

好的，那么在上一节中呢，评论功能的后台已经写好了，这一节，先把这部分后台代码和前台对接一下。

1. 评论功能实现

我们修改一下保存评论按钮的点击事件，用jQuery的方式获取文本框中的值，然后通过ajax方法，把数据传递到CommentController.jsp，jsp就是Servlet，这样写就和传递到Servlet是差不多的意思。

detail.jsp

代码：

```
$("#button").eq(0).on('click',function(){
    var txt = $('#commenttxt').val();
    $.post("${basePath}/controller/CommentController.jsp",{txt : txt},function(data){
        data = data.trim();
        if(data == '-1'){
            alert('请先登录!');
        }
    });
});
```

我们采用{}，也就是JSON的方式，将数据包裹一下，传递过去。

在JSP页面是这样接收的：

```
<%@page import="bean.User"%>
<%@ page language="java" import="java.util.*" pageEncoding="UTF-8"%>
<%
    //因为评论是需要登录的,如果没有登录我们就调整到登录页面去
    User user = (User)session.getAttribute("user");
    if(user==null){//代表用户没有登录
        out.print("-1");
        return;
    }
    String txt = request.getParameter("txt").toString();
    System.out.println(txt);
%>
```

它只是一种写法而已，如果在Servlet中，你是不是还要在doGet或者doPost方法中传递一个 HttpServletRequest request 参数呢？而当我们采用JSP的话呢，就不需要这么做了，这就是九大隐式对象的好处。

我们通过

```
request.getParameter("txt").toString();
```

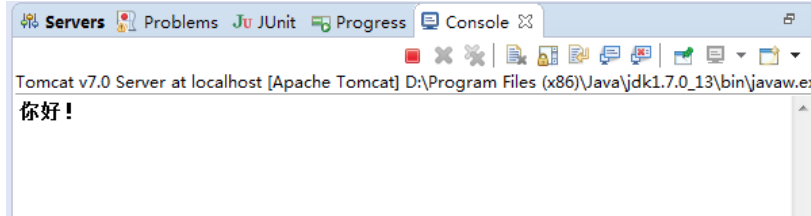
的方式，先获取到前台传过来的评论信息，并且用一个String数据类型的txt，保存下来。

接着，先不急着往下写，你可以先打印出来看看，看看它能不能取到，如果这都不能取到，就说明某个地方已经错了，再写下去也没有意义。

你好！

保存评论

我们随便写一个什么东西，点击保存评论。



控制台成功打印出来了，这说明代码没问题。

OK，继续，接下来，我们就需要把这个信息保存到数据库里了。

在调用CommentService之前，我们先看一下要保存哪些东西。（这些信息去CommentService里面看，之前写过一个saveComment方法）

保存信息的清单：

id,user_id,content,article_id,create_time,is_delete

id自然不用多说，我们采用UUID就行了。user_id是用户ID，就是当前登录的那个人。content的话，就是评论的内容，也就是我们刚刚取到的txt。article_id是文章的id，到时候加载评论的时候，肯定就是通过这个字段来查找的。create_time是创建日期，is_delete为是否删除，我们默认给他一个0，表示不删除。

好的，这些东西确认好了，就开始写代码吧。

通过以上的分析，我们是不是还要把文章ID从前台传过来呢？好的，说干就干。

回到detail.jsp，给点击事件中的JSON对象添加一个属性，把文章ID拿到：

```
$(".button").eq(0).on('click',function(){
    var txt = $('#commenttxt').val();
    $.post("${basePath}/controller/CommentController.jsp",{txt : txt , articleId : "${article.id}"},function(data){
        data = data.trim();
        if(data == '-1'){
            alert('请先登录！');
        }
    });
});
```

