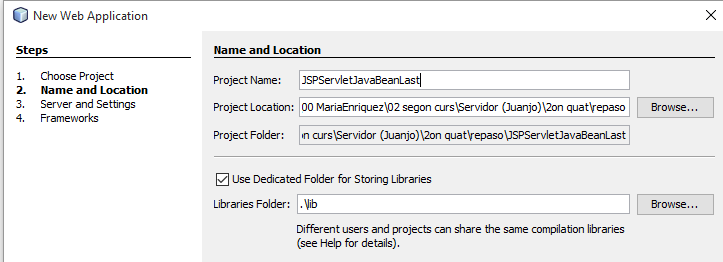
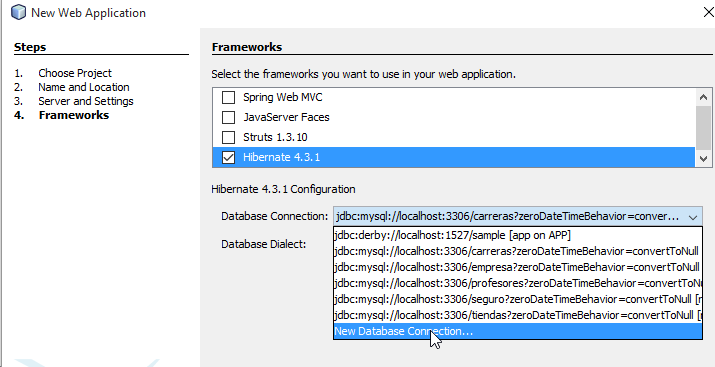
JSPSERVLETJAVABEAN **ENCIENDE EL XAMPP**

1 New Project, java web y eliges nombre, marcar Use Dedicated Folder **JavaHibernateRepasLast**

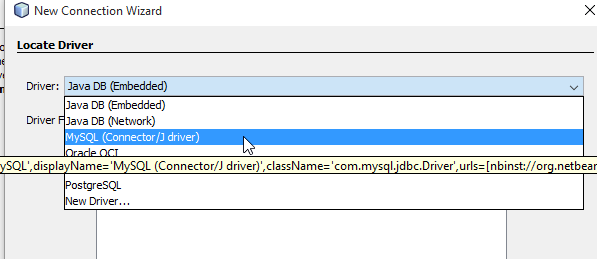
**SOLO LO CREAS DALE A ADD LIBRARY Y AÑADE MYSQL SINO NO TE FUNCIONARÁ**



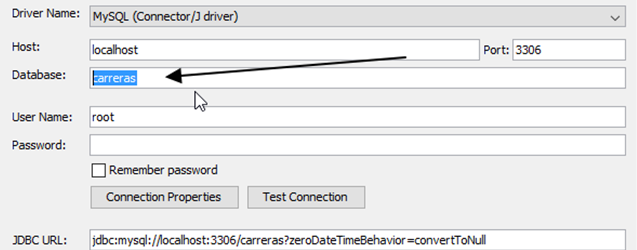
2 En Frameworks eliges Hibernate Y creas la conexión con la base de datos



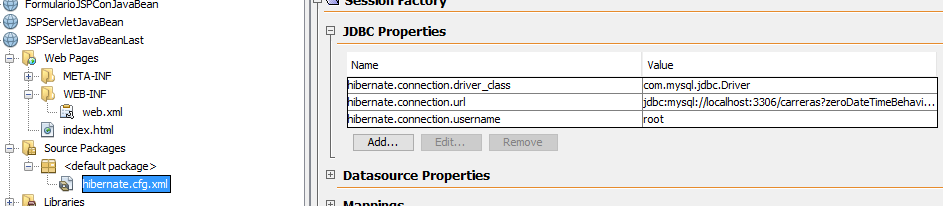
3 Seleccionas MySQL



4 Pones el nombre de la base de datos y haces un test



5 Comprobamos que en Default package/hibernate.cfg.xml se ha creado bien la conexión



El código debe ser algo así:

|  |
| --- |
| <hibernate-configuration>  <session-factory>  <property name="hibernate.dialect">org.hibernate.dialect.MySQLDialect</property>  <property name="hibernate.connection.driver\_class">com.mysql.jdbc.Driver</property>  <property name="hibernate.connection.url">jdbc:mysql://localhost:3306/carreras?zeroDateTimeBehavior=convertToNull</property>  <property name="hibernate.connection.username">root</property>  </session-factory>  </hibernate-configuration> |

