1. Css 中 Display属性

<html>

<head>

<style type="text/css">

span

{

display: block

}

</style>

</head>

<body>

<span>本例中的样式表把 span 元素设置为块级元素。</span>

<span>两个 span 元素之间产生了一个换行行为。</span>

</body>

</html>

二．z-index

<html>

<head>

<style type="text/css">

img

{

position:absolute;

left:0px;

top:0px;

z-index:-1;

}

</style>

</head>

<body>

<h1>This is a heading</h1>

<img src="/i/eg\_smile.gif" />

<p>由于图像的 z-index 是 -1，因此它在文本的后面出现。</p>

</body>

</html>

文本在图片上边显示。好看 !

三 伪类选择器。

|  |  |  |
| --- | --- | --- |
| **属性** | **描述** | **CSS** |
| [:first-letter](http://www.w3school.com.cn/cssref/pr_pseudo_first-letter.asp) | 向文本的第一个字母添加特殊样式。 | 1 |
| [:first-line](http://www.w3school.com.cn/cssref/pr_pseudo_first-line.asp) | 向文本的首行添加特殊样式。 | 1 |
| [:before](http://www.w3school.com.cn/cssref/pr_pseudo_before.asp) | 在元素之前添加内容。 | 2 |
| [:after](http://www.w3school.com.cn/cssref/pr_pseudo_after.asp) | 在元素之后添加内容。 | 2 |

<html>

<head>

<style type="text/css">

p:first-letter

{

color: #ff0000;

font-size:xx-large

}

</style>

</head>

<body>

<p>

shutdown! //首字母大写！

</p>

</body>

</html>

四入门javascript

1. console.log(alert('hi'));
2. toUpperCase 大写toLowerCase 小写
3. **var** s = 'Hello';
4. s.toUpperCase(); *// 返回'HELLO'*
5. indexOf()
   1. 来搜索一个指定的元素的位置，注意了 !！

var arr=[1,2,3];

console.log(arr.indexOf(2));

1

console.log(arr.indexOf(3));

2

1. **var** arr = [10, 20, '30', 'xyz'];
2. arr.indexOf(10); *// 元素10的索引为0*
3. arr.indexOf(20); *// 元素20的索引为1*
4. arr.indexOf(30); *// 元素30没有找到，返回-1*
5. arr.indexOf('30'); *// 元素'30'的索引为2*
6. 注意了，数字30 和字符串'30'是不同的元素。
   1. ．如果找到一个 searchvalue，则返回 searchvalue 的第一次出现的位置。stringObject 中的字符位置是从 0 开始的。Such as：

<script type="text/javascript">

var str="Hello world!"

document.write(str.indexOf("Hello") + "<br />")

document.write(str.indexOf("World") + "<br />")

document.write(str.indexOf("world"))

</script>

以上代码的输出：

0

-1

6

4, Push pop、

unshift和shift

* 1. **push()向Array的末尾添加若干元素，pop()则把Array的最后一个元素删除掉：**

**var arr[1,2,3];**

**arr.push(‘A’，‘B’);**

**arr；//此时是【1，2，3，A,B】**

**arr.pop();**

**arr;//此时是【1，2，3，A】**

* 1. **Array的头部添加若干元素，使用unshift()方法，shift()方法则把Array的第一个元素删掉：**

5.sort（） reverse（）

arr.sort();//重新进行排序

arr.reserve()//反转

6， $('#test-css li.dy>span').css('background-color', '#ffd351').css('color', 'red');//这句话是id选择器test-css下边的li里边的span中带有dy类的选择器中的

<ul id="test-css">

<li class="lang dy"><span>JavaScript</span></li>

<li class="lang"><span>Java</span></li>

<li class="lang dy"><span>Python</span></li>

<li class="lang"><span>Swift</span></li>

<li class="lang dy"><span>Scheme</span></li>

</ul>

输出结果是:

* JavaScript
* Java
* Python
* Swift
* Scheme

1. <input type="button" name="button1" value="please"onClick="window.open('2\_0.html','newWin')">

这是一个按钮的程序  
意识是说，做了一个按钮,定义名称为button1，按钮上面显示的文字是“please"，点击这个按钮之后，从新窗口打开网页"2\_0.html"  
  
<input type=“button”……>指的是类型为按钮  
name是你自己定义的按钮在你程序里的名字  
value是你按钮上默认显示的文字信息  
onClick的意识是点击后出现的内容  
window.open('2\_0.html','newWin')"的意识是从新窗口打开某网页，这里你打开的网页是2\_0.html

1. alert（‘‘）；

var name='frozen';//定义变量名字得用单引号

var age=19;

alert('hi,'+name+','+age+'岁了！');

五 1.Springmvc： 请求（需要在web.xml里边配置）→Dispatcherservlet

→处理器映射 handlermapping （添加springmvc的配置文件，判断去哪一个controller）是一个映射

→controller控制器

→ModelAndView 模型和视图

→Viewresolver视图解析器

2.Dao（持久层）是和数据库打交道的文件，在src-main-java-lab.s2jh-module-sys-

3.servlet:将前端页面和后端java结合起来

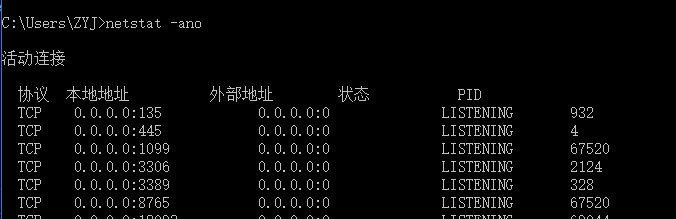
将请求的url映射到java类的一个方法上；往回传；控制页面的跳转；

今天（2017/7/2）谷歌就访问不了了 -----cn.bing.com

Springmvc.xml文件 先长按alt +Enter（取消注释这个东西）

（2017/7/6）今天遇见Address localhost:1099 is already in use 的错误：

第一步，win+R命令提示符号，[执行](http://www.haogongju.net/tag/%E6%89%A7%E8%A1%8C)命令：[n](http://www.haogongju.net/tag/net)etstat –ano