修改后的CommentController

```
<%@page import="bean.User"%>
<%@page import="bean.Comment"%>
<%@page import="service.CommentService"%>
<%@ page language="java" import="java.util.*" pageEncoding="UTF-8"%>
<%
    User user = (User)session.getAttribute("user");
    if(user==null){代表用户没有登陆
        out.print("-1");
        return;
    }
    String txt = request.getParameter("txt").toString();
    String articleId = request.getParameter("articleId").toString();

    CommentService commentService = new CommentService();
    Comment comment = new Comment();
    comment.setId(UUID.randomUUID().toString());
    comment.setUserId(user.getId());
    comment.setArticleId(articleId);
    comment.setContent(txt);
    commentService.saveComment(comment);
%>
```

测试：

OK

保存评论

General

Request URL: http://localhost/Article/controller/CommentController.jsp
Request Method: POST
Status Code: 200 OK
Remote Address: [::1]:80

Response Headers view source

Content-Length: 8
Content-Type: text/html; charset=UTF-8
Date: Wed, 30 Nov 2016 01:38:38 GMT
Server: Apache-Coyote/1.1

Request Headers (12)

Form Data view source view URL encoded

txt: OK
articleId: fb6b70ea-4023-45a3-9106-561649803805

哎哟不错哦~~ 貌似木有问题。

再看看数据库。

id	user_id	content	article_id	create_time	update_time	is_delete
1815f68d-0094-4687-8865-319600c3-550a-4f9f-80cf	319600c3-550a-4f9f-80cf	OK	fb6b70ea-4023-45a3-9106-561649803805	2016-11-30 09:41:06	2016-11-30 09:41:06	0
3f8a4c60-006c-4173-b70e-319600c3-550a-4f9f-80cf	319600c3-550a-4f9f-80cf	很不错的文章，赞一个！	fb6b70ea-4023-45a3-9106-561649803805	2016-11-17 11:17:06	2016-11-17 11:17:07	0

哎呀呀，真的有了！

那么这是不是就说明，评论功能到目前都是正确的？

接下来，我们是不是还应该给用户一点反馈表示评论已经成功保存了？

我们在CommentController里面，最后再加上返回一个标志位 1 即可。

```
out.print("1");
```

然后前台判断，如果返回的是1，那么就提示“保存成功！”

```
$(".button").eq(0).on('click',function(){
    var txt = $('#commenttxt').val();
    $.post("${basePath}/controller/CommentController.jsp",{txt : txt , articleId : "${article.id}"},function(data){
        data = data.trim();
        if(data == '-1'){
            alert('请先登录！');
        }else if(data == '1'){
            alert('保存成功！');
        }
    });
});
```

这样不就好了吗？O(∩_∩)O~~

localhost 显示：

保存成功！

☐ 禁止此页再显示对话框。

确定

2.评论加载

评论保存已经没问题了，接下来就是加载。

回到detail.jsp页面，第一步，是不是要导包啊？

```
<%@ page language="java" import="service.CommentService"%>
```

第二步，就是获取CommentService的实例对象。然后把我们需要评论都查询出来。（因为之前查询文章内容的时候，已经获取到id了，所以这里不再获取）

```
<%
    CommentService commentService = new CommentService();
    List<Map<String,Object>> list = commentService.getCommentsByArticleId(id);
    pageContext.setAttribute("comments", list);
    System.out.println(list);
%>
```

测试，刷新一下页面，要是不起作用的话呢，就重启一下tomcat。

控制台成功打印出了信息：

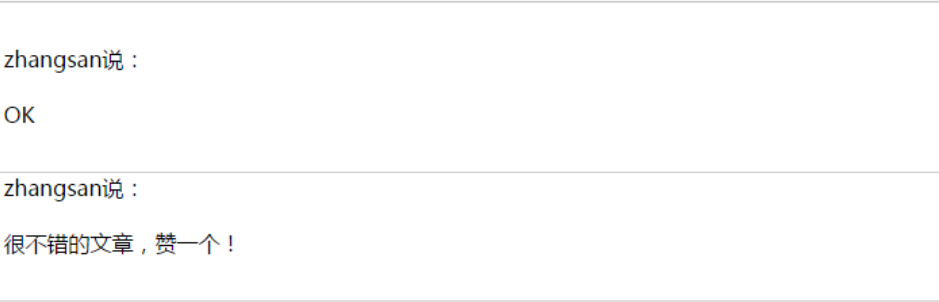
[{content=OK, username=zhangsan}, {content=很不错的文章，赞一个！, username=zhangsan}]

