Email Templating Reference

The Kibo eCommerce Core theme is the foundation upon which you can build your own Kibo eCommerce theme. You can view the latest Core theme files on Github.

Hypr and HyprLive Syntax

Hypr and HyprLive share many syntax features, but differ in a few areas. Also, not all tags or filters are supported on both Hypr and HyprLive.

Syntax Commonalities Between Hypr and HyprLive

Syntax Feature	Example
Tag delimiters	`{% %}`
Expression (variable, object lookup, literal, with any number of filters) delimiters	`{{ }}`
Pipe separator for filters	`{{ value|filter|filter2 }}`
Filter arguments enclosed in parentheses, separated by commas	`{{ value|filter(argument1, argument2) }}`
Tag arguments are key-value pairs, where the key is a simple string and the value (if needed) is any expression that could be placed within `{{ }}` delimiters (but without those delimiters)	`{% tag "key" expression %}`
Single-line comment delimiters	`{# #}`
Multi-line comment delimiters	`{% comment %}` ... `{% endcomment %}`
String literals (double-quotes)	`" "`
Local variables (simple barestrings)	`varname`
Object property dot lookup	`object.property`

Syntax Differences Between Hypr and HyprLive

Hypr and HyprLive support unique syntax for a small number of operations. Because existing themes might use these syntax features, Kibo eCommerce continues to support them, but whenever possible you should use a syntax that both rendering engines support.

- Index lookup
- Filter arguments

Index Lookup

Look up values within an indexed array.

Syntax Feature Recommended	`first` and `last` filters
Supported On	Hypr and HyprLive

`{{ model.items|first }} {# Access the first item in model.items #}`

To cache the result for further use, use a `with` tag.

```
{% with model.items|first as firstitem %}
   {{ firstitem.text }} {# Access the "text" property of the first item #}
{% endwith %}
```

To look up a value at a specific index, use a `for` tag.

```
{% for item in model.items %}
   {% if forloop.index == index %}
      {{ item.text }}
   {% endif %}
{% endfor %}
```

Syntax Feature	Dot notation

Supported On	Hypr

{{ model.items.0 }} {# Access the first item in model.items #}	
Syntax Feature	Brackets
Supported On	HyprLive

{{ model.items[0] }} {# Access the first item in model.items #}	

Filter Arguments

Specify a filter argument.

Syntax Feature Recommended	Parentheses
Supported On	Hypr and HyprLive
{{ bigList|find(777) }}	
Syntax Feature	Colon
Supported On	Hypr
{{ bigList|find:"777" }}	

Supported Tags and Filters

Some tags and filters are supported on both Hypr and HyprLive, only on Hypr, or only on HyprLive.

- View the list of supported tags.
- View the list of supported filters.

Hypr Tags

The following is the complete list of Hypr tags.

- all_scripts
- autoescape
- block
- cms_resources
- comment
- dropzone
- dump
- extends
- filter
- for

- header_content
- if
- include
- include_documents
- include_entities
- include_products
- inline_style
- json_attribute
- make_url

- now
- parent
- partial_cache
- preload_json
- require_script
- set_header
- set_var
- spaceless
- templatetag
- visitor_tracking_pixel
- with

Deprecated Tags
Click here to view a list of deprecated Hypr tags.

all-scripts

Returns an array of scripts marked as required by the `require_script` tag.

Supported On	Hypr

The output of this tag is a quote-delimited, comma-separated array of strings that you pass to the `require` function at or near the end of a top-level Hypr template's rendered HTML content. Through the combination of the `all_scripts` and `require_script` tags, the RequireJS module loader loads all the scripts that a template and its subtemplates need. For example, the `trailing-scripts.hypr` template, included near the end of `page.hypr` (the base Hypr template that most pages on your site extend from), passes the list of all required scripts to RequireJS by calling the `all_scripts` tag in the `require` function:

```
require(['jquery'], function() { require(['modules/common'], function() { require([{% all_scripts %}
]); }); });
```

autoescape

Useful for preventing cross-site scripting attacks, this tag converts special characters in every piece of content between the opening and closing tags into HTML entities, including the content of any subtemplates placed within the tags.

Supported On	Hypr, HyprLive

This is a tag that you must open, `{% autoescape on %}` or `{% autoescape off %}` , and close, `{% endautoescape % }`.

```
{% autoescape on %}
    <div>
    <h2>Header</h2>
    {% block article %}
    {% include "modules/article" %}
    {% endblock article %}
    </div>
{% endautoescape %}
```

To prevent a subset of the enclosed content from being autoescaped, use the `safe` filter.

block

Encloses content that overrides content in parent templates or defines content that child templates can override.

Supported On	Hypr, HyprLive

This is a tag that you must open, `{% block name %}` , and close, `{% endblock name % }`. This tag takes one argument, a name, as a string without quotation marks.

If the name you provide this tag does not match any name for a block in a parent template, you create a section of content that templates extending the current template can override.

If the name you provide this tag matches the name of a block present in a parent template, the current block overrides the content from the parent template. However, you can use the tag `{% parent %}` to output content from the parent block within the current block.

```
{% block product-code %}
{% if themeSettings.listProductCode %}
    <div class="mz-productlisting-productcode">{{model.productCode}}</div>
{% endif %}
{% endblock product-code %}
```

cms-resources

Enables widget and dropzone functionality in the Content Editor.

Supported On	Hypr

Place this tag between HTML `<head></head>` tags.

```
<head>
...
{% cms_resources %}
...
</head>
```

comment

Ignores every line of code between `{% comment %}` and `{% endcomment %}` .

Supported On	Hypr, HyprLive

```
{% comment %}
{% block breadcrumbs %}
    {% if themeSettings.showBreadcrumbs %}{% include "modules/breadcrumbs" %}
    {% endif %}
{% endblock breadcrumbs %}
{% endcomment %}
```

To include a comment within a single line of code, use `{#` and `#}` .

```
{# load a specific page #}{% include "modules/brenda" %}
```

dropzone

Renders an area (dropzone) in the Content Editor where Admin users can drag and drop widgets on a page.

Supported On	Hypr, HyprLive

On your live site, dropzones render `<div>` tags that contain live widget content. You must declare dropzones in areas wider than 320 pixels, which is generally the smallest breakpoint in responsive designs. This tag takes two arguments, `zoneId` and `scope` .

The `zoneId` is required and must be unique across all templates on the site. It is a best practice to name the `zoneId` according to the location of the dropzone, such as *footer-bottom-right.*

The `scope` specifies the context for the dropzone:

- Use "page" scope to target a widget only to the page where the widget is added. If you omit the `scope` parameter, "page" scope is the default value.
- Use "template" scope to target a widget to every page that uses the template to which the widget is added. This may not work on all sites, and it is recommended to use the page scope instead when possible.
- Use "site" scope to target a widget to all pages on the site.

```
{% dropzone zoneId="bodybottom" scope="page" %}
```

dump

Displays a formatted view useful for debugging of all the properties belonging to the argument you specify in the tag.

Supported On	Hypr, HyprLive

Remember to remove `dump` tags from a theme before releasing to a production environment.

```
{% dump product %}
```

extends

Specifies that the current template extends a parent template. Extended (or child) templates can access, display, and modify any content placed within `block` tags in the parent template.

Supported On	Hypr, HyprLive

You can use the `extends` tag in two ways:

`{% extends "page" %}` (with quotation marks) uses the literal value "page" as the filename of the parent template to extend. The tag applies the `.hypr` extension to the filename you provide and assumes the `templates` directory is the root folder for the path.

`{% extends variable %}` uses the value of `variable` as either the name of the parent template to extend (if it evaluates to a string) or as the parent template itself (if it evaluates to a template object).

Place the `extends` tag in the first line of a child template, and place all ensuing content in the child template inside `block` tags. Within the blocks you create in the child template, you can inherit or modify content from the parent template.

```
{% extends "page" %}
{% block body-content %}
…
{% endblock body-content %}
```

filter

Applies a Hypr filter(s) to the contents of the tag.

Supported On	Hypr, HyprLive

This is a tag that you must open, `{% filter filter1|filter2|filter3 … %}` , and close, `{% endfilter %}` . You provide the name of the filters you want to apply as arguments to the opening tag, separated by the pipe symbol (`|`) but without any spaces. Use this tag if you want to run a Hypr filter(s) on the outputs of a Hypr tag, which you can't do using the standard filter syntax.

```
{% filter escape|lower %}

"This text will be HTML-escaped and appear in lowercase characters"

{% endfilter %}
```

for

Loops over each item in an array.

Supported On	Hypr, HyprLive

This is a tag that you must open, `{% for item in list %}` , and close, `{% endfor %}` . The tag takes two arguments: an array to iterate through and a variable name for the items in the array. For example, to access the `name` property for each athlete in the array `athlete_list`:

```
{% for athlete in athlete_list %}
   {{ athlete.name }}
{% endfor %}
```

You can loop over a list in reverse by using `{% for item in list reversed %}` .

The following variables are available within the scope of the `for` tag to assist with operations related to the current iteration of the loop.

Variable	Description
`{{ forloop.counter }}`	The current iteration of the loop (1-indexed).
`{{ forloop.counter0 }}`	The current iteration of the loop (0-indexed).
`{{ forloop.revcounter }}`	The number of iterations from the end of the loop (1-indexed).
`{{ forloop.revcounter0 }}`	The number of iterations from the end of the loop (0-indexed).
`{{ forloop.first }}`	TRUE if the current iteration is the first time through the loop.
`{{ forloop.last }}`	TRUE if the current iteration is the last time through the loop.

header_content

Outputs the contents from the *Additional Header Tags* section of the Page Settings found in the Content Editor, without processing or escaping the content (so HTML tags are expressed as HTML tags).

Supported On	Hypr

In `page.hypr` , which is the base template that most other templates inherit from, the `header_content` tag is located at the end of the `<head></head>` section.

```
<head>
...
{% header_content %}
</head>
```

if

Evaluates the contents in the tag if the argument is true.

Supported On	Hypr, HyprLive

This is a tag you must open, `{% if something %}` and close, `{% endif %}` . You can complement the `if` tag with the `else` tag to provide logic to handle times when the argument is false.

```
{% if fruit_basket %}
   The fruit basket has fruit.
{% else %}
   The fruit basket is empty.
{% endif %}
```

You can also use equality and Boolean operators within `if` tags.

```
{% if fruit == "lucuma" or fruit == "kumquat" %}
   "That's an exotic fruit!"
{% endif %}
```

The operators you can use include:

Operator	Description	Precedence
`.`	dot operator	1
`==`	equal to	2
`!=`	not equal to	3
`<`	less than	4
`>`	greater than	5
`>=`	greater than or equal	6
`<=`	less than or equal	7

Operator	Description	Precedence
not	not operator	8
and	Boolean AND	9
or	Boolean OR	10
		pipe (for filters)	11

The operator precedence indicates which operations take place first. For example, for the statement `{% if items.length > 3 and flag == false or items.price >= 50 %}`, Hypr evaluates the components of the `if` tag in the following order. By the time Hypr evaluates the `or` operator, it has already evaluated the other components needed to determine if the statement is true or false.

1. `{% if items.length > 3 and flag == false or items.price >= 50 %}`
2. `{% if items.length > 3 and flag == false or items.price >= 50 %}`
3. `{% if items.length > 3 and flag == false or items.price >= 50 %}`
4. `{% if items.length > 3 and flag == false or items.price >= 50 %}`
5. `{% if items.length > 3 and flag == false or items.price >= 50 %}`
6. `{% if items.length > 3 and flag == false or items.price >= 50 %}`

include

Loads and renders an external template within the current template.

Supported On	Hypr, HyprLive

The `include` tag enables you to split your code into manageable pieces organized across separate templates.

`{% include "modules/web-fonts-loader" %}` (with quotation marks) uses the literal value `"modules/web-fonts-loader"` as the filename of the template to include. The tag applies the `.hypr` extension to the filename you provide and assumes the `templates` directory is the root folder for the path.

`{% include variable %}` uses the value of `variable` as either the name of the template to include (if it evaluates to a string) or as the template itself (if it evaluates to a template object).

This tag evaluates included templates with the same variables available to the parent template. However, you can pass additional variables to the included template scope manually by using name-value pairs within the tag. You can assign these variables any value of your choosing. For example:

```
{% include "modules/common/address-form" with model=model.billingAddress showAddressType
=false %}
```

In this case, for use within the included template's scope, the `model` variable is assigned the value of the `billingAddress` variable available from the current model and the `showAddressType` variable is assigned the value `false`.

include_documents

Retrieves a document(s) from the CMS using inline API calls and adds the returned document(s) to the page model of the specified template.

Supported On	Hypr

This tag is similar to the `include` tag but instead of setting the model manually or using the same model available to the parent template, the `include_documents` tag adds a collection of documents to the model of the specified template based on the document list, view, and any additional query arguments you specify.

