

API 정리 문서(v1.4.24)



2018. 08. 13

윌컴즈 웹에디터는  KukuDocs 엔진을 사용하고 있습니다.

문서 정보

문서 개요

이 문서는 "윌컴즈 웹에디터"에서 사용되는 API 정보를 담고 있습니다.

독자

이 문서의 독자는 "윌컴즈 웹에디터"를 이용하여 웹 사이트를 운영하고, 웹 에디터를 관리하는 사용자입니다.

문의처

이 문서의 내용에 오류가 있거나 내용과 관련된 의문사항이 있으면 담당자에게 문의하십시오.

윌커뮤니케이션즈 기술지원팀 : 031-330-8032

문서 버전 및 이력

버전	일자	이력사항
1.0	2017. 07. 19	Version 1.0 문서배포
1.0	2017. 07. 26	API 항목 추가 및 수정
1.1	2017. 08. 12	매뉴얼 상세 내용 추가

1.2	2017. 09. 01	이력 변경 없음
1.3	2017. 09. 15	동영상 삽입 이미지, emoticonList 관련 API 내용 추가
1.4	2017. 09. 29	인용구문, templateList 관련 API 설정 내용 추가
1.4.1	2017. 11. 15	IE 안정성 개선 및 API 문서 정리
1.4.4	2018. 01. 15	오타자 수정 및 문서 정리
1.4.5	2018. 01. 22	문서 버전 정리
1.4.6	2018. 01. 30	1.4.6 버전 정리
1.4.7	2018. 02. 02	오타자 수정 및 SetAttributeValueByBody 위치 이동(57->53)
1.4.8	2018. 02. 06	1.4.8 버전 정리
1.4.8	2018. 02. 06	getImageUploadURL / setImageUploadURL 누락 정정
1.4.9	2018. 02. 19	1.4.9 버전 정리
1.4.10	2018. 03. 09	1.4.10 버전정리 및 getPublicPathURL, setPublicPathURL 추가
1.4.11	2018. 03. 19	1.4.11 버전 정리
1.4.12	2018. 03. 23	1.4.12 버전 정리 및 GetEditorContent Parameter 추가
1.4.13	2018. 03. 29	1.4.13 버전 정리 및 getKeyEvent, setKeyEvent, getMouseEvent, setMouseEvent API 수정
1.4.14	2018. 04. 12	1.4.14 버전 정리
1.4.15	2018. 04. 20	1.4.15 버전 정리
1.4.16	2018. 04. 25	1.4.16 버전 정리
1.4.17	2018. 04. 25	1.4.17 버전 정리
1.4.18	2018. 05. 04	1.4.18 버전 정리
1.4.19	2018. 05. 10	1.4.19 버전 정리
1.4.20	2018. 05. 12	1.4.20 버전 정리 및 GetEditorContent API Parameter 수정
1.4.21	2018. 05. 17	1.4.21 버전 정리
1.4.22	2018. 06. 25	1.4.22 버전 정리

1.4.23	2018. 07. 04	1.4.23 버전 정리
1.4.24	2018. 07. 24	1.4.24 버전 정리 및 제거 API 수정 (getElementParentCells) / 오타자 수정

목차

1. A P I (Application Programming Interface).....	8
1.1. getElement().....	9
1.2. getToolbarElement().....	9
1.3. getContentViewElement()	10
1.4. getFooterElement()	10
1.5. getEditorWidth()	11
1.6. getEditorHeight()	11
1.7. getEditorContentWidth().....	12
1.8. getEditorContentHeight()	12
1.9. getOptions().....	13
1.10. getHelpURL()	13
1.11. setHelpURL(helpURL).....	14
1.12. getLoadingImageURL()	15
1.13. setLoadingImageURL(loadingImageURL)	15
1.14. getErrorImageURL().....	16
1.15. setErrorImageURL(errorImageURL).....	16
1.16. getPlayImageURL()	17
1.17. setPlayImageURL(playImageURL).....	17
1.18. getEmoticonList().....	18
1.19. setEmoticonList(emoticonList)	18
1.20. getTemplateList().....	19
1.21. setTemplateList(templateList)	19
1.22. getUseFooterMenu().....	20
1.23. setUseFooterMenu(useFooterMenu)	20

1.24.	getFileUploadURL().....	21
1.25.	setFileUploadURL(fileUploadURL)	21
1.26.	getImageUploadURL()	22
1.27.	setImageUploadURL(imageUploadURL).....	22
1.28.	getVideoUploadURL().....	23
1.29.	setVideoUploadURL(videoUploadURL)	23
1.30.	getKeyEvent()	24
1.31.	setKeyEvent(Key_event_Object).....	24
1.32.	getMouseEvent().....	25
1.33.	setMouseEvent(Mouse_event_Object)	25
1.34.	getCellLockName().....	26
1.35.	setCellLockName(cell_lock_name).....	26
1.36.	GetAllElements().....	27
1.37.	GetCurrentElement(tagName)	27
1.38.	GetElement(elementID).....	28
1.39.	SetEditorSize(width, height)	29
1.40.	SetEditorWidth(width)	30
1.41.	SetEditorHeight(height).....	30
1.42.	GetEditorSize()	31
1.43.	SetEditMode(modeType)	31
1.44.	GetEditorContent(isWrapping, isBackground).....	32
1.45.	GetEditorTextContent()	33
1.46.	GetContentHTMLFile().....	33
1.47.	GetAttribute(element, AttributeName)	34
1.48.	SetAttribute(element, AttributeName, AttributeValue)	35
1.49.	SetEditorContent(htmlString).....	36

