

Пользовательские BBCode

Создание/обновление BBCode

Для добавления нового BBCode используйте **sceditor.formats.bbcode.set()** соответствующую функцию.

 **Важно:** При создании или обновлении BBCode это необходимо сделать до создания экземпляра редактора, например, до вызова метода `sceditor.create()`.



Предупреждение: Эта функция обновит любой существующий BBCode с тем же именем.

Функция **sceditor.formats.bbcode.set()** принимает два аргумента, **name** и **bbcode**.

- **name** String

должна содержать имя BBCode, например **[b]**, «**b**». Это значение должно быть написано строчными буквами!

- Объект **bbcode**.

Объект BBCode. Дополнительную информацию см. ниже.

Структура объекта BBCode

[script.js](#)

```
{  
    styles: {  
        "stylename": null,  
        "another-style-name": ["value1", "value2"]  
    }  
    tags: {  
        "tag": null,  
        "another-tag": {  
            "attribute1": null,  
            "attribute2": ["value1", "value2"]  
        }  
    }  
    isSelfClosing: false,  
    isInline: true,  
    isHtmlInline: undefined,
```

```
        allowedChildren: null,  
        allowsEmpty: false,  
        excludeClosing: false,  
        skipLastLineBreak: false,  
        strictMatch: false,  
  
        breakBefore: false,  
        breakStart: false,  
        breakEnd: false,  
        breakAfter: false,  
  
        format: 'string|function',  
        html: 'string|function',  
  
        quoteType: sceditor.BBCodeParser.QuoteType.auto  
    }
```

Структура объекта BBCode

styles object По умолчанию используется null

Все совпадающие теги будут переданы в **format** функцию или строку для преобразования в BBCode.

Чтобы сопоставить все вхождения стиля CSS с определенным значением, выполните следующее:

style.css

```
"style-name": ["value1", "value2"]
```

Любой DOM-узел, имеющий стиль **style-name** со значением **value1** или **value2** будет сопоставлен и передан свойству **format** для преобразования.

Чтобы найти все вхождения стиля независимо от его значения, выполните следующие действия:

style.css

```
"style-name": null
```

Это nullозначает, что значение свойства CSS не имеет значения, важно лишь то, что узел обладает этим свойством style-name.



Предупреждение: В настоящее время невозможно одновременно применять свойства `styles` и `tags` к элементам styles и tags тегам.

Чтобы выбрать тег с определенными стилями, выберите его с помощью `tags` свойства, а затем в методе `format` проверьте, имеет ли узел нужные стили. Если да, обработайте запрос, в противном случае просто верните содержимое.

Например:

script.js

```
format: function(element, content) {
    // Only handle tags with the font-weight: bold style
    if(element.style.fontWeight !== 'bold') {
        return content;
    }
    return '[b]' + content + '[/b]';
}
```

Способ сопоставления можно изменить, изменив свойство `strictMatch`.

tags

tags object По умолчанию используется null

Все совпадающие теги будут переданы в `format` функцию или строку для преобразования в BBCode.

Чтобы найти все вхождения определенного тега, выполните следующее:

script.js

```
"tag-name": null
```

Любой DOM-узел, являющийся экземпляром, <tag-name> будет передан свойству `format` для преобразования.

Чтобы найти все вхождения тега, имеющего определенный атрибут, выполните следующее:

script.js

```
"tag-name": {
    "attribute-name": null,
```

```
        "another-attribute-name": null
    }
```

Это будет соответствовать любому тегу, который является экземпляром `<tag-name>` и имеет атрибут `attribute-name` или `another-attribute-name`.

Чтобы сопоставить все вхождения тега с атрибутом, имеющим определенное значение, выполните следующее:

script.js

```
"tag-name": {
    "attribute-name": ["value1", "value2"]
}
```

Это будет соответствовать любому тегу, который является экземпляром `<tag-name>` и имеет атрибут `attribute-name` со значением `value1` или `value2`.

Способ сопоставления можно изменить, изменив свойство [strictMatch](#).

isSelfClosing

isSelfClosing Bool По умолчанию false

Если этот BBCode является самозакрывающимся тегом (не имеет закрывающего тега `[hr]`).

isInline

isInline Bool По умолчанию true

Если этот BBCode является встроенным или блочным.

isHtmlInline

isHtmlInline — логическое значение или не определено. По умолчанию используется undefined

Определяет, является ли выходной HTML-код для данного BBCode встроенным или нет. Это значение необходимо задать только в том случае, если оно отличается от `isInline` значения в BBCode. Если значение не определено, оно игнорируется и `isInline` используется вместо него.

allowedChildren

allowedChildren — массив или null. По умолчанию — null.