As its first argument, this tag takes the name of a template (or alternatively, the template object itself) to whose model the document(s) should be added. As its second argument, this tag requires the fully qualified name of the document list that contains the document(s) to be returned. As its third argument, this tag requires the name of the view to apply to the document list. The full syntax looks like:

```
{% include_documents "templateName" listFQN="name@namespace" view="viewName" %}
```

You can use the following arguments with this tag.

Argument	Description

Argument	Description
templateName	The template in which to include the document(s). You can specify the template name (in quotes) or the template object itself (without quotes, such as `model.config.template`). If specifying the template name, the tag applies the proper extension to the filename and assumes the template is located in the `templates/pages` directory; if the template is located in a folder other than `pages` , specify the path starting after `templates` (for example, if the template is in the `templates/modules` directory, specify `modules/templateName` as the argument for the template name). **This argument is required and must be in the first position.**
listFQN	The fully qualified name of the document list. **This argument is required and must be in the second position.**
view	The name of the document list view that determines which document properties can be accessed. **This argument is required. If you don't specify a view, Kibo eCommerce assumes the name of the view is *default*. Many document lists do not contain a view named *default*, so it is recommended you always specify a view for this tag; otherwise, the API call may not return any data.**
pageWithUrl	Default: false. Specifies whether to use URL parameters for paging results. When false, you must specify the `startIndex` and `pageSize` parameters as arguments in the tag. If you set this argument to true, you specify the tag to use the existing values for `startIndex` and `pageSize` available from the URL parameters.
sortWithUrl	Default: false. Specifies whether to use URL parameters for sorting results. If you set this argument to true, you specify the tag to sort using the existing value of the `sortBy` URL parameter (either `asc` or `desc`).
startIndex	Default: 0 The item on which to begin listing items in the collection. Kibo eCommerce collections are zero-indexed, so `startIndex=0` starts at the first item in the collection.
pageSize	Default: 15 The maximum number of items to return in the model. **Note:** Setting this number too high may result in poor performance and a cumbersome UI experience.
filter	A filter expression for Kibo eCommerce collections. You can filter documents based on their properties by writing a string expression as your argument. For example: `firstname eq "Brenda"` .
query	A deprecated alias for `filter` . If both the `filter` and `query` arguments are present, `query` takes overrides `filter` .

Argument	Description
sort	Default: null A string representing how to sort the collection. You can sort on any property, date (such as `createDate`), or document name. After specifying the name of the property you are sorting on, include a space followed by " `asc` " or " `desc` " to specify sorting in ascending or descending order.
ids	Default: null A comma-separated list of individual document IDs. If you include this argument, Kibo eCommerce ignores the `filter` argument and the result set consists only of the documents whose IDs you provide.
useActiveDateRange	Default: false. If you set this argument to true, you specify the tag to use the existing active date range information present in the URL.

```
{% comment %}
"Include the last five sales updates added by a CMS business user in the model of the sale-updates
 template."
{% endcomment %}

{% include_documents "modules/sale-updates" listFQN="saleUpdates@company" view="admin"
pageSize=5 sort="createDate asc" %}
```

include_entities

Retrieves an entity(s) using inline API calls and adds the returned entity(s) to the page model of the specified template.

Supported On	Hypr

This tag is similar to the `include` tag but instead of setting the model manually or using the same model available to the parent template, the `include_entities` tag adds a collection of entities to the model of the specified template based on the entity list, view, and any additional query arguments you specify.

As its first argument, this tag takes the name of a template (or alternatively, the template object itself) to whose model the entity(s) should be added. As its second argument, this tag requires the fully qualified name of the entity list that contains the entity(s) to be returned. As its third argument, this tag requires the name of the view to apply to the entity list. The full syntax looks like:

`{% include_entities "templateName" listFQN="name@namespace" view="viewName" %}`

You can use the following arguments with this tag.

Argument	Description
templateName	The template in which to include the entity(s). You can specify the template name (in quotes) or the template object itself (without quotes, such as `model.config.template`). If specifying the template name, the tag applies the proper extension to the filename and assumes the template is located in the `templates/pages` directory; if the template is located in a folder other than `pages` , specify the path starting after `templates` (for example, if the template is in the `templates/modules` directory, specify `modules/templateName` as the argument for the template name). **This argument is required and must be in the first position.**
listFQN	The fully qualified name of the entity list. **This argument is required and must be in the second position.**

Argument	Description
view	The name of the entity list view that determines which document properties can be accessed. **This argument is required. If you don't specify a view, Kibo eCommerce assumes the name of the view is** *default***. Many entity lists do not contain a view named***default***, so it is recommended you always specify a view for this tag; otherwise, the API call may not return any data.**
pageWithUrl	Default: false. Specifies whether to use URL parameters for paging results. When false, you must specify the `startIndex` and `pageSize` parameters as arguments in the tag. If you set this argument to true, you specify the tag to use the existing values for `startIndex` and `pageSize` available from the URL parameters.
sortWithUrl	Default: false. Specifies whether to use URL parameters for sorting results. If you set this argument to true, you specify the tag to sort using the existing value of the `sortBy` URL parameter (either `asc` or `desc`).
startIndex	Default: 0 The item on which to begin listing items in the collection. Kibo eCommerce collections are zero-indexed, so `startIndex=0` starts at the first item in the collection.
pageSize	Default: 15 The maximum number of items to return in the model. **Note:** Setting this number too high may result in poor performance and a cumbersome UI experience.
filter	A filter expression for Kibo eCommerce collections. You can filter entities based on their properties by writing a string expression as your argument. For example: `firstname eq "Brenda"` .
query	A deprecated alias for `filter` . If both the `filter` and `query` arguments are present, `query` takes overrides `filter` .
sort	Default: null A string representing how to sort the collection. You can sort on any property, date (such as `createDate`), or entity name. After specifying the name of the property you are sorting on, include a space followed by " `asc` " or " `desc` " to specify sorting in ascending or descending order.
ids	Default: null A comma-separated list of individual entity IDs. If you include this argument, Kibo eCommerce ignores the `filter` argument and the result set consists only of the entities whose IDs you provide.

```
{% comment %}
"Include the last five sales quotas added by a business user in the model of the sales-quotas templ
ate."
{% endcomment %}

{% include_entities "modules/sales-quotas" listFQN="salesQuotas@company" view="admin" page
Size=5 sort="createDate asc" %}
```

include_products

Retrieves a product(s) using inline API calls and adds the returned product(s) to the page model of the specified template.

Supported On	Hypr

This tag is similar to the `include` tag but instead of setting the model manually or using the same model available to the parent template, the `include_products` tag adds a collection of products to the model of the specified template based on the arguments you specify in the tag.

As its first argument, this tag takes the name of a template (or alternatively, the template object itself) to whose model the product(s) should be added, based on the results of the other tag arguments.

`{% include_products "`*templateName* with argument1=value1 and argument2=value2 as_parameter `%}`

You can use the following arguments with this tag.

Argument	Description
templateName	The template in which to include the product(s). You can specify the template name (in quotes) or the template object itself (without quotes, such as `model.config.template`). If specifying the template name, the tag applies the proper extension to the filename and assumes the template is located in the `templates/pages` directory; if the template is located in a folder other than `pages` , specify the path starting after `templates` (for example, if the template is in the `templates/modules` directory, specify `modules/templateName` as the argument for the template name). **This argument is required and must be in the first position.**
filter	Default: "categoryId req *category ID of the current category page, if it exists*" A filter expression for Kibo eCommerce collections. You can [filter] products based on their properties by writing a string expression as your argument. For example: `firstname eq "Brenda"` . If you place the `include_products` tag on a Kibo eCommerce category template, then the default filter will be `categoryId req [`*current category id*`]`. For example, if the current category ID is 33, then the default filter will be `categoryId req 33` . This filter will automatically limit the displayed products to those belonging to the current category or any of its subcategories. You can override this behavior by supplying your own `filter` argument. If you place the `include_products` tag on a template other than a category template, then there is no default `filter` argument. If there is no `filter` argument then the first page of results from all products in the site will be displayed.
query	A deprecated alias for `filter` . If both the `filter` and `query` arguments are present, `query` takes overrides `filter` .
pageWithUrl	Default: false Specifies whether to use URL parameters for paging results. When false, you must specify the `startIndex` and `pageSize` parameters as arguments in the tag. If you set this argument to true, you specify the tag to use the existing values for `startIndex` and `pageSize` available from the URL parameters.
sortWithUrl	Default: false Specifies whether to use URL parameters for sorting results. If you set this argument to true, you specify the tag to sort using the existing value of the `sortBy` URL parameter (either `asc` or `desc`).

Argument	Description
startIndex	Default: 0 The item on which to begin listing items in the collection. Kibo eCommerce collections are zero-indexed, so `startIndex=0` starts at the first item in the collection.
suppressErrors	Default: false Prior to September 2016, the `include_products` tag suppressed errors related to product search, but this behavior changed in September 2016. In general, you should not have to enable this parameter, but if you want to make absolutely sure that errors on supplemental calls to the `include_products` tag do not obscure primary page content, you can set this parameter to true. For example, you can set the parameter to true when you display related products on a product details page to ensure that intermittent search errors on the related products call do not prevent the entire product details page from loading. **Note:** If you use enable this parameter and the `include_products` tag is located inside a `partial_cache` tag, then any error that occurs causes the page to cache as a blank page. The page will continue to display to shoppers as a blank page until the cache clears.
pageSize	Default: 15 The maximum number of items to return in the model. **Note:** Setting this number too high may result in poor performance and a cumbersome UI experience.
searchQuery	A search term to search on for the product results.
sort	Default: null A string representing how to sort the collection. You can sort on any property, date (such as `createDate`), or document name. After specifying the name of the property you are sorting on, include a space followed by " `asc` " or " `desc` " to specify sorting in ascending or descending order.
includeFacets	Default: false Specifies whether facets should be returned with the products.
productCodes	Default: null Provides a shortcut for building the `filter` argument with multiple products. This argument accepts a string that specifies a comma-separated list of product codes . For example, `productCodes="Product01,Product02,Product03"` is an equivalent shorthand syntax to `filter="ProductCode eq Product01 or ProductCode eq Product02 or ProductCode eq Product03"` . If present, this argument overrides the `filter` argument.
facetHierDepth	Default: 2 If filtering using category facets in a hierarchy, the number of category hierarchy levels to return for the facet. This option is only available for category facets.

Argument	Description
responseFields	Default: null A list or array of fields returned for a call. These fields may be customized and may be used for various types of data calls in Kibo eCommerce. For example, responseFields are returned for retrieving or updating specific attributes, carts, and messages in Kibo eCommerce rather than the whole object. To learn more about this field, refer to the Response Fields topic.
facet	Default: null Individually list the facet fields you want to display in a web storefront product search.
searchTuningRuleCode	Default: null The unique identifier of the search tuning rule.
enableSearchTuningRules	Default: null Enables search tuning rules.
searchTuningRuleContext	Default: null The category ID that the search tuning rule applies to.
facetTemplateExclude	Default: null A comma-separated list of the facets you want to exclude.
facetCategoryId	Default: null The unique identifier of a category for use in faceting.
categoryId	Default: pageContext.CategoryId This parameter is used as the default value for the `facetTemplate` , `facetHierValue` , and `searchTuningRuleContext` parameters. If the `productCodes` or `filter` parameters are not set, this parameter sets the `filter` parameter to " `categoryId req [value]` "
categoryCode	Default: null An alternative to `categoryId` . You can use this parameter to specify a category code instead of a category ID.
facetCategoryCode	Default: null Overrides the use of `categoryId` for the `facetTemplate` and `facetHierValue` parameters.

Argument	Description

facetPrefix	Default: null
	
	Filters facet values by a prefix. The syntax is `facetPrefix="`*facetName:prefix*`"` , and you must also specify either `facet` or `includeFacets` as a separate parameter.
	
	For example, to filter a brand entry facet by facet values that start with "2016", specify the following in the `include_products` parameter list:
	
	`facet="tenant~brandentry" and facetPrefix="tenant~brandentry:2016"`
	
	To set a prefix for more than one facet, separate the values with commas. For example:
	
	`includeFacets=themeSettings.showCategoryFacets and facetPrefix="tenant~color:b, tenant~size:s"`

```
{% comment %}
"Include products in the model of the template specified in the model.config object (or in the product-list-tiled template if the model.config object does not contain a template), based on the product codes listed in the model.config object, and sort the results based on the additional parameters."
{% endcomment %}

{% include_products model.config.template|default:"modules/product/product-list-tiled" with productCodes=model.config.productCodes and includeFacets=themeSettings.showCategoryFacets and pageWithUrl=true and sortWithUrl=true as_parameter %}
```

inline_style

Outputs the contents of a stylesheet as raw CSS.

Supported On	Hypr

Always enclose this tag within HTML `<style>` tags. This tag takes the name of a stylesheet in your theme (such as `"storefront.less"` or `"email.less"`) as its argument.

This tag is designed to support older email readers that cannot load and render external stylesheets.

```
<style>{% include_style "email.less" %}</style>
```

json_attribute

Encodes an object using HTML attribute encoding.

