

# Order Confirmation Pages (Before)

This action manipulates the HTTP request or response before an OrderConfirmation page loads on the live site.

<b>Action Type</b>	HTTP
<b>Full Action ID</b>	http.storefront.pages.orderConfirmation.request.before
<b>Runs multiple custom functions?</b>	Yes

## JavaScript File Structure

Action files share the following basic structure:

```
module.exports = function(context, callback) {  
  // Your custom code here  
  callback();  
};
```

When you code the custom function for an action, you have access to two arguments:

**callback** —This argument follows the established JavaScript callback pattern: it takes an error as the first argument (or null if there is no error) and a result as the second argument (if required).

**context** —This argument provides the function access to relevant objects and methods that interface with Kibo eCommerce.

## Context: HTTP

The following methods and objects are available to this action through the use of the **context** argument.

### Storefront Operation

This action occurs during the storefront operation that obtains and renders view and model data from Hypr particular to the requested page.

### Context Objects Available to All HTTP Actions

- [request](#)
- [response](#)
- [items.siteContext](#)
- [items.pageContext](#)
- [items.navigation](#)

### Context Methods Available to Storefront Actions

- [exec.loginUser](#)
- [exec.logOut](#)

### Context Methods Available to All HTTP Actions

- [request.get](#)
- [response.get](#)
- [response.set](#)
- [response.set](#)
- [response.remove](#)
- [response.redirect](#)
- [response.end](#)
- [get.resource](#)
- [get.resourceStatus](#)
- [items.urlHelper.getUrl](#)

### Context Objects Available to All Actions

- [apiContext](#)
- [configuration](#)

## Context Objects Available to All HTTP Actions

### request

Accesses the current HTTP request. In the case of Before actions, updates can be made to the request before Kibo eCommerce processes it.

Properties	Type	Description
params	object	The input parameters to the service or webpage.
headers	object	The request headers.
method	string	The request method.
url	string	The request URL.
path	string	The request path.

Properties	Type	Description
cookies	object	The request cookies. *Available only for Storefront HTTP actions.
query	object	The request query.
href	string	The request href.
secure	Boolean	Indicates whether the request uses HTTPS.
ip	string	The request IP address.
ips	string	The request secure IP address.
body	object	The request body of the API operation associated with this action.

Example:

```
context.request.url;
```

## response

Accesses the current HTTP response. For both Before and After actions, updates can be made to the response before Kibo eCommerce processes it.

Properties	Type	Description
header	object	The response header collection.
viewData	object	The viewData collection used by the storefront rendering engine. *Available only for Storefront HTTP actions.
viewName	string	The response viewName value. *Available only for Storefront HTTP actions.
body	object	The response body of the API operation associated with this action.
status	integer	The response status code.
message	string	The response message.
length	integer	The response length.

Properties	Type	Description
type	string	The response type.

Example:

```
context.response.header;
```

### items.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.

Properties	Type	Description
generalSettings	GeneralSettings	<p>An object the includes the following properties:</p> <ul style="list-style-type: none"> <li>• <code>websiteName</code> (string)—Name of the site as configured in the Admin general settings.</li> <li>• <code>timeZone</code> (string)—The site time zone as configured in general settings in admin. Stored as human-readable string, e.g. “Mountain Standard Time”.</li> <li>• <code>allowInvalidAddresses</code> (Boolean)—TRUE if address validation is enabled and invalid addresses are allowed, as configured in the Admin general settings.</li> <li>• <code>googleAnalyticsEcomEnabled</code> (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.</li> <li>• <code>googleAnalyticsEnabled</code> (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.</li> <li>• <code>googleAnalyticsId</code> (string)—The UA number provided by the Google Analytics account as configured by the site.</li> <li>• <code>isAddressValidationEnabled</code> (Boolean)—TRUE if address validation is enabled as configured in the Admin general settings.</li> </ul>

Properties	Type	Description
checkoutSettings	CheckoutSettings	<p>An object with the following properties:</p> <ul style="list-style-type: none"> <li>• <code>payByMail</code> (Boolean)—TRUE if pay-by-mail is enabled in the Admin checkout settings.</li> <li>• <code>isPayPalEnabled</code> (Boolean)—TRUE if PayPal Express is enabled in Admin checkout settings.</li> <li>• <code>supportedCards</code> (Dictionary&lt;string, string&gt;)—List of credit cards enabled in the Admin checkout settings.</li> </ul>
themeSettings	object	An object that contains the theme settings available in <code>theme.json</code> .
isEditMode	Boolean	TRUE if the current site is being rendered inside an editor frame. Use this to display preview content in widgets or templates that would not work properly in an editing session.
cdnPrefix	string	The URL prefix for CDN content, composed of the host name plus a unique CDN identifier for the site.
secureHost	string	<i>HTTPS</i> version of the requested host name.
supportsInStorePickup	Boolean	TRUE if store pickup is enabled in the Admin location settings.
domains	siteDomains	<p>An object that lists the Current domain and the Primary domain, each of which has the following properties:</p> <ul style="list-style-type: none"> <li>• <code>domainName</code> (string)—the name of the domain.</li> <li>• <code>isPrimary</code> (Boolean)—TRUE if the given domain is the primary one.</li> </ul>