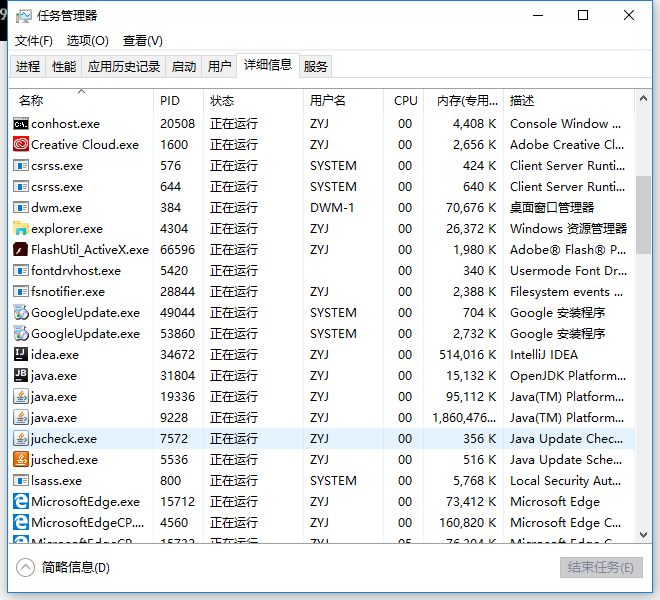


可见，占用1099端口的进程的PID是67520。

第二步，命令提示符号，执行命令：tasklist



可见，该占用8080端口的进程是java.exe  
第三步，通过任务管理器，终止进程java.exe



我们会看到开启了2个java.exe,统统都给它结束掉。

第四步，重新启动tomcat，即可正常启动

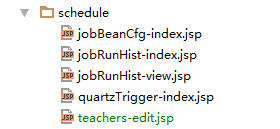
（2017/7/ 7）today分配了一个controller，弹出一个页面。

< 详情参照[http://www.cnblogs.7com/lonelyxmas/p/5397428.html](http://www.cnblogs.com/lonelyxmas/p/5397428.html)>

1.自己先创建一个controller，在src-main-java-support-web，（如下图）右击new-javaclass，自己起一个名字！！——没错我的就叫MainController

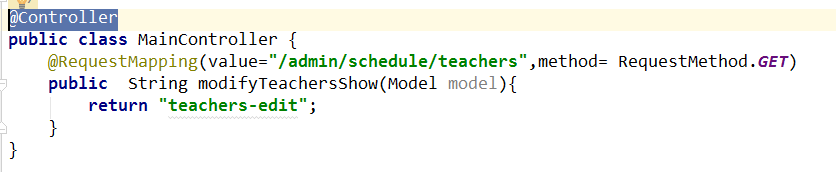
在MainController里边①@Controller

* 1. 根据网页显示可以知道在哪所以，右击new→teachers-edit.jsp

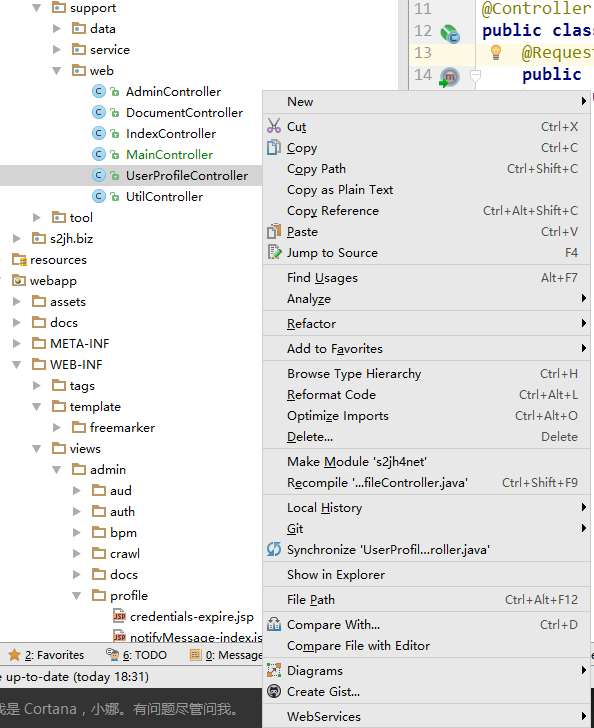


* 1. 自己写一个方法

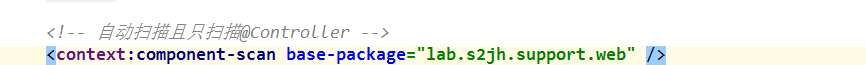
这儿是你写的那个jsp的文件所在位置(注意这个可以填一半名字的)



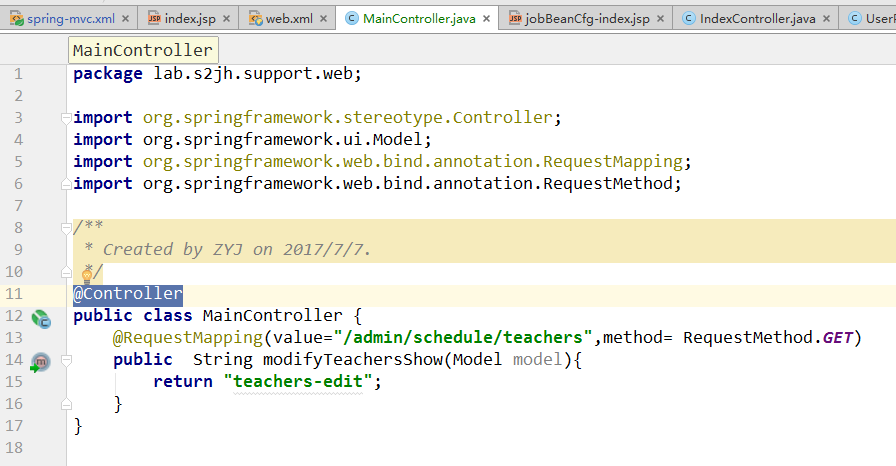
这个方法名字自己起的



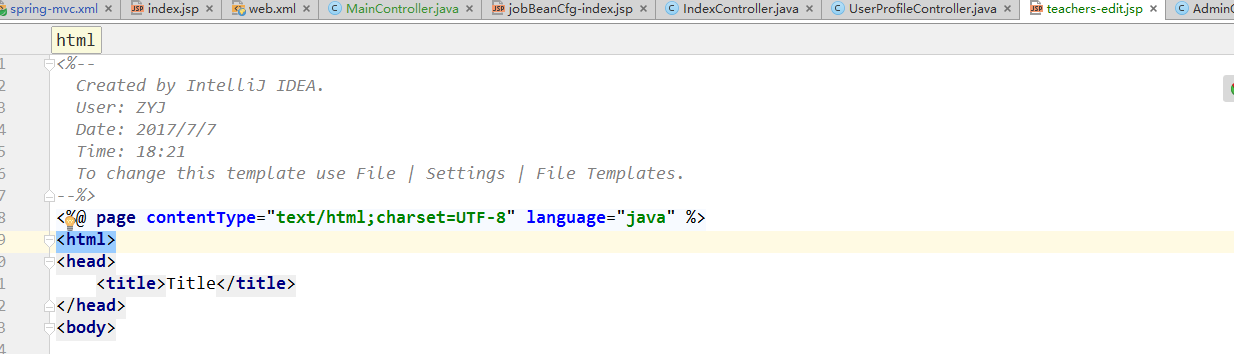
2.在springmvc-xml里边添加，在<beans></beans>之间写



