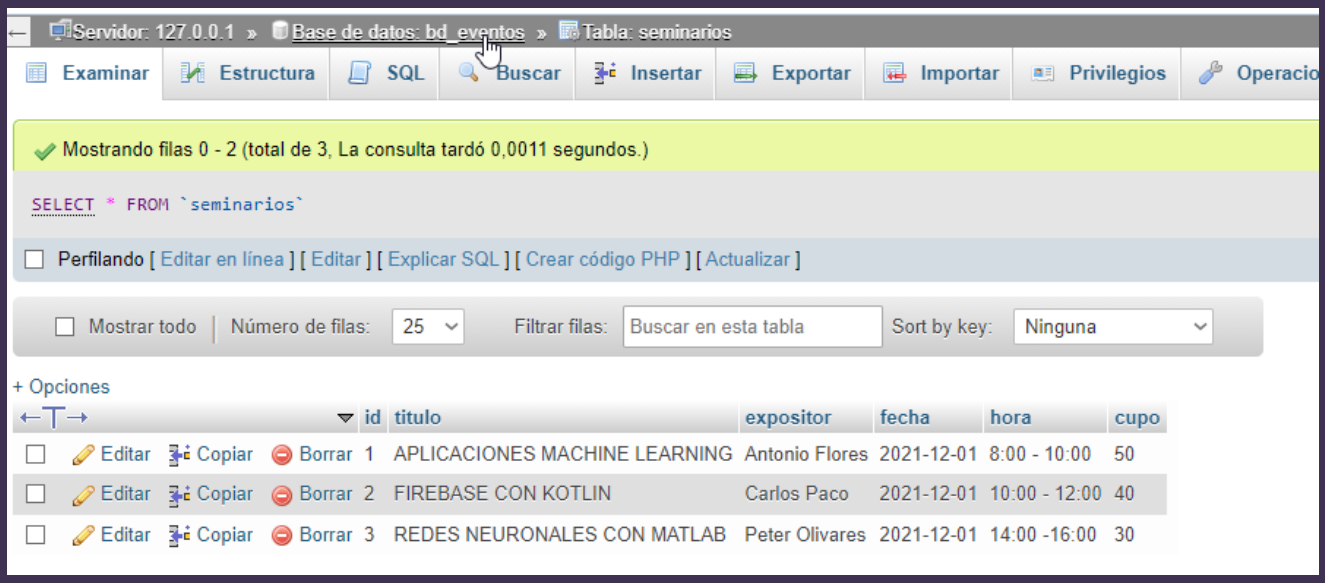
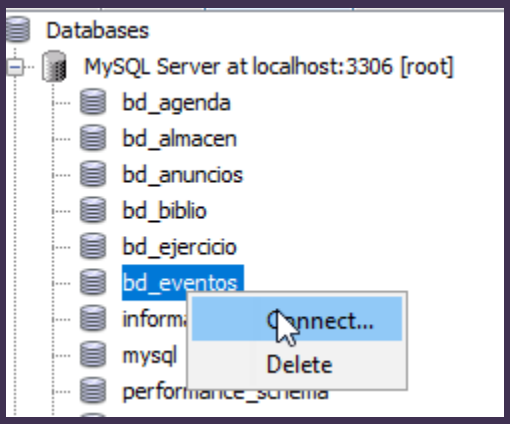
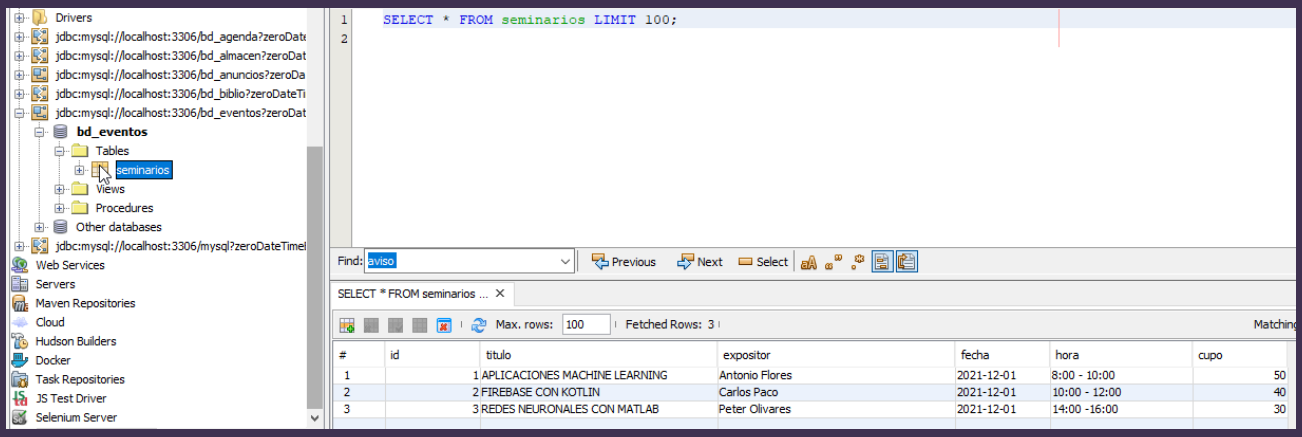
Estudiante: Mery Judith Mamani Mamani	**Fecha:** 09/11/2021	**M**
Paralelo: 7 c	**Materia:** Tec. Emergentes II	