Properties	Type	Description
currencyInfo	Currency	<p>An object with the following properties:</p> <ul style="list-style-type: none"> <li><code>EnglishName</code> (string)—the currency name.</li> <li><code>symbol</code> (string)—the currency symbol.</li> <li><code>precision</code> (integer)—the number of digits to display after the period.</li> <li><code>roundingType</code> (string)—the rounding type value: "UpToCurrencyPrecision", "NearestNickel", "DownToNearestNickel", "DownToCurrencyPrecisionMinusOne", or "NearestHalfUnit".</li> <li><code>currencyCode</code> (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".</li> </ul>

Example:

```
context.items.siteContext.siteId;
```

## items.pageContext

Accesses the current page context.

Properties	Type	Description
query	object	Read-only object of the key-value pairs in the current URL.
themeld	string	Unique identifier for the theme.
isDebugMode	Boolean	Indicates whether debug mode is enabled.

Properties	Type	Description
cdnCacheBustKey	string	The randomly generated number appended to the URL of CDN content. This number changes in order to refresh cached content every time a Kibo eCommerce Admin user clicks the Bust Cache button available in the General Settings.
isSecure	Boolean	TRUE if the current page is a secure (HTTPS) page.
pageType	string	The <code>documentType</code> of the current page, such as "web_page", "cart", "search", etc.
isCrawler	Boolean	TRUE if the current page is requested by a search engine crawler.
isMobile	Boolean	TRUE if the current page is requested by a mobile device.
isTablet	Boolean	TRUE if the current page is requested by a tablet.
isDesktop	Boolean	TRUE if the visitor's browser does not identify itself as a mobile or tablet device.
cmsContext	CmsPageContext	<p>An object with the following the <code>Page</code> , <code>Template</code> , and <code>Site</code> , which each have the following properties:</p> <ul style="list-style-type: none"> <li><code>path</code> (string)—name or ID of the CMS document.</li> <li><code>documentTypeFQN</code> (string)—the <code>documentType</code> such as "web_page".</li> <li><code>document</code> (pointer)—a pointer to the CMS document.</li> </ul>



Properties	Type	Description
search	SearchContext	<p>An object related to URL paging and URL queries of product collections on Search pages and Category pages. It contains the following properties:</p> <ul style="list-style-type: none"> <li>• <code>pageSize</code> (int)—the maximum number of items to return in the collection.</li> <li>• <code>query</code> (string)—A filter expression for Kibo eCommerce collections. You can <a href="#">filter products</a> based on their properties by writing a string expression as your argument. For example: <code>properties.firstname eq "Brenda"</code>.</li> <li>• <code>sortBy</code> (string)—A string representing how to sort the collection. You can sort on any property, date (such as <code>createDate</code>), or document name. After specifying the name of the property you are sorting on, include a space followed by "<code>asc</code>" or "<code>desc</code>" to specify sorting in ascending or descending order.</li> <li>• <code>categoryId</code> (int)—the category to facet products for.</li> <li>• A keyed collection of facets to filter on.</li> </ul>
visit	Visit	The unique visitor ID.
title	string	The title of the current page.
metaDescription	string	The contents of the description field entered into the SEO settings for the current page.
metaTitle	string	The contents of the title field entered into the SEO settings for the current page.
metaKeywords	string	The contents of the keywords field entered into the SEO settings for the current page.

Properties	Type	Description
user	User	<p>An object with the following properties:</p> <ul style="list-style-type: none"> <li><code>isAuthenticated</code> (Boolean)—TRUE if the user is logged in.</li> <li><code>userId</code> (string)—the unique identifier for the user.</li> <li><code>firstName</code> (string)—the user's first name.</li> <li><code>lastName</code> (string)—the user's last name.</li> <li><code>email</code> (string)—the user's email address.</li> <li><code>isAnonymous</code> (Boolean)—TRUE if the user is not logged in.</li> <li><code>accountId</code> (int)—the unique identifier of the user's account.</li> </ul>
isEditMode	Boolean	TRUE if the current site is being rendered inside an editor frame. Use this to display preview content in widgets or templates that would not work properly in an editing 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 site.
now	DateTime	The current server date/time when the page is viewed.
categoryCode	string	The category code for the current page if the page is a Category page or Search page.
categoryId	integer	The category ID for the current page if the page is a Category page or Search page.

Example:

```
context.items.pageContext.categoryCode;
```

### items.navigation

Accesses the current navigation context.

Properties	Type	Description
tree	NavigationNode array	<p>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:</p> <ul style="list-style-type: none"> <li>• <code>name</code> (string)—the name of the node.</li> <li>• <code>url</code> (string)—the URL to which the node links.</li> <li>• <code>index</code> (int)—the unique identifier for the node.</li> <li>• <code>isHidden</code> (Boolean)—TRUE if the node does not display in the site's navigation bar.</li> <li>• <code>parent</code> (NavigationNode)—the parent of the current node.</li> <li>• <code>items</code> (list of NavigationNode)—the children of the current node.</li> </ul>
rootCategories	NavigationNode array	This is the same as the <code>tree</code> , 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 <code>tree</code> but it excludes the <code>parent</code> and <code>items</code> properties.

