



# 개발자 가이드 (APIs 규격서)

v2.3

Professional

2024-03-07

Web Editor Team  
S-Core

# 문서 개정이력 표

[illegible]

## 목차

|  |    |
|--|----|
| I. APIs .....  | 4  |
| 1. Methods .....   | 4  |
| 1.1 clear .....  | 4  |
| 1.1.1 Plugin .....                                       | 4  |
| 1.1.2 Method.....  | 4  |
| 1.2 create .....   | 5  |
| 1.2.1 Plugin .....                                       | 5  |
| 1.2.2 Method.....  | 5  |
| 1.3 decodeMHTMLToEditor .....                            | 7  |
| 1.3.1 Plugin .....                                       | 7  |
| 1.3.2 Method.....  | 8  |
| 1.4 destroy .....  | 9  |
| 1.4.1 Plugin .....                                       | 9  |
| 1.4.2 Method.....  | 9  |
| 1.5 encodeMHTMLFromEditor .....                          | 10 |
| 1.5.1 Plugin .....                                       | 10 |
| 1.5.2 Method.....  | 10 |
| 1.6 encodeMHTMLFromEditorEx .....                        | 11 |
| 1.6.1 Plugin .....                                       | 11 |
| 1.6.2 Method.....  | 11 |
| 1.7 getCharCount.....                                    | 14 |
| 1.7.1 Plugin .....                                       | 14 |
| 1.7.2 Method.....  | 15 |
| 1.8 getContent .....                                     | 17 |
| 1.8.1 Plugin .....                                       | 17 |
| 1.8.2 Method.....  | 17 |
| 1.9 getContentSize ( <i>version added: 2.3.28</i> )..... | 18 |
| 1.9.1 Plugin .....                                       | 18 |
| 1.9.2 Method.....  | 18 |
| 1.10 isReadOnly .....                                    | 19 |
| 1.10.1 Plugin.....                                       | 19 |
| 1.10.2 Method .....                                      | 20 |
| 1.11 setReadOnly .....                                   | 21 |
| 1.11.1 Plugin.....                                       | 21 |

|   |    |
|---|----|
| 1.11.2 Method .....                               | 21 |
| 1.12 setContent.....                              | 23 |
| 1.12.1 Plugin.....                                | 23 |
| 1.12.2 Method .....                               | 23 |
| 1.13 setHeight .....                              | 24 |
| 1.13.1 Plugin.....                                | 24 |
| 1.13.2 Method .....                               | 24 |
| II. Custom APIs .....                             | 24 |
| 1. Methods .....                                  | 24 |
| 1.1 isEmpty ( <i>version added: 2.3.23</i> )..... | 25 |
| 1.1.1 Plugin .....                                | 25 |
| 1.1.2 Method.....                                 | 25 |

## I. APIs

아래는 CafeNote 에서 제공하는 APIs 에 대해서 기술되어 있습니다.

### 1. Methods

#### 1.1 clear

##### 1.1.1 Plugin

N/A

##### 1.1.2 Method

| Name        | clear   | Type | API |
|-------------|---|------|-----|
| Definition  | <div>clear( String id )</div> <div><b>Parameters:</b></div> <ul style="list-style-type: none"><li>id: 에디터의 ID.</li></ul> <div><b>Return value:</b></div> <ul style="list-style-type: none"><li>Boolean (true/false): 에디터가 정상적으로 초기화되었다면 true, 초기화에 실패했다면 false 반환</li></ul> |      |     |
| Description | 에디터 편집창 내 본문(HTML document) 전체 내용과 히스토리에 저장되어 있는 데이터를 초기화합니다.   |      |     |

|         |   |
|---------|---|
| Example | <pre> : &lt;script&gt;     var isClear = cafe.clear( "cafeDemo" );      if(!isClear) {         cafe.create(             "cafeDemo",             "/config.json",             "post",             function( editor ) {                 editor.setContent( "&lt;h1&gt;Hello&lt;/h1&gt;", {"format": "html"} );             }         );     } &lt;/script&gt; : </pre> |
|---------|---|

