entrenador);
}

RepostHapping nousages new*
public contrenador quardar(RequestBody Entrenador entrenador) {
 return servicio.guardar(entrenador);
}

RepostHapping nousages new*
public contrenador entrenador entrenador) {
 return servicio.guardar(entrenador);
}
```

```
spring.datasource.url=${DB_URL}

spring.datasource.username=${DB_USERNAME}

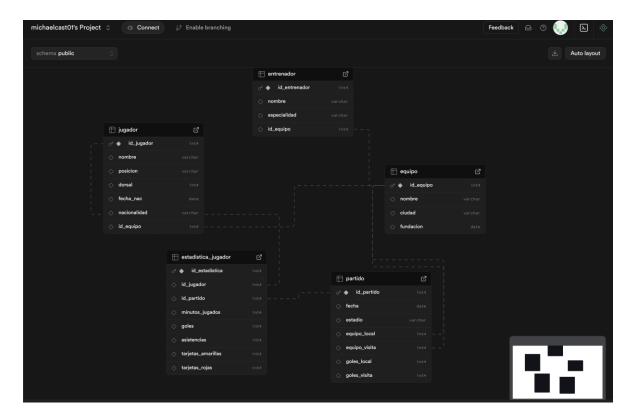
spring.datasource.password=${DB_PASSWORD}}