Supported On	Hypr

make_url

Generates a string that helps construct URLs for different Hypr objects, such as images or category models.

Supported On	Hypr, HyprLive

This tag is the recommended method for creating consistent URLs across your site. The syntax for the tag is:

`"`*mode*`" `*objectModel*` %}`

The mode is how you specify what type of URL you are constructing. Depending on which mode you select, you pass in different objects and optional parameters. The modes for this tag are:

- image
- product
- category
- sorting
- facet
- cdn
- paging
- cart
- document
- stylesheet

The result of the tag is a string that helps construct the URL in question. Depending on the mode, the URL may be *scheme-relative* (a relative URL that uses the current protocol, such as `https:` or `http:`, as its base) or *domain-relative* (a relative URL that uses the domain, such as `http://www.yourSite.com/`, as its base). For some modes, such as `"image"`, the resulting string appends a *cache key value*, which is a random number that is updated every time someone busts the cache through the Admin General Settings.

> ℹ️ ⊠Delete
>
> If you set up custom routes for your site, the results of the make_url tag as listed in the examples of this topic likely won't match your custom URLs. However, the make_url tag accounts for custom route changes and outputs a modified string according to the code you set up in the Custom Routing JSON Editor, so there is no need for further action on your part.

image

The `"image"` mode generates a path to the CDN location for the specified image model. You can add image fields to the URL, such as size and quality fields, by including the parameters between the keywords `with` and `as_parameter` at the end of the tag.

Cache key appended to result? **Yes**

Type of URL: **Scheme-relative**

Example:

```
<img src="" model.mainImage with max=themeSettings.listProductThumbSize quality=75 crop="10,10,10,10" as_parameter %}" />
```

```
{# Result: This code specifies the image location at "//cdn.mozu.com/tenantID-siteID/cms/files/sweetImage.jpg?max=150&quality=75&crop=10,10,10,10&_mzCb=90239082348020" #}
```

When you use the `make_url` tag on an image hosted on the Kibo eCommerce CDN, you can include the following additional fields at the end of the tag, between `with` and `as_parameter`:

> ℹ️ ⊠Delete
>
> To avoid incorrect image rendering, do not apply the image manipulation fields to images that contain a dimension greater than 30,000 pixels in length.

Field	Description
max	Specifies a pixel limitation for the largest side of an image.
maxWidth	Specifies a pixel limitation for the width of the image, preserving the aspect ratio if the image needs resizing.
maxHeight	Specifies a pixel limitation for the height of the image, preserving the aspect ratio if the image needs resizing.
width	Specifies an exact width dimension for the image, in pixels.
size	Same as `width`. Provides backwards-compatibility for an earlier syntax.
height	Specifies an exact height dimension for the image, in pixels.
crop	Usage: `crop="x1,y1,x2,y2"` Crops the image based on the specified coordinates. The reference point has positive coordinates only. You can use this field to easily crop an equal amount of pixels from every edge of the image. For example, `crop="10,10,10,10"` removes 10 pixels from all edges of the image (`crop="0,0,0,0"` leaves the image uncropped). You can also use this field to specify a subset of the image. For example, `crop="150,300,150,300"`. The only thing to remember is that you must always specify `x2` to the right of `x1` and `y2` to the bottom of `y1`, otherwise no cropping takes effect.
quality	Adjusts the image compression for JPEG files (other image types do not support this field). Accepts values from 0-100, where 100 = highest quality, least compression.

product

The `"product"` mode generates a path to a product page based on a product model, product code, or product ID. You can also specify a variant for a configurable product using the `variant` parameter.

Cache key appended to result? **No**

Type of URL: **Domain-relative**

Example:

```
{# "uses the product model available in the Hypr template" #}
"product" model %}
{# Result: "/p/my-product-code ...unless you have configured a canonical custom route, in which c
ase the result mirrors the custom route" #}

{# "uses a product code" #}
"product" "my-product-code" %}
{# Result: "/p/my-product-code" #}

{# "uses a product ID" #}
"product" 1234 %}
{# Result: "/p/1234" #}

{# "uses a product code and a variant code" #}
"product" "my-product-code" with variant="small-green" as_parameter %}
{# Result: "/p/my-product-code?vpc=small-green" #}
```

category

The `"category"` mode generates a path to a category page based on a category model, category code, or category ID.

Cache key appended to result? **No**

Type of URL: **Domain-relative**

Example:

```
{# uses the category model available in the Hypr template #}
"category" categoryModel %}
{# Result: "/c/my-category-code ...unless you have configured a canonical custom route, in which
case the result mirrors the custom route" #}

{# uses a category code #}
"category" "my-category-code" %}
{# Result: "/c/my-category-code" #}

{# uses a category ID #}
"category" 1234 %}
{# Result: "/c/1234" #}
```

sorting

The `"sorting"` mode generates a path to a sort query with URL-encoded sort parameters.

Cache key appended to result? **No**

Type of URL: **Domain-relative**

Example:

```
"sorting" "price:desc,rating:asc" %}
{# Result: "?sortBy=price%3Adesc%2Crating%3Aasc" #}
```

facet

The `"facet"` mode generates a path to a faceting query based on a facet value.

Cache key appended to result? **No**

Type of URL: **Domain-relative**

Example:

```
{% for facet in productSearch.facets %}

  <div>{{facet.label}}</div>

  {% for facetValue in facet.values %}

    <a href=""facet" facetValue %}">{{facetValue.label}}</a>

  {% endfor %}

{% endfor %}
```

{# Result (of tag): "This code creates a link that a shopper can use to apply a facet. For example, the facet label might be 'Small' and the result of the make_url tag might be '?facetValueFilter=size%3ASmall'" #}

cdn

The `"cdn"` mode generates a URL on the CDN domain for your tenant, based on a full path to a filename, relative to root.

Cache key appended to result? **Yes**

Type of URL: **Scheme-relative**

Example:

```
"cdn" "/cms/files/video.mp4" %}
{# Result: "//cdn.mozu.com/tenantID-siteID/cms/files/video.mp4?_mzCb=54654986689" #}
```

paging

The `"paging"` mode generates a path to a specific page within a product search model. You can specify to create a path to the previous page, next page, first page, last page, or a specific page with the following fields or page number, which you use at the end of the tag preceded by the `with page=` phrase:

- `"previous"`
- `"next"`
- `"first"`
- `"last"`
- zero-indexed page number

Cache key appended to result? **No**

Type of URL: **Domain-relative**

Example:

```
"paging" productSearch with page="previous" %}
```

```
"paging" productSearch with page=3 %}
```

cart

The `"cart"` mode generates a path to your cart page. This mode does not require you to pass it a Hypr object representing the cart.

Cache key appended to result? **No**

Type of URL: **Domain-relative**

Example:

```
"cart" %}
{# Result: "/cart" #}
```

document

The `"document"` mode generates a path to a document in the CMS based on a fully-qualified list name and a document name.

Cache key appended to result? **No**

Type of URL: **Domain-relative**

Example:

```
"document" listFQN="banners@namespace" name="banner007" %}
{# Result: "/cms/banners@namespace/banner007" #}
```

stylesheet

The `"stylesheet"` mode generates a stylesheet link based on the stylesheet location.

Cache key appended to result? **Yes** (and the resulting key is more complicated than the keys generated by the other modes)

Type of URL: **Scheme-relative**

Example:

```
"stylesheet" "stylesheets/storefront.less" %}
{# Result: "//cdn.mozu.com/tenantID-siteID//stylesheets/storefront.less?SBTHEME=~themeID&dt
=8D2D95D6D734CC0-8D2D3E168157400" #}
```

now

Renders the current date and time using PHP date format.

Supported On	Hypr

```
{% now "F, j, Y" %} {# Result: "Returns a date in the format of" January 1, 2016 #}
```

parent

Returns the output of the parent template block.

Supported On	Hypr, HyprLive

This tag only works in child templates that extend a parent template.

```
{% block title-tag-content %}{% if pageContext.metaTitle %}{{pageContext.metaTitle}}{% else
%}{{model.content.productName}} - {% parent %} {%endif%}{% endblock title-tag-content %}
```

partial_cache

Caches the contents in the tag as a string to reduce the amount of Hypr logic that must be evaluated during page loads. Subsequent page loads reuse the cached content so long as the value of the monitored parameters don't change.

Supported On	Hypr

This is a tag you must open, `{% partial_cache thingToMonitor1 thingToMonitor2 ... %}` , and close, `{% endpartial_cache %}` . You specify the properties to monitor as arguments to the tag. So long as these properties do not change, the contents in the tag are evaluated only once every five minutes.

The following example uses the `include_products` tag to request a list of products from the Kibo eCommerce API. Because server calls can take some time to execute, you can achieve better performance by caching the results of this call. Assuming the result of the query does not change so long as the category ID and the search query stay the same, you can set up the `partial_cache` tag with those two parameters as arguments. The cached result will remain in use for the default five minute interval unless the category ID or the query search change.

```
{% partial_cache model.categoryId pageContext.search.query %}
   {% include_products "modules/product/faceted-products" with pageWithUrl=true %}
{% endpartial_cache %}
```

⚠ ⊠Delete

> **Warning**
>
> Do not use tags whose effects should extend beyond the current template (such as the require_script and set_var tags) inside the partial_cache tag. To function properly, these types of tags should execute on every page load, which would be prevented by the partial_cache tag.

preload_json

Enables the full functionality of HyprLive by making the server-side model available on the client side.

Supported On	Hypr

This tag serializes an object into JSON and embeds the results in a `<script>` tag. The `preload_json` tag takes two arguments: an object that is serialized into JSON and a name that helps construct the ID of the `<script>` tag that contains the results of the tag.

```
{% preload_json model "product" %}
{# Result "The above code produces" <script id="data-mz-preload-product" type="text/json">Serialized JSON of Product Model #}
```

The `require` function exposed by Kibo eCommerce's RequireJS library has a static method that uses the scripts produced by this tag. This method is `require.mozuData(`*name*`)` , where `name` is the string you supply as the second argument to the `preload_json` tag. This function returns a deserialized JavaScript object representing the object you supply as the first argument to the `preload_json` tag. For the prior example, you could write:

```
var p = require.mozuData("product")
```

In this case, `p` is created as an object with the product code, options, and properties of the product model.

Use the `preload_json` tag on every page that requires complex JavaScript interaction with Kibo eCommerce data and also to fully enable the functionality of HyprLive. The core theme template, `template/modules/json-required-for-hyprlive.hypr` , contains many (but not all) of these required preloads. Omitting them from a theme results in errors whenever HyprLive is used.

require_script

Declares that a template depends on a particular JavaScript module to be fully functional.

Supported On	Hypr

This tag takes one argument as the filename of the required script: `{% require_script "`*script*`" %}` . The tag applies the `.js` extension to the filename you provide and assumes the `scripts` directory is the root folder for the path. Note that this tag does not work inside of the `partial_cache` tag.

This tag behaves like an HTML `<script src="">` tag but also guarantees that each script is loaded only once, per the benefits of Kibo eCommerce's RequireJS library. The tag does not output any HTML or text in place but does keep track of the arguments sent through each invocation of the tag. When the `all_scripts` tag is called near the end of the template, each argument passed to the `require_script` tag is output as a comma separated list of quote-delimited strings. If you use the `require_scripts` to require a tag that is not already a default script included by your theme's `build.js` file, you must add the script to that file.

You can load external scripts (so long as they do not explicitly require synchronous loading) by supplying the `require_script` tag with a fully qualified URL (*http://* or *https://*). Using the `require_script` tag instead of HTML `<script>` tags to load external scripts improves the load times of pages on your site.

```
{% require_script "modules/login-links" %
```

set_header

Adds/modifies http headers on the request response.

Supported On	Hypr

Examples:

Set the x-frame-options header

```
{% set_header "X-Frame-Options:SAMEORIGIN" %}
```

You can also use this tag with custom headers.

```
{% set_header "foo:bar" &}
```

This tag also supports named parameters:

```
{% set_header name="foo" value="bar" replage=false %}
```

Field	Description
Name	Name of header
value	value of the header
replace	(Optional) Used to replace a header if the page already has a header set.

set_var

Assigns a value to a new variable of your choice. The variable persists in the global Hypr context for the rest of the render cycle. Note that this tag is only supported on Hypr. In HyprLive, assign values to variables by using `set` . This discrepancy is due to `set_var` replacing the older `set` tag on the Hypr server but not replacing it on the client-side HyprLive.

Supported On	Hypr

When possible, use the `with` tag to assign values to variables. The `set_var` tag is designed for advanced use cases, such as setting global variables from within a `for` tag. The `set_var` tag can run in an inner template and, because of its global context, may have unexpected effects in the outer template (which calls the inner template via the `include` tag).