6 Creo el **Package Entidad**

7 Creo dentro de Entidad **- Java Class Participante** (le añado a la cabecera implements Serializable) le añado las variables que voy a necesitar

|  |
| --- |
| public class Participante implements Serializable{  private int dorsal;  private String nombre;  private String apellidos;  private String poblacion;  private String club; |



8 Creo 2 contructores:

-1 vacío

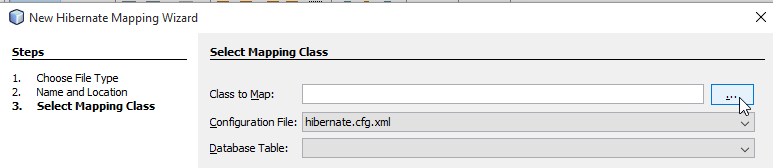
- otro con todos

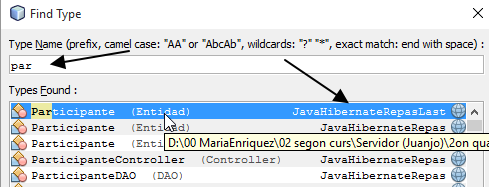
-Getters y setters

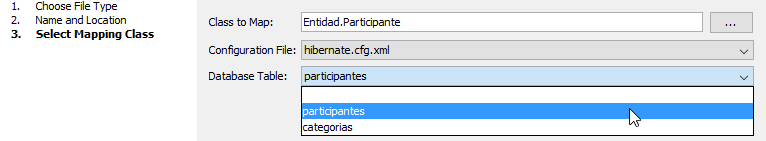
9 - En Entidad **Nuevo hibernate mapping wizard** -> Cambiar el **nombre** igual que la entidad **Participante.hbm**

-> Buscas el mismo nombre que toca en el Class to Map - > Entidad.Participante -> Database table la que necesitemos









Comprobamos que en hibernate.cfg aparece el mapping a lo que hemos creado <mapping resource="Entidad/Participante.hbm.xml"/>

10 - E Participante.hbm quitamos la barra de cerrado del class y dentro de </class> ponemos:

|  |
| --- |
| <class name="Entidad.Participante" table="participantes">  <id column="Dorsal" name="dorsal" type="integer"></id>  <property name="nombre" column="Nombre"></property>  <property name="apellidos" column="Apellidos"></property>  <property name="poblacion" column="Poblacion"></property>  <property name="club" column="CLUB"></property>  </class> |

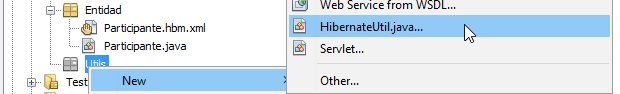
11 Creamos un WEB-PAGES/index.jsp

12 Package DAO

13 ParticipanteDAO

14 Necesitamos abrir conexión creamos un package Utils

15 New Other -> hibernateUtil FINISH



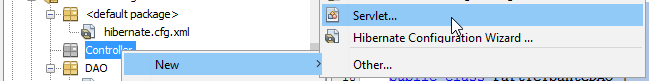
16 En ParticipanteDAO vamos a crear Guardar

