Content property | WeWeb Developer Docs

|  |  |
| --- | --- |
| URL source | <https://developer.weweb.io/api/content-property.html> |
| Date de scraping | 2025-04-08T15:52:33.942860+00:00 |

## Content property ​

Each content property is describe with an object which has the following properties.

## property.label ​

Type : translated text Mandatory Available for Section and Element Description : Label used by the editor on the edit panel interface

## property.section ​

Type : style or settings (default is style ) Available for Section and Element Description : Determine in which panel's tab the property will be display

## property.defaultValue ​

Type : any Available for Section and Element Description : This the default value of the property if empty.

## property.options ​

Type : Object or an EditorFunction returning an object Available for Section and Element Description : List of options pass to the ux component used in the edit panel.  
See available options (depending on the type) here

## property.type ​

Type : string Mandatory if your property is not hidden Available for Section and Element Description : The type of your property. It is used to know which user interface display to edit your property inside edit panel.  
A complete list of available types are availabe here

## property.multiLang ​

Type : boolean (default false) Available for Section and Element Description : Will treat your property as a translated object See also Add a translated text property

## property.responsive ​

Type : boolean (default false) Available for Section and Element Description : Indicate that your property can be different depending on the current screen See also Add a responsive property

## property.states ​

Type : boolean (default false) Available for Element only Description : Indicate that your property can be different depending on the current state (hover for example, or a custom one) See also Use element state

## property.bindable ​

Type : boolean | list | repeatable (default false) Available for Section and Element Description : Indicate that your property can be binded. The type of binding depend on the value true : bind to a primitive list : bind to an array repeatable : special binding to connect to a dropzone and repeat content See also Add a bindable property and Add a dropzone

## property.editorOnly ​

Type : boolean (default false) Available for Section and Element Description : Indicate if this property is only needed in editor. The data will not be saved inside content, and can be retrieve and modified in an other way. This is usefull for adding some editing logic See also Custom editor interface

## property.hidden ​

Type : boolean or an EditorFunction returning a boolean (default false) Available for Section and Element Description : Indicate if this property should be hidden from the EditorPanel. This is usefull for dropzone or if you provide yourself a custom way to edit this property See also Custom editor interface

## Editor Function ​

Some options of a content property can take a function rather than static value. This enable a better UX experience by making the sidepanel dynamic. The arguments of this function are:  
content : the current content of the component sidepanelContent : the current editor only content of the component boundsProps : an object with property name as prop, and true as value if the props is currently bound

## Available Types ​

## Array ​

### Code

Langage: unknown

{  
 type: "Array",  
 label: { "en": "List of items" },  
 options: {  
 item: {  
 [...] // a content property  
 },  
 fixed: false // if true, do not display add and remove button  
 }  
},

## BigButtons ​

Used to display a button on the edit panel.

### Code

Langage: unknown

{  
 label: { en: 'Presets' },  
 type: 'BigButtons',  
 options: {  
 action: 'setPreset',  
 values: [  
 { icon: 'two-columns-mosaic', id: 'two-columns-mosaic' },  
 { icon: 'three-columns-mosaic', id: 'three-columns-mosaic' },  
 { icon: 'four-columns-mosaic', id: 'four-columns-mosaic' },  
 { icon: 'five-columns-mosaic', id: 'five-columns-mosaic' },  
 ],  
 }  
}

## BigIconRadioGroup ​

### Code

Langage: unknown

{  
 label: { en: 'Display type', fr: "Type d'affichage" },  
 type: 'BigIconRadioGroup',  
 options: [  
 { icon: 'type-columns', value: 'columns', label: 'Columns' }]),  
 { icon: 'type-rows', value: 'rows', label: 'Rows' },  
 { icon: 'type-mosaic', value: 'mosaic', label: 'Mosaic' },  
 ],  
 defaultValue: 'columns',  
}

## Button ​

Used to display a button on the edit panel.

### Code

Langage: unknown

{  
 "type": "Button",  
 "label": null,  
 "options": {  
 // Translated text to be displayed inside button  
 "text": {  
 "en": "Click me"  
 },  
 "action": "display" // name of one of your component methods  
 }  
},

## Color ​

Return a string

### Code

Langage: unknown

{  
 "type": "Color",  
 "label": {  
 "en": "Text color",  
 "fr": "Couleur du texte"  
 },  
 "options": {  
 "nullable": true, // boolean (default: false). If true color can be null  
 "gradient": true // boolean (default: false). If true can return gradient string  
 }  
},

## CssFilter ​

Return a string

### Code

Langage: unknown

{  
 "type": "CssFilter",  
 "label": {  
 "en": "Filters",  
 "fr": "Filtres"  
 },  
},