PARCIAL 2

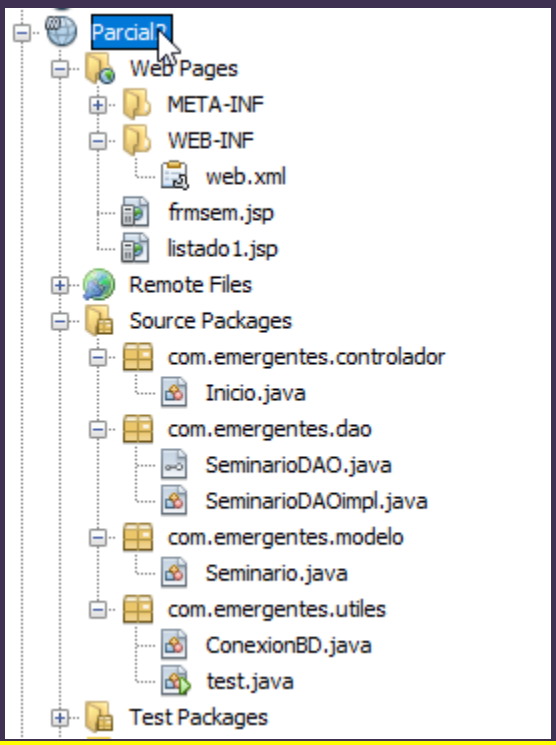
Base de datos: bd_eventos







ESTRUCTURA



Listado1.jsp (menú principal)

