public class AccesoDatos { //SQL AccesoDatos

private static String url\_mysql = "jdbc:mysql://localhost:3306/bd\_alumnos?user=root";

public static void insertarContacto(Contacto c)

{ try { Class.forName("com.mysql.cj.jdbc.Driver");

Connection conn = DriverManager.getConnection(url\_mysql);

Statement stmt = conn.createStatement();

String sql ="INSERT INTO t\_alumnos (nombre, edad, telefono) VALUES('" + c.getNombre() + "','" + c.getEdad() + "','" + c.getTelefono() + "')";

stmt.executeUpdate(sql);

} catch (SQLException e) {

e.printStackTrace();

} catch (ClassNotFoundException e) {

e.printStackTrace();}

} public static ArrayList<Contacto> recuperarContactos() {

ArrayList<Contacto> lista = new ArrayList<>();

try { Class.forName("com.mysql.cj.jdbc.Driver");

Connection conn = DriverManager.getConnection(url\_mysql);

Statement stmt = conn.createStatement();

String sql = "SELECT \* FROM t\_alumnos";

ResultSet rs = stmt.executeQuery(sql);

while (rs.next()) {

String nombre = rs.getString("nombre");

String edad = rs.getString("edad");

String telefono = rs.getString("telefono");

Contacto c = new Contacto(nombre, edad, telefono);

lista.add(c);

} } catch (SQLException e) {

e.printStackTrace(); }

catch (ClassNotFoundException e) {e.printStackTrace();}

return lista; }

public static ArrayList<Contacto> buscarContactos(String nombre) {

ArrayList<Contacto> lista = new ArrayList<>();

try { Class.forName("com.mysql.cj.jdbc.Driver");

Connection conn = DriverManager.getConnection(url\_mysql);

Statement stmt = conn.createStatement();

String sql = "SELECT \* FROM t\_alumnos WHERE nombre= '" + nombre + "'";

ResultSet rs = stmt.executeQuery(sql);

while (rs.next()) { String nombreAlumno = rs.getString("nombre");

String edad = rs.getString("edad");

String telefono = rs.getString("telefono");

Contacto c = new Contacto(nombreAlumno, edad, telefono);

lista.add(c);}

} catch (SQLException e) {

e.printStackTrace(); }

catch (ClassNotFoundException e) {e.printStackTrace();}

return lista;

} public static void borrarContacto(String tlf) {

try {

Class.forName("com.mysql.cj.jdbc.Driver");

Connection conn = DriverManager.getConnection(url\_mysql);

Statement stmt = conn.createStatement();

String sql = "DELETE FROM t\_alumnos WHERE telefono = '" + tlf + "'";

stmt.executeUpdate(sql);

stmt.close();

conn.close();

} catch (SQLException | ClassNotFoundException e) {

e.printStackTrace();

}}

------------ SERVLET

protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {

String accion = request.getParameter("accion");

if(accion != null)

{ if(accion.equals("escribir"))

{ String nombre = request.getParameter("name");

String edad = request.getParameter("age");

String telefono = request.getParameter("tel");

Contacto nuevoContacto = new Contacto(nombre, edad, telefono);

AccesoDatos.insertarContacto(nuevoContacto);

request.getRequestDispatcher("index.jsp").forward(request, response);

} else if(accion.equals("listar"))

{ ArrayList<Contacto> agenda = AccesoDatos.recuperarContactos();

request.setAttribute("listaContacto", agenda);

request.getRequestDispatcher("vercontacto.jsp").forward(request, response);

}else if (accion.equals("buscar"))

{ String nombreBuscado = request.getParameter("search");

ArrayList<Contacto> contactosEncontrados = AccesoDatos.buscarContactos(nombreBuscado);

request.setAttribute("contactosEncontrados", contactosEncontrados);

request.getRequestDispatcher("mostrarContacto.jsp").forward(request, response);

}else if(accion.equals("borrar"))

{ String tlf=request.getParameter("telefono");

AccesoDatos.borrarContacto(tlf);

ArrayList<Contacto> lista = AccesoDatos.recuperarContactos();

request.setAttribute("listaContacto", lista);

request.getRequestDispatcher("index.jsp").forward(request, response);