Example:

```
context.items.navigation.breadcrumbs;
```

## Context Methods Available to All Storefront Actions

### exec.loginUser

Asynchronously retrieves a user, creates an authentication ticket, and sets the authentication cookies.

Properties	Type	Description
------------	------	-------------

Properties	Type	Description
config	object	<p>An object that contains the following fields:</p> <ul style="list-style-type: none"> <li><code>userId</code> (string)—The 32-character alphanumeric ID of the user to authenticate.</li> <li><code>userName</code> (string)—The username of the user to authenticate (this field is not required if you provide a <code>userId</code>).</li> <li><code>rememberUser</code> (Boolean)—Specifies whether the authentication cookie should be persistent.</li> </ul>
callback	function	<p>A callback function that contains the following fields:</p> <ul style="list-style-type: none"> <li><code>err</code> An error object if the user is not found.</li> <li><code>data</code> The authentication ticket for the user.</li> </ul>

Example:

```
var config = {
  rememberUser:true,
  userId: '48577d74a86044bfb2872a4c184ce33c'
};

context.exec.loginUser(config, function (err, result){
//handle the possible error and call the callback
//if successful the result should be the customer auth ticket.
  if (err){
    //bubble up error to the platform
    callback (err);
    return;
  }

//return control flow to the platform
callback();
});
```

Response: NA

## exec.logout

Synchronously logs out the current user by resetting the user context to an anonymous user and updating the associated cookies.

N/A	N/A	N/A
Properties	Type	Description

Example:

```
context.exec.logout();
```

Response: NA

## Context Methods Available to All HTTP Actions

### request.get

Returns an HTTP header value for the specified header key.

Parameters	Type	Description
key	string	The header key.

Example:

```
context.request.get(field);
```

Response:

```
"object"
```

### response.get

Gets a field from the response.

Parameters	Type	Description
key	object	The field to obtain.

Example:

```
context.response.get(field);
```

Response:

```
"string"
```

### response.set

Sets the HTTP headers for the response.

Parameters	Type	Description
values	object	The values for the HTTP headers.

Example:

```
context.response.set({ myCustomHeader: "howdy" });
```

Response: N/A

## response.set2

Updates a field in the response.

Parameters	Type	Description
key	string	The key of the field.
value	string	The value of the field.

Example:

```
context.response.set("name", "stuff");
```

Response: N/A

## response.remove

Removes an HTTP header from the response.

Parameters	Type	Description
key	string	The HTTP header to remove.

Example:

```
context.response.remove(header);
```

Response: N/A

## response.redirect

Redirects the incoming URL.

Parameters	Type	Description
url	string	The destination URL.

Example:

```
context.response.redirect("http://someOtherSite/foo");
```

Response: N/A

## response.end

Ends the response so that other actions or Kibo eCommerce logic can run. Also, signals the callback to complete.

Parameters	Type	Description
N/A	N/A	N/A

Example:

```
context.response.end();
```

Response: N/A

## get.resource

Returns the currently persisted value of the requested resource. **\*\*Not available for all calls.**

Parameters	Type	Description
N/A	N/A	N/A

Example:

```
context.get.resource();
```

Response: N/A

## get.resourceStatus

Gets the HTTP resource status.

Parameters	Type	Description
N/A	N/A	N/A

Example:

```
context.get.resourceStatus();
```

Response: N/A

## items.urlHelper.getUrl

Gets the current URL.

Parameters	Type	Description
type		
object		
config		

Example:

```
context.items.urlHelper.getUrl(type, object, config);
```

## Context Objects Available to All Actions

### apiContext

Accesses Kibo eCommerce tenant information.

Properties	Type	Description
baseUrl	string	The base URL for the site.
basePciUrl	string	The base PCI URL for the site.
tenantPod	string	The name of the tenant pod in which the tenant resides.
appClaims	string	The application claims token.
appKey	string	The application key.
tenantId	integer	Unique identifier for the tenant.
siteId	integer	Unique identifier for the site. This ID is used at all levels of a store, catalog, and tenant to associate objects to a site.
masterCatalogId	integer	Unique identifier for the master catalog.
catalogId	integer	The unique identifier for the product catalog. Catalogs are part of a master catalog.
currencyCode	string	The default three-letter ISO currency code for monetary amounts.
previewDate	date/time	The date and time that the content is being viewed. This might be a future date if the content is previewed with an active date range set in the future.



Properties	Type	Description
localeCode	string	The locale code per the country code provided. This code determines the localized content to use and display.
correlationId	string	The unique identifier of the API request associated with the event action, which might contain multiple actions.
isAuthorizedAsAdmin	Boolean	Indicates whether the Dev Account user is authorized as an admin.
userClaims	string	The user claims token.

Example:

```
context.apiContext.baseUrl;
```

## configuration

Receives a JSON response that contains information about the configuration data set in the Action Management JSON editor.

Properties	Type	Description
Varies	object	Custom fields and values that you can set in the Action Management JSON Editor.

Example:

```
context.configuration.customData;
```