Código:

```
<%@taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!doctype html>
<html lang="en">
  <head>
    <!-- Required meta tags -->
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <!-- Bootstrap CSS -->
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css"
rel="stylesheet" integrity="sha384-
1BmE4kWBq78iYhFldvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3" crossorigin="anonymous">

    <title>Listado</title>
  </head>
  <div style="text-align:center">
      <font face="Verdana">
      <fieldset style="border: ridge #C0392B  2px;">
          <legend>SEGUNDO PARCIAL TEM-742</legend>
          <p>Nombres: Mery Judith Mamani Mamani</p>
          <p>Carnet: 9234793</p>
      </fieldset>
      </font>
  </div>

  <body>
      <div class="container">
          <h1 style="text-align:center">Registro de Seminarios</h1>
          <p><a href="Inicio?action=add" class="btn btn-primary btn-sm">Nuevo</a></p>
```

```
          <table class="table">
              <thead class="table-dark">
                  <tr>
                      <th>ID</th>
                      <th>TITULO</th>
                      <th>EXPOSITOR</th>
                      <th>FECHA</th>
                      <th>HORA</th>
                      <th>CUPO</th>
                      <th colspan="2">ACCIONES</th>
                  </tr>
              </thead>
              <tbody>
              <c:forEach var="item" items="${seminarios}">
                  <tr>
                      <td>${item.id}</td>
                      <td>${item.titulo}</td>
                      <td>${item.expositor}</td>
                      <td>${item.fecha}</td>
                      <td>${item.hora}</td>
                      <td>${item.cupo}</td>
                      <td><a href="Inicio?action=edit&id=${item.id}" class="btn btn-
warning">Editar</a></td>
                      <td><a href="Inicio?action=delete&id=${item.id}"
onclick="return(confirm('¿Está seguro de eliminar?'))" class="btn btn-danger">Eliminar</a></td>
                  </tr>
              </c:forEach>
              </tbody>
          </table>

      </div>
      <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js"
integrity="sha384-ka7Sk0Gln4gmtz2MlQnikT1wXgYsOg+OMhuP+IlRH9sENBO0LRn5q+8nbTov4+1p"
crossorigin="anonymous"></script>

  </body>
</html>
```

Frmsem.jsp

Código:

```
<%@taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!doctype html>
<html lang="en">
  <head>
    <!-- Required meta tags -->
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css"
rel="stylesheet" integrity="sha384-
1BmE4kWBq78iYhFldvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3" crossorigin="anonymous">
  </head>
  <body>
      <div class="container">
          <h1>
              <c:if test="${seminario.id==0}">NUEVO</c:if>
              <c:if test="${seminario.id!=0}">EDITAR</c:if>
```

```
            AVISO
          </h1>

         <form action="Inicio" method="post">
                <input class="form-control form-control-lg" type="hidden" name="id"
value=${seminario.id}>
                <table>
                <tr>
                    <td>Titulo:</td>
                    <td><input type="text" name="titulo" value="${seminario.titulo}"></td>
                </tr>
                <tr>
                    <td>Expositor: </td>
                    <td><input type="text" name="expositor"
value="${seminario.expositor}"></td>
                </tr>
                <tr>
                    <td>Fecha: </td>
                    <td><input type="date" name="fecha" value="${seminario.fecha}"></td>

                </tr>
                <tr>
                    <td>Hora: </td>
                    <td><input type="text" name="hora" value="${seminario.hora}"></td>

                </tr>
                <tr>
                    <td>Cupo: </td>
                    <td><input type="text" name="cupo" value="${seminario.cupo}"></td>

                </tr>
                <tr>
                    <td colspan="2"><input class="btn btn-primary btn-lg" type="submit"
value="Enviar" ></td>
                </tr>

            </table>
        </form>

      </div>
      <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js"
integrity="sha384-ka7Sk0Gln4gmtz2MlQnikT1wXgYsOg+OMhuP+IlRH9sENBO0LRn5q+8nbTov4+1p"
crossorigin="anonymous"></script>
  </body>
</html>
```