This tag takes the name of a new global variable as its argument, followed by an equal sign, and then the value you wish to set the variable to: `{% set_var fruit="apple" %}`

Simple examples:

```
{% set_var testBool=true %}
{{ testBool }} {# Renders true #}

{% set_var testFloat=123.456 %}
{{ testFloat }} {# Renders 123.456 #}

{% set_var testString="waffle cones" %}
{{ testString }} {# Renders "waffle cones" #}
```

The following, more in-depth example demonstrates how you can take advantage of the `set_var` tag's global scope to make the value of a Boolean available outside of the `for` tag where the Boolean is set. In this case, the Boolean is used to hide or display specific categories to shoppers depending on whether the shoppers are signed in or logged out.

```
{% for child in node.items %}
   {% if user.isAnonymous %}
      {% for hideId in themeSettings.hideIdsForAnonymousUsers %}
         {% if hideId == child.categoryId %}
            {% set_var hideThisCat=true %}
         {% endif %}
      {% endfor %}
   {% endif %}
{% endfor %}

{% if not hideThisCat %}
   {% include_products "modules/cat/cat" with pageSize=200 categoryId=child.categoryId %}
{% endif %}
```

spaceless

Removes tab characters, new lines, and the white space between HTML tags.

Supported On	Hypr, HyprLive

This tag does not remove spaces between text and HTML tags.

```
{% spaceless %}
<p>
<a href="">  Cool Gizmos  </a>
</p>
{% endspaceless %}

{% comment %}
"The prior example returns:
<p><a href="">  Cool Gizmos  </a></p>"
{% endcomment %}
```

templatetag

Displays the syntax characters used to construct a Hypr tag or variable.

Supported On	Hypr, HyprLive

Because the template system cannot escape, you must use the this tag to display Hypr syntax. Use this tag with the following arguments.

Argument	Output
openblock	{%
closeblock	%}
openvariable	{{
closevariable	}}
openbrace	{
closebrace	}
opencomment	{#
closecomment	#}

```
{% templatetag openblock %} require_script "modules/customScript"  {% templatetag closeblock %}
```

<

visitor_tracking_pixel

Required for every page on your site, this tag relays visitor tracking information for billing purposes.

Supported On	Hypr

This tag adds an HTML tag that renders a 1px by 1px transparent image on your page. Kibo uses the HTTP request for the image as a beacon to gather the necessary visitor information.

The tag is located near the end of the `page.hypr` template, as shown in the following example. If you don't extend from this template, you must add the tag manually.

```
...
{% visitor_tracking_pixel %}
</body>
{% endblock body-tag %}
</html>
```

with

Assigns the value of a Hypr expression to a local variable that you can use within the scope of the tag.

Supported On	Hypr, HyprLive

The local variable helps reduce the amount of times a complex piece of Hypr logic must be evaluated.

This is a tag you must open, `{% with HyprExpression as variable %}`, and close `{% endwith %}`. You provide the tag an expression whose value you assign to a variable of your choice. Separate the expression and the variable with the keyword `as`.

```
{% with order.items|findwhere("fulfillmentMethod", "Digital") as digitalFulfillmentItem %}
    {# "Everything between here and the {% endwith %} tag will have access to a new local variab
le called "digitalFulfillmentItem", which contains the first item in the order that is to be digitally fulf
lled" #}

    {% if digitalFulfillmentItem %} {# "use the variable to avoid repeating the same operation mult
iple times" #}
        {# "do stuff..." #}
    (% endif %}
{% endwith %}
```

Deprecated Hypr Tags

This topic includes a list of deprecated Hypr tags.

• cycle	• ifchanged	• json
• debug	• ifequal	• raw
• firstof	• ifnotequal	• widthratio

cycle

DEPRECATED. Modern web technologies allow you to achieve the effects of this tag using CSS.

Supported On	Hypr

This tag is used in legacy systems to apply a particular CSS class to alternating rows in a table or list.

debug

DEPRECATED. Use the `dump` tag instead.

Supported On	Hypr

Outputs a verbose amount of debugging information, including the current context and imported modules.

firstof

DEPRECATED

Supported On	Not supported in most versions of Hypr.

Outputs the first argument that is not empty or false from a variable number of arguments. If all passed arguments are false, this tag outputs nothing.

```
{% firstof var1 var2 var3 %}
```

You can use a literal string as a fallback value if all passed arguments are false:

```
{% firstof var1 var2 var3 "fallback value" %}
```

ifchanged

DEPRECATED

Supported On	Hypr

Determines whether a value has changed from the last iteration of a loop. This tag checks its rendered contents against the previous state and displays the content if it has changed. For example, the following code displays a list of days but only displays the month if it has changed:

```
{% for date in days %}
    {% ifchanged %} {{ date|date:"F" }} {% endifchanged %}
    {{ date|date:"M/d"|lower }}
    {{ date|date:"j" }}
{% endfor %}
```

If given a variable, this tag checks whether that variable has changed. For example, the following shows the date every time it changes, but only shows the hour if both the hour and the date change:

```
{% for date in days %}
    {% ifchanged date.date %} {{ date.date }} {% endifchanged %}
    {% ifchanged date.hour date.date %}
        {{ date.hour }}
    {% endifchanged %}
{% endfor %}
```

ifequal

DEPRECATED. Not supported in HyprLive. Use an equal sign within an `if` tag instead.

Supported On	Hypr

Outputs the contents of the block if the two arguments equal each other.

ifnotequal

DEPRECATED. Not supported in HyprLive. Use the not equal operator inside an `if` tag instead.

Supported On	Hypr

Outputs the contents of the block if the two arguments do not equal each other.

json

DEPRECATED. Refer to the `json_attribute` and `preload_json` tags as alternatives, which also process JSON.

Supported On	Hypr

Outputs an object formatted in JSON.

```
{% json model %}
```

raw

DEPRECATED. This is a legacy tag that was used for complicated debugging scenarios.

Supported On	Hypr, HyprLive

Eliminates the processing of content inside tags. This is a tag you must open, `{% raw %}` , and close `{% endraw %}` .

```
{% raw %}
{% if %}
"Hello sweet world."
{{a|b}}
{% endif %}
{% endraw %}

{% comment %}
Outputs as:
"{% if %}
Hello sweet world.
{{a|b}}
{% endif %}"
{% endcomment %}
```

widthratio

DEPRECATED

Supported On	Hypr

Calculates the ratio of a given value to a maximum value, and then applies that ratio to a constant. For example:

```
<img src="" height="10" width="{% widthratio this_value max_value 100 %}" />
```

If `this_value` is 10 and `max_value` is 20, then the image in the above example is 50 pixels wide because (10 / 20) * 100 = 50.

Hypr Filters

This topic includes a complete list of Hypr filters.

	• divisibleby		
	• escape		
	• find	• linenumbers	
• add	• findwhere	• lower	
• add_time	• first	• mod	• subtract
• currency	• fix_ampersands	• multiply	• timesince
• date	• floatformat	• prop	• timeuntil
• default	• get_product_attribute	• replace	• truncatewords
• dictsort	• get_product_attribute_value	• safe	• upper
• dictsortreversed	• get_product_attribute_values	• slugify	• urlencode
• divide	• is_after	• split	
	• is_before	• string_format	
	• join		
	• last		

Deprecated Filters
Click here to view a list of deprecated Hypr filters.

add

Adds the filtered value to the argument.

Filtered Value	type: number If this is not a number, Hypr tries to parse it as a number.
Argument	type: number If this is not a number, Hypr tries to parse it as a number.
Supported On	Hypr, HyprLive

```
{% comment %}
model.items=[
    {'name': 'painting', 'ID': 123},
    {'name': 'sculpture', 'ID':456},
    {'name': 'tapestry', 'ID': 789},
]
{% endcomment %}
{{ model.items.length|add(1) }}

{# Result: 4 #
```

add_time

Adds a specified amount of seconds to a date and returns a new date. You can pass the results of this filter to the date filter to format the resulting date.

Filtered Value	type: DateTime The date to add seconds to.
Argument	type: integer The number of seconds to add.
Supported On	Hypr

```
{{ now|add_time(86400)|date("l") }}

{# Result: Wednesday (assuming today = Tuesday) #}
```

currency

Formats the filtered value as currency according to the rules of the locale for the site.

Filtered Value	type: number The number to format into currency. If this is not a number, Hypr attempts to parse it as a number.
Argument	N/A
Supported On	Hypr, HyprLive

```
{{ 15.5|currency }}

{# Result (EN_US locale): "$15.50" #}
```

date

Formats the filtered value as a date and time using PHP date format.

Filtered Value	type: DateTime The date to format. If this is not a date, Hypr attempts to parse it as a date.
Argument	type: string The PHP formatting convention to format the date into.
Supported On	Hypr, HyprLive

```
{{ model.auditInfo.createDate|date("F j, Y") }}

{# Result: January 1, 2016 #}
```

default

Returns the filtered value if it exists. If the filtered value is null, this filter returns the argument.

Filtered Value	type: any type The value to return, if it exists.
Argument	type: any type The value to return as a fallback when the filtered value does not exist.
Supported On	Hypr, HyprLive

```
Welcome back, {{ user.firstName|default(user.emailAddress) }}!

{# Result: "Welcome back, Brenda!" or "Welcome back, brenda@mozu.com!" (depending on whether user.firstName exists) #}
```

dictsort

Sorts a list of objects in ascending order based on a property that all the objects have.

Filtered Value	type: list of objects The list to sort.
Argument	type: string The property to sort the list on.
Supported On	Hypr, HyprLive

```
foo = [
    {'name': 'zed', 'age': 19},
    {'name': 'amy', 'age': 22},
    {'name': 'joe', 'age': 31},
]
{% for item in foo|dictsort("name") %}

{{ item.name }}

{% endfor %}

{% comment %}
Result:
    amy
    joe
    zed
{% endcomment %}
```

dictsortreversed

Same as `dictsort` , but sorts in descending order.

divide

Divides the filtered value by the argument.

Filtered Value	type: number The number to divide. If this is not a number, Hypr attempts to parse it as a number.
Argument	type: number The number to divide by. If this is not a number, Hypr attempts to parse it as a number.
Supported On	Hypr, HyprLive

```
{{ 21|divide(7) }}

{# Result: 3 #}
```

divisibleby

Returns TRUE if the argument divides into the filtered value evenly with zero remainder.

Filtered Value	type: number The number to divide. If this is not a number, Hypr attempts to parse it as a number.
Argument	type: number The number to divide by. If this is not a number, Hypr attempts to parse it as a number.
Supported On	Hypr, HyprLive

```
Is six divisible by three? {{ 6|divisibleby(3) }} -- Is six divisible by four? {{ 6|divisibleby(4) }}

{# Result: "Is six divisible by three? True -- Is six divisible by four? False #}
```

escape

Converts HTML special characters into HTML entities.

This filter is useful for sanitizing potentially malicious code supplied by users or third-parties. Note, however, that it is not necessary to use this filter on content enclosed by the `autoescape` tag.

Filtered Value	type: string The string to escape.
Argument	N/A
Supported On	Hypr, HyprLive

```
{{ "

Hello Friends & Family

"|escape }}

{# Result: "<p>Hello Friends & Family</p>" #}
```

find

Searches a list of objects by the property `ID` and returns the first object that matches the ID.

Filtered Value	type: list of objects The list to search.
Argument	type: number The ID to match.

Supported On	Hypr

```
foo = [
    {'name': 'painting', 'ID': 123},
    {'name': 'sculpture', 'ID':456},
    {'name': 'tapestry', 'ID': 789},
]
{{ foo|find(789) }}

{# Result: {'name': 'tapestry', 'ID': 789} #}
```

findwhere

Searches a list of objects to find the object that contains a matching property value.

Filtered Value	type: list of objects The list to search.
Argument	type: string The first argument is the name of a property to search. Every object in the list should have this property.
	type: string The second argument is the value of the property that results in a match.
	type: Boolean (Optional) If TRUE, the search is case-sensitive.
Supported On	Hypr, HyprLive

```
foo = [
    {'name': 'painting', 'ID': 123},
    {'name': 'sculpture', 'ID':456},
    {'name': 'tapestry', 'ID': 789},
    {'name': 'Tapestry', 'ID': 012},
]
{{ foo|findwhere("name", "Tapestry") }}
{# Result: {'name': 'tapestry', 'ID': 789} #}

{{ foo|findwhere("name", "Tapestry", True) }}
{# Result: {'name': 'Tapestry', 'ID': 012} #}
```

first

Returns the first value in a list or the first character in a string.

Filtered Value	type: list or string The list or string to apply the filter to.
Argument	N/A
Supported On	Hypr, HyprLive

```
{{ "Hypr"|first }}

{# Result: "H" #}
```

fix_ampersands

Replaces unescaped ampersand characters with HTML entities.

Filtered Value	type: string The HTML string to apply the filter to.
Argument	N/A
Supported On	Hypr

```
{{ "<p>Hello Friends & Family</p>"|fix_ampersands }}

{# Result: "<p>Hello Friends & Family</p>" #
```

floatformat

Formats the filtered value into a floating point number.