## 1.2 create

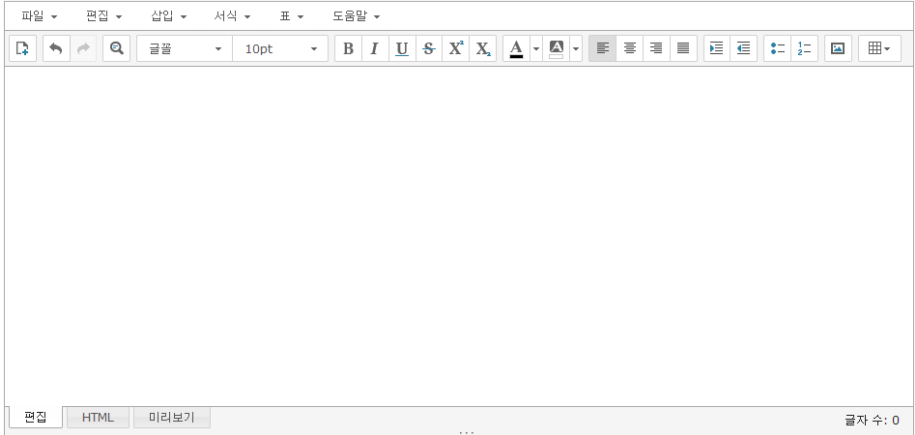
### 1.2.1 Plugin

N/A

### 1.2.2 Method

| Name       | create   | Type | API |
|------------|--|------|-----|
| Definition | <p><b>1. 환경 설정 파일 로딩(loading)을 통한 에디터 생성:</b></p> <p>create( String id, String url, String type, Function callback )</p> <p><b>Parameters:</b></p> <ul style="list-style-type: none"> <li>id: 에디터 ID</li> <li>url: 에디터 환경 설정 파일(config.json)이 있는 경로</li> <li>type (optional): 환경 설정 파일(config.json) 전송(HTTP Request) 방식 <ul style="list-style-type: none"> <li>"GET" (default)</li> <li>"POST"</li> </ul> </li> <li>callback (optional): 에디터가 생성된 뒤 실행되는 콜백(callback) 함수</li> </ul> <p><b>Return value:</b></p> <ul style="list-style-type: none"> <li>undefined: 없음</li> </ul> <p><b>2. 환경 설정 정보가 담긴 JSON Object 를 통한 에디터 생성:</b></p> |      |     |

|             |  |
|-------------|--|
|             | <p>create( String id, Object jsonConfig, Function callback )</p> <p><b>Parameters:</b></p> <ul style="list-style-type: none"> <li>• id: 에디터 ID</li> <li>• jsonConfig: 에디터 환경 설정 정보가 담긴 JSON Object 명</li> <li>• callback (optional): 에디터가 생성된 뒤 실행되는 콜백(callback) 함수</li> </ul> <p><b>Return value:</b></p> <ul style="list-style-type: none"> <li>• undefined: 없음</li> </ul>  |
| Description | 에디터를 입력된 설정에 따라 생성하고, 생성 뒤 콜백 함수를 호출합니다.   |
| Example     | <p>1) config.json 파일로써 에디터를 생성하는 경우:</p> <pre> : &lt;script&gt;     cafe.create( "cafeDemo", "/config.json", "GET" ); &lt;/script&gt; : </pre> <p>2) config.json 파일로써 에디터를 생성하고, 에디터 생성 직후 본문 내용을 삽입하는 경우:</p> <pre> : &lt;script&gt;     cafe.create(         "cafeDemo",         "/config.json",         "GET",         function() {             cafe.setContent( "cafeDemo",                 "&lt;p&gt;&lt;span style='color: blue; font-size: 16pt;'&gt;Hello&lt;/span&gt;&lt;/p&gt;",                 {"format": "html"} );         }     ); &lt;/script&gt; : </pre> <p>3) JSON Object로써 에디터를 생성하는 경우:</p> <pre> : &lt;script&gt;     var objConfigJSON = {         "height": 350,         "plugins": "autolink dragdrop ... subscript table",         :         "invalid_elements": "script"     };      cafe.create( "cafeDemo",objConfigJSON ); &lt;/script&gt; : </pre> <p>4) JSON Object로써 에디터를 생성하고, 에디터 생성 직후 본문 내용을 삽입하는 경우:</p> |

|        |   |
|--------|---|
|        | <pre> : &lt;script&gt;   var objConfigJSON = {     "height": 350,     "plugins": "autolink dragdrop ... subscript table",     :     "invalid_elements": "script"   };    cafe.create(     "cafeDemo",     objConfigJSON,     function() {       cafe.setContent( "cafeDemo",         "&lt;p&gt;&lt;span style='color: blue; font-size:         16pt;'&gt;Hello&lt;/span&gt;&lt;/p&gt;",         {"format": "html"} );     }   ); &lt;/script&gt; : </pre> |
| Result | <p>에디터 화면 (config.json 파일로써 에디터를 생성하는 경우) :</p>    |

## 1.3 decodeMHTMLToEditor

### ※ 주의사항:

- 해당 메소드를 사용하기 위해서는 반드시 MHTMLLoader 플러그인을 사용하도록 config 에 명시되어 있어야만 합니다.

### 1.3.1 Plugin

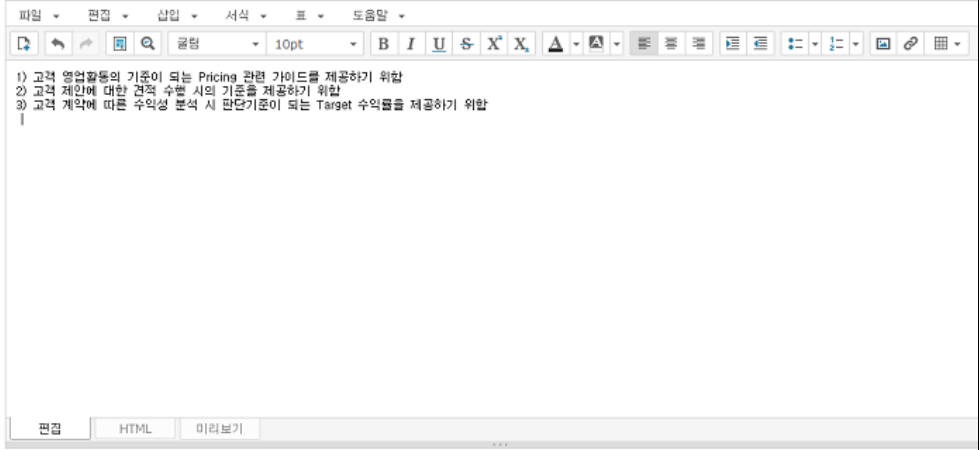
| Name          | mhtmlloader                                  | Type | Plugin |
|---------------|--|------|--------|
| Configuration | config.json 에 mhtmlloader 플러그인을 사용하도록 명시합니다. |      |        |



|             |  |
|-------------|--|
|             | <pre>{   :   "plugins": "mhtmlloader",   : }</pre> |
| Description | (※ 플러그인에 대한 설명은 관리자 가이드를 참고하시기 바랍니다.)              |