接下来，就是如何把这些数据贴到页面上的问题了。

怎么贴呢，是不是还要循环一下？

```
<div class='mb64' class="comment_list">
    <c:forEach items="${comments}" var="comment">
        <div class="comment_infor clearfix">
            <div style='border-bottom:1px solid #ccc' class="comment_word">
                <p style='border-bottom:20px solid #fff'>${comment.username}说: </p>
                <p class='mb32'>${comment.content}</p>
            </div>
        </div>
    </c:forEach>
</div>
```

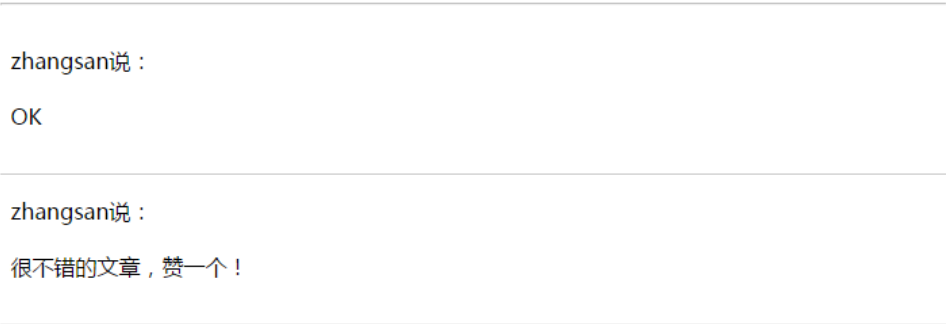
这样就行了，刷新：



样式稍微有点问题，稍微调一下：


```
.comment_word {
    margin-top:16px;
}
```

这样好些了。



最后，当提示保存成功的时候，顺便把页面刷新一下，这样方便我们直接看到效果：

```
$(".button").eq(0).on('click',function(){  
    var txt = $('#commenttxt').val();  
    $.post("${basePath}/controller/CommentController.jsp",{txt : txt , articleId : "${article.id"}",  
        data = data.trim();  
        if(data == '-1'){  
            alert('请先登录!');  
        }else if(data == '1'){  
            alert('保存成功!');  
            location.reload();  
        }  
    });  
});
```



本文结束...

最近事情太多，一方面项目工期加紧，空余时间越来越少。另一方面，我自己的个人网站-小兔博客刚刚起步，一大堆BUG要调整，真的有些分身乏术的感觉。

这个系列，到这一节，一共写了12章，虽然不够完整，但是基本的前后台已经有了。我想，对于新人而言，还是有一定的借鉴意义的。我发现之前的这段日子里，自己开坑无数，却都不系统，所以，我打算在今后的文章中，做一个系统的归类，否则的话显得有些杂乱。

这个系列，这应该算是最后一章，没法子，个人精力有限，不可能真的把一个完完整整的网站做出来，配上博客讲解，那样的话工作量实在太大了。不过这个系列我还是倾注了很多心血的，一路上，也感谢你们的陪伴，这个小网站也是在诸位的眼皮子底下一步一步成长起来的。最后，还是非常感谢大家的陪伴。好的，那么这个系列到此为止，结束了。

小兔博客：<http://www.xiaotublog.com/>

文章发布系统demo（雏形）：<http://www.xiaotublog.com/8088/Article/>（测试用户：zhangsan/123456）

技术文章搜索（简书、博客园）：<http://www.xiaotublog.com/search.html>