Inicio.java

Código:

```
package com.emergentes.controlador;

import com.emergentes.dao.SeminarioDAO;
import com.emergentes.dao.SeminarioDAOimpl;
import com.emergentes.modelo.Seminario;
import java.io.IOException;
import java.io.PrintWriter;
import java.util.List;
import javax.persistence.metamodel.SetAttribute;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

@WebServlet(name = "Inicio", urlPatterns = {"/Inicio"})
public class Inicio extends HttpServlet {

    @Override
    protected void doGet(HttpServletRequest request, HttpServletResponse response)
            throws ServletException, IOException {
        try {
            SeminarioDAO dao = new SeminarioDAOimpl();
            int id;
            Seminario sem= new Seminario();
            String action = (request.getParameter("action") != null) ?
request.getParameter("action") : "view";

            switch (action) {
                case "add":
                    request.setAttribute("seminario", sem);
                    request.getRequestDispatcher("frmsem.jsp").forward(request, response);
                    break;
                case "edit":
                    id=Integer.parseInt(request.getParameter("id"));
                    sem = dao.getById(id);
                    request.setAttribute("seminario", sem);
                    request.getRequestDispatcher("frmsem.jsp").forward(request, response);
                    break;

                case "delete":
                    id = Integer.parseInt(request.getParameter("id"));
                    dao.delete(id);
                    response.sendRedirect("Inicio");
                    break;

                default:
                    //listar
                    List<Seminario> lista = dao.getAll();
                    request.setAttribute("seminarios",lista);
                    request.getRequestDispatcher("listado1.jsp").forward(request, response);
                    break;
            }
        } catch (Exception e) {
            System.out.println("error"+e.getMessage());
```

```
        }
    }

    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
            throws ServletException, IOException {
        SeminarioDAO dao = new SeminarioDAOimpl();

        int id = Integer.parseInt(request.getParameter("id"));
        String titulo = request.getParameter("titulo");
        String expositor = request.getParameter("expositor");
        String fecha = request.getParameter("fecha");
        String hora = request.getParameter("hora");
        int cupo = Integer.parseInt(request.getParameter("cupo"));

        Seminario sem = new Seminario();

        sem.setId(id);
        sem.setTitulo(titulo);
        sem.setExpositor(expositor);
        sem.setFecha(fecha);
        sem.setHora(hora);
        sem.setCupo(cupo);

        if(id==0){
            //nuevo
            try {
                dao.insert(sem);
                response.sendRedirect("Inicio");
            } catch (Exception e) {
                System.out.println("error" + e.getMessage());
            }
        } else {
            //modificacion
            try {
                dao.update(sem);
                response.sendRedirect("Inicio");
            } catch (Exception e) {
                System.out.println("error" + e.getMessage());
            }
        }
    }

}
```

SeminarioDAO.java

Código:

```
package com.emergentes.dao;
import com.emergentes.modelo.Seminario;
import java.util.List;

public interface SeminarioDAO {
    public void insert(Seminario ev)throws Exception;
    public void update(Seminario ev)throws Exception;
    public void delete(int id)throws Exception;
    public Seminario getById(int id)throws Exception;
    public List<Seminario> getAll()throws Exception;
}
```