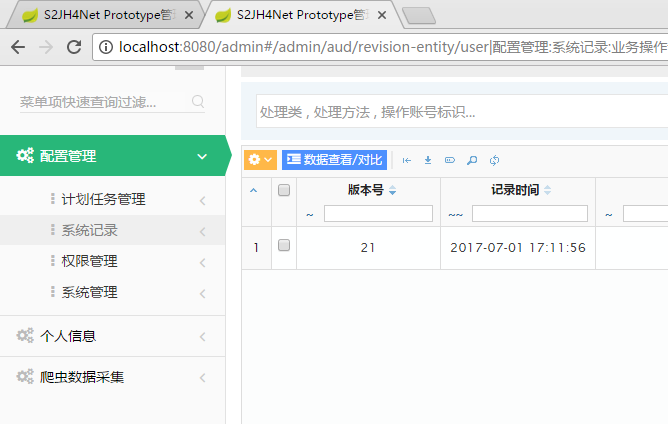
这里是你所添加的controller即MainController里边第一句



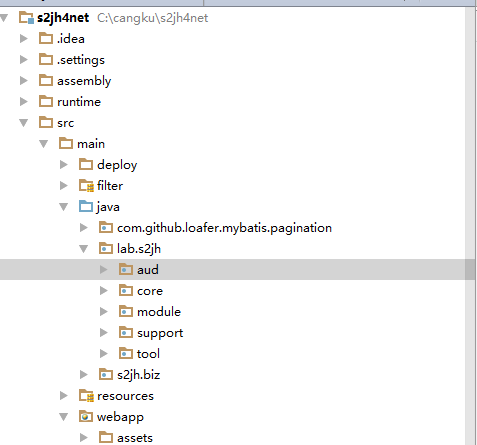
1. 然后就可以在你所表达的网页jsp进行描述了



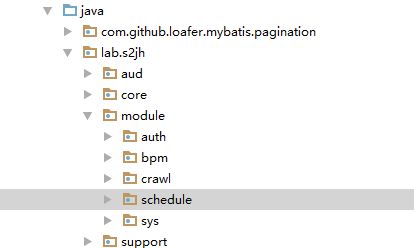
（2017/7/8）今天恍惚之间貌似发现什么:



点开其中一小节，你可以观察他的 url，（such as <http://localhost:8080/admin#/admin/sys/config-property|配置管理:系统管理:参数配置>） 你会发现其实这4个分别对应的在（src-main-java-lab.s2jh）里边的schedule；aud；auth；sys



（注意，aud单独一个文件夹而其他在module里边组件里边）



那么 ，接下来要干一件大事情了（/发呆）就是将昨天自己分配的那个controller弄到左侧，令其产生一个栏目。对没错！



其实只要在你建的那个Controller里边添加、



其中必须是英文的 冒号这是一个三级菜单，。。。结果如上所示。

则点开java-lab.s2jh-module-sys-web，里边很多controller

(2017/7/9) 今天呀 要开始话页面、

1.datagrid比较常 用也比较复杂，它有其中有load,reload,loadData这三个方法，它们都有相同的功能，都是加载数据的。

* 1. load方法，比如我已经定义一个datagrid的id为grid，那这个方法的使用方式为

$(“#grid”).datagrid(“load”,{ });

它的参数为一个json对象，里面写的是你要传输的参数的键值对，调用这个方法来加载数据的时候，它传给后台的分页信息是从第一页开始的。

* 1. reload方法

$(“#grid”).datagrid(“reload”,{ });



但它传给后台的分布信息是当前的页码，就是实现刷新当前页的功能

③ loadData方法，这个方法有比较大的不同，首先它加载的本地数据，就是不会跟后台什么的有交互，它的使用方式是

$(“#grid”).datagrid(“loadData”,{ “total”:”30″,rows:[] });

看到没，它是比较灵活的，可以设置分页信息的，total就是行数嘛，当然你也可以设置它的当前页，而它那个rows属性设的就是你所要加载的行的集合。

2.colModel设置表格的 属性

<http://blog.csdn.net/wpcxyking/article/details/5909283>

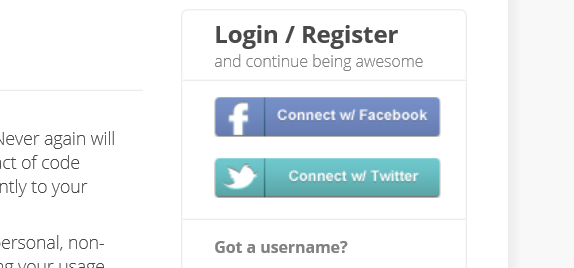
3. [java enum的用法详解](http://blog.csdn.net/liaoqianwen123/article/details/23561935)，列举

<http://blog.csdn.net/yechaodechuntian/article/details/23561935>

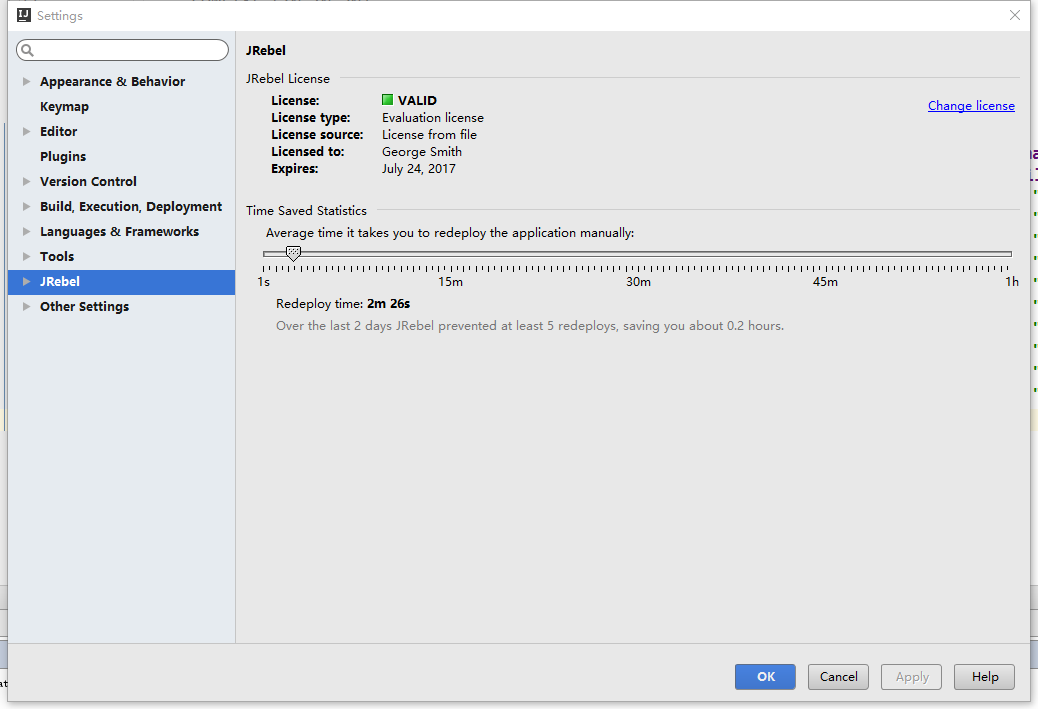
（2017/7/10）日子过得还真快，要问我想不想回家，说实话也想也不想，额

安装插件JRebel

1. setting→查找jrebel→选择for intellij→install→去my jrebel官网<https://my.jrebel.com/>
2. 会找到这个→点击 connect w/ Twitter→注册→提交后→就会出现注册码→