Filtered Value	type: number The number to apply the filter to.
Argument	type: number (Optional) Specifies rounding options.
Supported On	Hypr, HyprLive

When used without an argument, this filter rounds a floating-point number to one decimal place (but only if there's a non-zero decimal place to be displayed).

Value	Filter	Result
34.23234	`{{ value|floatformat }}`	34.2
34.00000	`{{ value|floatformat }}`	34
34.5600	`{{ value|floatformat }}`	34.6

If you specify a positive integer argument, this filter rounds the value to the specified number of decimal places (including zeroes).

Because of the differences in math handling between servers, the conversion accuracy is guaranteed up to the fifth decimal place only.

Value	Filter	Result
34.23234	`{{ value|floatformat(3) }}`	34.232
34.00000	`{{ value|floatformat(3) }}`	34.000
34.5600	`{{ value|floatformat(3) }}`	34.560

If you specify a zero as the argument, this filter rounds the value to the nearest integer.

Value	Filter	Result
34.23234	`{{ value|floatformat(0) }}`	34
34.00000	`{{ value|floatformat(0) }}`	34
34.56000	`{{ value|floatformat(0) }}`	35

If you specify a negative argument, this filter rounds the value to the specified number of decimal places (but only if there's a non-zero decimal place to be displayed).

Using this filter without an argument is equivalent to using this filter with an argument of `-1` .

Value	Filter	Result
34.23234	`{{ value|floatformat(-3) }}`	34.232
34.00000	`{{ value|floatformat(-3) }}`	34
34.5600	`{{ value|floatformat(-3) }}`	34.560

get_product_attribute

Returns an attribute object for a specified product.

Filtered Value	type: object A Kibo eCommerce product, such as the `model` on a Product page or an item in a product list.
Argument	type: string The fully-qualified name of a product attribute.
Supported On	Hypr, HyprLive

```
{% with model|get_product_attribute("tenant~color") as colorAttr %}
  {{ colorAttr.attributeDetail.name }}:{{ colorAttr.values|first|prop("value") }}
{% endwith %}

{# Result: "Color: Blue" #}
```

get_product_attribute_value

Returns the first string value of a product attribute.

Filtered Value	type: object A Kibo eCommerce product, such as the `model` on a Product page or an item in a product list.
Argument	type: string The fully-qualified name of a product attribute.
	type: Boolean (Optional, and applicable only to string attributes) If TRUE, returns the value of the attribute instead of the string value. The string value is the user-facing label of the

Supported On	Hypr, HyprLive

```
{{ model|get_product_attribute_value("tenant~color") }}
{# Result: "Aqua Blue" #}
{{ model|get_product_attribute_value("tenant~color", true) }}
{# Result: "aqua-blue" #}
```

get_product_attribute_values

Returns the list of string values for a product attribute.

Filtered Value	type: object A Kibo eCommerce product, such as the `model` on a Product page or an item in a product list.
Argument	type: string The fully-qualified name of a product attribute.
	type: Boolean (Optional, and applicable only to string attributes) If TRUE, returns the value of the attribute instead of the string value. The string value is the user-facing label of the
Supported On	Hypr, HyprLive

```
{{ model|get_product_attribute_values("tenant~color")|join(", ") }}
{# Result: "Aqua blue, Mariner Green, Catawampus Orange" #}
{{ model|get_product_attribute_values("tenant~color", true)|join(", ") }}
{# Result: "aqua-blue, mariner-green, catawampus-orange" #}
```

is_after

Returns TRUE if the filtered value occurs after the argument.

Filtered Value	type: DateTime This date is later than the argument when this filter returns TRUE.
Argument	type: DateTime This date is earlier than the filtered value when this filter returns TRUE.
Supported On	Hypr, HyprLive

```
{{ laterDate|is_after(earlierDate) }}
{# Result: True #}
{{ earlierDate|is_after(laterDate) }}
{# Result: False #}
```

is_before

Returns TRUE if the filtered value occurs before the argument.

Filtered Value	type: DateTime This date is earlier than the argument when this filter returns TRUE.
Argument	type: DateTime This date is later than the filtered value when this filter returns TRUE.
Supported On	Hypr, HyprLive

```
{{ earlierDate|is_before(laterDate) }}
{# Result: True #}
{{ laterDate|is_before(earlierDate) }}
{# Result: False #}
```

join

Joins the values of a list with the characters specified in the argument.

Filtered Value	type: list of strings or numbers The list of values to join. If the values are numbers, Hypr converts them to strings.
Argument	type: string The joining characters.
Supported On	Hypr, HyprLive

```
{{ product.fulfillmentTypesSupported|join(" or " }}
{# Result: "InStorePickup or DirectShip" #}
```

last

Returns the last value in a list or the last character in a string.

Filtered Value	type: list or string The list or string to apply the filter to.
Argument	N/A
Supported On	Hypr, HyprLive

```
{{ "Hypr"|last }}
{# Result: "r" #}
```

linenumbers

Prepends text with line numbers, calculating the maximum line number width to align text with correct padding.

Filtered Value	type: string The text to apply line numbers to.
Argument	N/A
Supported On	Hypr

```
text = "The
quick
brown
fox"
{{ text|linenumbers }}

{% comment %}
Result:
    "1 The
     2 quick
     3 brown
     4 fox"
{% endcomment %}
```

lower

Converts a string to lowercase.

Filtered Value	type: string The string to lowercase.
Argument	N/A

Supported On	Hypr, HyprLive

```
{{ "Hypr"|lowercase }}
{# Result: "hypr" #}
```

mod

Returns the modulus (remainder) of dividing the filtered value by the argument.

Filtered Value	type: number The number to divide. If this is not a number, Hypr attempts to parse it as a number.
Argument	type: number The number to divide by. If this is not a number, Hypr attempts to parse it as a number.
Supported On	Hypr, HyprLive

```
{{ 7|mod(3) }}
{# Result: 1 #}
```

multiply

Multiplies two numbers

Filtered Value	type: number A number to multiply. If this is not a number, Hypr attempts to parse it as a number.
Argument	type: number A number to multiply. If this is not a number, Hypr attempts to parse it as a number.
Supported On	Hypr, HyprLive

```
{{ 7|multiply(3) }}
{# Result: 21 #}
```

prop

Returns the value of an object property.

This filter may not seem immediately useful because you can use dot-lookup notation to access object properties in Hypr. For instance, *object*.length returns the length property for an object in Hypr. However, you can not use dot-lookup notation on the result of a filter unless you first assign the result to a variable. In these cases, the prop filter comes in handy by shortening the code you have to write.

Filtered Value	type: object The object that contains the property.
Argument	type: string The name of the property.
Supported On	Hypr, HyprLive

```
{% comment %}
  model.items=[
    {'name': 'painting', 'ID': 123},
    {'name': 'sculpture', 'ID':456},
    {'name': 'tapestry', 'ID': 789},
  ]
{% endcomment %}

{{ model.items|first|prop("name") }}

{# Result: "painting" #}
```

replace

Removes or replaces all instances of the provided argument from a string.

Filtered Value	type: string The string to remove or replace the argument in.
Argument	type: string If you provide one argument, this filter removes all instances of the argument from the filtered value. If you provide two arguments, this filter replaces all instances of the first argument with the second argument.
Supported On	HyprLive

```
{{ "HyprLive"|replace("Live") }}
{# Result: "Hypr" #}

{{ "HyprLive"|replace("Live", " has many filters") }}
{# Result: "Hypr has many filters" #}
```

safe

Excludes the filtered value from being escaped by the `{% autoescape %}` tag. Use this tag to output raw HTML from inside the body of an `{% autoescape %}` tag.

Filtered Value	type: string The content to exclude from HTML escaping.
Argument	N/A
Supported On	Hypr, HyprLive

```
{% autoescape %}
{{ "

I won't be escaped!

"|safe }}
{% autoescape off %}

{# Result: "

I won't be esacped!

" #}
```

slugify

Converts text to conventional URL slug format to benefit SEO.

This filter works by converting text to lowercase and replacing special characters and spaces with hypens.

Filtered Value	type: string The string to slugify.
Argument	N/A
Supported On	Hypr, HyprLive

```
{{ "Brenda is a closer"|slugify }}

{# Result: "brenda-is-a-closer" #}
```

split

Splits a string wherever there is a space and returns a list of the split components.

Filtered Value	type: string The string to split.
Argument	type: string (Optional) The character to split the string on. If you leave this argument blank, the default is to split the string on the space character.
Supported On	Hypr, HyprLive

```
{{ "Brenda is a closer"|split }}
{# Result: ["Brenda", "is", "a", "closer"] #}

{{ "Brenda is a closer"|split("a") }}
{# Result: ["Brenda is ", " closer"] #}
```

string_format

Formats a string that contains placeholders (such as " `{0}` " and " `{1}` ") by injecting the filter arguments into the string according to the numbering of the placeholder and the order of the arguments.

Filtered Value	type: string The string to apply the filter to.
Argument	type: string The value(s) to inject into the placeholder(s) located in the string. This filter takes a variable amount of arguments, separated by commas, to match the number of place
Supported On	Hypr, HyprLive

```
labels.optIn = "Yes, I would like to receive offers from {0}. Send these bad boys to {1}!"

{{ labels.optIn|string_format(siteContext.generalSettings.websiteName, user.emailAddress) }}

{# Result: "Yes, I would like to receive offers from Your Awesome Site. Send these bad boys to my Email@coolShopper.com!" #}
```

subtract

Subtracts the argument from the filtered value.

Filtered Value	type: number The number to subtract from. If this is not a number, Hypr attempts to parse it as a number.
Argument	type: number This number to subtract. If this is not a number, Hypr attempts to parse it as a number.
Supported On	Hypr, HyprLive

```
{{ 11|subtract(3) }

{# Result: 8 #}
```

timesince

Returns the elapsed time in seconds between the argument and the filtered value.

Filtered Value	type: DateTime The end time to measure to.
Argument	type: DateTime The start time to measure from.

Supported On	Hypr, HyprLive

{{ timeOfShopperPurchase|timesince(timeOfShopperVisit) }}

{# Result: 9000 #}

timeuntil

Same as the `timesince` filter, except reversed. This filter returns the elapsed time in seconds between the filtered value and the argument.

Filtered Value	type: DateTime The start time to measure from.
Argument	type: DateTime The end time to measure to.
Supported On	Hypr, HyprLive

{{ timeOfShopperVisit|timeuntil(timeOfShopperPurchase) }}

{# Result: 9000 #}

truncatewords

Truncates a string if it is longer than the specified number of words. Truncated strings end with an ellipsis.

Filtered Value	type: string The string to truncate.
Argument	type: integer The number of total words in the resulting string.
Supported On	Hypr, HyprLive

{{ "My Site is Awesome"|truncatewords(3) }}

{# Result: "My Site is..." #}

upper

Converts a string to uppercase.

Filtered Value	type: string The string to uppercase.
Argument	N/A
Supported On	Hypr, HyprLive

{{ "Hypr"|upper }}

{# Result: "HYPR" #}

urlencode

Converts special characters in a string into their URL-encoded equivalents.

Filtered Value	type: string The string to apply the filter to.
Argument	N/A
Supported On	Hypr

```
urlText = "firstParam=1&secondParam=2"
{{ urlText|urlencode}}

{# Result: "firstParam%3D1%26secondParam%3D2" #}
```

Deprecated Hypr Filters

This topic includes a list of deprecated Hypr filters.

• absolute_url • addslashes • capfirst • center • collection_filter • cut • default_edit	• escapejs • filesizeformat • force_escape • get_digit • length • length_is • linebreaks • linebreaksbr	• ljust • make_list • number_format • phone2numeric • pluralize • random • removetags • rjust • slice • striptags	• stylesheet_tag • time • title • truncatechars • urlize • urlizetrunc • wordcount • wordwrap • yesno

absolute_url

DEPRECATED

Validates and returns an absolute URL assembled by prepending the argument to the filtered value.

Filtered Value	type: string The latter part of the URL.
Argument	type: string The origin of the URL, which should include the protocol and host name.
Supported On	Hypr

```
{{ "/product/12345"|absolute_url(pageContext.url) }}

{# Result: "http://yourSite.com/product/12345" #}
```

addslashes

DEPRECATED

Adds back slashes before quotation marks in order to escape strings.

Filtered Value	type: string The string to apply the filter to.
Argument	N/A
Supported On	HyprLive

```
{{ "I'm being escaped!"|addslashes }}

{# Result: "I\'m being escaped!" #}
```

capfirst

DEPRECATED

Capitalizes the first character of the filtered value.

Filtered Value	type: string The string to apply the filter to.
Argument	N/A
Supported On	Hypr

```
{{ "hypr"|capfirst }}
{# Result: "Hypr" #}
```

center

DEPRECATED

Centers the filtered value by within a set amount of total characters.