SeminarioDAOimpl.java

Código:

```
package com.emergentes.dao;
import com.emergentes.modelo.Seminario;
import com.emergentes.utiles.ConexionBD;
import com.mysql.cj.protocol.Resultset;
import com.mysql.cj.xdevapi.PreparableStatement;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.util.ArrayList;
import java.util.List;

 public class SeminarioDAOimpl extends ConexionBD implements SeminarioDAO{

    @Override
    public void insert(Seminario sem) throws Exception {
        try {
            this.conectar();
            String sql="insert into seminarios (titulo,expositor,fecha,hora,cupo) values
(?,?,?,?,?)";
            PreparedStatement ps = this.conn.prepareStatement(sql);

            ps.setString(1, sem.getTitulo());
            ps.setString(2, sem.getExpositor());
            ps.setString(3, sem.getFecha());
            ps.setString(4, sem.getHora());
            ps.setInt(5, sem.getCupo());

            //ejecutamos la consulta
            ps.executeUpdate();
        } catch (Exception e) {
            throw e;
        }finally{
            this.desconectar();
        }
    }

    @Override
    public void update(Seminario sem) throws Exception {
        try {
            this.conectar();
            String sql="update seminarios set titulo=?,expositor=?,fecha=?,hora=?,cupo=? where
id=?";
```

```
        PreparedStatement ps = this.conn.prepareStatement(sql);

        ps.setString(1, sem.getTitulo());
        ps.setString(2, sem.getExpositor());
        ps.setString(3, sem.getFecha());
        ps.setString(4, sem.getHora());
        ps.setInt(5, sem.getCupo());
        ps.setInt(6, sem.getId());
        //ejecutamos la consulta
        ps.executeUpdate();
    } catch (Exception e) {
        throw e;
    }finally{
        this.desconectar();
    }
}

@Override
public void delete(int id) throws Exception {
    try {
        this.conectar();
        String sql="delete from seminarios where id=?";
        PreparedStatement ps = this.conn.prepareStatement(sql);
        ps.setInt(1, id);
        //ejecutamos la consulta
        ps.executeUpdate();
    } catch (Exception e) {
        throw e;
    }finally{
        this.desconectar();
    }
}

@Override
public Seminario getById(int id) throws Exception {
    Seminario sem = new Seminario();
    try {
        this.conectar();
        String sql="select * from seminarios where id=?";
        PreparedStatement ps = this.conn.prepareStatement(sql);
        ps.setInt(1, id);
        ResultSet rs= ps.executeQuery();

        if(rs.next()){
            sem.setId(rs.getInt("id"));
            sem.setTitulo(rs.getString("titulo"));
            sem.setExpositor(rs.getString("expositor"));
            sem.setFecha(rs.getString("fecha"));
            sem.setHora(rs.getString("hora"));
            sem.setCupo(rs.getInt("cupo"));
        }
    } catch (Exception e) {
        throw e;

    } finally {
        this.desconectar();
    }
    return sem;
}
```

```
    @Override
    public List<Seminario> getAll() throws Exception {
       List<Seminario> lista = null;
        try {
            this.conectar();
            String sql="select * from seminarios";
            PreparedStatement ps = this.conn.prepareStatement(sql);
            ResultSet rs= ps.executeQuery();
            lista = new ArrayList<Seminario>();

            while(rs.next()){
                Seminario sem = new Seminario();
                sem.setId(rs.getInt("id"));
                sem.setTitulo(rs.getString("titulo"));
                sem.setExpositor(rs.getString("expositor"));
                sem.setFecha(rs.getString("fecha"));
                sem.setHora(rs.getString("hora"));
                sem.setCupo(rs.getInt("cupo"));

                lista.add(sem);
            }
        } catch (Exception e) {
            throw e;

        } finally {
            this.desconectar();
        }
        return lista;
    }
}
```

Seminario.java

Código:

```
package com.emergentes.modelo;

public class Seminario {
     private int id;
    private String titulo;
    private String expositor;
    private String fecha;
    private String hora;
    private int cupo;

    public int getId() {
        return id;
    }

    public void setId(int id) {
        this.id = id;
    }

    public String getTitulo() {
        return titulo;
    }

    public void setTitulo(String titulo) {
        this.titulo = titulo;
```

```
    }

    public String getExpositor() {
        return expositor;
    }

    public void setExpositor(String expositor) {
        this.expositor = expositor;
    }

    public String getFecha() {
        return fecha;
    }

    public void setFecha(String fecha) {
        this.fecha = fecha;
    }

    public String getHora() {
        return hora;
    }

    public void setHora(String hora) {
        this.hora = hora;
    }

    public int getCupo() {
        return cupo;
    }

    public void setCupo(int cupo) {
        this.cupo = cupo;
    }

    @Override
    public String toString() {
        return "Evento{" + "id=" + id + ", titulo=" + titulo + ", expositor=" + expositor + ",
fecha=" + fecha + ", hora=" + hora + ", cupo=" + cupo + '}';
    }
}
```