1. 3.可以点击提醒的 activation（激活）→早已有注册码→成功了→license是valid



4.成功了！

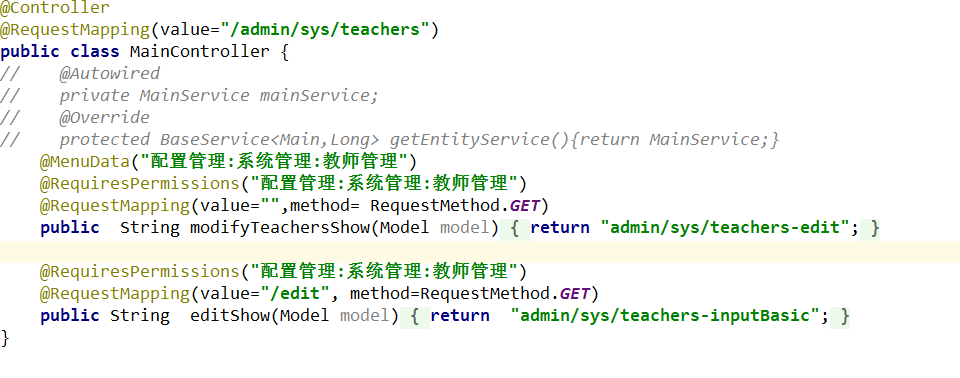
<http://manuals.zeroturnaround.com/jrebel/ide/intenllij.html>

comodel

做那个教师管理的静态页。。。（<http://localhost:8080/admin#/admin/sys/config-property|配置管理:系统管理:参数配置> ） 应该以这个网页为主，进行修改。

（2017/7/11）

1.要知道在浏览器中访问的 时候输入的是url ① + ② such as



2.RGBA是代表Red（红色） Green（绿色） Blue（蓝色）和 Alpha的色彩空间。虽然它有的时候被描述为一个颜色空间，但是它其实仅仅是RGB模型的附加了额外的信息。

3. <http://runjs.cn/code/jahijctd>

.do是一种网页后台程序，\*.jsp或者struts的组件文件\*.do ，它不是一个文件,并没有一个真正的.do文件存在。

4. <a href="test.html" target="\_blank"></a>重新开一个窗口

5.注意，前方有兔子出没！！！

/edit必须与controller里边 value值 就是上边说的

<**a href="/admin/sys/teachers/edit"**>添加</**a**>

6.alt+左箭头 在网页里边是后退

Alt+右箭头 是前进

7. http://www.cnblogs.com/liukemng/p/3754211.html

表单标签<form:form>

(2017/7/12)回家倒计时

1. Css里边的margin
2. ：margin:10px 5px 15px 20px;

分别代表： 上 右 下 左

1. ： margin：10px 5px 15px；

分别代表： 上 右左 下

1. ：margin：10px 5px;

分别代表：上下 左右

1. ：margin：5px;

代表全部

1. 可以这样说吧，设置段落 p {margin:0 auto;}。 在IE与FF中，段落P均实现了居中对齐。
2. 而 设置图片标签img {margin:0 auto;} ，就错了，img类于内联对象，不可以设置图片img标签的margin属性，如果一定要设置，那么先将它的属性转变为块元素，如下面的代 码：img {display:block; margin:0 auto;}
3. 注意这儿的display：block；（换行的意思）。

display: inline; 内联元素又名行内[元素](http://baike.baidu.com/item/%E5%85%83%E7%B4%A0)(inline element)，和其对应的是块元素(block element)，内联元素的显示,可以形象的称为“文本模式”,即一个挨着一个，都在同一行按从左至右的顺序显示，不单独占一行。而当加入了css控制以后，块元素和内联元素的这种[属性](http://baike.baidu.com/item/%E5%B1%9E%E6%80%A7)差异就不成为差异了。比如，我们完全可以把内联元素加上display:block这样的属性，让它也有每次都从新行开始的属性，即成为块元素同样我们可以把块元素加上display:inline这样的属性，让它也在一行上排列。

1. （可以看看这个<http://blog.dimpurr.com/css-before-after/>专门说了before::,after::）样式表里边：a:before{color:red;}有时候在。。之前加东西嗯！！没错

<a href=”#”>试试</a>

<style>

p:before{

content: "H" /\*:before和:after必带技能，重要性为满5颗星\*/

}

p:after{

content: "d" /\*:before和:after必带技能，重要性为满5颗星\*/

}

</style>

<p>ello Worl</p>

[复制代码](javascript:void(0);)

以上的代码将会在页面中展现的是"Hello World"。我们通过浏览器的"审查元素"看到的内容是：

<p>

::before

"ello Worl"

::after

</p>

6.spring 表单标签<http://blog.csdn.net/miketom155/article/details/45058179>

7. <http://www.cnblogs.com/ninsh1992/p/6232013.html>

## [SPRING MVC中的MODELMAP作用及用法](http://www.cnblogs.com/ninsh1992/p/6232013.html)

8. 这个东西我会弄出来的



9.具体实现点一个东西弹一个框；

*<a data-toggle="modal" data-target="#addUser">增加用户</a>*

(2017/7/13)传说，在古老的东方，有那么yi

1. jQuery.param()函数用于**将一个JS数组或纯粹的对象序列化为字符串值，以便用于URL查询字符串或AJAX请求**。

如果传入的不是数组或"纯粹的对象"，则返回空字符串("")；如果传入的是null、undefined等无法访问属性的值，则直接报错。

所谓"纯粹的对象"，就是通过{}或new Object()自行创建的对象。JS内置的Boolean、Number、String、Date、RegExp、Function、Window等类型的对象都不算是"纯粹的对象"。

返回的字符串已经过URL编码处理(采用的字符集为UTF-8)。

该函数属于**全局**的jQuery对象。

2. <http://blog.csdn.net/suyu_yuan/article/details/50680024>

<c:if test="${requestScope.username = = 'admin' }">

　　ADMIN您好!! //body部分

　　</c:if>　如果名称等于admin，则会显示"ADMIN您好!! "的动作，如果相反，则不会执行<c:if>的body部分，所以不会显示"ADMIN您好!! //body部分"。

1. 今天算是圆满解决了，这儿得再写的controller里边写一下映射@requestmapping。

<a href="${ctx}/admin/sys/teachers/edit" data-toggle="modal-ajaxify" title="添加用户" data-modal-size="1400px">添加</a>



没错就是这样！

1. <http://www.trirand.com/blog/jqgrid/jqgrid.html>表格 终于弄出来了 (还有一个疑问 ，spring标签such as (form：form)没弄懂)