Filtered Value	type: string The string to center.
Argument	type: integer The total amount of ASCII characters in the resulting string, including the centered value.
Supported On	Hypr

```
{{ "Brenda"|center(10) }}
{# Result:"  Brenda  " (there are two spaces on either side) #}
```

collection_filter

DEPRECATED

Filters a collection (list of objects) to satisfy the string predicate in the first argument.

Filtered Value	type: list of objects The list to filter.
Argument	type: string The string predicate to use as a filter for the list.
Supported On	Hypr

```
foo = [
    {'name': 'zed', 'age': 19},
    {'name': 'amy', 'age': 22},
    {'name': 'joe', 'age': 31},
]
{% for item in foo|collection_filter("age gt 20") %}
{% endfor %}

{% comment %}
Result:
    [
        {'name': 'amy', 'age': 22},
        {'name': 'joe', 'age': 31},
    ]
{% endcomment %}
```

cut

DEPRECATED. Use the `replace` filter instead.

Removes all instances of the argument from the filtered value.

Filtered Value	type: string The string to remove the argument from.
Argument	type: string The string to remove from the filtered value.
Supported On	Hypr

```
{{ "Hyper"|cut("e") }}
{# Result: "Hypr" #}
```

default_edit

DEPRECATED

Similar to the default filter, except this filter only displays the fallback argument when the filtered value does not exist AND when the current page is in Edit mode, meaning that the page is being accessed through the Content Editor. You can use this filter to provide placeholder data such as usernames or product information, which helps users editing the site work within a realistic-looking page.

Supported On	Hypr

escapejs

DEPRECATED

Replaces single and double quotes with backslashes so that the resulting string can be safely embedded in JavaScript code and stored as a string in a variable.

Supported On	Hypr

filesizeformat

DEPRECATED

Formats a number into a human-readable file size in powers of 1024.

Filtered Value	type: number The number to format into a file size. If this is not a number, Hypr attempts to parse it as a number.
Argument	N/A
Supported On	Hypr

```
{{ 123456789|filesizeformat }}
{# Result: "117.7 MB" #}
```

force_escape

DEPRECATED

Immediately escapes HTML characters into HTML entities. This filter is useful in the rare cases where you need multiple escaping or want to apply other filters to the escaped results.

Supported On	Hypr

get_digit

DEPRECATED

Given a number, returns the digit in the position specified by the argument.

Filtered Value	type: number or string The number (or string formatted as a number) to apply the filter to.
Argument	type: integer The position of the digit to return. 1 returns the rightmost digit. 2 returns the tenths place. 3 returns the hundredths place. And so on.
Supported On	Hypr

```
{{ 107.853|get_digit(3) }}
{# Result: 8 #}
```

length

DEPRECATED. All Hypr collections have a `.length` property you can use instead.

Supported On	Hypr

Returns the length of a string or the number of items in a list.

length_is

DEPRECATED

Supported On	Hypr

Returns TRUE if the filtered value's length matches the argument.

linebreaks

DEPRECATED

Supported On	Hypr

Replaces line breaks with HTML `<br>` tags, unless the line is surrounded by blank lines, in which case this filter wraps the line in HTML `<p>` tags.

linebreaksbr

DEPRECATED

Supported On	Hypr

Same as the `linebreaks` filter, except this filter never outputs HTML `<p>` tags.

ljust

DEPRECATED

Left-aligns the filtered value within a set amount of total characters.

Filtered Value	type: string The string to left-align.
Argument	type: integer The total amount of ASCII characters in the resulting string, including the aligned value.
Supported On	Hypr

```
{{ "Brenda"|ljust(10) }}
{# Result: "Brenda    " (contains four trailing spaces) #}
```

make_list

DEPRECATED. Use the `split` filer instead.

Returns the filtered value in list form. For integers, it returns a list of digits. For strings, it returns a list of characters.

Filtered Value	type: string or number The value to convert into a list.
Argument	N/A
Supported On	Hypr

String list: {{ "Hypr"|make_list }} -- Number list: {{ 123|make_list }}

{# Result: String list: ['H', 'y', 'p', 'r'] -- Number list: [1, 2, 3] #}

number_format

DEPRECATED. Use the `floatformat` filter instead.

Supported On	Hypr

phone2numeric

DEPRECATED

Converts a phone number that contains letters into its numeric equivalent.

Filtered Value	type: string The phone number to apply the filter to.
Argument	N/A
Supported On	Hypr

{{ "1-800-PIRATES"|phone2numeric }}

{# Result: "1-800-747-2837" #}

pluralize

DEPRECATED

Returns a plural suffix if the filtered value is not one. By default, " s " is used as the suffix.

Filtered Value	type: number The string to apply the filter to.
Argument	type: string (Optional) An alternate suffix to use, such as " es ".
Supported On	Hypr

shopper{{ 0|pluralize }} {# Result: "shoppers" #}

shopper{{ 1|pluralize }} {# Result: "shopper" #}

shopper{{ 2|pluralize }} {# Result: "shoppers" #}

business{{ 3|pluralize("es") }} {# Result: "businesses" #}

random

DEPRECATED

Returns a random item from a list.

Filtered Value	type: list The list to extract a random value from.
Argument	N/A
Supported On	Hypr

```
list = ["a", "b", "c", "d"]
{{ list|random }}

{# Result: "c" #}
```

removetags

DEPRECATED

Also deprecated in Django. Not recommended for use.

rjust

DEPRECATED

Right-aligns the filtered value within a set amount of total characters.

Filtered Value	type: string The string to right-align.
Argument	type: integer The total amount of ASCII characters in the resulting string, including the aligned value.
Supported On	Hypr

```
{{ "Brenda"|rjust(10) }}

{# Result: "    Brenda" (preceded by four spaces) #}
```

slice

DEPRECATED

Returns a slice of the list. This filter uses the same syntax as Python's list slicing.

Filtered Value	type: list The list to return a slice from.
Argument	type: string The index to slice to, assuming the first index is 0. If you provide more than one index (e.g. 1:3), then the slice happens between the first and second indices.
Supported On	Hypr

```
list = ["a", "b", "c", "d"]
{{ list|slice("2") }}

{# Result: ["a", "b"] #}
```

striptags

DEPRECATED

Strips all HTML and XHTML tags.

Filtered Value	type: string The string to remove HTML/XHTML tags from.
Argument	N/A
Supported On	HyprLive

```
{{ "
```

Brenda is a Champion Who Deserves Her Own H1 Tags

```
"|striptags}}
```

{# Result: "Brenda is a Champion Who Deserves Her Own H1 Tags" #}

stylesheet_tag

DEPRECATED. Use the `%}` tag instead.

Supported On	Hypr

Takes a relative URL as the filtered value and returns an HTML link tag with appropriate URL parameters for theme, CDN, and cache content.

time

DEPRECATED. Use the `date` filter instead.

Supported On	Hypr

title

DEPRECATED

Capitalizes every word in the supplied string.

Filtered Value	type: string The string to convert to title case.
Argument	N/A
Supported On	HyprLive

{{ "my site delivers an outstanding shopper experience"|title }}

{# Result: "My Site Delivers An Outstanding Shopper Experience" #}

truncatechars

Truncates a string if it is longer than the specified number of characters. Truncated strings end with an ellipsis.

Filtered Value	type: string The string to truncate.
Argument	type: integer The number of total characters in the resulting string, including the ellipsis if truncated.
Supported On	Hypr

```
{{ "HyprLive"|truncatechars(7) }}
{# Result: "Hypr..." #}
```

urlize

Converts URLs embedded within plain text into clickable links.

Filtered Value	type: string The string to apply the filter to.
Argument	N/A
Supported On	Hypr

```
{{ "Get started at www.mozu.com!"|urlize}}
{# Result: "Get started at www.mozu.com!" #}
```

urlizetrunc

Same as urlize , but truncates URLs after the specified character limit. This filter adds an ellipsis to truncated URLs.

Filtered Value	type: string The string to apply the filter to.
Argument	type: integer The total number of characters in the URL, including the ellipsis for truncated URLs.
Supported On	Hypr

```
{{ "Get started at www.mozu.com!"|urlizetrunc(9)}}
{# Result: "Get started at www.mo...!" #}
```

wordcount

Returns the number of words in the string.

Filtered Value	type: string The string to apply the filter to.
Argument	N/A
Supported On	Hypr

```
{{ "I love to count words"|wordcount }}
{# Result: 5 #}
```

wordwrap

DEPRECATED

Wraps lines of text after a specified character limit.

Filtered Value	type: string The string to wrap text on.
Argument	type: integer The character limit per line.
Supported On	Hypr

```
{{ "I don't like long lines at all"|wordwrap(8) }}

{% comment %}
Result:
    "I don't
    like
    long
    lines
    at all"
{% endcomment %}
```

yesno

DEPRECATED

Maps values for TRUE and FALSE to the strings "yes", "no", or a custom mapping passed as a comma-separated list in the argument, and returns one of those strings according to the filtered value.

Filtered Value	type: Boolean The Boolean value to map a string output to.
Argument	type: string (Optional) Custom words to map to TRUE and FALSE.
Supported On	Hypr

```
{{ True|yesno }}
{# Result: "yes" #}

{{ False|yesno }}
{# Result: "no" #}

{{ True|yesno("yeah, nope") }}
{# Result: "yeah" #}

{{ False|yesno("yeah, nope") }}
{# Result: "nope" #}
```

Global Variables Available in Hypr Templates

In addition to the `model` variable and local variables set by `include` tags or JavaScript, Hypr templates have access

to the following global variables:

- pageModel
- siteContext
- pageContext
- navigation
- user
- categories
- themeSettings
- labels
- routeData

You can access these globals inside tags, e.g. `{% if user.isAuthenticated %}` ... , or variables, e.g. `{{ labels.checkout }}` .

pageModel

Accesses the current top-level model data. The model data changes depending on the template. For example, it is different depending on if the current template is for the home page, a category page, a product page, or a widget editor.

To learn about the model data that you can access with `pageModel` for a given template, use the `dump` tag. For example, you can include the following code to output the contents of the model within a template you are editing:

```
{% dump pageModel %}
```

siteContext

Accesses the current site context.

Properties	Type	Description
tenantId	integer	Unique identifier for the tenant.
siteId	integer	Unique identifier for the site.
hashString	string	A string to append to URLs that will change when cache is invalidated, either by a change to catalog or a publish of content.
labels	object	The theme labels, which are key-value pairs used for localization.
themeId	string	Unique identifier for the theme.
		An object that includes the following properties: • `websiteName` (string)—Name of the site as configured in the Admin general settings. • `siteTimeZone` (string)—The site time zone as configured under General Settings in Admin. Stored as human-readable string, e.g. "Mountain Standard Time". • `allowInvalidAddresses` (Boolean)—TRUE if address validation is enabled and invalid addresses are allowed, as configured in the Admin general settings. • `isGoogleAnalyticsEcommerceEnabled`

Properties	Type	Description
generalSettings	GeneralSettings	(Boolean)—TRUE if Google Analytics is enabled and Google Analytics eCom tracking parameters are also enabled. In the Core theme and Core-derived themes, this results in a set of extra calls to the Google Analytics tracking beacon on the Order Confirmation page. • `isGoogleAnalyticsEnabled` (Boolean)—TRUE if Google Analytics is enabled. In the Core theme and Core-derived themes this results in a call to the Google Analytics tracking beacon on every page. • `googleAnalyticsCode` (string)—The UA number provided by the Google Analytics account as configured by the site. • `isAddressValidationEnabled` (Boolean)—TRUE if address validation is enabled as configured in the Admin general settings.
checkoutSettings	CheckoutSettings	An object with the following properties: • `payByMail` (Boolean)—TRUE if pay-by-mail is enabled in the Admin checkout settings. • `isPayPalEnabled` (Boolean)—TRUE if PayPal Express is enabled in Admin checkout settings. • `supportedCards` (Dictionary<string, string>)—List of credit cards enabled in the Admin checkout settings.
themeSettings	object	An object that contains the theme settings available in `theme.json` .
cdnPrefix	string	The URL prefix for CDN content, composed of the host name plus a unique CDN identifier for the site.
secureHost	string	*HTTPS* version of the requested host name.
supportsInStorePickup	Boolean	TRUE if store pickup is enabled in the Admin location settings.
domains	siteDomains	An object that lists the Current domain and the Primary domain, each of which has the following properties: • `domainName` (string)—the name of the domain. • `isPrimary` (Boolean)—TRUE if the given domain is the primary one.
currencyInfo	Currency	An object with the following properties: • `englishName` (string)—the currency name. • `symbol` (string)—the currency symbol. • `precision` (integer)—the number of digits to display after the period. • `roundingType` (string)—the rounding type value: "UpToCurrencyPrecision", "NearestNickel", "DownToNearestNickel", "DownToCurrencyPrecisionMinusOne", or "NearestHalfUnit". • `currencyCode` (string)—for example, "USD" for U.S. dollars. Other values include: "EGP", "GBP", "TZS", "UYU", "UZS", "WST", "YER", "ZMK", "TWD", "GHS", "VEF", "SDG", "RSD", "MZN", and "AZN".