1.50.	SetEditorTextContent(textString)	36
1.51.	SetFocus(element).....	37
1.52.	IsExistsElement(elementID)	37
1.53.	GetAttributeValueByBody(attributeName)	38
1.54.	SetAttributeValueByBody(attributeName, attributeValue)	39
1.55.	IsCellLockById(elementID).....	40
1.56.	SetCellLockById(elementID, isLock)	41
1.57.	IsCellLockByFocus().....	42
1.58.	SetCellLockByFocus(isLock)	42
1.59.	GetAttributeValueById(elementID, attributeName)	43
1.60.	SetAttributeValueById(elementID, attributeName, attributeValue).....	44
1.61.	GetHtmlById(elementID).....	45
1.62.	SetHtmlById(elementID, htmlString)	46
1.63.	GetTextById(elementID)	47
1.64.	SetTextById(elementID, textValue).....	48
1.65.	GetAttributeValueByFocus(attributeName, tagName)	49
1.66.	SetAttributeValueByFocus(attrName, attrValue, tagName)	50
1.67.	getPublicPathURL()	51
1.68.	setPublicPathURL(publicPathURL).....	51

1. A P I (Application Programming Interface)

이 장에서는 윤컴즈 웹에디터의 API를 설명합니다.

1.1. getRootElement()

getRootElement()

editor의 전체 DOM을 get 한다.

(※ editor란 toolbar, Content, Footer를 포함한 영역을 의미한다.)

Syntax

```
Editor_Instance.getRootElement();
```

return

Type: Element

전체 Editor영역의 DOM을 반환

1.2. getToolbarElement()

getToolbarElement ()

Toolbar영역의 DOM을 get 한다.

(※ Toolbar란 Content 영역을 편집할 때 사용하는 기능들의 UI를 의미한다.)

Syntax

```
Editor_Instance.getToolbarElement();
```

return

Type: Element

Toolbar영역의 DOM을 반환

1.3. getContentViewElement()

getContentViewElement()

Content 영역의 DOM을 get 한다.
(※ Content란 Editor를 편집하는 영역을 의미한다.)

Syntax

```
Editor_Instance.getContentViewElement();
```

return

Type: Element
Content 영역의 DOM을 반환

1.4. getFooterElement()

getFooterElement()

Footer 영역의 DOM을 get 한다.
(※ Footer란 Editor, HTML, TEXT, Preview의 Mode를 선택하는 영역을 의미한다.)

Syntax

```
Editor_Instance.getFooterElement();
```

return

Type: Element
Footer 영역의 DOM을 반환

1.5. getWidth()

getWidth()

editor의 전체 Width Size를 get 한다.

(※ editor란 toolbar, Content, Footer를 포함한 영역을 의미한다.)

Syntax

```
Editor_Instance.getWidth();
```

return

Type: Int (Number)

Editor영역의 전체 Width를 반환

1.6. getHeight()

getHeight()

editor의 전체 Height Size를 get 한다.

(※ editor란 toolbar, Content, Footer를 포함한 영역을 의미한다.)

Syntax

```
Editor_Instance.getHeight();
```

return

Type: Int (Number)

Editor영역의 전체 Height를 반환

1.7. getEditorContentWidth()

getEditorContentWidth()

Content 영역의 Width Size를 get 한다.
(※ Content란 Editor를 편집하는 영역을 의미한다.)

Syntax

```
Editor_Instance.getEditorContentWidth();
```

return

Type: Int (Number)
Content 영역의 Width를 반환

1.8. getEditorContentHeight()

getEditorContentHeight()

Content 영역의 Height Size를 get 한다.
(※ Content란 Editor를 편집하는 영역을 의미한다.)

Syntax

```
Editor_Instance.getEditorContentHeight();
```

return

Type: Int (Number)
Content 영역의 Height를 반환

1.9. getOptions()

getOptions()

Editor에 설정된 Options의 객체 정보를 get 한다.
(※ Options이란 Editor를 생성할 때 설정한 정보들을 의미한다.)

Syntax

```
Editor_Instance.getOptions();
```

return

Type: Object
Editor의 Option 정보를 가지는 객체 반환

1.10. getHelpURL()

getHelpURL()

Editor에 설정된 도움말 웹사이트의 정보를 get한다.
(※ 도움말 웹사이트이란 사용자에게 기능적 설명을 제공하는 웹사이트를 의미한다.)