## 模态框 modal data-toggle data-target

1. 这个模态框高深莫测呀 ，功能还不少

<http://www.cnblogs.com/xiaoyuandxiaobai/articles/6021680.html> modal 主要讲解

（没怎么看懂。。）

2.  //data-toggle

data-toggle="modal" data-target="#myModal"

data-toggle 表示触发类型  
data-target 表示触发的节点

如果不是使用<button>， 而是<a>， 其中 data-target 也可以使用 href="#myModal"

取代。当然，我们建议使用 data-target。除了 data-toggle 和 data-target 两个声明属性外，还有一些可以用选项。

3.     Data-show显示当前模态窗，如果说 你要单独引用该插件你需要引用bootstrap.js,.min.css,.jquery。

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8">

<title>Bootstrap 实例 - 模态框（Modal）插件</title>

<link rel="stylesheet" href="https://cdn.bootcss.com/bootstrap/3.3.7/css/bootstrap.min.css">

<script src="https://cdn.bootcss.com/jquery/2.1.1/jquery.min.js"></script>

<script src="https://cdn.bootcss.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>

</head>

<body>

**<!—点击关闭按钮后出来一个警告 -->**

<script>

$(function () { $('#myModal').on('hide.bs.modal', function () {

alert('嘿，我听说您喜欢模态框...');})

});

</script>

**<!-- 按钮触发模态框 -->**

<button class="btn btn-primary btn-lg" data-toggle="modal" data-target="#myModal">点我</button>

**<!—data-target="#myModal"用于切换指定的模态框（带有id=myModel）-->**

**<!-- 模态框（Modal） fade当模态框被切换时，它会引起内容淡入淡出；-->**

<div class="modal fade" id="myModal" tabindex="-1" role="dialog" aria-labelledby="myModalLabel" aria-hidden="true">**<!-- 模态声明，aria-labelledby="myModalLabel"，该属性引用模态框的标题；属性 aria-hidden="true" 用于保持模态窗口不可见，直到触发器被触发为止（比如点击在相关的按钮上）-->**

<div class="modal-dialog">**<!-- 窗口声明 -->**

<div class="modal-content">**<!-- 内容声明 -->**

**<!—总结一下总共用到了三个div声明 -->**

<div class="modal-header">

<button type="button" class="close" data-dismiss="modal" aria-hidden="true">

&times;

</button>

<h4 class="modal-title" id="myModalLabel">

模态框（Modal）标题（比如：添加用户）

</h4>

</div>

<div class="modal-body">

在这里添加一些文本

</div>

<div class="modal-footer">

<button type="button" class="btn btn-default" data-dismiss="modal">关闭

</button>

<button type="button" class="btn btn-primary">

提交更改

</button></div>

</div><!-- /.modal-content -->

</div><!-- /.modal -->

</div>

</body>

</html>

（2017/7/14）今天，想尝试这个自己弄一个 模拟框出来、

1.首先得回顾一下样式的引用：

（1）行内式

<p style=”color:red”>网页中css的导入方式</p>

（2）嵌入式

<style type=”text/css”>

    P{ color:red }

</style>

**嵌入式一般写在head中，对于单个页面来说，这种方式很方便。**（3）导入式

<!-- 导入外部样式：在内部样式表的<style></style>标记之间导入一个外部样式表，导入时用@import。 -->

<style type="text/css">

 @import "jisuan.css";

</style>

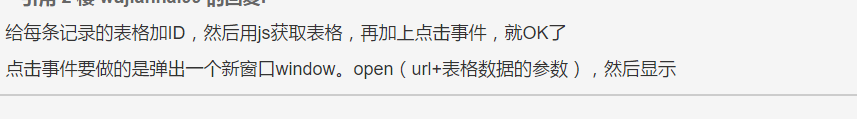
（4）链接式

<link href="jisuan.css" rel=”stylesheet” type=”text/css” />

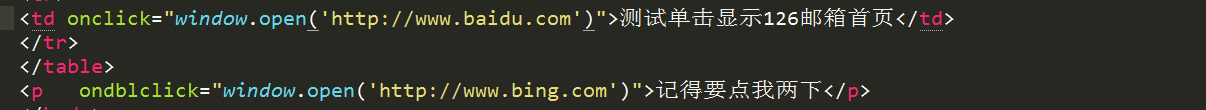
导入式和链接式差不多，都是从外部引入css文件。但是链接式对于客户端用户浏览网站时，效果会好些。

2.引用jquery<script src=" https://code.jquery.com/jquery-3.2.1.js "></script>

3.怎么看都感觉。。。的吧那个spring标签库给弄懂。。



GET 新技能：



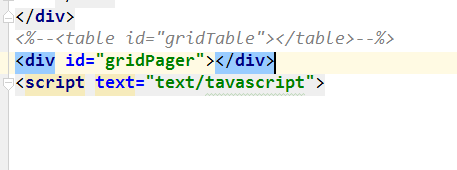
（2017/7/15）colmodal小demo具体可以看一下

<http://www.cnblogs.com/mycoding/archive/2011/07/07/2099878.html>

[http://mj4d.iteye.com/blog/1628851、（](http://mj4d.iteye.com/blog/1628851、（这个) 这个更详细吧）

1. 底下那个 翻页的那个 看见了没

<http://blog.csdn.net/polaris1119/article/details/5089611?locationNum=14&fps=1>具体借鉴这个网址



在这添加一句话<div id=”gridPager”></div>,然后再colmodel里边页面id





2.colmodal详情概述

**multiselect**: **true**,*<!--支持多选-->***caption**: **"教师信息"**,*<!--标题-->***rownumbers**: **true** ,*//是否显示行数！！***rowNum**:10, *<!--每页能看多少条，比如10条翻一页-->***sortname**:**'id'**,  
**sortorder**:**'asc'**,  
**viewrecords**:**true**,*<!--是否显示总记录信息 such as在这个例子中就是总共有几行就写1-12总共12条 -->***rowList**:[10,20,30],*<!--可以改变每页的条数，比如20条翻一页-->***pager**:**"#gridPager"**,  
**pgtext** : **"第{0}页 总页数 {1}"***//页数显示格式*

（2017/7/16）

1. ctrl+alt+L 可将代码整齐化
2. 默认字体大小是16px，则该样式1em=16px，0.5em=8px。
3. 设置css3圆角

只需要一个属性border-radius：5px；可以看一下底下这个网页里边说的圆角：

<http://www.ruanyifeng.com/blog/2010/12/detailed_explanation_of_css3_rounded_corners.html>