} } }

-------- PINTAR HTML

public class PintarHTML {

public static String crearTabla(ArrayList<Contacto> lista)

{String tabla="<table class='tabla\_contactos'><tr><th>NOMBRE</th><th>EDAD</th><th>TELEFONO</th><th>BORRAR</th></tr>";

for(int i = 0; i<lista.size(); i++)

{Contacto c = lista.get(i);

tabla += "<tr><td>" + c.getNombre()+"</td><td>"+ c.getEdad() + "</td><td>" + c.getTelefono() + "</td><td><a href='Servlet?accion=borrar&telefono="+c.getTelefono()+"'><img src='./imagenes/basura.jpg' width='20px'></td></tr>"; }

tabla += "</table>";

return tabla;} }

--------- MOSTRAR CONTACTOS

ESTAS EN TUS CONTACTOS BUSCADOS

<% ArrayList<Contacto> contactos =(ArrayList<Contacto>) request.getAttribute("contactosEncontrados");

String tabla\_html = PintarHTML.crearTabla(contactos);%>

<%=tabla\_html %>

<a href="index.jsp">

<button type="button">Volver al menu principal</button></a>

public class AccesoDatos {static String route = "C:\\Users\\madrid\\eclipse-workspace\\AplicacionServer\\datos.csv";

public static String insertarContacto(Contacto c)

{try {

FileWriter fw = new FileWriter(route, true);

fw.write(c.getNombre() + "," + c.getEdad() + "," + c.getTelefono() + "\n");

fw.close();

} catch(IOException e) {

System.out.println("La ruta no existe"); }

return "OK"; }

public static ArrayList<Contacto> recuperarContactos(){

ArrayList<Contacto> lista = null;

try {lista = new ArrayList<Contacto>();

FileReader fr=new FileReader(route);

BufferedReader br = new BufferedReader(fr);

String linea = br.readLine();

while(linea!=null) {String[] datos = linea.split(",");

String nombre = datos[0];

String edad = datos[1];

String telefono = datos[2];

Contacto c = new Contacto(nombre, edad, telefono);

lista.add(c);

linea = br.readLine();

} } catch (FileNotFoundException e) {

e.printStackTrace();

} catch (IOException e) {e.printStackTrace();}

return lista;

} public static ArrayList<Contacto> buscarContactos(String nombre, String route) {

ArrayList<Contacto> lista = new ArrayList<>();

try { FileReader fr = new FileReader(route);

BufferedReader br = new BufferedReader(fr);

String linea;

while ((linea = br.readLine()) != null) {

String[] datos = linea.split(",");

if (datos.length >= 3 && datos[0].equalsIgnoreCase(nombre)) {

String nombreContacto = datos[0];

String edad = datos[1];

String telefono = datos[2];

Contacto contacto = new Contacto(nombreContacto, edad, telefono);

lista.add(contacto);

} } } catch (IOException e) { e.printStackTrace(); }

return lista;

}public static ArrayList<Contacto> Borrar(String tlf) { ArrayList<Contacto> lista = new ArrayList<Contacto>();

lista = recuperarContactos(); try {FileWriter fw = new FileWriter(route, false);

fw.write("");

fw.close();

fw = new FileWriter(route,true);

for (int i = 0; i< lista.size(); i++) {

if(!lista.get(i).getTelefono().equals(tlf)) {

Contacto c = new Contacto(lista.get(i).getNombre(),lista.get(i).getEdad(),lista.get(i).getTelefono());

insertarContacto(c);

} }fw.close(); } catch (IOException e) { e.printStackTrace(); } return lista;} }

---------------------------- CLASE CONTACTO public class Contacto { String nombre; String edad; String telefono; public Contacto(String nombre, String edad, String telefono) {super();

this.nombre = nombre; this.edad = edad; this.telefono = telefono; }

public String getNombre() {

return nombre;

}public void setNombre(String nombre) {

this.nombre = nombre;

}public String getEdad() {

return edad;

}public void setEdad(String edad) {

this.edad = edad;

}public String getTelefono() {

return telefono;

}public void setTelefono(String telefono) {

this.telefono = telefono; } }