### 1.3.2 Method

| Name        | decodeMHTMLToEditor   | Type | API |
|-------------|---|------|-----|
| Definition  | <p>decodeMHTMLToEditor( String id, String content )</p> <p><b>Parameters:</b></p> <ul style="list-style-type: none"> <li>id: 에디터 ID</li> <li>content: MHTML 양식의 데이터</li> </ul> <p>※ 참고사항:</p> <ul style="list-style-type: none"> <li>BASE64 및 quoted-printable 방식으로 인코딩된 MHTML 의 디코딩을 지원합니다.</li> <li>Character Set 은 UTF-8 을 지원하며, 디코딩 시에만 EUC-KR 을 추가 지원합니다.</li> </ul> <p><b>Return value:</b></p> <ul style="list-style-type: none"> <li>Boolean (true/false): 디코딩이 정상적으로 되었다면 true, 디코딩이 실패했다면 false 반환</li> </ul> |      |     |
| Description | <p>API 호출 시 입력된 MHTML 양식의 문자열을 HTML 로 파싱(Parsing)하여 에디터에 삽입합니다.</p> <p>만약, 에디터에 기 입력된 데이터가 있는 경우, 기존 데이터는 삭제되고 디코딩(decoding)된 데이터로 교체됩니다.</p>   |      |     |
| Example     | <p>1) config.json 파일에 아래와 같이 작성합니다.</p> <pre>: "plugins": "mhtmlloader" :</pre> <p>2) 에디터를 생성한 페이지에 아래와 같이 decodeMHTMLToEditor() 를 호출합니다.</p> <pre>: &lt;script&gt; var mhtmlData = "MIME-Version: 1.0" + "\n" +                 "Content-Type: text/html;" + "\n" +                 ...</pre>  |      |     |

|        |   |
|--------|---|
|        | <pre> "vPbAzbf8wLsmb...SJsaW5lLWhlaWdo" + "\n" + "dDogMTIwJTsgb...j4mbmJzcDs8L3A+" + "\n" + "PC9ib2R5PjwvaHRtbD4=" + "\n";  cafe.decodeMHTMLToEditor( "cafeDemo", mhtmlData ); &lt;/script&gt; : </pre> |
| Result | <p>에디터 화면:</p>    |

## 1.4 destroy

### 1.4.1 Plugin

N/A

### 1.4.2 Method

| Name        | destroy  | Type | API |
|-------------|--|------|-----|
| Definition  | <p>destroy( String id )</p> <p><b>Parameters:</b></p> <ul style="list-style-type: none"> <li>id: 에디터 ID</li> </ul> <p><b>Return value:</b></p> <ul style="list-style-type: none"> <li>Boolean (true/false): 에디터가 정상적으로 제거 되었다면 true, 이미 제거되었거나 실패했다면 false 반환</li> </ul> |      |     |
| Description | 해당 ID 값으로 웹 페이지에 생성된 에디터를 제거합니다.   |      |     |

|         |   |
|---------|---|
| Example | <pre> : &lt;script&gt;     cafe.destroy( "cafeDemo" ); &lt;/script&gt; : </pre> |
|---------|---|

## 1.5 encodeMHTMLFromEditor

※ **주의사항:**

- 해당 메소드를 사용하기 위해서는 반드시 *MHTMLLoader* 플러그인을 사용하도록 *config* 에 명시되어 있어야만 합니다.

### 1.5.1 Plugin

(※ MHTMLLoader 플러그인 설정은 상기 decodeMHTMLToEditor 메소드의 Plugin 설정 내용을 참고하시기 바랍니다.)

### 1.5.2 Method

| Name        | encodeMHTMLFromEditor  | Type | API |
|-------------|--|------|-----|
| Definition  | <p>encodeMHTMLFromEditor( String id )</p> <p><b>Parameters:</b></p> <ul style="list-style-type: none"> <li>id: 에디터 ID</li> </ul> <p><b>Return value:</b></p> <ul style="list-style-type: none"> <li>String: 변환된 MHTML 양식의 문자열</li> </ul> <p>※ 참고사항:</p> <ul style="list-style-type: none"> <li>BASE64 방식의 인코딩을 지원합니다.</li> <li>Character Set 은 UTF-8 을 지원합니다.</li> </ul> |      |     |
| Description | <p>에디터 편집창에 입력되어 있는 내용을 MHTML 양식의 문자열로 반환합니다.</p> <p>※ 참고사항:</p> <ul style="list-style-type: none"> <li>본문 내 포함된 이미지들(단, &lt;img&gt;tag 로 삽입된 경우) 중, 이미지 소스가 base64 인코딩(Encoding)된 Data URI 값인 경우에는 MHTML 양식으로 변환 시에 CID 처리됩니다.</li> </ul>   |      |     |
| Example     | 1) config.json 파일에 아래와 같이 작성합니다.   |      |     |

