

Custom elements JS API

March 9, 2022 • Jan Cerman • 11 min read

The Custom elements API is a JavaScript API that contains methods needed for implementing custom elements in Kontent by Kentico. The API serves as a tool for communication between a custom element and the Kontent app.

Inclusion in your app

When implementing your custom elements, you need to reference the Custom Elements JS API in your HTML5 app.

JavaScript

For more details, see our tutorial on [Integrating your own content editing features](#).

init method

Use the `init` method to initialize your custom element.

The `init` method expects a callback function which takes `context` and `data` as parameters. This callback function is called once your custom element is registered within Kontent.

JavaScript

Parameter	Type	Description
callback	function	A callback function that receives the <code>context</code> and <code>data</code> objects which provide information from Kontent about the custom element and about where it is displayed in the UI.

Context object

When using the custom element in a content component, the `context` object refers to the parent content item of the content component, not the component itself. It's currently not possible to retrieve information about the component or access its other elements.

Property	Type	Description
projectId	string	The project's ID.
item.id	string	The ID of the content item that contains the custom element.
item.codename	string	The codename of the content item that contains the custom element.
item.collection	reference object	The object represents an identifier of the item's collection, for example <code>{ collection: '...' }</code>
item.name	string	The display name of the content item that contains the custom element
variant.id	string	The language's ID.
variant.codename	string	The language's codename, for example, <code>en-us</code> .

Element object

Property	Type	Description
value	string, null	The custom element's initial value.
disabled	boolean	Indicates whether an element is disabled for editing.
config	object, null	Parameters of the custom element specified as a JSON object in the UI (in a content type or a content type snippet)

setValue method

Use the `setValue` method to set a custom element value. When the value is set to `''`, the element is considered empty.

If you want the custom element content to be searchable in the content items list, specify the keywords or the plain text using [value object](#).

If a custom element is set as required in the UI, Kontent checks its value. Any value other than `''` is considered a valid value and won't generate a validation error message.

Custom elements can contain up to 100,000 characters. If an element exceeds this number, Kontent will display an error.

JavaScript

Parameter	Type	Description
value	string, null, object	Custom element value to be set.

ValueObject object

Parameter	Type	Description
valueObject.value	string, null	Custom element value to be set.
valueObject.searchableValue	string, null	Plain text searchable value used for searching in the content items list. Do not use any markup or JSON here to allow a correct search.

onDisabledChanged method

Use the `onDisabledChanged` method to define what should happen with a custom element when it's disabled for editing. The method takes a callback function.

When disabled, the element is in read-only mode. This means that the element is shown in a content item but cannot be changed.

For example, custom elements can be disabled in these cases:

- The content item is published.
- User is viewing an older version of a content item.
- User doesn't have editing permissions.

JavaScript

Parameter	Type	Description
callback	function	A callback function that receives a boolean indicating whether the element is disabled for editing. <pre> function callback(disabled) { // ... } </pre>

setHeight method

Use the `setHeight` method to set a custom height of a custom element. The value will be set in pixels. See [specifying the height of a custom element](#) when creating a custom element for more context.

JavaScript

Parameter	Type	Description
height	number	Custom element height to be set in the Kontent UI. Height must be a positive integer.

getElementValue method

Use the `getElementValue` method to retrieve the value of a different element. You can retrieve another element's value even if the custom element is disabled.

The `getElementValue` method works on elements that are:

- Used in the same [content group](#) as the custom element itself.
- Specified in the [custom element's configuration](#), specifically, in the *Allow the custom element to read values of specific elements* field.

If you need to get values of multiple elements, call the `getElementValue` method for each such element. If you need to get the value multiple times, call the method multiple times.

JavaScript

Parameter	Type	Description
elementCodename	string	Codename of the content element from which the value should be read.
callback	function	A callback function that receives the value of the specified element. The data type of the value varies based on the type of the content element . <pre> function callback(value) { // ... } </pre>

Data type of content elements

- Custom element – string
- Date & Time – [ISO-8601](#) formatted string
- Multiple choice – array of selected options. Each option is returned as an object based on the following schema: `{}`.
- Text – string
- URL slug – string

Other content elements are currently not supported.

observeElementChanges method

Use the `observeElementChanges` method to receive notifications about changes in other elements.

