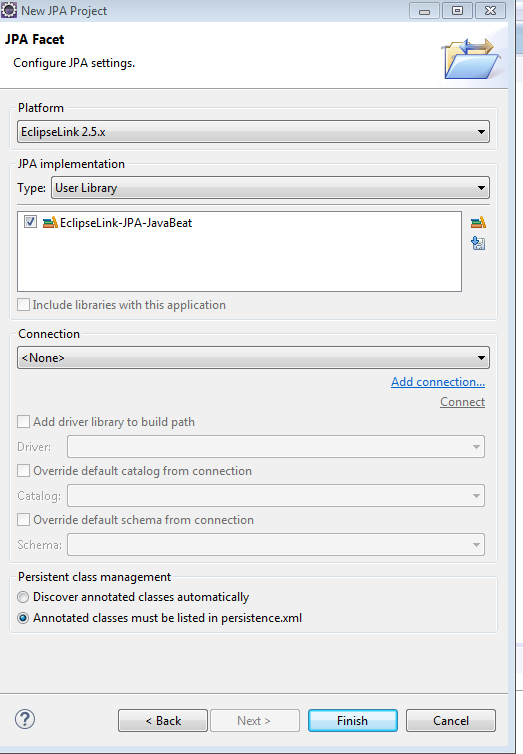
**File new JPA Project**, ponudio mi moj user library, kreiran po uputstvima sa <http://javabeat.net/eclipselink-jpa-installation-configuration/>



https://www.eclipse.org/eclipselink/downloads/ link. That contains the implementation for all persistence services that EclipseLink provides.

The JAR files that should be used for JPA persistence service are:

* eclipselink.jar
* javax.persistence.source\_2.1.0.v201304241213.jar
* javax.persistence\_2.1.0.v201304241213.jar

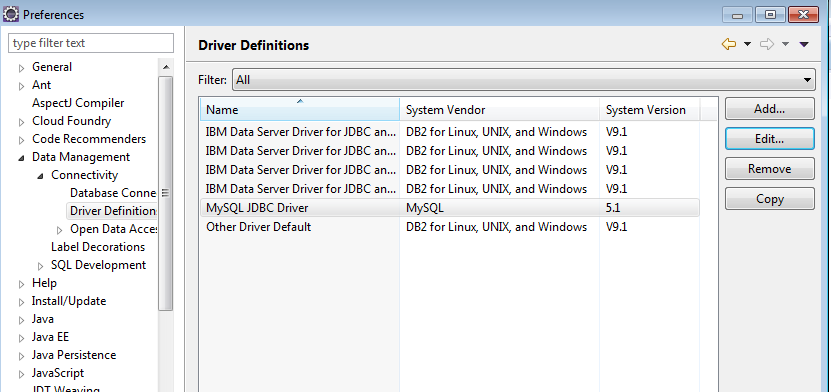
**Kreiranje baze**

create database **jpastore**

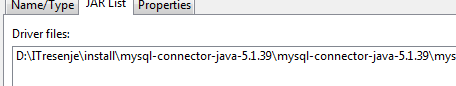
**1. Select Window > Preferences from the main menu bar.**

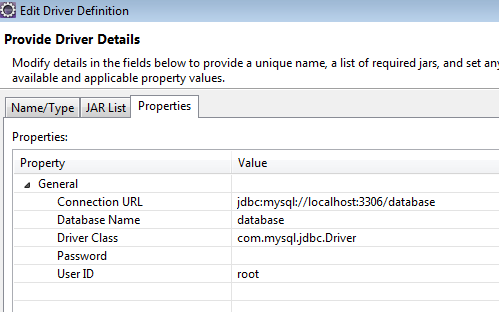
**2. Expand Data Management > Connectivity and select Driver Definitions.**

**ja tu vec imam moj drajver MySQL JDBC Driver**

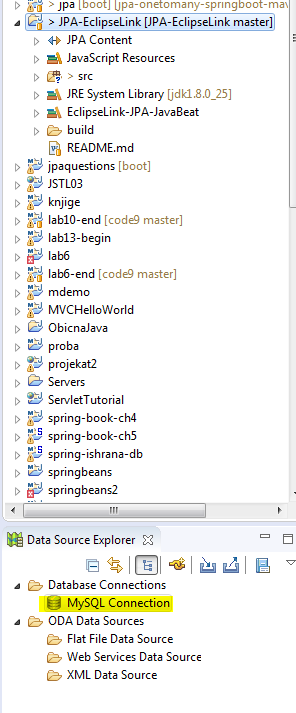
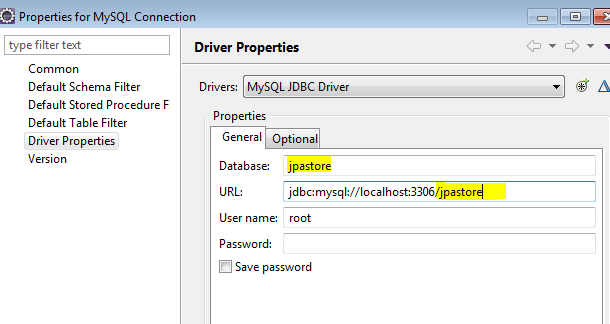


sa podesavanjima:





Kod kreiranja JPA projekta otvorila mi se ova perspektiva sa prozorom DataSourceExplorer:

Testiraj konekciju klikom na test connection

**persistance.xml**

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<persistence version=*"2.1"* xmlns=*"http://xmlns.jcp.org/xml/ns/persistence"* xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"* xsi:schemaLocation=*"http://xmlns.jcp.org/xml/ns/persistence http://xmlns.jcp.org/xml/ns/persistence/persistence\_2\_1.xsd"*>

<persistence-unit name=*"JPA-EclipseLink"* transaction-type=*"RESOURCE\_LOCAL"*>

<provider>org.eclipse.persistence.jpa.PersistenceProvider</provider>

<exclude-unlisted-classes>false</exclude-unlisted-classes>

<properties>

<property name=*"javax.persistence.jdbc.url"* value=*"jdbc:mysql://localhost:3306/jpastore"*/>

<property name=*"javax.persistence.jdbc.user"* value=*"root"*/>

<property name=*"javax.persistence.jdbc.password"* value=*""*/>

<property name=*"javax.persistence.jdbc.driver"* value=*"com.mysql.jdbc.Driver"*/>

<property name=*"javax.persistence.schema-generation.database.action"* value=*"create"*/>

</properties>

</persistence-unit>

</persistence>

**Test klasa**

package com.jelena.tests;

import javax.persistence.EntityManagerFactory;

import javax.persistence.Persistence;

public class JPAImpl {

public static void main(String[] args) {

EntityManagerFactory factory = Persistence.createEntityManagerFactory("**JPA-EclipseLink**");

System.out.println("Is opened connection :: "+ factory.createEntityManager().isOpen());

}

}

[EL Info]: 2017-06-09 10:41:43.046--ServerSession(942804532)--EclipseLink, version: Eclipse Persistence Services - 2.5.1.v20130918-f2b9fc5

[EL Info]: connection: 2017-06-09 10:41:43.45--ServerSession(942804532)--file:/C:/Users/Win7/workspace/JPA-EclipseLink/build/classes/\_JPA-EclipseLink login successful

[EL Warning]: metamodel: 2017-06-09 10:41:43.475--The collection of metamodel types is empty. Model classes may not have been found during entity search for Java SE and some Java EE container managed persistence units. Please verify that your entity classes are referenced in persistence.xml using either <class> elements or a global <exclude-unlisted-classes>false</exclude-unlisted-classes> element

**Is opened connection :: true**

**JPA Entity**

package com.jelena.business;

import javax.persistence.Column;

import javax.persistence.Entity;

import javax.persistence.GeneratedValue;

import javax.persistence.GenerationType;

import javax.persistence.Id;

import javax.persistence.Table;

@Entity

@Table(name="customers")

public class Customer {

@Id

@GeneratedValue(strategy = GenerationType.AUTO)

@Column(name = "customer\_id")

private Long customerId;

@Column(name = "first\_name")

private String firstName;

@Column(name = "last\_name")

private String lastName;

private String email;

public Long getUserId() {

return customerId;

}

public void setUserId(Long customerId) {

this.customerId = customerId;

}

public String getFirstName() {

return firstName;

}

public void setFirstName(String firstName) {

this.firstName = firstName;

}

public String getLastName() {

return lastName;

}

public void setLastName(String lastName) {

this.lastName = lastName;

}

public String getEmail() {

return email;

}

public void setEmail(String email) {

this.email = email;

}

}

**How to get an entity manager factory**

package com.jelena.data;

import javax.persistence.EntityManagerFactory;

import javax.persistence.Persistence;

public class DBUtil {

private static final EntityManagerFactory emf =

Persistence.createEntityManagerFactory("JPA-EclipseLink");

public static EntityManagerFactory getEmFactory() {

return emf;

}

}

**CustomersServlet (url pattern /customers)**

index.jsp

Show all customers---> doGet(), /customers, action = null

Add new customer---> doGet(), /customers?action=display\_empty\_customer

customer.jsp

Join Now (submit dugme)---> doPost(), /customers, action=add\_customer

CustomersServlet

*action = display\_customers*

difoltna akcija je display\_customers, ako stigne action=null on se pretvara u display\_customers

citaju se svi customers iz baze, postavljaju se u request kao atribut customers, request se forvarduje na index.jsp

*action = display\_empty\_customer*

kreira se prazan customer, postavlja u session kao atribut customer, request se forvarduje na customer.jsp koji prikazuje formu za unos podataka o customer-u.

*action = add\_customer*

dolazi klikom na submit dugme forme customer.jsp, podaci o customer-u se citaju iz requesta preko request.getParameter, tim podacima se puni novi objekat customer koji se upisuje u bazu ukoliko prodje validaciju. Validacija podrazumeva proveru da li email novog customera vec postoji u bazi, ukoliko postoji ne upisuje se u bazu i salje se poruka o tome. U suprotnom se upisuje u bazu i salje se poruka zahvalnosti. Potom se vrsi citanje baze i lista customera se salje kao request atribut ka index.jsp.