|        |   |
|--------|---|
|        | <pre> : "plugins": "mhtmlloader" : </pre> <p>2) 에디터를 생성한 페이지에 아래와 같이 <code>encodeMHTMLFromEditor()</code>를 호출합니다.</p> <pre> : &lt;script&gt;     var mhtmlData = cafe.encodeMHTMLFromEditor( "cafeDemo" );     console.log( mhtmlData ); &lt;/script&gt; : </pre>   |
| Result |  <pre> &lt;top frame&gt; &gt; var mhtmlData = cafe.encodeMHTMLFromEditor( "cafeDemo" );   console.log( mhtmlData ); MIME-Version: 1.0 X-Generator: S-Core CAFE Note Content-Type: text/html;   charset="utf-8" Content-Transfer-Encoding: base64 PCFET0NUWVBFIGh0bWw+CjxodG1sPgo8aGVhZD48bWV0YSBodHRwLWVxdWl2PSJDb250ZW50LVR5c RleHQvaHRtbDsgY2hhcnNldD1VVEYtOCIGY2xhc3M9ImN1aS1jb250ZW50LWR1ZmF1bHQiPjxsaW5 </pre> |

## 1.6 encodeMHTMLFromEditorEx

### ※ 주의사항:

- 해당 메소드를 사용하기 위해서는 반드시 *ExMHTMLLoader* 플러그인을 사용하도록 *config*에 명시되어 있어야만 합니다.

### 1.6.1 Plugin

| Name          | exmhtmlloader   | Type | Plugin |
|---------------|---|------|--------|
| Configuration | <p>config.json에 exmhtmlloader 플러그인을 사용하도록 명시합니다.</p> <pre> {     :     "plugins": "exmhtmlloader",     : } </pre> |      |        |
| Description   | (※ 플러그인에 대한 설명은 관리자 가이드를 참고하시기 바랍니다.)   |      |        |

### 1.6.2 Method

| Name       | encodeMHTMLFromEditorEx  | Type | API |
|------------|--|------|-----|
| Definition | <p>encodeMHTMLFromEditorEx( String id, Function callback )</p> <p><b>Parameters:</b></p> <ul style="list-style-type: none"> <li>id: 에디터 ID</li> <li>callback: 편집창 내용이 MHTML 문자열로 변환된 뒤 실행되는 콜백(callback) 함수</li> </ul> <p><b>Return value:</b></p> <ul style="list-style-type: none"> <li>Boolean (true/false): 인코딩 요청이 정상적으로 되었다면 true, 인코딩 요청에 실패했다면 false 반환</li> </ul> |      |     |

|             |   |
|-------------|---|
| Description | <p>에디터 편집창에 입력되어 있는 내용을 MHTML 양식의 문자열로 변환하여, callback 함수를 통해 반환합니다.</p> <ol style="list-style-type: none"> <li>CID(Content-ID) 지원 범위:<br/>Same-origin policy 에 위배되지 않고, base64 인코딩(Encoding)이 가능한 이미지.<br/>즉, <ul style="list-style-type: none"> <li>본문 내 포함된 &lt;img&gt;tag 를 사용하여 삽입된 이미지들의 CID 처리</li> <li>본문 내 포함된 배경이미지(background image)들의 CID 처리 <ul style="list-style-type: none"> <li>&lt;td&gt;,&lt;th&gt;의 background 속성(비표준)으로 삽입된 이미지</li> <li>CSS 의 background 속성으로 삽입된 이미지</li> <li>CSS 의 background-image 속성으로 삽입된 이미지</li> </ul> </li> </ul> </li> <li>공통 정책 <ol style="list-style-type: none"> <li>이미지 소스(src 또는 url())값이 "http://"또는 "https://"를 포함한 URL 인 경우, CID 처리 지원 <div style="border-left: 2px solid gray; padding-left: 10px; margin-left: 20px;"> <p>※ 주의사항:</p> <ul style="list-style-type: none"> <li>이미지 소스가 URL 인 경우: <ol style="list-style-type: none"> <li>Same-origin policy 에 위배되지 않은 동일도메인 URL 만 가능함</li> <li>http://(또는 https://)를 포함한 절대경로이어야만 함</li> <li>인코딩 시 MIME Type 처리 (※ 고객사와 협의된 내용): <ul style="list-style-type: none"> <li>아래의 특정 패턴인 경우에만 MIME Type 추출</li> <li>GIF/PNG 는 PNG 로, JPEG/JPG 는 JPG 로 인코딩(Encoding)</li> <li>만약, 추출된 값이 비정상적이거나 패턴을 위반한 경우, 기본값인 JPEG (JPG) type 으로 인코딩(Encoding)</li> <li>패턴:<br/>http://동일도메인/~~~.확장자?~~~~~ 또는<br/>https://동일도메인/~~~.확장자?~~~~~</li> </ul> </li> </ol> </li> </ul> </div> </li> <li>이미지 소스(src 또는 url())값이 base64 인코딩(Encoding)된 dataURI 값인 경우, CID 처리 지원</li> </ol> </li> </ol> |
| Example     | <ol style="list-style-type: none"> <li>config.json 파일에 아래와 같이 작성합니다. <pre> : "plugins": "exmhtmlloader" : </pre> </li> <li>에디터를 생성한 페이지에 아래와 같이 encodeMHTMLFromEditorEx() 를 호출합니다. <pre> : &lt;script&gt; var mhtmlData = "", </pre> </li> </ol>  |

