

The REST service allows you to retrieve page and object data from Kentico instances.

Send requests using the **GET** [HTTP method](#) to a URL in format: **<REST base URL>/<resource path>**

The base URL of the Kentico REST service is **<site domain name>/rest**. For example, if your site is running at <http://localhost/Kentico>, use <http://localhost/Kentico/rest> as the base URL of the service. To learn about the available **resource paths** for pages and objects, see the tables in the sections below. The requests return data in either XML, JSON, RSS or Atom format (see [Example of data retrieved via the REST service](#) for more information).

REST base URL

Sending a GET request to the base URL of the REST service exposes the service document (for ODATA browsing). The document contains a list of all available primary object types (without child and binding object types) and the URLs under which the objects can be accessed. See also: [ODATA service documents](#)

Character encoding


Data retrieval requests return the results in the server's default character encoding. To get the results in a different encoding type, set the encoding in the **Accept-Charset** field of the GET request's HTTP header. If the specified encoding is not available, the system uses the **Default encoding** configured in **Settings -> Integration -> Rest**.

Example

```
GET http://localhost/Kentico/rest/cms.user/administrator HTTP/1.1
Authorization: Basic UmVzdENsaWVudDpNeVBhc3N3b3Jk
Accept-Charset: utf-8
Content-Type: text/xml
```

Getting page data

To load the data of pages from Kentico websites, send GET requests to the appropriate URL – append the **resource paths** described below to the base URL of your REST service.

 You can further configure the data retrieval by adding [query string parameters](#) to the URL.

Resource format and example	Description
Single page	
/content/currentsite/<culture>/document/<alias path> /content/currentsite/en-us/document/company /careers	A single page in the given culture from the site running on the domain in the base URL. Identify the page using its alias path .
/content/site/<site name>/<culture>/document /<alias path> /content/site/corporatesite/en-us/document /company/careers	A single page in the given culture on the specified site.
Multiple pages	

/content/currentsite/<culture>/all/<alias path> <i>/content/currentsite/en-us/all/news</i>	All pages starting with the specified alias path .
/content/site/<sitename>/<culture>/all/<alias path> <i>/content/site/corporatesite/en-us/all/news</i>	
/content/currentsite/<culture>/childrenof/<alias path> <i>/content/currentsite/en-us/childrenof/news</i>	All child pages under the specified parent page. Does not include the parent page itself.
/content/site/<site name>/<culture>/childrenof/<alias path> <i>/content/site/corporatesite/en-us/childrenof/news</i>	

Culture constants

You can use the following constants instead of culture codes in page REST calls:


- **defaultculture** - the page version in the site's default culture
- **allcultures** - page versions in all available cultures

Example: */content/currentsite/defaultculture/document/company/careers*

Getting object data

To load the data of objects from Kentico, send GET requests to the appropriate URL – append the **resource paths** described below to the base URL of your REST service.

Most object resources start with an **object type** value. To find the value for specific object types, open the **System** application in the Kentico administration interface and select the **Object types** tab.

 You can further configure the data retrieval by adding [query string parameters](#) to the URL.

Resource format and example	Description
Multiple objects	
/<object type> <i>/cms.country</i>	All objects of the given type assigned to the site running on the domain in the base URL. If the object type does not use site bindings, returns all objects of the given type.
/<object type>/site/<site name> <i>/cms.country/site/corporatesite</i>	All objects of the given type assigned to the specified site. If the object type does not use site bindings, returns all objects of the given type.
/<object type>/global <i>/cms.emailtemplate/global</i>	All global objects of the given type (objects not assigned to any site). If the object type does not use site bindings, returns all objects of the given type.