Example:

```
{{ siteContext.generalSettings.isAddressValidationEnabled }}
```

pageContext

Accesses the current page context.

Properties	Type	Description
themeId	string	Unique identifier for the theme.
isDebugMode	Boolean	Indicates whether debug mode is enabled.
cdnCacheBustKey	string	The randomly generated number appended to th CDN content. This number changes in order to re cached content every time a Admin user clicks th Cache button available in the General Settings.
isSecure	Boolean	TRUE if the current page is a secure (HTTPS) page
pageType	string	The documentType of the current page, such "web_page", "cart", "search", etc.
isCrawler	Boolean	TRUE if the current page is requested by a search crawler.
isMobile	Boolean	TRUE if the current page is requested by a mobile
isTablet	Boolean	TRUE if the current page is requested by a tablet.
isDesktop	Boolean	TRUE if the visitor's browser does not identify itse mobile or tablet device.
cmsContext	CmsPageContext	An object with the following the Page , Templ Site , which each have the following properties: • path (string)—name or ID of the CMS doc • documentTypeFQN (string)—the documentType such as "web_page".
search	SearchContext	An object related to URL paging and URL queries collections on Search pages and Category pages. the following properties: • startIndex (int)—The first item in the pa results. • pageSize (int)—The maximum number return in the collection. • query (string)—A filter expression for Kib eCommerce collections. You can filter prod on their properties by writing a string expre your argument. For example: firstname "Brenda" . • sortBy (string)—A string representing ho the collection. You can sort on any property as createDate), or document name. Aft the name of the property you are sorting o space followed by " asc " or " desc " to s sorting in ascending or descending order. • categoryId (int)—The category to facet for. • facets (collection)—A keyed collection o filter on.
title	string	The title of the current page.

Properties	Type	Description
metaDescription	string	The contents of the description field entered into settings for the current page.
metaTitle	string	The contents of the title field entered into the SE(for the current page.
metaKeywords	string	The contents of the keywords field entered into t settings for the current page.

user	User	An object with the following properties:
		
		• isAuthenticated (Boolean)—TRUE if th(logged in.
		• userId (string)—the unique identifier for
		• firstName (string)—the user's first name
		• lastName (string)—the user's last name.
		• email (string)—the user's email address.
		• isAnonymous (Boolean)—TRUE if the u: logged in.
		• accountId (int)—the unique identifier of account.
		• segments (string array)—the list of cust(segments the user belongs to. You can use display or hide content from users based o segments they belong to. For example:
		
		`{% if pageContext.user.segments|contair %}`
		`<div class="silver"><h3>Enjoy yo Benefits!</h3></div>`
		`{% endif %}`

isEditMode	Boolean	TRUE if the current site is being rendered inside a Editor frame. Use this to display preview content or templates that would not work properly in a C(Editor session.
url	string	The canonical URL for the current page.
dataViewMode	string	Either "live" or "pending".
secureHost	string	The fully-qualified secure CDN domain for the sit(
now	DateTime	The current server date/time when the page is vie
categoryCode	string	The category code for the current page if the pag(Category page or Search page.
categoryId	integer	The category ID for the current page if the page is page or Search page.
feedUrl	string	If an RSS feed is configured for this site, this is the that feed.
storefrontOrderAttributes	StorefrontOrderAttributes	A collection of active order attributes.
		An object with the following properties: • isCrawler (Boolean)—TRUE if the curren requested by a search engine crawler. • canonical (string)—The canonical URL, i the start index appended as a URL paramet CMS, product, or category page. If the page of these page types, this property is empty. has a filter applied or if the default sort and parameters are changed, this property is er

Properties	Type	Description
crawlerInfo	CrawlerInfo	• **next** (string)—The URL for the next categ[...] it exists. If the page has a filter applied or if [...] sort and page size parameters are changed[...] property is empty. • **previous** (string)—The URL for the previ[...] category page, if it exists. If the page has a f[...] applied or if the default sort and page size p[...] are changed, this property is empty. • **noIndex** (Boolean)—TRUE if crawlers are [...] from indexing the page in search engine res[...] default this property is not initialized, so it [...] an empty value. • **noFollow** (Boolean)—TRUE if crawlers ar[...] disallowed from following links on the page[...] this property is not initialized, so it may ret[...] empty value.

Example:

```
{{ pageContext.categoryCode }}
```

navigation

Accesses the current navigation context.

Properties	Type	Description
tree	NavigationNode array	The navigation tree used to build the main navigation bar for the site. Consists of a list of navigation nodes, each of which may contain children nodes which also are a list of nodes, and so on. The navigation nodes contain the following properties: • **name** (string)—the name of the node. • **url** (string)—the URL to which the node links. • **isHomePage** (Boolean)—TRUE if the node refers to the homepage. • **index** (int)—the unique identifier for the node. • **isHidden** (Boolean)—TRUE if the node does not display in the site's navigation bar. • parent (NavigationNode)—the parent of the current node, if applicable. • **items** (list of NavigationNode)—the children of the current node, if applicable.
rootCategories	NavigationNode array	This is the same as the `tree`, except it only includes the root-level or top level of the tree, and also excludes CMS pages and external links.
currentNode	NavigationNode	The current node in the tree.
breadcrumbs	NavigationNode array	Shows the "breadcrumbs" that navigate to the current page. This is the same as the `tree` but it excludes the `parent` and `items` properties.

Example:

```
{{ navigation.currentNode.url }}
```

user

Accesses the current user context.

Properties	Type	Description
accountId	integer	The customer ID of the logged-in user.

Properties	Type	Description
userName	string	The username of the logged-in user (generally equal to the email).
email	string	The email of the logged-in user.
firstName	string	The first name of the logged-in user.
lastName	string	The last name of the logged-in user.
isAnonymous	Boolean	TRUE if the current user is not logged in.
isAuthenticated	Boolean	TRUE if the user is logged in.

Example:

```
{{ user.userName }}
```

categories

Accesses the full list of categories for a site.

Combine this variable with the find filter to retrieve a particular category by category code (string) or category ID (number), such as:

```
{{ categories|find('Bicycles') }}
{{ categories|find(107) }}
```

Properties	Type	Description
all	Category array	All the categories for the site.
top	Category array	The top-level categories for the site.

Example:

```
{{ categories.all }}
```

themeSettings

Reflects the contents of the settings collection in the `theme.json` file as well as any runtime overrides established through Admin. You can view the `theme.json` file for the Core theme at https://github.com/Mozu/core-theme/blob/master/theme.json.

labels

Reflects the contents of the JSON file for the current locale in the `labels` directory of the current theme. You can view the `labels` directory for the Core theme at https://github.com/Mozu/core-theme/tree/master/labels.

routeData

Includes key-value pairs related to the route that retrieved the page, including any custom key-value pairs specified in the `"defaults"` object of a custom route or in the `"mapTo"` field of a regex mapping.

The key-value pairs within this variable are useful for rendering custom content through your Hypr template. You can also access this data for use in an API Extension application.

Form Controls

Form controls are inputs that display to Admin users in widget editors, document editors, and theme setting fields. Admin users make selections or enter text in form controls, and you can leverage these selections in other parts of your theme. For example, you can provide a checkbox that lets Admin users choose whether to enable wishlists, and include logic in your Hypr templates that displays wishlists on the storefront depending on the user's selection.

Example
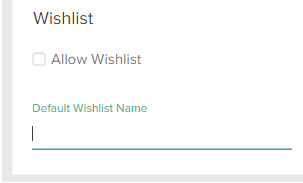
```
{
    "xtype": "panel",
    "title": "Wishlist",
    "collapsed": false,
    "items": [
        {
            "xtype": "mz-input-checkbox",
            "name": "allowWishList",
            "fieldLabel": "Allow Wishlist"
        },
        {
            "xtype": "mz-input-text",
            "name": "wishlistName",
            "fieldLabel": "Default Wishlist Name"
        }
```

```
    ]
}
```

Wishlist

☐ Allow Wishlist

Default Wishlist Name

_____

Access Form Control Values

All form controls (with the exception of `panel`) have an `xtype` field, which specifies the type of form control, a `name` field, which uniquely identifies an instance of a form control, and a `fieldLabel` field, which displays a descriptive label about the control to Admin users. You can access a form control's user-specified values by referencing the `name` field in Hypr templates, stylesheets, or scripts, as explained in the Theme Settings topic.
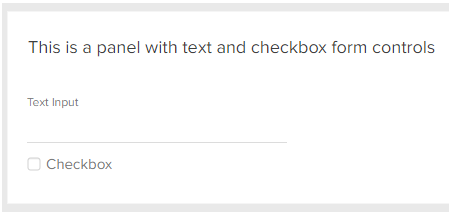
Types of Form Controls

The following is the complete list of form controls:

- Panel
- Date Picker
- Code Editor
- Rich Text Editor
- Text Area
- Text Input
- Dropdown Menu
- Multiselect Menu

- Checkbox
- Number Input
- Advanced Image Input
- Intermediate Image Input
- Basic Image Input
- Product Input
- Product Multiselect

- Category Input
- Category Multiselect
- Discount Input
- Discount Multiselect
- Navigation Node Input
- Navigation Node Multiselect
- Color Picker

Tip: You can hide and display form controls depending on whether users have provided a value for another form control.

Panel

A panel provides a visual grouping and title for other form controls.

This is a panel with text and checkbox form controls

Text Input

_____

☐ Checkbox

Form Control Fields:

`xtype`	`panel`
`title`	A title to display at the top of the panel.
`collapsed`	A Boolean that specifies whether the panel options are collapsed (usually `false`).
`items`	A collection of form controls that display as a group within the panel.

Example Code:

```
{
    "xtype": "panel",
    "title": "This is a panel with text and checkbox form controls",
    "collapsed": false,
    "items": [
        {
            "xtype": "mz-input-text",
            "name": "text",
            "fieldLabel": "Text Input"
        },
        {
            "xtype": "mz-input-checkbox",
            "name": "checkbox",
            "fieldLabel": "Checkbox"
        }
    ]
}
```

Date Picker

A date picker with direct text input and calendar select.

Date Input

mm/dd/yy 12:00... 🔢

Form Control Fields:

xtype	mz-input-date
name	A unique name of your choice.
fieldLabel	A unique label of your choice.
(Optional) emptyText	A string to display in the text input area by default. If you omit this field, the default is blank.

Example Code:

```
{
"xtype": "mz-input-date",
"name": "date",
"fieldLabel": "Date Input"
}
```

Code Editor

A code editor with syntax highlighting

Code Input

```
1 |
```

Form Control Fields:

xtype	mz-input-code
name	A unique name of your choice.
fieldLabel	A unique label of your choice.
(Optional) width	A width in pixels. If you omit this field, the default is `360` .
(Optional) mode	The programming language to use for syntax highlighting (for example, `javascript`). If you omit this field, the default is `html` .

Example Code:

```
{
"xtype": "mz-input-code",
"name": "code",
"fieldLabel": "Code Input"
}
```

Rich Text Editor

A rich text editor with formatting controls and HTML input.

Rich Text Input

B *I* U 🔗 ≔ ≣ <>

Form Control Fields:

xtype	mz-input-richtext
name	A unique name of your choice.
fieldLabel	A unique label of your choice.
(Optional) width	A width in pixels. If you omit this field, the default is `360` .
(Optional) enableAlignments	A Boolean that specifies whether to provide text alignment options. If you omit this field, the default is `false` .
(Optional) enableColors	A Boolean that specifies whether to enable font color and highlighting options. If you omit this field, the default is `false` .
(Optional) enableFont	A Boolean that specifies whether to enable font type options. If you omit this field, the default is `false` .
(Optional) enableFontSize	A Boolean that specifies whether to enable font size options. If you omit this field, the default is `false` .

Example Code:

```
{
"xtype": "mz-input-richtext",
"name": "html",
"fieldLabel": "Rich Text Input"
}
```

Text Area

A multi-line text area input.

Text Area

Form Control Fields:

xtype	mz-input-textarea
name	A unique name of your choice.
fieldLabel	A unique label of your choice.
(Optional) width	A width in pixels. If you omit this field, the default is `360` .
(Optional) emptyText	A string to display in the text input area by default. If you omit this field, the default is blank.

Example Code:

```
{
"xtype": "mz-input-textarea",
"name": "textArea",
"fieldLabel": "Text Area"
}
```

Text Input

A single-line text input.

Text Input

Form Control Fields:

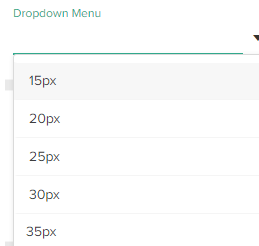| xtype	mz-input-text
name	A unique name of your choice.
fieldLabel	A unique label of your choice.

(Optional) width	A width in pixels. If you omit this field, the default is `260` .
(Optional) emptyText	A string to display in the text input area by default. If you omit this field, the default is blank.