Syntax

```
Editor_Instance.getHelpURL();
```

return

Type: String
Editor의 도움말 URL의 정보를 반환

1.11. setHelpURL(helpURL)

setHelpURL(helpURL)

Editor에 설정된 도움말 웹사이트의 정보를 set한다.

(※ 도움말 웹사이트이란 사용자에게 기능적 설명을 제공하는 웹사이트를 의미한다.)

Syntax

```
Editor_Instance.setHelpURL("http://solution.yooncoms.com/help");
```

Parameter

helpURL

- Type : String
- description : 도움말을 표시할 웹사이트 주소를 설정 한다.

return

1.12. getLoadingImageURL()

getLoadingImageURL()

Image Upload시 사용하는 Loading 이미지 정보를 get한다.

Syntax

```
Editor_Instance.getLoadingImageURL();
```

return

Type: String

Editor의 Loading Image URL 정보를 반환

1.13. setLoadingImageURL(loadingImageURL)

setLoadingImageURL(loadingImageURL)

Image Upload시 사용하는 Loading 이미지 정보를 set한다.

Syntax

```
Editor_Instance.setLoadingImageURL("./images/loading.gif");
```

Parameter

helpURL

- Type : String
- description : Loading Image로 사용할 이미지의 주소를 설정 한다.

return

1.14. getErrorImageUrl()

getErrorImageUrl()

Image Upload시 발생하는 Error에 사용하는 Error 이미지 정보를 get한다.

Syntax

```
Editor_Instance.getErrorImageUrl();
```

return

Type: String

Editor의 Error Image URL 정보를 반환

1.15. setErrorImageUrl(errorImageUrl)

setErrorImageUrl(errorImageUrl)

Image Upload시 발생하는 Error에 사용하는 Error 이미지 정보를 set한다.

Syntax

```
Editor_Instance.setErrorImageUrl("./images/error.png");
```

Parameter

errorImageUrl

- Type : String
- description : Error Image로 사용할 이미지의 주소를 설정 한다.

return

1.16. getPlayImageURL()

getPlayImageURL()

동영상 Upload시 편집화면에서 사용할 동영상 표현 이미지의 URL 정보를 get한다.

Syntax

```
Editor_Instance.getPlayImageURL();
```

return

Type: String

동영상 임을 표현하는 Image URL 정보를 반환

1.17. setPlayImageURL(playImageURL)

setPlayImageURL(playImageURL)

동영상 Upload시 편집화면에서 사용할 동영상 표현 이미지의 URL 정보를 set한다.

Syntax

```
Editor_Instance.setPlayImageURL("./images/play.png");
```

Parameter

playImageURL

- Type : String
- description : 동영상 표시 Image로 사용할 이미지의 주소를 설정 한다.

return

1.18. getEmoticonList()

getEmoticonList()

이모티콘으로 사용하는 이미지의 URL List 정보를 get한다.

Syntax

```
Editor_Instance.getEmoticonList();
```

return

Type: String

이모티콘으로 사용하는 Image URL List 정보를 반환

1.19. setEmoticonList(emoticonList)

setEmoticonList(emoticonList)

이모티콘으로 사용하는 이미지의 URL List 정보를 set한다.

Syntax

```
Editor_Instance.setEmoticonList( [ "./emoticon1.png", "./emoticon1.png" ] );
```

Parameter

emoticonList

- Type : Array[String]
- description : 이모티콘으로 사용할 Image들을 설정 한다.

return

1.20. getTemplateList()

getTemplateList()

Template으로 사용하는 Template List(Object) 정보를 get한다.

Syntax

```
Editor_Instance.getTemplateList();
```

return

Type: Array[Objecy]

Template로 사용하는 Template List(Object List) 정보를 반환

1.21. setTemplateList(templateList)

setTemplateList(templateList)

template으로 사용하는 Template List(Object Array) 정보를 set한다.

Syntax

```
Editor_Instance.setTemplateList( [{  
    name : '부서양식',  
    items : [ {  
        name : '회의록',  
        type : 'url',  
        value : '/template/meeting_log.html'  
    } ]  
} ] );
```

Parameter

templateList

- Type : Array[Object]
- description : Template로 사용할 내용들에 대해서 Template List 정보를 설정 한다.

1.22. getUseFooterMenu()

getUseFooterMenu()

Editor에 설정된 Footer Menu의 사용 유무를 get한다.

Syntax

```
Editor_Instance.getUseFooterMenu();
```

return

Type: Boolean(사용 : true, 미사용 : false)

Footer Menu에 대한 사용 유무를 반환

1.23. setUseFooterMenu(useFooterMenu)

setUseFooterMenu(useFooterMenu)

Editor에 설정된 Footer Menu의 사용 유무를 set한다.

(※ 사용(true)를 Parameter로 전달시 Footer Menu가 나타나고, 미사용(false)를 전달시에
는 Footer Menu가 사라진다..)

Syntax

```
Editor_Instance.setUseFooterMenu(true);
```

Parameter

useFooterMenu

- Type : Boolean
- description : Footer Menu의 사용 유무를 설정 한다.

return

1.24. getFileUploadURL()

getFileUploadURL()

Editor에 설정된 File Upload URL정보를 get한다.