/<object type>/all <i>/cms.emailtemplate/all</i>	<p>All objects of the given type (site-related objects from all sites and global objects).</p> <p>Note:</p> <ul style="list-style-type: none"> • <i>/all</i> object retrieval requests only work if the user account used for authentication has the <i>Global administrator</i> privilege level. • The REST service does not allow <i>/all</i> object retrieval for requests that use the hash URL parameter for authentication. <p>These are intentional security limitations that protect global data.</p>
Single object	
/<object type>/<id> <i>/cms.country/271</i>	<p>Object of the given type with the specified identifier (primary key ID).</p> <p>Ignores site bindings – object IDs are unique across all sites in the system.</p>
/<object type>/<code name> <i>/cms.country/usa</i>	<p>Object of the given type with the specified code name.</p> <p>For object types with site bindings, always returns the object assigned to the site running on the domain in the base URL.</p>
/<object type>/site/<site name> /<object code name or guid> <i>/cms.emailtemplate/site/corporatesite</i> <i>/Blog.NotificationToModerators</i> <i>/cms.emailtemplate/site</i> <i>/corporatesite/e431b7e6-9e6c-409d-a1d9-748cbf51b5d6</i>	<p>Object of the given type with the specified code name or GUID value, assigned to the specified site.</p>
/<object type>/global/<object code name or guid> <i>/cms.emailtemplate/global/Blog.NotificationToModerators</i> <i>/cms.emailtemplate/global/607ed253-7842-48be-98d7-63e0714a6ad1</i>	<p>Global object of the given type with the specified code name or GUID value.</p>
Child objects	
/<object type>/<id>/children <i>/cms.country/271/children</i>	<p>All object types that are supported as child types for the specified object. Only ID values can be used to identify the parent object.</p>
/<object type>/<id>/children/<child object type> <i>/cms.country/271/children/cms.state</i>	<p>All objects of the given type that are children of the specified parent object. Only ID values can be used to identify the parent object.</p>
Binding objects	
/<object type>/<id>/bindings <i>/cms.user/53/bindings</i>	<p>All object types that are used as binding types (represent relationships with other objects) for the specified object. Only ID values can be used to identify the object.</p>



/<object type>/<id>/bindings /<binding object type> <i>/cms.user/53/bindings/cms.usersite</i>	All objects of the given binding type that exist for the specified object. Only ID values can be used to identify the object.
Custom tables	
/cms.customtable/<custom table code name> <i>/cms.customtable/customtable.sampletable</i>	The class definition of the specified custom table (includes data such as the custom table's names, field definitions and search settings).
/customtableitem.<custom table code name> <i>/customtableitem.customtable.sampletable</i>	The data records stored in the specified custom table.
Forms	
/cms.form/<form code name> <i>/cms.form/contactus</i>	The class definition of the specified form (includes data such as the form's names, basic settings and field definitions).
/bizformitem.bizform.<form code name> <i>/bizformitem.bizform.contactus</i>	The data records stored for the specified form.
Other	
/<object type>/site <i>/cms.country/site</i>	Lists all sites on which the specified object type is available and provides the REST URLs under which the object data can be retrieved for individual sites (for ODATA browsing).
/macro/<expression> <i>/macro/CurrentSite.SiteName</i>	<p>Evaluates the given macro expression and serializes the result. Only works for macros that return a serializable result.</p> <p>When creating the macro expression:</p> <ul style="list-style-type: none"> Do NOT add the encapsulating bracket characters Use URL encoding for forbidden URL characters

Data loading parameters

When loading data via REST, you can append the following query string parameters to the request URL:

Parameter	Value (default bold)	Description
General		
format	xml /json /atom10 /rss20	Sets the format of the retrieved data. For example, append <i>?format=json</i> to the request URL to get data in JSON format. See also: Examples of data retrieved via the REST service