spring.datasource.driver-class-name=org.postgresql.Driver

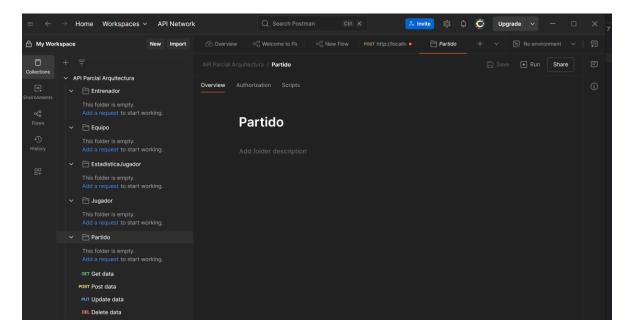
spring.jpa.hibernate.ddl-auto=update

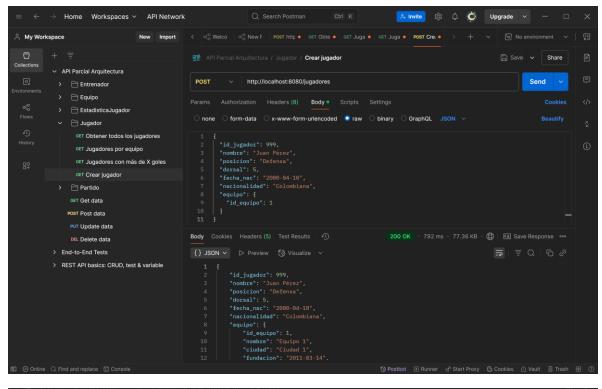
spring.jpa.show-sql=true

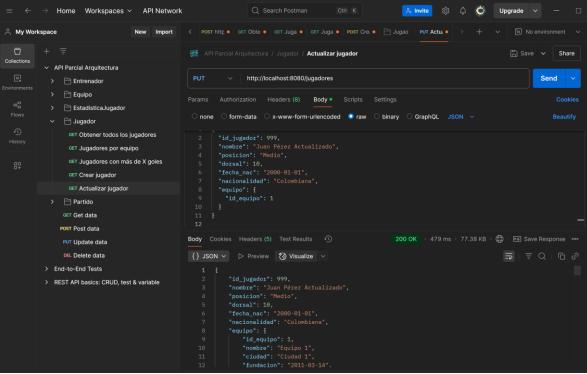
spring.jpa.properties.hibernate.dialect=org.hibernate.dialect.PostgreSQLDialect
```

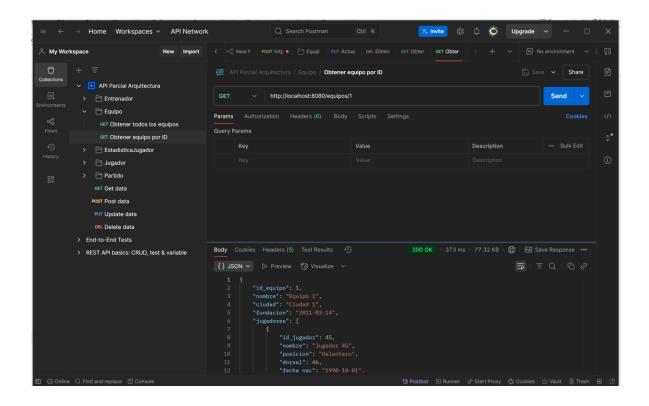


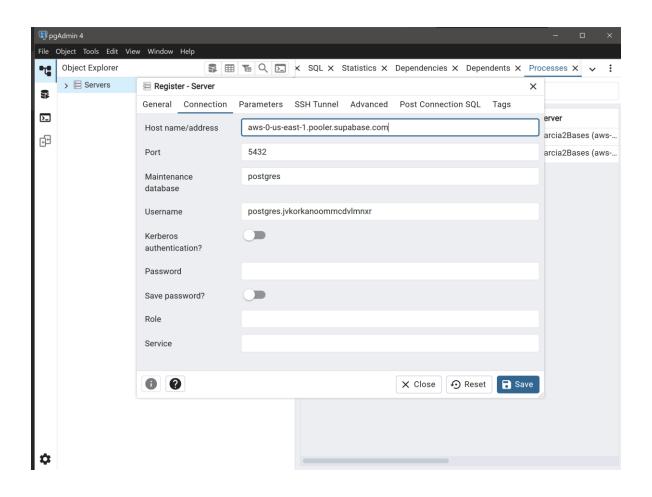
Crud Postman

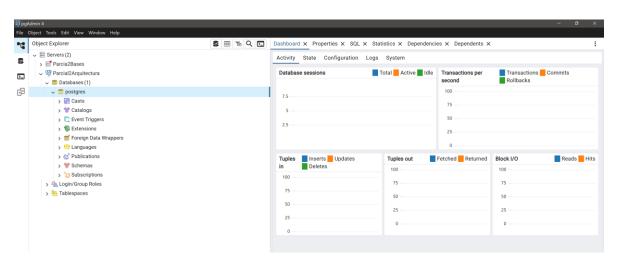


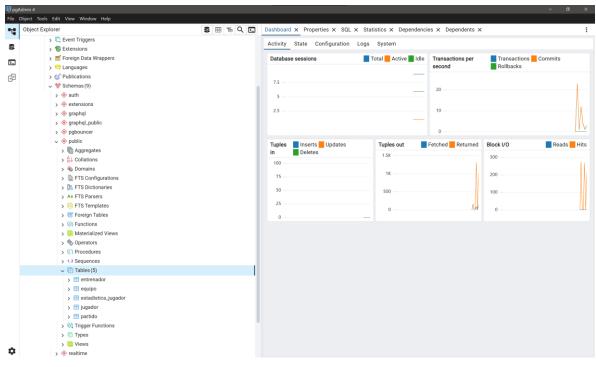




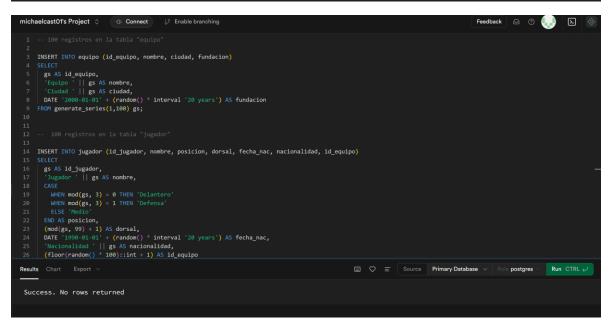


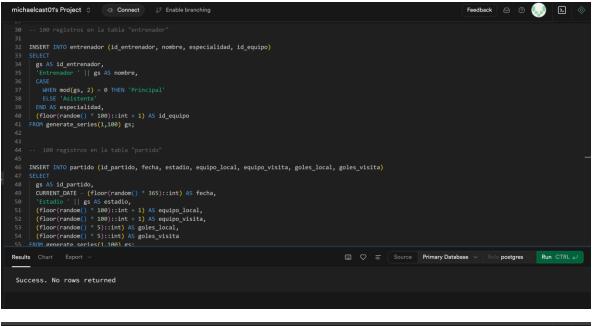


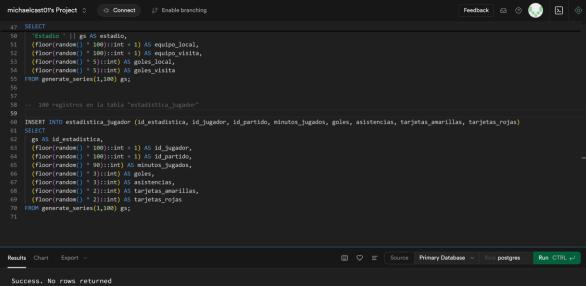


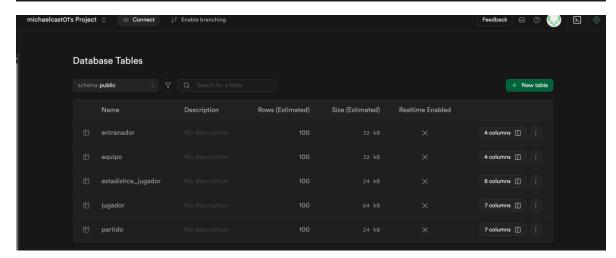


```
michaelcast01's Project \( \begin{array}{c} & - \text{Connect} \\ \end{array} \) Enable branching
29 CREATE TABLE partido (
     id_partido INT PRIMARY KEY,
        fecha DATE,
        estadio VARCHAR(100),
        equipo_local INT REFERENCES equipo(id_equipo),
         equipo_visita INT REFERENCES equipo(id_equipo),
         goles_local INT,
         goles_visita INT
40 CREATE TABLE estadistica_jugador (
         id estadistica INT PRIMARY KEY,
         id_jugador INT REFERENCES jugador(id_jugador),
         id_partido INT REFERENCES partido(id_partido),
        minutos_jugados INT,
         goles INT,
         asistencias INT,
         tarjetas_amarillas INT,
         tarjetas_rojas INT
```









✓ Consultas Nativas

- Esta consulta retorna todos los jugadores que pertenecen a un equipo específico.
 Se utiliza un parámetro id que representa el identificador del equipo.
 Se ejecuta mediante un método GET en /jugadores/equipo/{id}.
- Esta consulta devuelve los jugadores que han marcado más de una cierta cantidad de goles.

Se suman los goles desde la tabla estadistica_jugador y se filtran los que superan el parámetro min.

Se accede con GET usando /jugadores/goles?min=3 (o el número que el usuario quiera).

- Consulta el número total de goles anotados por un equipo, considerando todos sus partidos como local o visitante.
 - Se utiliza un parámetro idEquipo que se pasa en la URL.
 - Se accede por GET en /partidos/goles/{idEquipo}.
- Esta consulta muestra todos los partidos jugados incluyendo la fecha, el nombre del equipo local, el visitante, y los goles de cada uno.
 - Utiliza joins para relacionar los IDs de equipos con sus nombres reales.
 - Se consulta con un GET a /partidos/resultados.