(※ File Upload URL이란 사용자가 파일을 업로드할 위치의 URL 정보로 업로드하는 파일을 받아들이는 서버의 처리 주소를 의미한다. / Image, Video업로드 주소가 없을 때 기본값으로 사용되어 진다.)

Syntax

```
Editor_Instance.getFileUploadURL();
```

return

Type: String

File Upload URL의 정보를 반환

1.25. setFileUploadURL(fileUploadURL)

setFileUploadURL(fileUploadURL)

Editor에 설정된 File Upload URL정보를 set한다.

(※ File Upload URL이란 사용자가 파일을 업로드할 위치의 URL 정보로 업로드하는 파일을 받아들이는 서버의 처리 주소를 의미한다. / Image, Video업로드 주소가 없을 때 기본값으로 사용되어 진다.)

Syntax

```
Editor_Instance.setFileUploadURL("http://solution.yooncoms.com/upload");
```

Parameter

fileUploadURL

- Type : String
- description : File Upload를 처리할 서버의 주소를 설정 한다.

return

1.26. getImageUploadURL()

getImageUploadURL()

Editor에 설정된 Image Upload URL정보를 get한다.

(※ Image Upload URL이란 사용자가 Image를 업로드할 위치의 URL 정보로 업로드하는 Image를 받아들이는 서버의 처리 주소를 의미한다.)

Syntax

```
Editor_Instance.getImageUploadURL();
```

return

Type: String

Image Upload URL의 정보를 반환

1.27. setImageUploadURL(imageUploadURL)

setImageUploadURL(imageUploadURL)

Editor에 설정된 File Upload URL정보를 set한다.

(※ Image Upload URL이란 사용자가 Image를 업로드할 위치의 URL 정보로 업로드하는 Image를 받아들이는 서버의 처리 주소를 의미한다.)

Syntax

```
Editor_Instance.setImageUploadURL("http:// solution.yooncoms.com/imageUpload");
```

Parameter

imageUploadURL

- Type : String
- description : Image Upload를 처리할 서버의 주소를 설정 한다.

return

1.28. getVideoUploadURL()

getVideoUploadURL()

Editor에 설정된 Video Upload URL정보를 get한다.

(※ Video Upload URL이란 사용자가 동영상을 업로드할 위치의 URL 정보로 업로드하는 동영상을 받아들이는 서버의 처리 주소를 의미한다.)

Syntax

```
Editor_Instance.getVideoUploadURL();
```

return

Type: String

Video Upload URL의 정보를 반환

1.29. setVideoUploadURL(videoUploadURL)

setVideoUploadURL(videoUploadURL)

Editor에 설정된 Video Upload URL정보를 set한다.

(※ Video Upload URL이란 사용자가 동영상을 업로드할 위치의 URL 정보로 업로드하는 동영상을 받아들이는 서버의 처리 주소를 의미한다.)

Syntax

```
Editor_Instance.setVideoUploadURL("http://solution.yooncoms.com/videoUpload");
```

Parameter

videoUploadURL

- Type : String
- description : Video Upload를 처리할 서버의 주소를 설정 한다.

return

1.30. getKeyEvent()

getKeyEvent()

Editor에 Key Event발생시 실행되는 Key Event Function를 get 한다.

(※ KeyEvent는 키보드의 입력이 감지되었을 경우 사용자가 외부에서 삽입한 KeyEvent 실행 함수가 존재할 때 호출된다.)

Syntax

```
Editor_Instance.getKeyEvent();
```

return

Type: {object [Key : Function]}

사용자가 입력한 KeyEvent 함수 정보를 반환

1.31. setKeyEvent(Key_event_Object)

setKeyEvent(Key_event_Object)

Editor에 Key Event발생시 실행되는 Key Event Function를 set 한다.

(※ KeyEvent는 키보드의 입력이 감지되었을 경우 사용자가 외부에서 삽입한 KeyEvent 실행 함수가 존재할 때 호출된다.)

Syntax

```
Editor_Instance.setKeyEvent( { 'keydown' : keyDownFn ... } );
```

Parameter

Key_event_Object

- Type : Object ('event_name' : callbackFunction)
- description : Editor의 Key Event 발생시 호출될 외부의 함수를 설정 한다.

return

1.32. getMouseEvent()

getMouseEvent()

Editor에 Mouse Event 발생시 실행되는 Mouse Event Function를 get 한다.
(※ MouseEvent는 마우스의 동작이 감지되었을 경우 사용자가 외부에서 삽입한 MouseEvent 실행 함수가 존재할 때 호출된다.)

Syntax

```
Editor_Instance.getMouseEvent();
```

return

Type: Function

사용자가 입력한 MouseEvent 함수 정보를 반환

1.33. setMouseEvent(Mouse_event_Object)

setMouseEvent(Mouse_event_Object)

Editor에 Mouse Event 발생시 실행되는 Mouse Event Function를 set 한다.
(※ MouseEvent는 마우스의 동작이 감지되었을 경우 사용자가 외부에서 삽입한 MouseEvent 실행 함수가 존재할 때 호출된다.)