|        |  |
|--------|--|
|        | <pre> result = false;  function callbackMHTML( convertData ) {     mhtmlData = convertData;      console.log( mhtmlData ); }  result = cafe.encodeMHTMLFromEditorEx(     "cafeDemo",     callbackMHTML ); &lt;/script&gt; : </pre>   |
| Result | <pre> &lt;top frame&gt; &gt; var mhtmlData = "",     result = false;  function callbackMHTML( convertData ) {     mhtmlData = convertData;      console.log( mhtmlData ); }  result = cafe.encodeMHTMLFromEditorEx( "cafeDemo", callbackMHTML ); &lt; true MIME-Version: 1.0 X-Generator: S-Core CAFE Note vdev Content-Type: text/html;     charset="utf-8" Content-Transfer-Encoding: base64  PCFET0NUWVBFIGh0bWw+CjxodG1sPgo8aGVhZD48bWV0YSBodHRwLWVxdWl2PSJDb250Z hcnNldD1VVEYtOCIGY2xhc3M9ImN1aS1jb250ZW50LWR1ZmF1bHQiPjxsaW5rIHJ1bD0i NzPSJjdWktY29udGVudC1kZWZhdWx0IiBocmVmPSJjYWZ1L215c2luZ2x1L3Nraw5zL2R GllcV2xhc3M9TmN1aS1jb250ZW50LWV0YSBodHRwLWVxdWl2PSJDb250Z </pre> |

## 1.7 getCharCount

### ※ 주의사항:

- 해당 메소드를 사용하기 위해서는 반드시 *charcount* 플러그인을 사용하도록 *config* 에 명시되어 있어야만 합니다.

### 1.7.1 Plugin

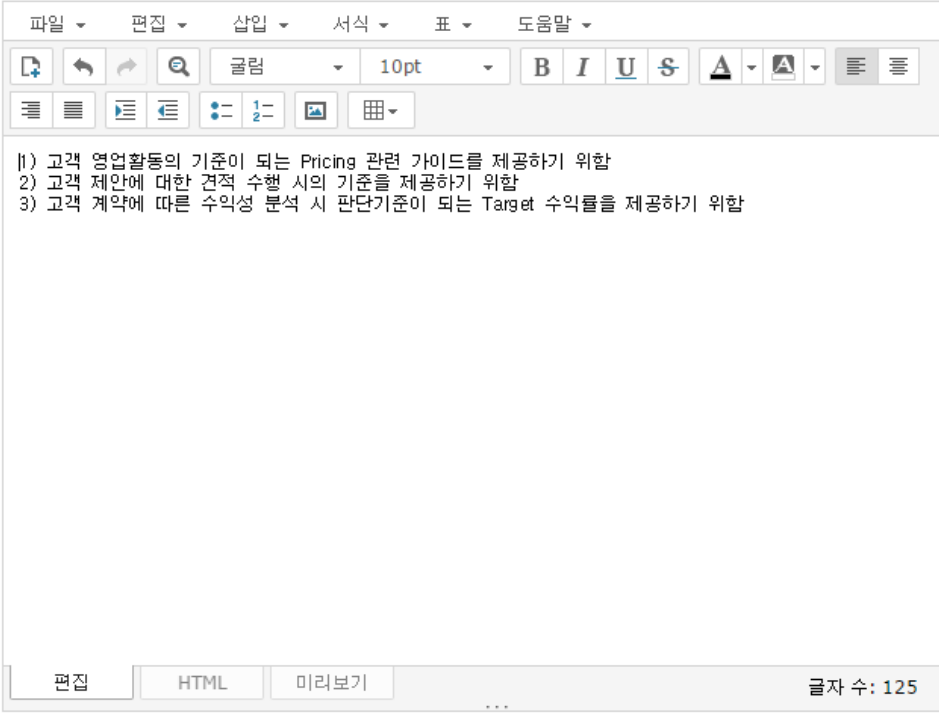
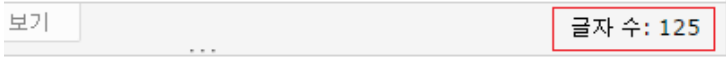
| Name          | charcount  | Type | Plugin |
|---------------|--|------|--------|
| Configuration | <p>config.json 에 charcount 플러그인을 사용하도록 명시합니다.</p> <pre> {     :     "plugins": "charcount", </pre> |      |        |

|             |                                       |
|-------------|---------------------------------------|
|             | <pre> : } </pre>                      |
| Description | (※ 플러그인에 대한 설명은 관리자 가이드를 참고하시기 바랍니다.) |

