CKEditor的基本使用及上传图片功能说明

1.基本使用

官网下载CKEditor,可选Basic, Standard, Full

解压放置其WebRoot下

JSP中引入以下文件：

1. <script type="text/javascript" src="ckeditor/ckeditor.js"></script>
2. <script type="text/javascript" src="ckeditor/config.js"

有以下几种方法可实现CKEditor

**一：通过设置class样式**

1. **<textarea** class="ckeditor" name="editor1"**></textarea>**

**二：使用JS代码**

1. $(document).ready(**function**(){
2. CKEDITOR.replace('content');  //entity\_content为textarea元素ID
3. });
4. 或者
5. <textarea id="content" name="content">${jzbHtml.content}</textarea>
6. <script type="text/javascript">CKEDITOR.replace("content");</script>

**三：使用jquery**

**使用jquery之前一定要先引入jquery文件，另外还要引入ckeditor/adapters下的一个jquery.js**

**(两者不一样，务必要引入)**

1. $(document).ready(**function**(){
2. $('textarea#editor1').ckeditor();
3. });

**另外配置config.js以满足自己的项目需求。**

1. CKEDITOR.editorConfig = **function**( config ) {
2. // Define changes to default configuration here.
3. // For the complete reference:
4. // http://docs.ckeditor.com/#!/api/CKEDITOR.config
6. // The toolbar groups arrangement, optimized for a single toolbar row.
7. config.toolbarGroups = [
8. { name: 'document',    groups: [ 'mode', 'document', 'doctools' ] },
9. { name: 'clipboard',   groups: [ 'clipboard', 'undo' ] },
10. { name: 'editing',     groups: [ 'find', 'selection', 'spellchecker' ] },
11. { name: 'forms' },
12. { name: 'basicstyles', groups: [ 'basicstyles', 'cleanup' ] },
13. { name: 'paragraph',   groups: [ 'list', 'indent', 'blocks', 'align', 'bidi' ] },
14. { name: 'links' },
15. { name: 'insert' },
16. { name: 'styles' },
17. { name: 'colors' },
18. { name: 'tools' },
19. { name: 'others' },
20. { name: 'about' }
21. ];
23. // The default plugins included in the basic setup define some buttons that
24. // we don't want too have in a basic editor. We remove them here.
25. config.removeButtons = 'Cut,Copy,Paste,Undo,Redo,Anchor,Underline,Strike,Subscript,Superscript';
27. // Let's have it basic on dialogs as well.
28. config.removeDialogTabs = 'link:advanced';
29. };

**根据需要 具体配置查阅其他资料**

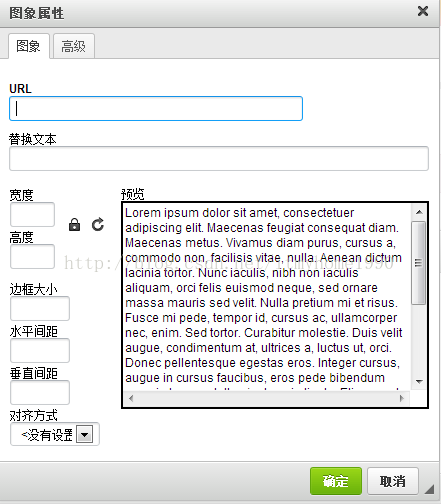
**PS：需要注意的是路径问题，在引入js的时候 如果ckeditor放在了WebRoot根目录下。**

**src应该是src="ckeditor/ckeditor.js" ，不能为src="/ckeditor/ckeditor.js"**

2.上传图片

基本配置：

本人用的CKEditor版本为4.3 CKEditor配置和部署参考[CKEditor4.x部署和配置](http://blog.csdn.net/itmyhome1990/article/details/17266045)。

CKEditor编辑器的工具栏中初始的时候应该是这样子的，没有图片上传按钮  


并且预览中有一堆火星文，可以修改相应配置删除它。

**第一种方法**：打开ckeditor/plugins/image/dialogs/image.js文件，搜索“b.config.image\_previewText”，(b.config.image\_previewText||'')单引号中的内容全删了，注意别删多了。

**第二种方法：**打开config.js文件，加入下面一句话（推荐）

config.image\_previewText=' '; //预览区域显示内容

图片上传：

Ckeditor隐藏了上传按钮，要想出现上传按钮，两种方法

第一种：还是刚才那个image.js  
搜索“upload”可以找到这一段 id:'Upload',hidden:true，而我使用的4.3的是

id:"Upload",hidden:!0，反正改为false就行了，(遗憾的是此种方法对我这个版本不起作用)

第二种：打开config.js文件，加入下面一句话

config.filebrowserImageUploadUrl= "admin/UserArticleFileUpload.do"; //待会要上传的action或servlet

1. **config.js**
2. var jzbHtmlId=$("#jzbHtmlId").val();
3. CKEDITOR.editorConfig = function( config ) {
4. config.image\_previewText=' '; //预览区域显示内容
5. var pathName = window.document.location.pathname;
6. //获取带"/"的项目名，如：/vtigu
7. var projectName = pathName.substring(0, pathName.substr(1).indexOf('/') + 1);
8. //上传请求  可传递参数
9. config.filebrowserUploadUrl=projectName+"/uploadImgForJzbHtmlCKEditor?jzbHtmlId="+jzbHtmlId;
10. };