Syntax

```
Editor_Instance.setMouseEvent( { 'mousedown' : mouseDownFn ... } );
```

Parameter

Mouse_event_Object

- Type : Object ('event_name' : callbackFunction)
- description : Editor의 Mouse Event 발생시 호출될 외부의 함수를 설정 한다.

return

1.34. getCellLockName()

getCellLockName ()

Editor의 Content 중 Cell에 해당하는 Element의 편집을 제한하기 위해 설정하는 Options의 Attribute Name을 반환.

Syntax

```
Editor_Instance.getCellLockName();
```

return

Type: String

Cell_Lock을 지정하는 Attribute Name 정보를 반환

1.35. setCellLockName(cell_lock_name)

setCellLockName(cell_lock_name)

Editor의 Content 중 Cell에 해당하는 Element의 편집을 제한하기 위해 설정하는 Options의 Attribute Name을 설정.

Syntax

```
Editor_Instance.setCellLockName('free');
```

Parameter

cell_lock_name

- Type : String

- description : Content영역에 입력된 Cell Element의 편집을 방지하는 Attribute Name을 설정 한다.

return

1.36. GetAllElements()

GetAllElements()

Content 영역에 존재하는 모든 Element들을 찾아서 반환한다.
(※ Content란 Editor를 편집하는 영역을 의미한다.)

Syntax

```
Editor_Instance.GetAllElements();
```

return

Type: Array [Element]

사용자가 입력한 MouseEvent 함수 정보를 반환

1.37. GetCurrentElement(tagName)

GetCurrentElement(tagName)

현재 Cursor 또는 Selection 위치의 Element를 가져온다.
(※ Selection 상태일 경우에는 복수의 Element List로 반환한다.)

Syntax

```
Editor_Instance.GetCurrentElement(tagName);
```

Parameter

tagName [option]

- Type : String
- description : 현재 Cursor 또는 Selection에 해당하는 Element를 반환할 tagName

return

Type: null or Element or Array[Element]

elementID에 해당하는 Element를 반환한다. (Element가 존재하지 않을 시 null 반환)

1.38. GetElement(elementID)

GetElement(elementID)

Element의 ID가 Parameter로 전달된 elementID에 해당하는 Element를 찾는다.

Syntax

```
Editor_Instance.GetElement("elementID");
```

Parameter

elementID

- Type : String
- description : Element의 ID에 해당하는 String형의 값.

return

Type: Element or null

elementID에 해당하는 Element를 반환한다. (Element가 존재하지 않을 시 null 반환)

1.39. SetEditorSize(width, height)

SetEditorSize(width, height)

Editor의 전체 width, height의 사이즈를 변경하는 기능

Syntax

```
Editor_Instance.SetEditorSize("100px", "300px");
```

or

```
Editor_Instance.SetEditorSize(100, 300);
```

Parameter

width

- Type : String or Number
- description : Editor 전체의 Width를 지정하는 String 또는 Number 형의 값.

height

- Type : String or Number
- description : Editor 전체의 height를 지정하는 String 또는 Number 형의 값.

return

1.40. SetEditorWidth(width)

SetEditorWidth(width)

Editor의 width의 사이즈만 변경하는 기능

Syntax

```
Editor_Instance.SetEditorWidth("100px"); // 또는 SetEditorWidth(100);
```

Parameter

width

- Type : String or Number
- description : Editor 전체의 Width를 지정하는 String 또는 Number 형의 값

return

1.41. SetEditorHeight(height)

SetEditorHeight(height)

Editor의 height의 사이즈만 변경하는 기능

Syntax

```
Editor_Instance.SetEditorHeight("300px"); // 또는 SetEditorHeight(300);
```

Parameter

height

- Type : String or Number
- description : Editor 전체의 height를 지정하는 String 또는 Number 형의 값

return

1.42. GetEditorSize()

GetEditorSize()

Editor의 width, height 사이즈를 가져온다.

Syntax

```
var editorSize = Editor_Instance.GetEditorSize();
```

```
console.log( editorSize.width, editorSize.height );
```

return

Type: Object { width, height }

Editor의 width와 height의 정보를 Object 형태 Wrapping하여 반환한다.

1.43. SetEditMode(modeType)

SetEditMode(modeType)

Editor의 편집Mode(Editor, HTML, TEXT, Preview)를 설정하는 기능

Syntax

```
Editor_Instance.SetEditMode(0);
```

```
// ModeType ( 0 : Editor, 1 : HTML, 2 : TEXT, 3 : Preview )
```

Parameter

modeType

- Type : Number
- description : Editor의 편집 Mode를 설정하는 Number형의 값.
(0 : Editor, 1 : HTML, 2 : TEXT, 3 : Preview)

Return

1.44. GetEditorContent(isWrapping, isBackground)

GetEditorContent(isWrapping, isBackground)

Content 영역내에 입력되어 있는 HTML의 정보를 가져오는 기능
(※ Content란 Editor를 편집하는 영역을 의미한다.)