-Creamos estos métodos estáticos

|  |
| --- |
| import Entidad.Participante;  import Utils.HibernateUtil;  import java.util.ArrayList;  import org.hibernate.Query;  import org.hibernate.Session;  public class ParticipanteDAO {  **public static void guardar(Participante participante){**  // Participante participante=new Participante(13456,"juanje","hibernate manual","gata","club");  Session session=HibernateUtil.getSessionFactory().openSession();  try {  session.beginTransaction();  session.saveOrUpdate(participante);  session.getTransaction().commit();  } catch (Exception e) {  session.getTransaction().rollback();  }finally{  session.close();  }  }  public static void **borrar**(Participante participante){  // Participante participante=new Participante(13456,"juanje","hibernate manual","gata","club");  Session session=HibernateUtil.getSessionFactory().openSession();  try {  session.beginTransaction();  session.delete(participante);  session.getTransaction().commit();  } catch (Exception e) {  session.getTransaction().rollback();  }finally{  session.close();  }  }  public static ArrayList<Participante> **getAllParticipantes**(){  Session session=HibernateUtil.getSessionFactory().openSession();  String hql="SELECT p FROM Participante p ORDER BY apellidos"; //el select es sobre la clase, los nombres son los mismos que hay en la clase y no la BD  Query query=session.createQuery(hql);  ArrayList<Participante> listaParticipantes=null;  listaParticipantes=(ArrayList<Participante>) query.list();  session.close();  return listaParticipantes;  }  public static Participante **getParticipanteByDorsal**(int dorsal){  Session session=HibernateUtil.getSessionFactory().openSession();  Participante participante=null;  String hql="SELECT p FROM Participante p WHERE dorsal=:dorsal";  Query query=session.createQuery(hql);  query.setParameter("dorsal", dorsal);  participante=(Participante) query.uniqueResult();  session.close();  return participante;  }  } |

17 Package Controller

- New Servlet ParticipanteController -> next -> add information -> finish





18 Borramos toda la KAKA del try -> Solo pongo esto:

|  |
| --- |
| protected void processRequest(HttpServletRequest request, HttpServletResponse response)  throws ServletException, IOException {  ArrayList<Participante> listaPart = ParticipanteDAO.getAllParticipantes();  request.setAttribute("listaPart", listaPart);  request.getRequestDispatcher("/index.jsp").forward(request, response);  } |

19 Ahora en index.jsp ponemos toda la pesca para que haga la lista (importamos arraylist y entidad.participante)

|  |
| --- |
| <body>  <h1>Lista de participantes</h1>  <a href="insertar.jsp">Añadir participante</a>  <table border="1">  <th>Dorsal</th>  <th>Nombre</th>  <th>Apellidos</th>  <th>Poblacion</th>  <th>Club</th>  <th>Editar</th>  <th>Borrar</th>  <!-- esto es un scriptlet para poder usar funciones de java -->  <% ArrayList<Participante> listaPart = (ArrayList<Participante>) request.getAttribute("listaPart");  for (Participante part : listaPart) {  if(part!=null){  %>  <tr>  <td><%= part.getDorsal()%></td>  <td><%= part.getNombre()%></td>  <td><%= part.getApellidos()%></td>  <td><%= part.getPoblacion()%></td>  <td><%= part.getClub()%></td>  <td><a href="EditarParticipante?dorsal=<%=part.getDorsal()%>">Editar</a></td>  <td><a href="BorrarParticipante?dorsal=<%=part.getDorsal()%>">Borrar</a></td>  </tr>  <% }//if distinto de null  }//for  %>  </table>  </body> |

20 por ultimo en /webinf/web.xml cambiamos el wellcome file y ponemos dentro ParticipanteController

21 creamos archivo insertar.jsp para que se cargue cuando le damos a añadir registro con un formulario con todas las variables en la entidad, en action lo enviamos a registrarParticipante

|  |
| --- |
| <body>  <h1>Alta participante</h1>  <form name="formulario" method="POST" action="RegistrarParticipante">  Dorsal: <input type="text" name="dorsal" value="" /><br>  Nombre:<input type="text" name="nombre" value="" /><br>  Apellidos:<input type="text" name="apellidos" value="" /><br>  Poblacion: <input type="text" name="poblacion" value="" /><br>  Club: <input type="text" name="club" value="" /><br>  <input type="submit" value="Guardar" name="guardar" />  </form>  </body> |