## 1.7.2 Method

| Name        | getCharCount  | Type | API |
|-------------|---|------|-----|
| Definition  | getCharCount( String id )<br><br><b>Parameters:</b> <ul style="list-style-type: none"> <li>id: 에디터 id</li> </ul> <b>Return value:</b> <ul style="list-style-type: none"> <li>Number: 본문 내 글자 수</li> </ul> |      |     |
| Description | 에디터에 작성되어 있는 내용의 글자 수를 반환합니다.   |      |     |



|         |  |
|---------|--|
| Example | <p>1) config.json 파일에 아래와 같이 작성합니다.</p> <pre> : "plugins": "charcount" : </pre> <p>2) 에디터를 생성한 페이지에 아래와 같이 본문을 작성합니다.</p>  <p>3) 본문을 작성한 에디터에 대해서 getCharCount 메소드를 호출합니다.</p> <pre> : &lt;script&gt;   var charCount = cafe.getCharCount( "cafeDemo" ); &lt;/script&gt; : </pre> |
| Result  | <p>에디터 화면:</p> <p>에디터 오른쪽 하단에 글자 수가 표시됩니다.</p>  <p>getCharCount 메소드를 호출한 결과값(변수 charCount 값):</p> <p><u>125</u></p>  |

## 1.8 getContent

### 1.8.1 Plugin

N/A

### 1.8.2 Method

| Name        | getContent  | Type | API |
|-------------|---|------|-----|
| Definition  | <p>getContent( String id, Object args )</p> <p><b>Parameters:</b></p> <ul style="list-style-type: none"><li>• id: 에디터 ID</li><li>• args (optional): 반환 받을 문자열의 형식<ul style="list-style-type: none"><li>■ HTML 양식({ "format": "html" }) 또는 TEXT 양식({ "format": "text" })</li></ul></li></ul> <p>※ <b>주의사항:</b></p> <ul style="list-style-type: none"><li>● 반환 받을 문자열 형식은 HTML 양식과 TEXT 양식만 지원합니다.</li></ul> <p>따라서, { "format": "html" } 또는 { "format": "text" } 이외의 다른 값을 기입할 경우, 결과값이 정상적으로 반환되지 않거나 오동작할 수 있으니, 주의하시기 바랍니다.</p> <p><b>Return value:</b></p> <ul style="list-style-type: none"><li>• String: 에디터 본문(HTML/TEXT 양식의 데이터)</li></ul> |      |     |
| Description | 에디터에 작성되어 있는 데이터를 HTML 양식 또는 TEXT 양식의 문자열로 반환합니다.   |      |     |
| Example     | <pre>: &lt;script&gt; // 에디터에 작성된 내용을 추출해야 할 때 아래 API를 호출 하여야 한다. var content = cafe.getContent( "cafeDemo" ); console.log( content ); &lt;/script&gt; :</pre>  |      |     |

|        |   |
|--------|---|
| Result | <pre> &gt; var content = cafe.getContent( "cafeDemo" );   console.log( content );  &lt;!DOCTYPE html&gt; &lt;html&gt; &lt;head&gt; &lt;meta http-equiv="Content-Type" content="text/html; charset=UTF-8" class="cui- content-default"&gt;&lt;style class="cui-content-default" data-cafe- default="true"&gt;/*! cafe note v2.1.47   Copyright 2014, S-Core, Inc. All Right Reserved. */     ...코드 종락... height:1.2;}&lt;/style&gt; &lt;/head&gt; &lt;body&gt;&lt;div id="cafe-note-contents" style="font-family:'굴림',Gulim,Arial;font- size:10pt;line-height:1.2;"&gt;&lt;p&gt;&amp;nbsp;&lt;/p&gt;&lt;/div&gt;&lt;/body&gt; &lt;/html&gt; </pre> |
|--------|---|

## 1.9 getContentSize (version added: 2.3.28)

### 1.9.1 Plugin

N/A

### 1.9.2 Method

| Name        | getContentSize  | Type | API |
|-------------|---|------|-----|
| Definition  | <p>getContentSize( String id )</p> <p><b>Parameters:</b></p> <ul style="list-style-type: none"> <li>id: 에디터 id</li> </ul> <p><b>Return value:</b></p> <ul style="list-style-type: none"> <li>Number: 편집창 내, HTML 문서의 전체 크기(byte)</li> </ul>                         |      |     |
| Description | <p>에디터 편집창 내의 HTML 문서의 전체 크기(byte)를 반환합니다.</p> <p>※ <b>주의사항:</b></p> <ul style="list-style-type: none"> <li>본문 &lt;body&gt; 영역의 HTML 크기가 아니라, 편집창 내 HTML 문서의 전체 크기, 즉, DOCTYPE 및 &lt;html&gt; tag, &lt;head&gt; 영역, &lt;body&gt; 영역의 크기(byte)를 모두 포함합니다.</li> </ul> |      |     |
| Example     | <p>1) 에디터를 생성한 페이지에 아래와 같이 본문을 작성합니다.</p>   |      |     |

|        |  |
|--------|--|
|        | <div> 파일 ▾ 편집 ▾ 삽입 ▾ 서식 ▾ 표 ▾ 도움말 ▾ </div> <div> </div> <div> 11) 고객 영업활동의 기준이 되는 Pricing 관련 가이드를 제공하기 위함<br/> 2) 고객 제안에 대한 견적 수행 시의 기준을 제공하기 위함<br/> 3) 고객 계약에 따른 수익성 분석 시 판단기준이 되는 Target 수익률을 제공하기 위함 </div> <div> <div> 편집 HTML 미리보기 ... </div> 글자 수: 125 </div> |
|        | 2) 본문을 작성한 에디터에 대해서 getContentSize 메소드를 호출합니다.<br><pre> : &lt;script&gt;     var contentSize = cafe.getContentSize( "cafeDemo" ); &lt;/script&gt; : </pre>   |
| Result | 변수 contentSize 에 할당된 값:<br><b>1070</b>   |

## 1.10 isReadOnly

※ **주의사항:**

- 해당 메소드를 사용하기 위해서는 반드시 `ReadOnly` 플러그인을 사용하도록 `config` 에 명시되어 있어야만 합니다.

