**实验2**

/\*创建学生类\*/

@Entity

public class Student extends Model{

@Required

public String nameString;

@Required

public String xuehao;

@Required

public double feiyong;

public Student(String nameString, String xuehao, double feiyong) {

this.nameString = nameString;

this.xuehao = xuehao;

this.feiyong = feiyong;

}

@Override

public String toString() {

return nameString;

}

}

/\*创建本科生类\*/

@Entity

public class BKStudent extends Student {

public BKStudent(StringnameString, String xuehao, double feiyong){

super(nameString, xuehao, feiyong);

}

}

/\*创建研究生类\*/

@Entity

public class YJStudent extends Student {

public YJStudent(String nameString, String xuehao,double feiyong) {

super(nameString, xuehao, feiyong);

}

}

/\*创建管理员类\*/

@Entity

public class User extends Model{

@Required

public String password;

public String fullname;

public boolean isAdmin;

public User(String password, String fullname, boolean isAdmin) {

super();

this.password = password;

this.fullname = fullname;

this.isAdmin = isAdmin;

}

public static User connect(String fullname, String password) {

return *find*("byFullnameAndPassword", fullname, password).first();

}

@Override

public String toString() {

return "isAdmin";

}

}

/\*把User类添加到CRUD模块上\*/

package controllers;

public class Users extends CRUD{

}

/\*把BKStudent类添加到CRUD模块上\*/

package controllers;

public class BKStudents extends CRUD{

}

/\*把YJStudent类添加到CRUD模块上\*/

package controllers;

public class YJStudents extends CRUD {

}

**实验3:查看学生信息**

1. 利用CRUD将本科生所有信息添加到数据库中去。
2. 在Controllers相应目录下写一个相应的学生信息控制类

public class GLbk extends Controller {

public static void index()

{

java.util.List<JPABase> BKStudents = BKStudent.*findAll*();

*render*("Student/BKStudent.html",BKStudents);

}

public static void info(Long id) {

BKStudent bk = BKStudent.*findById*(id);

*render*("Student/bkinfo.html",bk);

}

public static void update(Long id,String name,String xuehao,float feiyong) {

BKStudent bk = BKStudent.*findById*(id);

bk.nameString = name;

bk.xuehao=xuehao;

bk.feiyong=feiyong;

bk.save();

*render*("Student/bkinfo.html",bk);

}

public static void delete(Long id) {

BKStudent bk = BKStudent.*findById*(id);

bk.delete();

*index*();

}

}

1. 在Views下写一个相应的界面

<div id="login">

<center>

<h1>&{'欢迎进入本科生管理系统'}</h1>

#{form @GLbk.update()}

<p id="username-field">

<label for="id">&{'id号：'}</label>

<input type="text" name="id" id="username" value="${bk.\_key().toString()}" />

</p>

<p id="username-field">

<label for="name">&{'姓名：'}</label>

<input type="text" name="name" id="username" value="${bk.nameString}" />

</p>

<p id="username-field">

<label for="feiyong">&{'费用：'}</label>

<input type="text" name="feiyong" id="password" value="${bk.feiyong}" />

</p>

<p id="username-field">

<label for="xuehao">&{'学号：'}</label>

<input type="text" name="xuehao" id="password" value="${bk.xuehao}" />

</p>

<p id="signin-field">

<input type="submit" id="signin" value="&{'更改'}" />

</p>

#{/form}

</div>

/\*自动生成学号\*/

package controllers;

import java.util.List;

import models.BKStudent;

import play.db.jpa.JPABase;

import play.mvc.Controller;

public class Addstudent extends Controller{

public static void add(String name,double feiyong) {

if(name != null && feiyong != 0) {

BKStudent bk =BKStudent.find("order by xuehao desc").first();

Integer xuehao = Integer.valueOf(bk.xuehao)+1;

String nextXueHao = xuehao.toString();

BKStudent bk1 =new BKStudent(name,nextXueHao,feiyong);

bk1.save();

render("Student/AddStudent.html",bk1);

}

render("Student/AddStudent.html",null);

}

}

**实验4:Secure模块**

package controllers;

import models.User;

public class Security extends Secure.Security{

public static boolean authenticate(String fullname,String password) {

return User.*connect*(fullname, password)!=null;

}

public static boolean check(String profile){

if(Security.*isConnected*()){

String user = Security.*connected*();

boolean rst = false;

if(profile.equals("studentNameModify") && user.equals("stu"))

rst = true;

return rst;

}

else{

return false;

}

}

}