Syntax

```
Editor_Instance.GetEditorContent(/*option*/ true , /*option*/ true);  
//parameter true시 DIV Wrapping / 배경이미지 포함 HTML을 반환
```

Parameter

isWrapping

- Type : Boolean
- description : Export HTML를 DIV 감싸서 내려주는 옵션

isBackground

- Type : Boolean
- description : Export HTML에 Background이미지를 포함시키는 옵션

return

Type: String

Content영역에 편집되어 있는 콘텐츠를 HTML Code형식의 String으로 반환한다.

1.45. GetEditorTextContent()

GetEditorTextContent()

Content 영역내에 입력되어 있는 콘텐츠를 Text로 가져오는 기능
(※ Content란 Editor를 편집하는 영역을 의미한다.)

Syntax

```
Editor_Instance.GetEditorTextContent();
```

return

Type: String

Content영역에 편집되어 있는 콘텐츠를 Plain Text형식의 String으로 반환한다.

※ 객체형식(Table, Image, Hyperlink 등)의 내용은 제거되어 순수 Text로만 반환한다.

1.46. GetContentHTMLFile()

GetContentHTMLFile()

Content 영역내에 입력되어 있는 HTML의 정보를 가져오는 기능
(※ Content란 Editor를 편집하는 영역을 의미한다.)

Syntax

```
Editor_Instance.GetContentHTMLFile();
```

return

Type: String

Content영역에 편집되어 있는 콘텐츠를 DocType, Language, Title, Encoding 정보를 포함하는 HTML Code형식의 String으로 반환한다. (HTML 파일 형태의 내용으로 반환)

1.47. GetAttribute(element, AttributeName)

GetAttribute(element, AttributeName)

Parameter로 전달되어지는 Element의 AttributeName의 Value값을 찾는 기능

Syntax

```
Editor_Instance.GetAttribute(element, "id");
```

Parameter

element

- Type : Element
- description : attribute의 value를 찾을 대상이 되는 Element.

AttributeName

- Type : String
- description : element에서 찾을 attribute name 정보 값

Return

Type: String or null

Attribute name에 해당하는 value 값을 반환한다. (attribute name이 없을 경우 null)

1.48. SetAttribute(element, AttributeName, AttributeValue)

SetAttribute(element, AttributeName, AttributeValue)

Parameter로 전달되어지는 Element를 대상으로 AttributeName 속성에 AttributeValue값을 지정하는 기능

Syntax

```
Editor_Instance.SetAttribute(element, 'id', 'YooncomsWebeditor');
```

Parameter

element

- Type : Element
- description : attribute를 변경할 대상이 되는 Element

AttributeName

- Type : String
- description : element에 지정할 attribute name 정보 값

AttributeValue

- Type : String
- description : element에 지정할 attribute Value 값

Return

Type: Boolean (true : 성공, false : 실패)

Element에 Attribute정보가 정상적으로 적용되었는지에 대한 여부 반환

1.49. SetEditorContent(htmlString)

SetEditorContent(htmlString)

Content 영역에 HTML Code를 삽입하는 기능

Syntax

```
Editor_Instance.SetEditorContent("<div> <span>Hello</span> </div>");
```

Parameter

htmlString

- Type : String
- description : Content영역에 삽입할 HTML Code 형식의 String 값.

Return

Type: Boolean (true : 성공, false : 실패) : 성공유무 정보 반환

1.50. SetEditorTextContent(textString)

SetEditorTextContent(textString)

Content 영역에 Plain Text를 삽입하는 기능

Syntax

```
Editor_Instance.SetEditorTextContent("Hello, Yooncoms WebEditor!");
```

Parameter

htmlString

- Type : String
- description : Content영역에 삽입할 Plain Text 형식의 String 값.

Return

Type: Boolean (true : 성공, false : 실패) : 성공유무 정보 반환

1.51. SetFocus(element)

SetFocus(element)

Parameter로 전달된 Element에 Focus를 지정하는 기능

Syntax

```
Editor_Instance.SetFocus(element);
```

Parameter

element

- Type : Element
- description : Focus를 지정할 Element.

Return

1.52. IsExistsElement(elementID)

IsExistsElement(elementID)

Parameter로 전달된 elementID를 가지는 Element의 존재 여부를 확인하는 기능

Syntax

```
Editor_Instance.IsExistsElement("editorSampleID");
```

Parameter

elementID

- Type : String
- description : Element를 찾기 위한 Element의 ID에 해당하는 String 값

Return

Type: Boolean (true : 존재, false : 미존재)

ID에 해당하는 Element가 존재하는지에 대한 여부 반환

1.53. GetAttributeValueByBody(attributeName)

GetAttributeValueByBody(attributeName)