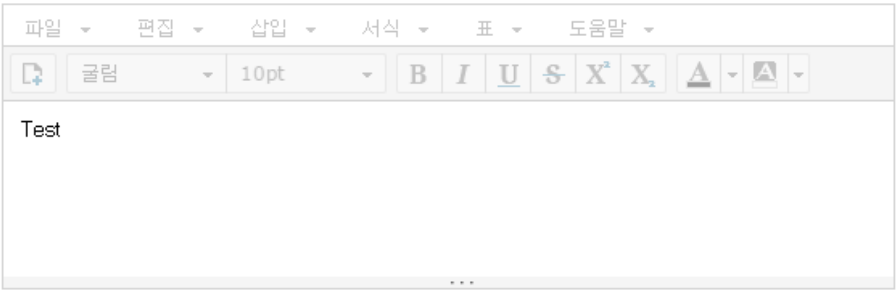
### 1.10.1 Plugin

| Name | readonly | Type | Plugin |
|------|----------|------|--------|
|------|----------|------|--------|

|               |  |
|---------------|--|
| Configuration | <p>config.json 에 readonly 플러그인을 사용하도록 명시합니다.</p> <pre>{   :   "plugins": "readonly",   : }</pre> |
| Description   | (※ 플러그인에 대한 설명은 관리자 가이드를 참고하시기 바랍니다.)  |

## 1.10.2 Method

| Name        | isReadOnly  | Type | API |
|-------------|---|------|-----|
| Definition  | <p>isReadOnly( String id )</p> <p><b>Parameters:</b></p> <ul style="list-style-type: none"> <li>id: 에디터 ID</li> </ul> <p><b>Return value:</b></p> <ul style="list-style-type: none"> <li>Boolean (true/false): 에디터가 현재 "읽기 전용 모드"이면 true, "편집 모드(기본)"이면 false 반환</li> </ul> |      |     |
| Description | 현재 에디터의 모드가 "읽기 전용 모드"인지 여부를 확인합니다.   |      |     |
| Example     | <pre>: &lt;script&gt;   var checkReadOnly = cafe.isReadOnly( "cafeDemo" );   console.log( checkReadOnly ); &lt;/script&gt; :</pre>  |      |     |

|        |   |
|--------|---|
| Result | <p>에디터 화면 (읽기 전용 모드):</p>  <p>API 실행 결과:</p> <pre> &gt; var checkReadOnly = cafe.isReadOnly( "cafeDemo" );   console.log( checkReadOnly );  true </pre> |
|--------|---|

## 1.11 setReadOnly

※ **주의사항:**

- 해당 메소드를 사용하기 위해서는 반드시 *ReadOnly* 플러그인을 사용하도록 *config* 에 명시되어 있어야만 합니다.

### 1.11.1 Plugin

(※ *ReadOnly* 플러그인 설정은 상기 *isReadOnly* 메소드의 Plugin 설정 내용을 참고하시기 바랍니다.)

### 1.11.2 Method

| Name       | setReadOnly   | Type | API |
|------------|---|------|-----|
| Definition | <p>setReadOnly( String id, Boolean isOn )</p> <p><b>Parameters:</b></p> <ul style="list-style-type: none"> <li>• id: 에디터 ID</li> <li>• isOn: <ul style="list-style-type: none"> <li>■ true: 읽기 전용 모드로 설정<br/>(편집 모드에서 읽기 전용 모드로 전환)</li> <li>■ false: 읽기 전용 모드 해제<br/>(읽기 전용 모드에서 편집 모드로 전환)</li> </ul> </li> </ul> |      |     |

|             |  |
|-------------|--|
|             | <p><b>Return value:</b></p> <ul style="list-style-type: none"> <li>Boolean (true/false): 모드 전환 결과를 반환 <ul style="list-style-type: none"> <li>true: 읽기 전용 모드로 설정/해제 성공 시</li> <li>false: 읽기 전용 모드로 설정/해제 실패 시</li> <li>null: 입력 받은 에디터 ID 를 갖는 에디터가 존재하지 않거나 isOn 에 대한 입력값의 type 이 Boolean 이 아닐 경우</li> </ul> </li> </ul>                     |
| Description | 읽기 전용 모드로 설정하거나, 읽기 전용 모드를 해제하여 편집 모드로 전환을 할 수 있습니다.   |
| Example     | <pre> : &lt;script&gt;     var result = cafe.setReadOnly( "cafeDemo", true );     console.log( result ); &lt;/script&gt; : </pre>  |
| Result      | <p>1) 에디터 화면 (편집 모드):</p>  <p>2) API 호출</p>  <p>3) 결과 (읽기 전용 모드로 전환된 에디터 화면)</p>  |