1. 20：01，此时此刻 前面做的那些，可以说是。。重新推翻了 ！

只需要全部 粘贴拿过来用就行了。。哎 只需要改三个地方！！ ，具体要和你的controller一样！！！



1. Ctrl+shift+/ 多行注释

Ctrl+/ 单行注释

（2017/7/17）

1. **subGrid** : **true**,*//首先必须将jqGrid的subGrid选项设置为true，默认为false；当此项设为true的时候，Grid表格的最左边将会添加一列，里面有一个“+”图标，用于展开子格；*subGridRowExpanded : **function**(subgrid\_id, row\_id) {  
 ***Grid***.initSubGrid(subgrid\_id, row\_id, {  
 **url** : ***WEB\_ROOT*** + **"/admin/sys/notify-message/read-list?search['EQ\_notifyMessage.id']="** + row\_id,  
 **colNames** : [ **'阅读用户'**, **'首次阅读时间'**, **'最后阅读时间'**, **'总计阅读次数'** ],  
 **colModel** : [ {  
 **name** : **'readUserLabel'**,  
 **width** : 150  
 }, {  
 **name** : **'firstReadTime'**,  
 **formatter** : **'timestamp'** }, {  
 **name** : **'lastReadTime'**,  
 **formatter** : **'timestamp'** }, {  
 **name** : **'readTotalCount'**,  
 **align** : **'center'** } ]  
 });  
 }*//开启子列表详情参照*[*http://blog.sina.com.cn/s/blog\_573a052b0100q7tf.html*](http://blog.sina.com.cn/s/blog_573a052b0100q7tf.html)

2. nullable:true；充许数据为空，也就是说这一列可以放数据，也可以不放。  
没有这个标志的列，则必须有数据才行，要不然数据保存不了

3.现在是有了一个思路：先entity-dao-service（controller也要改）

(2017/7/18)

1. API就是我们设计程序用的接口和方法，API文档就是API接口的使用说明；接口是一种特殊的类。

<http://blog.csdn.net/wenzhi20102321/article/details/52431519>（可以看看这个网址）

1. java里边有一个注解@Accessors(chain = **true**)

chain如果为真，产生的setter返回的this而不是void。默认是假。如果fluent=true，那么chain默认为真。

1. 缓存可以简单的看成一个 Map ，通过 key 在缓存里面找 value 。

@Cache(usage = CacheConcurrencyStrategy.***READ\_WRITE***)

读写模式 在更新缓存的时候会把缓存里面的数据换成一个锁，其它事务如果去取相应的缓存数据，发现被锁了，直接就去[**数据库**](http://lib.csdn.net/base/mysql)查询；

1. 今天看了一下这个网页 。。和数据库一样

<https://www.ibm.com/developerworks/cn/java/j-lo-jpasimpemap/>

1. ① @Access(AccessType.***FIELD***)

--> 直接访问Entity的变量，可以不定义getter和setter方法，但是需要将变量定义为public；

--> 需要在变量上定义字段的属性；

1. ② @Access(AccessType.**PROPERTY**)

--> 通过getter和setter方法访问Entity的变量，可以把变量定义为private；  
     --> 需要在getter方法上定义字段的属性；

7.有一句话：用类去引用方法

8. @Audited hibernate 自动审计，记录历史表，标记该实体类或属性支持数据修改记录。

9.@GeneratedValue具体参照<http://blog.csdn.net/canot/article/details/51455967>

### hibernate主键策略生成器

hibernate提供多种主键生成策略，有点是类似于JPA，有的是hibernate特有，下面列出几个Hibernate比较常用的生成策略:

* native: 对于 oracle 采用 Sequence 方式，对于MySQL 和 SQL Server 采用identity（自增主键生成机制），native就是将主键的生成工作交由数据库完成，hibernate不管
* uuid: 采用128位的uuid算法生成主键，uuid被编码为一个32位16进制数字的字符串。占用空间大（字符串类型）。
* assigned: 在插入数据的时候主键由程序处理（即程序员手动指定），这是 <generator>元素没有指定时的默认生成策略。等同于JPA中的AUTO。
* identity: 使用SQL Server 和 MySQL 的自增字段，这个方法不能放到 Oracle 中，Oracle 不支持自增字段，要设定sequence（MySQL 和 SQL Server 中很常用）。 等同于JPA中的INDENTITY。
* increment: 插入数据的时候hibernate会给主键添加一个自增的主键，但是一个hibernate实例就维护一个计数器，所以在多个实例运行的时候不能使用这个方法

### @GenericGenerator

自定义主键生成策略，由@GenericGenerator实现。   
hibernate在JPA的基础上进行了扩展，可以用一下方式引入hibernate独有的主键生成策略，就是通过@GenericGenerator加入的。

比如说，JPA标准用法

@Id

@GeneratedValue(GenerationType.AUTO)

就可以用hibernate特有以下用法来代替:

@Id

@GeneratedValue(generator = "paymentableGenerator")

@GenericGenerator(name = "paymentableGenerator", strategy = "assigned"

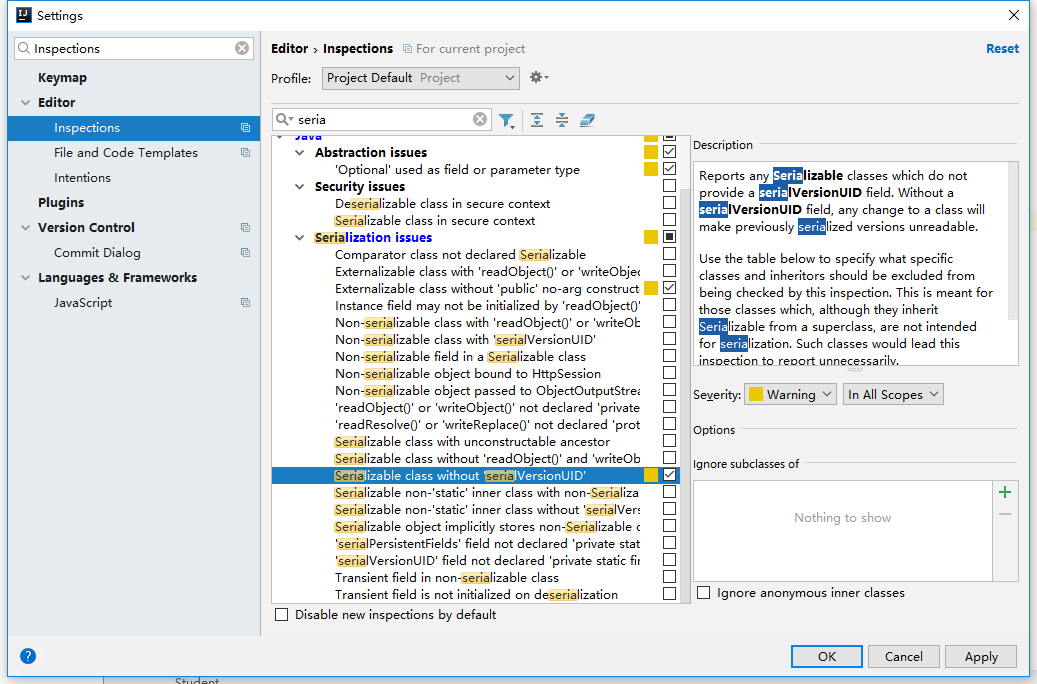
1. **private static final long *serialVersionUID*** = 4034691676061136485L;话说这句话是什么意思。。。