## FontFamily ​

Return a string

### Code

Langage: unknown

{  
 "type": "FontFamily",  
 "label": {  
 "en": "Font Family",  
 },  
},

## Icon ​

Return a string

### Code

Langage: unknown

{  
 "type": "Icon",  
 "label": {  
 "en": "Icon",  
 },  
},

## Image: ​

Return a string representing an image path

### Code

Langage: unknown

{  
 "type": "Image",  
 "label": {  
 "en": "Image",  
 "fr": "Image"  
 },  
},

## Info: ​

### Code

Langage: unknown

{  
 label: {  
 en: 'Text',  
 fr: 'Texte',  
 },  
 type: 'Info',  
 options: {  
 text: { en: 'Click text to edit' },  
 },  
},

## Length ​

Return a string representing a length (value + unit)

### Code

Langage: unknown

{  
 "type": "Length",  
 "label": {  
 "en": "Size",  
 "fr": "Taille"  
 },  
 "options": {  
 "unitChoices": [  
 { "value": "px", "label": "px", "min": 1, "max": 1000 },  
 { "value": "%", "label": "%", "min": 1, "max": 100 }  
 ]  
 }  
}

## Number ​

Return a Number

### Code

Langage: unknown

{  
 "type": "Number",  
 "label": {  
 "en": "Number of columns",  
 "fr": "Nombres de colonnes"  
 },  
 "options": {  
 "min": 1, // default 0  
 "max": 10, // default 100,  
 "step": 1 // default 1  
 }  
}

## Object ​

This is usefull if you want nested properties. We still encourage you to use as flat as possible content.

### Code

Langage: unknown

{  
 type: "Object",  
 label: { "en": "Parameters" },  
 options: {  
 item: {  
 {  
 prop1: [...] // Content property  
 prop2: [...] // Content property  
 }  
 },  
 }  
}

## OnOff ​

Return a Boolean

### Code

Langage: unknown

{  
 "type": "OnOff",  
 "label": {  
 "en": "Use icon?",  
 "fr": "Utiliser un icône?"  
 }  
}

## Script ​

Return a string

### Code

Langage: unknown

{  
 "type": "Script",  
 "label": {  
 "en": "Custom js"  
 }  
}

## Section ​

Return a section id

### Code

Langage: unknown

{  
 "type": "Section",  
 "label": {  
 "en": "Section"  
 }  
}

## Shadows ​

Return a string

### Code

Langage: unknown

{  
 "type": "Shadows",  
 "label": {  
 "en": "Shadows",  
 "fr": "Ombres"  
 },  
 "options": {  
 "isText": true // default: false. If true return text-shadow css compatible value  
 }  
}

## Text ​

Return a string

### Code

Langage: unknown

{  
 "label": { "en": "Alt", "fr": "Alt" },  
 "type": "Text",  
 "options": {  
 "placeholder": "Image description" // all input[type = text] options are supported  
 }  
}

## Textarea ​

Return a string

### Code

Langage: unknown

{  
 "label": { "en": "Description", "fr": "Description" },  
 "type": "Textarea",  
 "options": {  
 "placeholder": "Image description" // all textarea options are supported  
 }  
}

## TextRadioGroup ​

Return type of value

### Code

Langage: unknown

{  
 "type": "TextRadioGroup",  
 "label": {  
 "en": "Alignement",  
 "fr": "Alignement"  
 }  
 "options": {  
 "choices": [  
 {  
 "value": "left",  
 "default": true,  
 "title": { "en": "Left", "fr": "Gauche" },  
 "icon": "text-align-left"  
 },  
 { "value": "center", "title": { "en": "Center", "fr": "Milieu" }, "icon": "text-align-center" },  
 { "value": "right", "title": { "en": "Right", "fr": "Droite" }, "icon": "text-align-right" },  
 {  
 "value": "justify",  
 "title": { "en": "Justify", "fr": "Justifié" },  
 "icon": "text-align-justify"  
 }  
 ]  
 }  
}

## TextSelect ​

### Code

Langage: unknown

{  
 "label": {  
 "en": "Font weight",  
 "fr": "Graisse"  
 },  
 "type": "TextSelect",  
 "options": {  
 "options": [  
 { "value": null, "label": { "en": "Default", "fr": "Par défaut" } },  
 { "value": 100, "label": { "en": "100 - Thin" } },  
 { "value": 200, "label": { "en": "200 - Extra Light" } },  
 { "value": 300, "label": { "en": "300 - Light" } },  
 { "value": 400, "label": { "en": "400 - Normal" } },  
 { "value": 500, "label": { "en": "500 - Medium" } },  
 { "value": 600, "label": { "en": "600 - Semi Bold" } },  
 { "value": 700, "label": { "en": "700 - Bold" } },  
 { "value": 800, "label": { "en": "800 - Extra Bold" } },  
 { "value": 900, "label": { "en": "900 - Black" } }  
 ]  
 }  
}

## Video ​

Return a string

### Code

Langage: unknown

{  
 "label": { "en": "Video url", "fr": "Video url" },  
 "type": "Video"  
}