Body(Editor)에 존재하는 Attribute 중 Parameter로 전달된 name에 해당하는 Value값을 찾는 기능

Syntax

```
Editor_Instance.GetAttributeValueByBody("id");
```

Parameter

attributeName

- Type : String
- description : Body(Editor)의 Attribute를 찾을 name에 대한 String 값

Return

Type: String

Body(Editor)에 지정된 Attribute Name이 존재할 경우 Attribute Value값을 반환

1.54. SetAttributeValueByBody(attributeName, attributeValue)

SetAttributeValueByBody(attributeName, attributeValue)

Body(Editor)에 Attribute name에 Value값을 지정하는 기능

Syntax

```
Editor_Instance.SetAttributeValueByBody("id", "YooncomsWebeditor");
```

Parameter

attributeName

- Type : String
- description : Body(Editor)의 Attribute를 찾을 name에 대한 String 값

attributeValue

- Type : String
- description : Body(Editor)의 Attribute Name에 지정할 Value 대한 String 값

Return

Type: Boolean (true : 성공, false : 실패)

Body(Editor)에 지정된 Attribute Name에 Attribute Value값 지정 성공여부 반환

1.55. IsCellLockByID(elementID)

IsCellLockByID(elementID)

지정된 ID의 Cell에 쓰기금지(Lock) 적용 여부에 대한 결과를 가져온다.
(※ 쓰기금지에 대한 Attribute 기준은 free Attribute가 없으면 Lock, 있으면 쓰기가능)

Syntax

```
Editor_Instance.IsCellLockByID('cell');
```

Parameter

elementID

- Type : String
- description : 지정할 Cell의 ID 값.

Return

Type: Boolean (true : 쓰기금지, false : 쓰기가능)

지정된 elementID에 해당하는 Cell에 free Attribute가 존재하면 쓰기가능(false) 없으면 쓰기금지(true)를 반환

1.56. SetCellLockByID(elementID, isLock)

SetCellLockByID(elementID, isLock)

지정된 ID의 Cell에 쓰기금지(Lock) 적용

(※ 쓰기금지시 free Attribute제거, 쓰기가능시 free Attribute 삽입)

Syntax

```
Editor_Instance.SetCellLockByID('cell', true);
```

Parameter

elementID

- Type : String
- description : Lock의 여부를 설정할 Cell의 ID 값.

isLock

- Type : Boolean
- description : Lock의 여부 (true : Lock 설정, false : Lock 해제).

Return

Type : Boolean (true : 설정 성공, false : 설정 실패)

Lock 설정여부에 대해 성공여부를 반환한다.

1.57. IsCellLockByFocus()

IsCellLockByFocus()

선택한 Focus에 쓰기금지 적용이 되어있는지에 대한 여부 반환
(※ 쓰기금지에 대한 Attribute 기준은 free Attribute가 없으면 Lock, 있으면 쓰기가능)

Syntax

```
Editor_Instance.IsCellLockByFocus()
```

Return

Type: Boolean (true : 쓰기금지, false : 쓰기가능)

지정된 Focus에 해당하는 Cell에 free Attribute가 존재하면 쓰기가능(false) 없으면 쓰기금지(true)를 반환

1.58. SetCellLockByFocus(isLock)

SetCellLockByFocus(isLock)

지정된 Focus의 Cell에 쓰기금지(Lock) 적용
(※ 쓰기금지시 free Attribute제거, 쓰기가능시 free Attribute 삽입)

Syntax

```
Editor_Instance.SetCellLockByFocus(true);
```

Parameter

isLock

- Type : Boolean
- description : Lock의 여부 (true : Lock 설정, false : Lock 해제)

Return

Type : Boolean (true : 설정 성공, false : 설정 실패)

Lock 설정여부에 대해 성공여부를 반환한다.

1.59. GetAttributeValueByID(elementID, attributeName)

GetAttributeValueByID(elementID, attributeName)

elementID를 가지는 Element의 Attribute 중 Parameter로 전달된 name에 해당하는 Value값을 찾는 기능

Syntax

```
Editor_Instance.GetAttributeValueByID("elementID", "class");
```

Parameter

elementID

- Type : String
- description : elementID를 ID로 가지고 있는 Element를 찾기 위한 String 값

attributeName

- Type : String
- description : Body(Editor)의 Attribute를 찾을 name에 대한 String 값

Return

Type: String

elementID에 해당하는 Element의 Attribute Name이 존재할 경우 Attribute Value값 반환

1.60. SetAttributeValueByID(elementID, attributeName, attributeValue)

SetAttributeValueByID(elementID, attributeName, attributeValue)

elementID를 ID로 가지고 있는 Element의 Attribute name에 Value값을 지정하는 기능

Syntax

```
Editor_Instance.SetAttributeValueByID("elementID", "class", "blue_theme");
```

Parameter

elementID

- Type : String
- description : elementID를 ID로 가지고 있는 Element를 찾기 위한 String 값

attributeName

- Type : String
- description : Body(Editor)의 Attribute를 찾을 name에 대한 String 값

attributeValue

- Type : String
- description : Body(Editor)의 Attribute Name에 지정할 Value 대한 String 값