The `observeElementChanges` method can listen to any element changes and does not depend on the list of elements allowed for reading.

The `observeElementChanges` method does not return a specific element value but only notifies that the change occurred. The method takes a callback function as a second parameter. Use this method together with the `getElementValue` method to keep updated information about an element value.

JavaScript

Parameter	Type	Description
elementCodenames	array of strings	Array of codenames of the content elements to observe. When an empty array is provided, all elements are observed.
callback	function	A callback function that receives codenames of changed elements as an array of strings. <pre>function (codenames) { // ... }</pre>

observeItemChanges method

Use the `observeItemChanges` method to receive notifications about changes in the item.

The `observeItemChanges` method can listen to changes of the item's attributes – its name, codename, or collection.

The method only notifies about the change, it doesn't return the new value of the changed attributes. It takes a callback function that is called whenever any of the attributes is changed. The callback function is called

with a parameter that contains the item's new details.

JavaScript

Parameter	Type	Description
callback	function	A callback function that receives object with current item data. <pre> function callback(item) { // ... } </pre>

ItemChangedDetails object

Property	Type	Description
item.codename	string	The codename of the content item that contains the custom element.
item.collection	reference object	The object represents an identifier of the item's collection, for example <code>{ id: '...', name: '...' }</code> .
item.name	string	The display name of the content item that contains the custom element.

selectAssets method

Use the `selectAssets` method to enable the custom element to select assets from Kontent.

The method takes a `config` object and, if a user selects assets from the dialog, returns a promise with an array of [asset references](#) containing the selected asset's ID. If the selection dialog is closed, the method returns `undefined`.

JavaScript

Config object

Property	Type	Description
allowMultiple	boolean	Specifies whether multiple assets can be selected.
fileType	string	Specifies what type of images can be selected; allowed values are <code>'image'</code> and <code>'video'</code> .

AssetReference object

The object containing an ID reference to the selected asset.

Property	Type	Description
id	GUID	The asset's internal ID.

getAssetDetails method

Use the `getAssetDetails` method to get metadata about a specific asset. After getting IDs with the `getAssetIds` method, you can then get more information about each of the assets returned.

The `getAssetDetails` method takes an array of asset IDs as strings and returns a promise with an array of [asset objects](#). The method returns a `Promise` if the specified items don't exist.

JavaScript

Parameter	Type	Description
assetIds	array of GUIDs	The internal IDs of the assets to be returned.

Asset object

Property	Type	Description
id	GUID	The asset's internal ID.
descriptions	array of asset description objects	The descriptions of the asset in all available languages.
fileName	string	The name of the file.
imageHeight	integer	The asset's height in pixels.
imageWidth	integer	The asset's width in pixels.
name	string	The asset's display name.
size	integer	The size of the asset in bytes.
thumbnailUrl	string	The URL for the thumbnail of the asset.
title	string	The asset's title.
type	string	The type of the file.
url	string	The URL of the asset.

Asset description object

Property	Type	Description
language.id	GUID	The description's language ID.
language.codename	string	The description's language codename.
description	string	The asset's description for a specific language.

selectItems method

Use the `selectItems` method to enable the custom element to select content items from Kontent.

The method takes a `selectItemsOptions` object and, if a user selects content items from the dialog, returns a promise with an array of [content item references](#) containing the selected item's ID. If the selection dialog is cancelled, the method returns `undefined`.

JavaScript

Config object

Property	Type	Description
allowMultiple	boolean	Specifies whether multiple items can be selected.

ContentItemReference object

Property	Type	Description
id	GUID	The content item's internal ID.

getItemDetails method

Use the `getItemDetails` method to get details about a specific content item. After getting IDs with the `getContentItemIds` method, you can then get details about each of the items returned.

The `getItemDetails` method takes an array of content item IDs as strings and returns a promise with an array of [content item objects](#). The method returns a `Promise` if the specified items don't exist.

JavaScript

Parameter	Type	Description
itemIds	array of GUIDs	The internal IDs of the items to be returned.

ContentItem object

Property	Type	Description
id	GUID	The item's internal ID.
codename	string	The item's codename.
name	string	The item's display name.
collection	reference object	The item's collection
type	reference object	The item's type.