Example Code:

```
{
"xtype": "mz-input-text",
"name": "text",
"fieldLabel": "Text Input"
}
```

Dropdown Menu

A dropdown menu that allows for one selection.
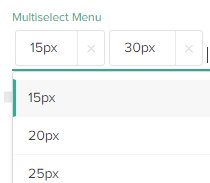

Form Control Fields:

xtype	mz-input-dropdown
name	A unique name of your choice.
fieldLabel	A unique label of your choice.
store	A collection of values for the menu.
(Optional) storeFrom	The `name` of another form control that contains an existing `store` that you want to reuse. If you use this field, omit the `store` field in the current form control (if you leave the field in place, it is ignored).
(Optional) width	A width in pixels. If you omit this field, the default is `260` .
(Optional) typeAhead	A Boolean that specifies whether users can type into the dropdown search box and see filtered results in real time. If you omit this field, the default is `false` .
(Optional) emptyText	A string to display in the text input area by default. If you omit this field, the default is blank.

Example Code:

```
{
"xtype": "mz-input-dropdown",
"name": "dropdown",
"fieldLabel": "Dropdown Menu",
"store": [
      "15px",
      "20px",
      "25px",
      "30px",
      "35px"
   ]
}
```

Multiselect Menu

A dropdown menu that allows for multiple selections.

30px
35px

Form Control Fields:

xtype	mz-input-selectmulti
name	A unique name of your choice.
fieldLabel	A unique label of your choice.
store	A collection of values for the menu.
(Optional) width	A width in pixels. If you omit this field, the default is `260` .
(Optional) typeAhead	A Boolean that specifies whether users can type into the dropdown search box and see filtered results in real time. If you omit this field, the default is `false` .
(Optional) emptyText	A string to display in the text input area by default. If you omit this field, the default is blank.

Example Code:

```
{
"xtype": "mz-input-selectmulti",
"name": "multiselect",
"fieldLabel": "Multiselect Menu",
"store": [
      "15px",
      "20px",
      "25px",
      "30px",
      "35px"
   ]
}
```

Checkbox

A standard checkbox.

☐ Checkbox

Form Control Fields:

xtype	mz-input-checkbox
name	A unique name of your choice.
fieldLabel	A unique label of your choice.

Example Code:

```
{
"xtype": "mz-input-checkbox",
"name": "checkbox",
"fieldLabel": "Checkbox"
}
```

Number Input

An input that only accepts numbers.

Number Input

Form Control Fields:

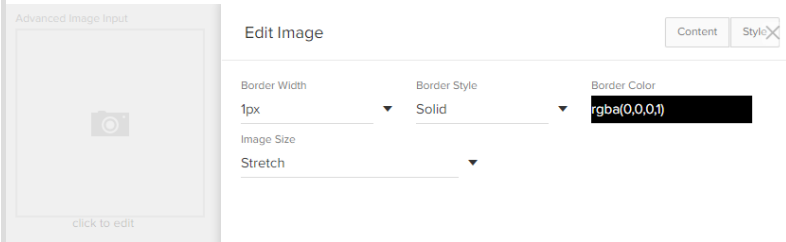| xtype	mz-input-number
name	A unique name of your choice.
fieldLabel	A unique label of your choice.
(Optional) width	A width in pixels. If you omit this field, the default is `260` .
(Optional)	A Boolean that specifies whether users can enter decimal numbers. If you omit this field

(Optional) allowDecimals	A Boolean that specifies whether users can enter decimal numbers. If you omit this field, the default is `false` .
(Optional) emptyText	A string to display in the text input area by default. If you omit this field, the default is blank.

Example Code:

```
{
"xtype": "mz-input-number",
"name": "number",
"fieldLabel": "Number Input"
}
```

Advanced Image Input

An image uploader that also allows editing the properties and style of the image
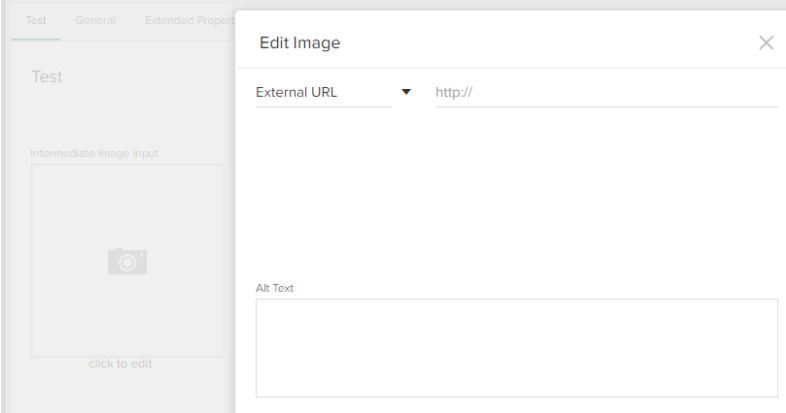

Form Control Fields:

xtype	mz-input-image
name	A unique name of your choice.
fieldLabel	A unique label of your choice.

Example Code:

```
{
"xtype": "mz-input-image",
"name": "advancedImage",
"fieldLabel": "Advanced Image Input"
}
```

Intermediate Image Input

An image uploader that allows editing the properties but not the style of the image.
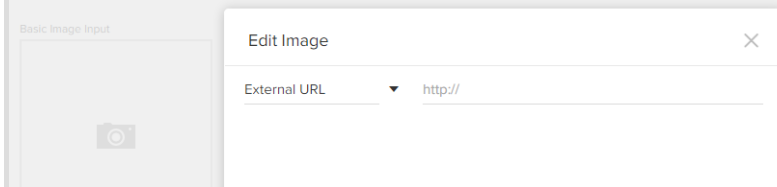

Form Control Fields:

xtype	mz-input-image-nostyle
name	A unique name of your choice.
fieldLabel	A unique label of your choice.

Example Code:

```
{
"xtype": "mz-input-image-nostyle",
"name": "intermediateImage",
"fieldLabel": "Intermediate Image Input"
}
```

Basic Image Input

An input that allows you to upload an image, but does not let you edit the image properties or style.

Basic Image Input

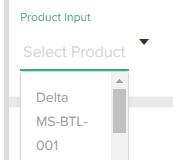Edit Image   ✕

External URL   ▼   http://

Form Control Fields:

xtype	mz-input-imageurl
name	A unique name of your choice.
fieldLabel	A unique label of your choice.

Example Code:

```
{
"xtype": "mz-input-imageurl",
"name": "basicImage",
"fieldLabel": "Basic Image Input"
}
```

Product Input

An input that allows you to select an existing product from a catalog.

Product Input

Select Product   ▼

Delta
MS-BTL-
001

Form Control Fields:

xtype	mz-input-product
name	A unique name of your choice.
fieldLabel	A unique label of your choice.
(Optional) emptyText	A string to display in the text input area by default. If you omit this field, the default is Select Product .

Example Code:

```
{
"xtype": "mz-input-product",
"name": "productPicker",
"fieldLabel": "Product Input"
}
```

Product Multiselect

An input that allows you to select multiple products from a catalog.

Product Multiselect

Delta   ✕   ▼

SoftBottle   ✕
Water
Bottle

Form Control Fields:

xtype	mz-input-productmulti
name	A unique name of your choice.
fieldLabel	A unique label of your choice.
(Optional) emptyText	A string to display in the text input area by default. If you omit this field, the default is Select Products .

Example Code:

```
{
"xtype": "mz-input-productmulti",
"name": "productMultiselect",
"fieldLabel": "Product Multiselect"
}
```

Category Input

An input that allows you to select an existing category from a catalog.

Category Input

[   ] [Add]

Form Control Fields:

xtype	mz-input-category
name	A unique name of your choice.
fieldLabel	A unique label of your choice.

Example Code:

```
{
"xtype": "mz-input-category",
"name": "categoryPicker",
"fieldLabel": "Category Input"
}
```

Category Multiselect

An input that allows you to select multiple categories from a catalog.

Category Multiselect

[Cycling (38) ×] [Jackets (53) ×]   [Add]

Form Control Fields:

xtype	mz-input-categorymulti
name	A unique name of your choice.
fieldLabel	A unique label of your choice.

Example Code:

```
{
"xtype": "mz-input-categorymulti",
"name": "categoryMultiselect",
"fieldLabel": "Category Multiselect"
}
```

Discount Input

An input that allows you to select an existing discount.

Discount Input

[Select Discount   ▼]
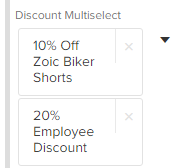
Form Control Fields:

xtype	mz-input-discount
name	A unique name of your choice.
fieldLabel	A unique label of your choice.
(Optional) emptyText	A string to display in the text input area by default. If you omit this field, the default is `Select Discount` .

Example Code:

```
{
"xtype": "mz-input-discount",
"name": "discountPicker",
"fieldLabel": "Discount Input"
}
```

Discount Multiselect

An input that allows you to select multiple discounts.

Discount Multiselect

10% Off Zoic Biker Shorts	×	▼

20% Employee Discount	×

Form Control Fields:

xtype	mz-input-discountmulti
name	A unique name of your choice.
fieldLabel	A unique label of your choice.
(Optional) emptyText	A string to display in the text input area by default. If you omit this field, the default is `Select Discounts` .

Example Code:

```
{
"xtype": "mz-input-discountmulti",
"name": "discountMultiselect",
"fieldLabel": "Discount Multiselect"
}
```

Navigation Node Input

An input that allows you to select a navigation node (such as a page, template, or email template) from your Content Editor tree.

Navigation Node Input

Add

Form Control Fields:

xtype	mz-input-navnode
name	A unique name of your choice.
fieldLabel	A unique label of your choice.
(Optional) emptyText	A string to display in the text input area by default. If you omit this field, the default is `Select Navigation Nodes` .

Example Code:

```
{
"xtype": "mz-input-navnode",
"name": "navNodePicker",
"fieldLabel": "Navigation Node Input"
}
```

Navigation Node Multiselect

An input that allows you to select multiple navigation node (such as a pages, templates, or email templates) from your Content Editor tree.

Navigation Node Multiselect

| Sign Up | × |

| Order Confirmation | × | | | Add |

Form Control Fields:

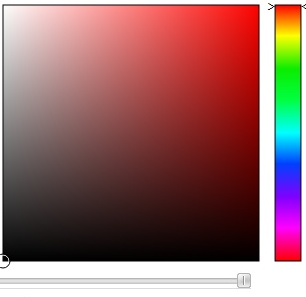| xtype	mz-input-navnodemulti
name	A unique name of your choice.
fieldLabel	A unique label of your choice.
(Optional) emptyText	A string to display in the text input area by default. If you omit this field, the default is `Select Navigation Nodes` .

Example Code:

```
{
"xtype": "mz-input-navnodemulti",
"name": "navNodeMultiselect",
"fieldLabel": "Navigation Node Multiselect"
}
```

Color Picker

A color picker that allows you to specify a color manually or using a GUI.

Color Picker

rgba(0, 0, 0, 1)

General

☑ Show Address Type dropdown in Address

☑ Enable partial caching using the Hypr {%

Extended Properties

☐ Enable Extended Properties (Tracking St

Query Parameters (comma separated)

Select Color ×

rgba(0, 0, 0, 1)

Form Control Fields:

xtype	mz-input-color
name	A unique name of your choice.
fieldLabel	A unique label of your choice.
(Optional) width	A width in pixels. If you omit this field, the default is `260` .

Example Code:

```
{
"xtype": "mz-input-color",
"name": "colorPicker",
"fieldLabel": "Color Picker"
}
```

Hide and Display Form Controls

You can hide, show, enable, or disable form controls using the following Boolean fields:

Field Name	Description
enableIf	Enables the form control if `true` . Otherwise, the form control is disabled (grayed out).
disableIf	Disables (grays out) the form control if `true` . Otherwise, the form control is enabled.

Field Name	Description
shown	Displays the form control if `true`. Otherwise, the form control is not displayed.
hideIf	Hides the form control if `true`. Otherwise, the form control is displayed.

Example

Let's assume you want to enable a checkbox only if a user enters text into a text input. Furthermore, let's assume you only want to display an image input only if the user enables the checkbox. Here's code that would accomplish that:

```
{
    "xtype": "panel",
    "title": "Store Name and Logo",
    "collapsed": false,
    "items": [
        {
            "xtype": "mz-input-text",
            "name": "storeName",
            "fieldLabel": "Store Name"
        },
        {
            "xtype": "mz-input-checkbox",
            "name": "displayLogo",
            "fieldLabel": "Display Logo?",
            "enableIf": "storeName"
        },
        {
            "xtype": "mz-input-image",
            "name": "logo",
            "fieldLabel": "Logo",
            "showIf": "displayLogo"
        }
    ]
}
```

Results

Before the User Completes the Form:

Store Name and Logo

Store Name

☐ Display Logo?

After the User Completes the Form:

Store Name and Logo

Store Name

Virtual Reality Superstore

☑ Display Logo?

Logo

click to edit