Return

Type: Boolean (true : 성공, false : 실패)

elementID로 지정된 Element의 Attribute Name에 Attribute Value값 지정 성공여부 반환

1.61. GetHtmlById(elementID)

GetHtmlById(elementID)

elementID를 가지는 Element의 innerHTML을 가져오는 기능

Syntax

```
Editor_Instance.GetHtmlById(elementID);
```

Parameter

elementID

- Type : String
- description : elementID를 ID로 가지고 있는 Element를 찾기 위한 String 값

Return

Type: String

elementID에 해당하는 Element의 InnerHTML 정보 반환

1.62. SetHtmlById(elementID, htmlString)

SetHtmlById(elementID, htmlString)

elementID를 가지는 Element에 HTML Code를 삽입하는 기능

Syntax

```
Editor_Instance.SetHtmlById("elementID", "<div><span>hello</span></div>");
```

Parameter

elementID

- Type : String
- description : elementID를 ID로 가지고 있는 Element를 찾기 위한 String 값

htmlString

- Type : String
- description : elementID를 가지는 Element에 삽입할 HTML Code 형식의 String 값.

Return

Type: Boolean (true : 성공, false : 실패)

elementID로 지정된 Element에 HTML 삽입 성공여부 반환

1.63. GetTextByID(elementID)

GetTextByID(elementID)

elementID를 가지는 Element의 Plain Text 값을 가져오는 기능

Syntax

```
Editor_Instance.GetTextByID(elementID);
```

Parameter

elementID

- Type : String
- description : elementID를 ID로 가지고 있는 Element를 찾기 위한 String 값

Return

Type: String

elementID에 해당하는 Element의 Plain Text값 반환

1.64. SetTextById(elementID, textValue)

SetTextById (elementID, textValue)

elementID를 가지는 Element에 Plain Text 값을 삽입하는 기능

Syntax

```
Editor_Instance.SetTextById("elementID", "Hello,YooncomsWebeditor");
```

Parameter

elementID

- Type : String
- description : elementID를 ID로 가지고 있는 Element를 찾기 위한 String 값

SetTextById

- Type : String
- description : elementID를 가지는 Element에 삽입할 Plain Text 형식의 String 값.

Return

Type: Boolean (true : 성공, false : 실패)

elementID로 지정된 Element에 Plain Text 삽입 성공여부 반환

1.65. GetAttributeValueByFocus(attributeName, tagName)

GetAttributeValueByFocus(attributeName, tagName)

선택한 Element의 AttValue 가져오는 기능

(※ 단 tagName은 Optional한 Parameter이며, 인자로 전달하지 않을시에는 TD, TH의 Cell Type Element를 기본으로 탐색한다.)

Syntax

```
Editor_Instance.GetAttributeValueByFocus("id");
```

Parameter

attributeName

- Type : String
- description : 현재 선택되어진 element의 attribute Value를 찾기 위한 Attribute Name 값

tagName [optional]

- Type : String
- description : 현재 Cursor 또는 Selection에 해당하는 Element를 반환할 tagName

Return

Type: String

현재 선택되어진 Element의 Attribute Name에 해당하는 Value값을 반환

1.66. SetAttributeValueByFocus(attrName, attrValue, tagName)

SetAttributeValueByFocus(attrName, attrValue, tagName)

선택한 Element의 AttValue 삽입하는 기능

(※ 단 tagName은 Optional한 Parameter이며, 인자로 전달하지 않을시에는 TD, TH의 Cell Type Element를 기본으로 탐색한다.)

Syntax

```
Editor_Instance.SetAttributeValueByFocus("id", 'sample');
```

Parameter

attributeName

- Type : String
- description : 현재 선택되어진 element의 attribute Value를 찾기 위한 Attribute Name 값

AttributeValue

- Type : String
- description : Attribute Name에 적용할 attribute Value 값

tagName [option]

- Type : String
- description : 현재 Cursor 또는 Selection에 해당하는 Element를 반환할 tagName

Return

Type: Boolean (성공:true, 실패 : false)

현재 선택되어진 Element의 Attribute Name에 Value값을 삽입 성공 여부 반환

1.67. getPublicPathURL()

getPublicPathURL()

Image 등과 같은 Public(Static) Resource를 사용시 사용하는 리소스의 웹경로의 설정 주소 정보를 get한다.

Syntax

```
Editor_Instance.getPublicPathURL();
```

return

Type: String

Editor의 Public Resource의 위치 웹경로에 대한 URL 정보를 반환

1.68. setPublicPathURL(publicPathURL)

setPublicPathURL(publicPathURL)

Image 등과 같은 Public(Static) Resource를 사용시 사용하는 리소스의 웹경로의 설정 주소 정보를 set한다.

Syntax

//예)현재 웹 주소의 경로에서부터 접근 가능한 Public 리소스에 대한 웹 경로

```
Editor_Instance.setPublicPathURL("./");
```

Parameter

helpURL

- Type : String
- description : Public Path로 사용할 시작 경로를 설정 한다.

return