啊哈哈哈，这是Java里面[新建](https://www.baidu.com/s?wd=%E6%96%B0%E5%BB%BA&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1YkPhc3nvf3n1DdrHN-Pj7-0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EnW0dn1RzP1fkn1Rsnj01rHRYPs)一个类会默认添加的一个字段，表示这个类的版本及[ID](https://www.baidu.com/s?wd=ID&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1YkPhc3nvf3n1DdrHN-Pj7-0ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EnW0dn1RzP1fkn1Rsnj01rHRYPs)信息，这个码是唯一的，建议不要删除，虽然没有什么实际用处。（咳咳，具体可以参照<http://blog.csdn.net/liuzongl2012/article/details/45168585>）

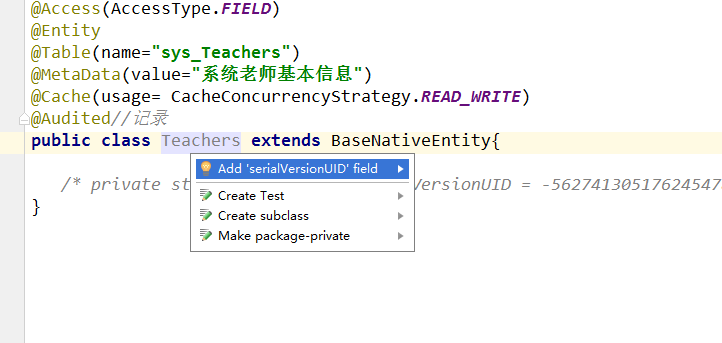
那么，话又说回来了这个码怎么出来的，

首先①点击File->Setting->Plugins->Browse Repositories，然后搜索GenerateSerialVersionUID的插件，下载、安装后关闭IDEA，然后再打开项目。

Ⅰ、File->setting->Inspections->Serialization issues，将其展开后将serialzable class without "serialVersionUID"打上勾；（默认是关闭的）



Ⅱ、将光标放到类名上，按alt＋enter键，就会提示生成serialVersionUID了。看图好了。。



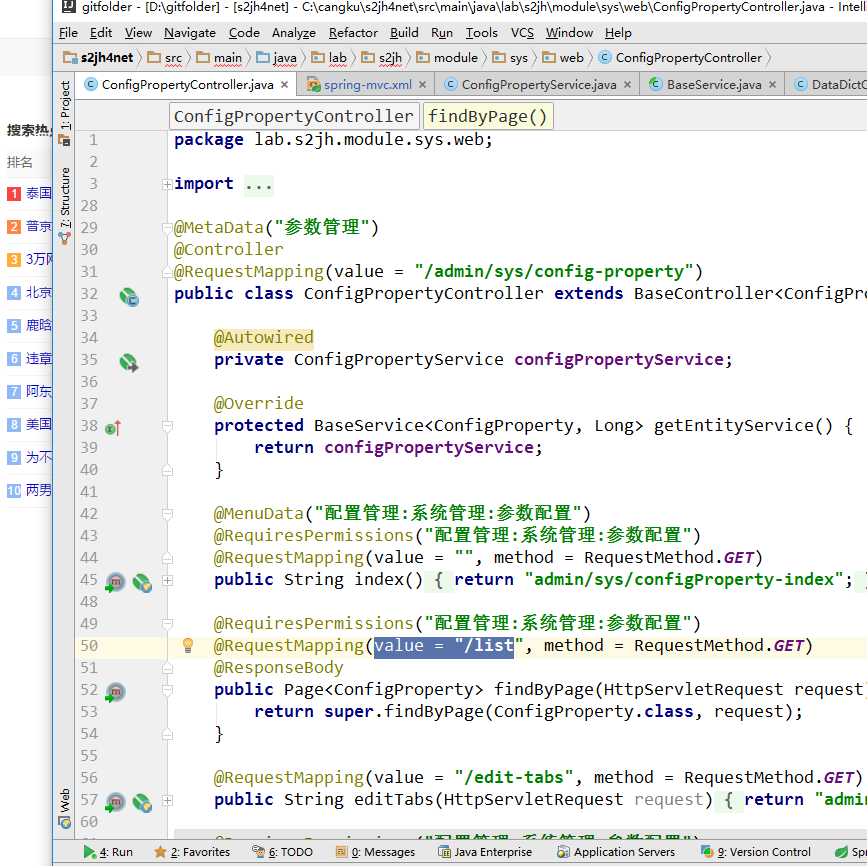
（2017/7/19）**啦啦啦today，which is the last day for staying at the university, there is no volatility in my heart. Volatility**

1. <http://www.jianshu.com/p/633d83dd303b>

@JsonView(JsonViews.List.**class**)

<http://dev.dafan.info/detail/214263?p=68>（这个网址我感觉不错！）

问题：value=“/list”是什么意思



# [Java中@Override的作用](http://blog.csdn.net/zht666/article/details/7869383)

@Override是伪、代码,表示重写(当然不写也可以)，不过写上有如下好处:   
1、可以当注释用,方便阅读；  
2、编译器可以给你验证@Override下面的方法名是否是你父类中所有的，如果没有则报错。例如，你如果没写@Override，而你下面的方法名又写错了，这时你的编译器是可以编译通过的，因为编译器以为这个方法是你的子类中自己增加的方法。  
   
举例：在重写父类的onCreate时，在方法前面加上@Override 系统可以帮你检查方法的正确性。  
@Override  
public void onCreate(Bundle savedInstanceState)

{…….}  
这种写法是正确的，如果你写成：

@Override  
public void oncreate(Bundle savedInstanceState)  
{…….}  
编译器会报如下错误：The method oncreate(Bundle) of type HelloWorld must override or implement a supertype method，以确保你正确重写（人家本来c是大写）onCreate方法（因为oncreate应该为onCreate）。而如果你不加@Override，则编译器将不会检测出错误，而是会认为你为子类定义了一个新方法：oncreate

六today（2017/7/5/18:25）

**JSP 标准标签库（JSTL）**

1.使用时需要在在JSP页面中声明标记库:  
<%@ taglib prefix="c" uri="8r<http://java.sun.com/jstl/core>" %>