22 Creo el servlet en controller que se llame RegistrarParticipante -> next -> add information

- quito todo lo del try de registrarParticipante

- recojo todo lo del form

|  |
| --- |
| int dorsal = Integer.parseInt(request.getParameter("dorsal"));  String nombre = request.getParameter("nombre");  String apellidos = request.getParameter("apellidos");  String poblacion = request.getParameter("poblacion");  String club = request.getParameter("club");  Participante part = new Participante(dorsal, nombre, apellidos, poblacion, club);  ParticipanteDAO.guardar(part);  response.sendRedirect("exito.jsp"); |

23 creamos exito.jsp

24 Volvemos al Index.jsp vamos a hacer el EditarParticipante

|  |
| --- |
| <a href="EditarParticipante?dorsal=<%=part.getDorsal()%>">Editar</a> |

25 Creamos el controller (servlet) EditarParticipante -> add information

* borramos todo y recogemos al participante

|  |
| --- |
| protected void processRequest(HttpServletRequest request, HttpServletResponse response)  throws ServletException, IOException {  int dorsal=Integer.parseInt(request.getParameter("dorsal"));  Participante part= ParticipanteDAO.getParticipanteByDorsal(dorsal); //Aquí hemos guardado el javabean  //gravem un javabean  request.setAttribute("participante", part);  request.getRequestDispatcher("/editar.jsp").forward(request, response);  } |

26 Creamos el editar.jsp -> le ponemos el mismo formulario de insertar -> UseBean (Id: participante (da igual))

Class: Entidad.Participante (igual que lo tenemos)

Scope: request

GetbeanProperty

Bean name: participante

Property Name: dorsal

y así con los demás

|  |
| --- |
| **<jsp:useBean id="participante" scope="request" class="Entidad.Participante" />**  <%@page contentType="text/html" pageEncoding="UTF-8"%>  <!DOCTYPE html>  <html>  <head>  <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">  <title>JSP Page</title>  </head>  <body>  **<h1>Editar participante</h1>**  **<form name="formulario" method="POST" action="RegistrarParticipante">**  **Dorsal: <input disabled type="text" name="dorsal" value="<jsp:getProperty name="participante" property="dorsal" />" /><br>**  **Nombre:<input type="text" name="nombre" value="<jsp:getProperty name="participante" property="nombre" />" /><br>**  **Apellidos:<input type="text" name="apellidos" value="<jsp:getProperty name="participante" property="apellidos" />" /><br>**  **Poblacion: <input type="text" name="poblacion" value="<jsp:getProperty name="participante" property="poblacion" />" /><br>**  **Club: <input type="text" name="club" value="<jsp:getProperty name="participante" property="club" />" /><br>**  **<input type="submit" value="Guardar" name="guardar" />**  **</form>**  </body>  </html> |

27 Creamos en el participanteDAO el método borrar (ya lo tenía desde el principio)

28 Ahora ponemos el link borrar en index.jsp

|  |
| --- |
| <a href="BorrarParticipante?dorsal=<%=part.getDorsal()%>">Borrar</a> |

29 creamos controlador servlet BorrarParticipante en Controller -> add information

Borro todo lo del try y pongo

|  |
| --- |
| protected void processRequest(HttpServletRequest request, HttpServletResponse response)  throws ServletException, IOException {  int dorsal=Integer.parseInt(request.getParameter("dorsal"));  Participante part= ParticipanteDAO.getParticipanteByDorsal(dorsal);  ParticipanteDAO.borrar(part);  response.sendRedirect("borrar.jsp");  } |

30 Creo borrar.jsp

|  |
| --- |
| <h1>Registro borrado satisfactoriamente</h1> |