

# getUploadedFiles

RELEASE 2.2.1 OR ABOVE

Return the path of uploaded files by their file types (path) and whether they are deleted (isDeleted) in Object form.

Parameters:

Name	Type	Description
fileType	string	File type whose information shall be imported. ("image", "video", "file")

Return:

Type	Attribute	Description
Array	<pre>[   { path: "/upload/path/filename.png", isDeleted: false },   { path: "/upload/path/filename2.png", isDeleted: true },   .... ]</pre>	path and isDeleted data

Example:

```
var filesData = editor.getUploadedFiles();
```

*getUploadedFiles()*

RELEASE 2.2.1 OR ABOVE

Return the path of uploaded files by their file types (path) and whether they are deleted (isDeleted) in Object form.

Parameters:

Name	Type	Description
fileType	String	When file type information is not entered

Return:

Type	Attribute	Description
------	-----------	-------------

Object	<pre>{   image: [     { path: "/upload/path/filename.png", isDeleted: false },     { path: "/upload/path/filename2.png", isDeleted: true },     ....   ],   video: [],   file: [] }</pre>	path and isDeleted data
--------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------

#### Example:

```
var filesData = editor.getUploadedFiles();
```

## Example

### 1. Limiting the Number of Image Upload

You may limit the maximum number of image upload using API related to the file upload provided by Synap Editor.

```
var editorId = 'synapEditor';
var editorConfig = {};
var html = '';
var editor = new SynapEditor(editorId, editorConfig, html);

editor.setEventListener('afterUploadImage', function (e) {
  var fileType = e.fileType;
  var uploadPath = e.path;
  e.editor.addUploadPath(fileType, uploadPath); // You must use this to check the file uploaded through
editor.getUploadedFiles()
});

editor.setEventListener('beforeUploadImage', function (e) {
  var fileType = e.fileType;
  var uploadCount = e.uploadCount; //Number of files being uploaded (the number of files to be uploaded
whose upload is not completed yet)
  var imageCount = 0;
  e.editor.getUploadedFiles(fileType).forEach(function (info) {
    if (!info.isDeleted) {
      imageCount++; // Count the number of images not deleted
    }
  });
  if (imageCount + uploadCount >= 5) { // Allow up to 5 uploads
    e.returnValue = false; // Cancel the upload by returning false through returnValue
    alert('Five images can be uploaded at most');
  }
});
```

#### Relevant APIs

- [getUploadedFiles](#)
- [addUploadPath](#)