Если значение равно null/undefined, то будут разрешены все дочерние элементы. Если это массив, то будут разрешены только указанные теги. Чтобы разрешить использование обычного текста, используйте # в качестве имени тега.

Чтобы разрешить только обычный текст: allowedChildren: ['#']

Разрешить использование жирного и курсивного шрифта только для дочерних элементов: allowedChildren: ['b', 'i']

allowsEmpty

allowsEmpty Bool По умолчанию false

Если этот тег может быть пустым (не иметь дочерних элементов или контента).

excludeClosing

excludeClosing Bool По умолчанию false

Если не добавлять закрывающий тег. В основном для того, чтобы [*] можно было использовать без /*] .

skipLastLineBreak

skipLastLineBreak - логическое значение, по умолчанию false

 В блочных тегах во всех браузерах, кроме Internet Explorer, в конце добавляется дополнительный перенос строки . Если этот параметр установлен в значение true, дополнительный перенос строки добавляться не будет.

strictMatch

strictMatch Bool По умолчанию false

Следует ли использовать строгий поиск по атрибутам и стилям.

Если значение истинно, это приведет к выполнению операции И, требующей совпадения всех атрибутов тегов и стилей.

Если значение равно `false`, будет выполнено сопоставление по оператору ИЛИ, и совпадение будет обнаружено, если совпадает хотя бы один из атрибутов или стилей тега.

breakBefore

breakBefore - логическое значение, по умолчанию false

Если необходимо вставить новую строку перед открывающим тегом.

breakStart

breakStart Bool По умолчанию false

Если нужно вставить новую строку после начального тега.

Примечание: Это не относится к самозакрывающимся тегам.

breakEnd

breakEnd Bool По умолчанию false

Если необходимо вставить новую строку перед закрывающим тегом.

Примечание: Это не относится к самозакрывающимся тегам.

breakAfter

breakAfter Bool По умолчанию false

Если необходимо вставить новую строку после закрывающего тега.

format

format String или function

Это должна быть либо строка в формате «`[b]{0}/b]`», где `{0}` будет заменена содержимым тегов BBCode.

Или функция, которая принимает два аргумента, `element&`, `content` и возвращает отформатированную строку BBCode.

- `element` HTMLElement —

объект DOM HTMLElement, который необходимо преобразовать.

- **content String** —

строка, содержащая содержимое BB-кодов. например:

script.js

```
function(element, content) {
    if(!element.attr('data-youtube-id'))
        return content;

    return '[youtube]' + element.attr('data-youtube-id') +
'[/youtube]';
}
```

html

HTML - String или function

Это должна быть либо строка в формате «`<>{0}`», где `{0}` будет заменено содержимое HTML-тегов.

Или функция, которая принимает 3 аргумента (токен, атрибуты, содержимое) и возвращает HTML-строку.

- **token** Object

TokenizeToken

- **attrs** Object

атрибутов. Атрибут по умолчанию будет `[tag=default]` установлен на `defaultattr`

- **content** String

HTML-содержимое этого тега например:

script.js

```
html: function(token, attrs, content) {
    if(typeof attrs.defaultattr !== 'undefined')
        content = '<cite>' + attrs.defaultattr + '</cite>' + content;
```

```
        return '<blockquote>' + content + '</blockquote>';  
    }
```

quoteType

quoteType sceditor.BBCodeParser.QuoteType

По умолчанию используется sceditor.BBCodeParser.QuoteType.auto

Тип цитирования атрибута.

Это должна быть либо функция, либо одно из следующих значений:

- **sceditor.BBCodeParser.QuoteType.always**

Всегда заключать значение атрибута в кавычки

- **sceditor.BBCodeParser.QuoteType.never**

Никогда не заключайте значение атрибута в кавычки.

- **sceditor.BBCodeParser.QuoteType.auto**

Заключайте значение атрибута в кавычки только в том случае, если оно содержит пробелы или знак равенства.

Дополнения и Файлы

[Ссылка на оригинальную статью](#)

From:
<https://wwoss.ru/> - worldwide open-source software

Permanent link:
https://wwoss.ru/doku.php?id=software:development:web:docs:web:wysiwyg:sceditor_doc_custom_bbcode

Last update: 2026/01/07 12:16