## 1.12 setContent

### 1.12.1 Plugin

N/A

### 1.12.2 Method

| Name        | setContent  | Type | API |
|-------------|---|------|-----|
| Definition  | <p>setContent( String id, String content, Object args )</p> <p><b>Parameters:</b></p> <ul style="list-style-type: none"><li>• id: 에디터 id</li><li>• content: 교체할 HTML/TEXT 양식의 데이터</li><li>• args: { "format": "html" }</li></ul> <p><b>Return value:</b></p> <ul style="list-style-type: none"><li>• String: 입력 받은 content 값(HTML/TEXT 양식의 데이터)</li></ul> |      |     |
| Description | <p>HTML/TEXT 양식의 데이터를 에디터에 삽입합니다.</p> <p>만약, 에디터에 기 입력된 데이터가 있는 경우, 해당 데이터는 삭제되고 API 호출 시 입력된 데이터로 교체됩니다.</p>   |      |     |
| Example     | <pre>: &lt;script&gt;     var content = "&lt;p&gt;Hello CafeNote&lt;/p&gt;";      cafe.setContent( "cafeDemo", content ); &lt;/script&gt; :</pre>   |      |     |



## 1.13 setHeight

### 1.13.1 Plugin

N/A

### 1.13.2 Method

| Name        | setHeight  | Type | API |
|-------------|--|------|-----|
| Definition  | <p>setHeight( String id, Number height )</p> <p><b>Parameters:</b></p> <ul style="list-style-type: none"><li>id: 에디터 id</li><li>height: 변경할 에디터의 높이 값 (단, 최소값: 100px)</li></ul> <p><b>Return value:</b></p> <ul style="list-style-type: none"><li>Number: 입력 받은 height 값</li></ul> |      |     |
| Description | <p>에디터의 높이를 변경합니다.<br/>에디터 편집창의 영역을 감안하여 최소 설정 가능한 값은 100px 로 제한됩니다.</p> <p>※ <b>주의사항:</b></p> <ul style="list-style-type: none"><li>만약, 최소값보다 작은 값을 입력할 경우, 최소값으로 강제 적용됩니다.</li></ul>   |      |     |
| Example     | <pre>:<br/>&lt;script&gt;<br/>    cafe.setHeight( "cafeDemo", 300 );<br/>&lt;/script&gt;<br/>:</pre>   |      |     |

## II. Custom APIs

아래는 고객사의 요청으로 추가된 Customizing 기능 관련 APIs 에 대해 기술되어 있습니다.

### 1. Methods

## 1.1 isEmpty (version added: 2.3.23)

### ※ 주의사항:

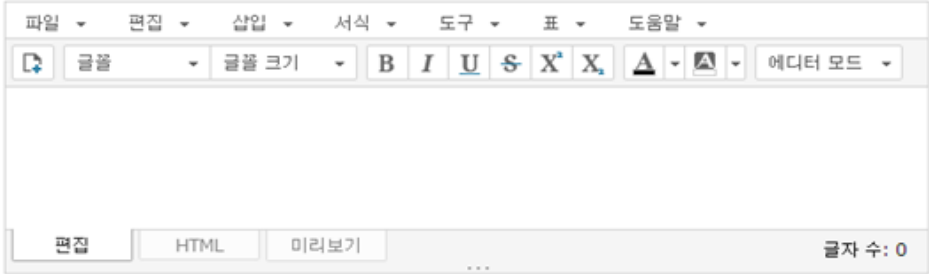
- 해당 메소드를 사용하기 위해서는 반드시 CustomAPI 플러그인을 사용하도록 config 에 명시되어 있어야만 합니다.

### 1.1.1 Plugin

| Name          | customapi  | Type | Plugin |
|---------------|--|------|--------|
| Configuration | config.json 에 customapi 플러그인을 사용하도록 명시합니다.<br><pre>{<br/>  :<br/>  "plugins": "customapi",<br/>  :<br/>}</pre> |      |        |
| Description   | (※ 플러그인에 대한 설명은 관리자 가이드를 참고하시기 바랍니다.)  |      |        |

### 1.1.2 Method

| Name        | isEmpty  | Type | API |
|-------------|--|------|-----|
| Definition  | <p>isEmpty( String id )</p> <p><b>Parameters:</b></p> <ul style="list-style-type: none"><li>• id: 에디터 ID</li></ul> <p><b>Return value:</b></p> <ul style="list-style-type: none"><li>• Boolean (true/false):<br/>에디터에 본문이 존재하지 않으면 true,<br/>본문이 존재한다면 false 반환</li></ul> <p>※ 주의사항:</p> <ul style="list-style-type: none"><li>● 본문 내 텍스트 없이 서식만 적용되어 있는 경우에도 본문이 존재하는 것으로 판단하여 false 반환</li><li>● 에디터 ID 값이 잘못 지정되었거나 텍스트모드일 경우에는 null 반환</li></ul> |      |     |
| Description | 에디터의 편집창 내에 본문이 존재하는지 여부를 확인합니다.   |      |     |

|         |   |
|---------|---|
| Example | <pre> : &lt;script&gt;     console.log( cafe.isEmpty( "cafeDemo" ) ); &lt;/script&gt; : </pre>  |
| Result  | <p>에디터 화면:</p>  <p>API 실행 결과:</p> <pre> &gt; cafe.isEmpty( "cafeDemo" ); &lt; true </pre> |

끝.