ConexionBD.java

Código:

```
package com.emergentes.utiles;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;
import java.util.logging.Level;
import java.util.logging.Logger;

public class ConexionBD {

    static String driver = "com.mysql.jdbc.Driver";
    static String url = "jdbc:mysql://localhost:3306/bd_eventos";
    static String usuario = "root";
    static String password = "";
   protected Connection conn = null;
    public ConexionBD() {
        try {
            // Especificación del driver
            Class.forName(driver);
            // Establece la conexión a la base de datos
            conn = DriverManager.getConnection(url, usuario, password);
            // Verificar si la conexión fue exitosa
            if (conn != null) {
                System.out.println("Conexión Exitosa :" + conn);
            }
        } catch (SQLException ex) {
            System.out.println("Error de SQL" + ex.getMessage());

        } catch (ClassNotFoundException ex) {
            Logger.getLogger(ConexionBD.class.getName()).log(Level.SEVERE, null, ex);
        }
}
public Connection conectar() {
        return conn;
    }

    public void desconectar() {
        try {
            conn.close();
        } catch (SQLException ex) {
            Logger.getLogger(ConexionBD.class.getName()).log(Level.SEVERE, null, ex);
        }
    }
}
```

test.java

Código:

```
package com.emergentes.utiles;
public class test {
    public static void main(String[] args) {
        ConexionBD con = new ConexionBD();
        con.conectar();
    }

}
```

pom.xml

```
    <properties>
        <endorsed.dir>${project.build.directory}/endorsed</endors
        <project.build.sourceEncoding>UTF-8</project.build.source
    </properties>

    <dependencies>
        <dependency>
            <groupId>javax</groupId>
            <artifactId>javaee-web-api</artifactId>
            <version>7.0</version>
            <scope>provided</scope>
        </dependency>
        <dependency>
            <groupId>javax.servlet</groupId>
            <artifactId>jstl</artifactId>
            <version>1.2</version>
        </dependency>
        <dependency>
            <groupId>mysql</groupId>
            <artifactId>mysql-connector-java</artifactId>
            <version>8.0.26</version>
        </dependency>
    </dependencies>
```