Mybatis-config.xml

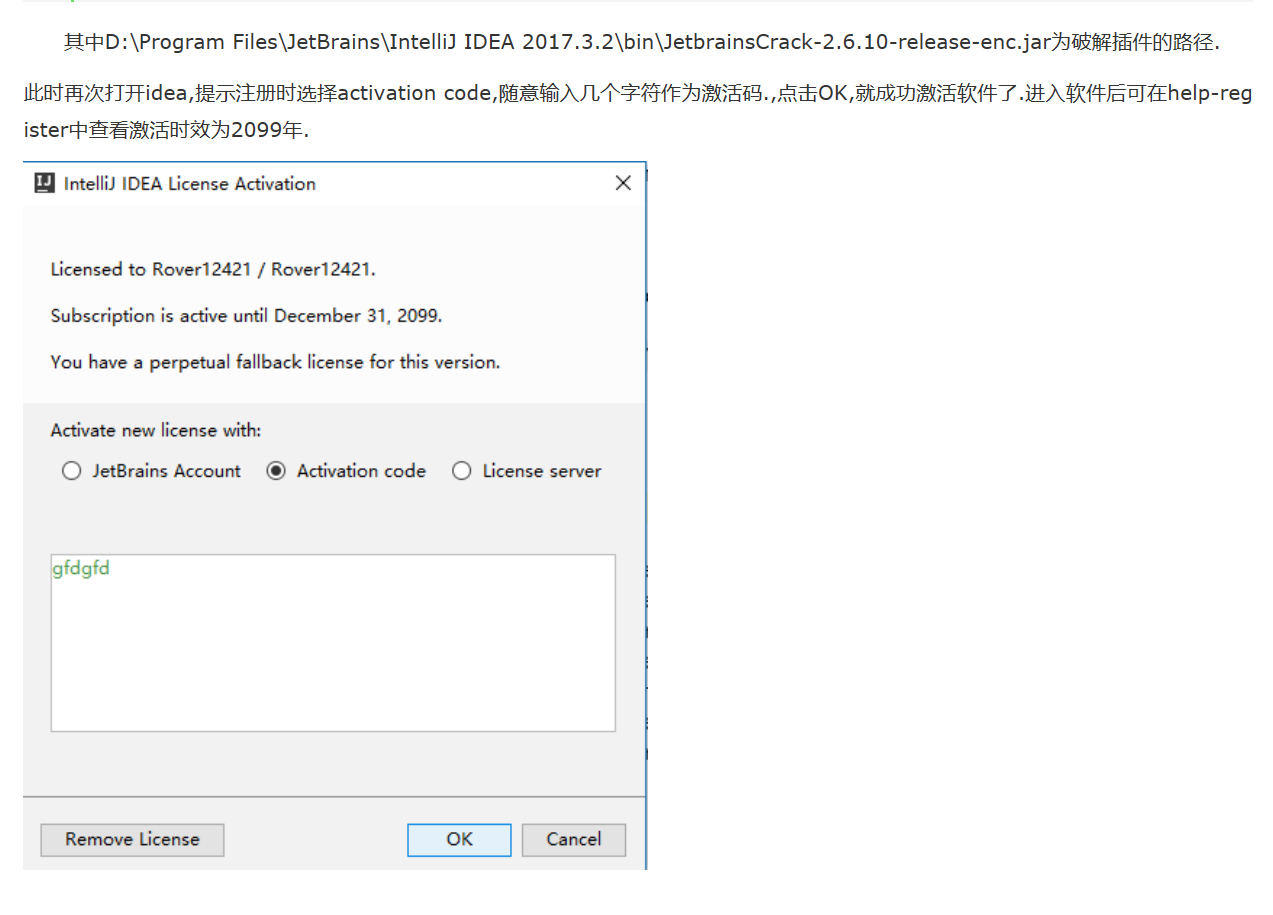
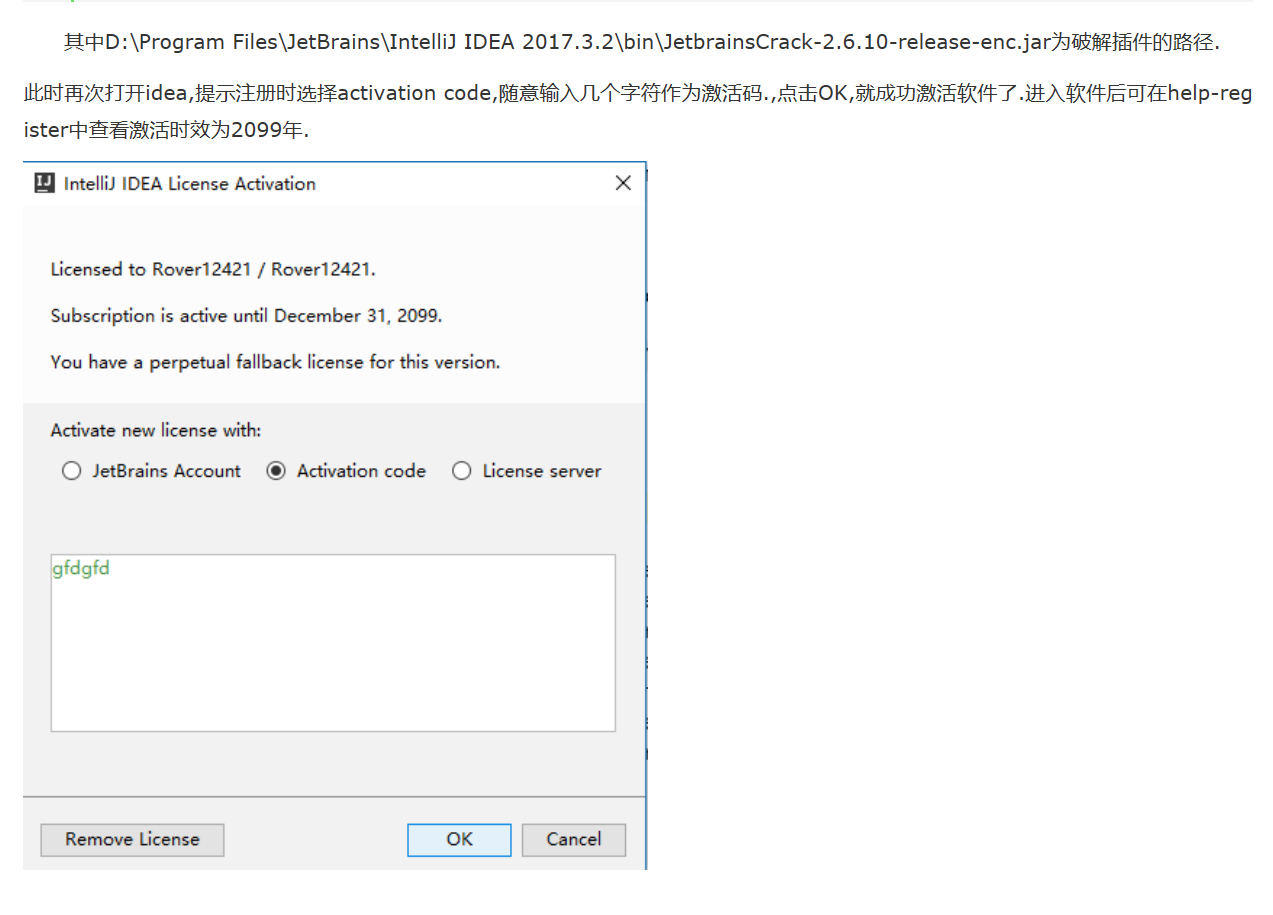
安装idea

1.需要下载去官网下载.exe,弹出一个框选第三个输入激活码：Idea免费激活码<http://www.cnblogs.com/suiyueqiannian/p/6754091.html>

2.2017.3以后的不能用以上方法了：<http://www.cnblogs.com/legoo/p/8214165.html>

D:\东西\idea的破解里边的有一个jar包





获奖等级省级一等奖金→国赛二等奖金