OK现在基本上是下面这个样子的了



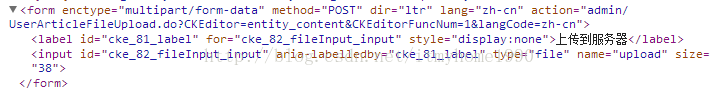


上面的只是一个上传页面。也就相当于一个HTML的form表单，

要配置点击"上传到服务器上"按钮后请求的Action。已在ckeditor/config.js中配置。

就是上面的 config.filebrowserImageUploadUrl = "admin/UserArticleFileUpload.do";

可使用chrome审查元素查看代码



最后上传图片成功  


\* 注意：图片上传后会返回到图像信息的选项卡里，并且会自动将服务器返回的imgUrl填充到URL框中

\* 有两个关键:

\* １.是返回的内容必须为　text/html　格式，并且内容为：

（error　message可以去掉，为空表示没有错误，不为空则会弹出一个对话框显error message的内容）

<script type="text/javascript">window.parent.CKEDITOR.tools.callFunction(callback,imgUrl,'error message')</script>

\* ２.其中，imgUrl　中的路径名必须以斜杠“/”分隔

接下来就是action中的上传方法：

1. **import** java.io.File;
2. **import** java.io.FileInputStream;
3. **import** java.io.FileOutputStream;
4. **import** java.io.IOException;
5. **import** java.io.InputStream;
6. **import** java.io.OutputStream;
7. **import** java.io.PrintWriter;
9. **import** javax.servlet.http.HttpServletRequest;
10. **import** javax.servlet.http.HttpServletResponse;
12. **import** org.apache.struts2.ServletActionContext;
14. **public** **class** ImgUploadAction {
15. **private** File upload; // 文件
16. **private** String uploadContentType; // 文件类型
17. **private** String uploadFileName; // 文件名
19. /\*\*
20. \* 图片上传
21. \*
22. \* @return
23. \* @throws IOException
24. \*/
25. **public** String imgUpload() **throws** IOException {
27. // 获得response,request
28. HttpServletResponse response = ServletActionContext.getResponse();
29. HttpServletRequest request = ServletActionContext.getRequest();
31. response.setCharacterEncoding("utf-8");
32. PrintWriter out = response.getWriter();
33. // CKEditor提交的很重要的一个参数
34. String callback = request.getParameter("CKEditorFuncNum");
35. String expandedName = ""; // 文件扩展名
36. **if** (uploadContentType.equals("image/pjpeg")
37. || uploadContentType.equals("image/jpeg")) {
38. // IE6上传jpg图片的headimageContentType是image/pjpeg，而IE9以及火狐上传的jpg图片是image/jpeg
39. expandedName = ".jpg";
40. } **else** **if** (uploadContentType.equals("image/png")
41. || uploadContentType.equals("image/x-png")) {
42. // IE6上传的png图片的headimageContentType是"image/x-png"
43. expandedName = ".png";
44. } **else** **if** (uploadContentType.equals("image/gif")) {
45. expandedName = ".gif";
46. } **else** **if** (uploadContentType.equals("image/bmp")) {
47. expandedName = ".bmp";
48. } **else** {
49. out.println("<script type=\"text/javascript\">");
50. out.println("window.parent.CKEDITOR.tools.callFunction(" + callback
51. + ",''," + "'文件格式不正确（必须为.jpg/.gif/.bmp/.png文件）');");
52. out.println("</script>");
53. **return** **null**;
54. }
55. **if** (upload.length() > 600 \* 1024) {
56. out.println("<script type=\"text/javascript\">");
57. out.println("window.parent.CKEDITOR.tools.callFunction(" + callback
58. + ",''," + "'文件大小不得大于600k');");
59. out.println("</script>");
60. **return** **null**;
61. }
63. InputStream is = **new** FileInputStream(upload);
64. //图片上传路径
65. String uploadPath = ServletActionContext.getServletContext().getRealPath("/img/uploadImg");
66. String fileName = java.util.UUID.randomUUID().toString(); // 采用时间+UUID的方式随即命名
67. fileName += expandedName;
68. File file = **new** File(uploadPath);
69. **if** (!file.exists()) { // 如果路径不存在，创建
70. file.mkdirs();
71. }
72. File toFile = **new** File(uploadPath, fileName);
73. OutputStream os = **new** FileOutputStream(toFile);
74. **byte**[] buffer = **new** **byte**[1024];
75. **int** length = 0;
76. **while** ((length = is.read(buffer)) > 0) {
77. os.write(buffer, 0, length);
78. }
79. is.close();
80. os.close();
82. // 返回"图像"选项卡并显示图片  request.getContextPath()为web项目名
83. out.println("<script type=\"text/javascript\">");
84. out.println("window.parent.CKEDITOR.tools.callFunction(" + callback
85. + ",'" + request.getContextPath() + "/img/uploadImg/" + fileName + "','')");
86. out.println("</script>");
87. **return** **null**;
88. }
90. **public** File getUpload() {
91. **return** upload;
92. }
94. **public** **void** setUpload(File upload) {
95. **this**.upload = upload;
96. }
98. **public** String getUploadContentType() {
99. **return** uploadContentType;
100. }
102. **public** **void** setUploadContentType(String uploadContentType) {
103. **this**.uploadContentType = uploadContentType;
104. }
106. **public** String getUploadFileName() {
107. **return** uploadFileName;
108. }
110. **public** **void** setUploadFileName(String uploadFileName) {
111. **this**.uploadFileName = uploadFileName;
112. }
113. }