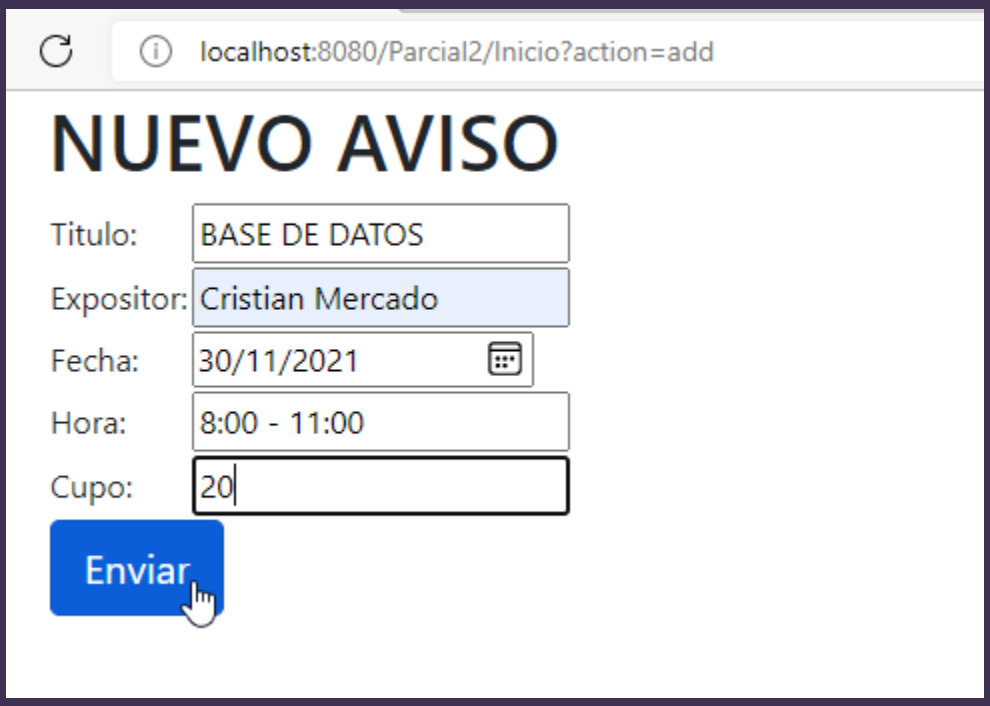
web.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app version="3.1" xmlns="http://xmlns.jcp.org/xml/ns/javaee" xmlns:xsi="http://www.w
    <session-config>
        <session-timeout>
            30
        </session-timeout>
    </session-config>
    <welcome-file-list>
        <welcome-file>Inicio</welcome-file>
    </welcome-file-list>
</web-app>
```

FUNCIONAMIENTO

Visualización de registros:

Nuevo registro:

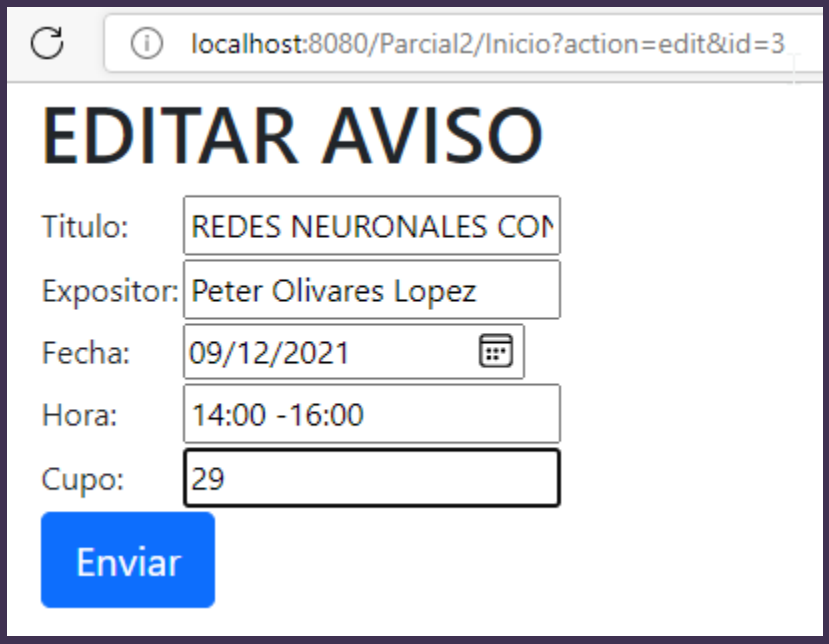


Edición de registro:

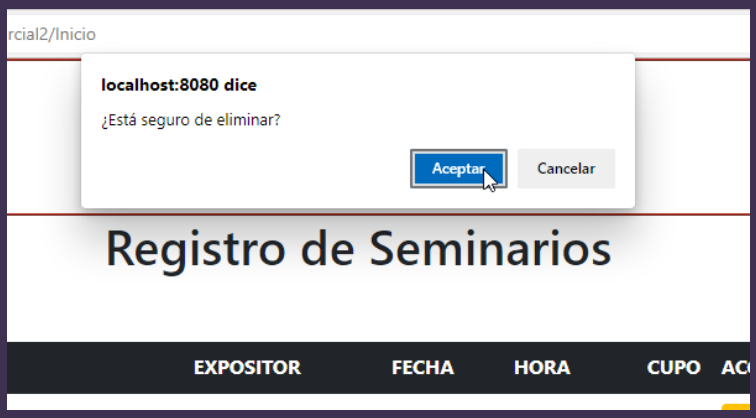
Registro de Seminarios

Nuevo

ID	TITULO	EXPOSITOR	FECHA	HORA	CUPO	ACCIONES	
1	APLICACIONES MACHINE LEARNING	Antonio Flores	2021-12-01	8:00 - 10:00	50	Editar	Eliminar
2	FIREBASE CON KOTLIN	Carlos Paco	2021-12-01	10:00 - 12:00	40	Editar	Eliminar
3	REDES NEURONALES CON MATLAB	Peter Olivares	2021-12-01	14:00 -16:00	30	Editar	Eliminar
5	BASE DE DATOS	Cristian Mercado	2021-11-30	8:00 - 11:00	20	Editar	Eliminar



Eliminación de registro:



LINK - GITHUB

https://github.com/ByMeyJudy/parcial-2.git