localize	culture code	<p>If added, the system resolves all localization expressions inside the returned data. Without this parameter, requests always return localization expressions in unresolved format. Supported by both page and object retrieval requests.</p> <p>Specify the target language by entering the corresponding <i>culture code</i> into the parameter's value. You can find the available culture codes in the Localization application on the Cultures tab.</p> <p>For example, append <i>?localize=fr-fr</i> to the request URL to resolve all localization expressions into their French value (or the value in the default UI culture if the expression is not defined in French).</p>
hash	hash string	<p>Allows you to authenticate the request without requiring an authentication header.</p> <p>See Authenticating REST requests to learn how to generate the hash value.</p>
Pages		
classnames	page type code name (all)	<p>Limits which page types the request returns. Specified as a list of page type code names separated by semicolons.</p> <p>For example, to retrieve only <i>cms.article</i> pages, append <i>?classnames=cms.article</i> to the request URL.</p>
coupleddata	true/false	<p>Determines if the request retrieves data stored in the fields of specific page types (coupled data).</p> <p>The default behavior depends on the type of the page request:</p> <ul style="list-style-type: none"> • Requests for single pages contain coupled data by default • Requests for multiple pages that do not specify the page type (<i>classnames</i> parameter) do NOT load coupled data by default <p>For example, to load pages with coupled data, append <i>?coupleddata=true</i> to the request URL.</p>
combinewithdefaultculture	true/false	<p>Indicates if the request loads the default language versions of pages if they do not exist in the specified culture.</p> <p>To load the default language versions of pages, append <i>?combinewithdefaultculture=true</i> to the request URL.</p>
selectonlypublished	true/false	<p>Determines if the request loads only pages that are published on the live site.</p> <p>To also retrieve unpublished pages, append <i>?selectonlypublished=false</i> to the request URL.</p>
version	published/last	<p>Determines if the request returns the page versions that are published on the live site or the latest version that is being edited in the Pages application (when using Workflow).</p> <p>To load the latest page versions, append <i>?version=last</i> to the request URL.</p>
Multiple pages or objects		

where	SQL code (empty)	<p>SQL WHERE condition for filtering the loaded data. The parameter only allows the following types of SQL syntax:</p> <ul style="list-style-type: none"> column names, values and basic operators: =, !=, >, < AND & OR operators, parentheses column BETWEEN value AND value column LIKE value column IN (values) column IS NULL NOT keyword for the above expressions (NOT BETWEEN, NOT LIKE, NOT IN, IS NOT NULL) <p>Other expressions and SQL functions are not supported.</p> <p>Example: <code>~/rest/cms.user?Where=UserPrivilegeLevel=1</code></p> <p>Note: To avoid ambiguity, ensure that <i>where</i> parameter values are URL encoded. For example, occurrences of the % character in LIKE expressions should be replaced by the encoded equivalent – %25.</p>
orderby	SQL code (empty)	<p>SQL ORDER BY clause for modifying the order of the items in the data. The parameter allows only the following values:</p> <ul style="list-style-type: none"> one or more column names (separated by commas) the ASC and DESC keywords <p>Other SQL expressions and functions are not supported.</p> <p>Append <code>?orderby=##default##</code> to use alphabetical order based on the object display name.</p>
columns	column names (all columns)	<p>Limits which data columns of the object or page are loaded. The parameter value must be one or more column names (separated by commas).</p> <p>For example, to load only the <i>UserName</i> and <i>UserID</i> columns when retrieving users, append <code>?columns=UserName,UserID</code> to the request URL.</p> <p>If you add the <i>columns</i> parameter, the <i>binary</i> parameter is ignored (you can choose whether to include binary columns by enumerating the corresponding columns).</p>
topn	integer (all records)	<p>SQL TOP N clause for filtering the loaded data.</p> <p>For example, to load only the first 10 records, append <code>?topn=10</code> to the request URL.</p>
Multiple objects		
offset	integer (first record)	<p>Sets the number of the first record that the request returns (according to the order of the data). Allows you to implement paging of the data.</p> <p>For example, to load data starting from the third item in the dataset, append <code>?offset=2</code> to the request URL.</p> <p>Note: The <i>offset</i> parameter only works when getting object data (not supported for pages).</p>
maxrecords	integer (all records)	<p>Limits the maximum number of retrieved records. You can use the <i>maxrecords</i> parameter in combination with the <i>offset</i> parameter to retrieve a specific range of records.</p> <p>For example, to load items 11–20 from the data, append <code>?offset=10&maxrecords=10</code> to the request URL.</p> <p>Note: The <i>maxrecords</i> parameter only works when getting object data (not supported for pages).</p>



Objects		
objectdata	true/false	<p>Indicates whether the request retrieves the data fields of objects.</p> <p>To load only the metadata of an object, append <i>?objectdata=false&metadata=true</i> to the request URL.</p>
metadata	true/false	<p>Determines if the request retrieves the metadata of objects (type, list of properties / columns).</p> <p>To load the metadata, append <i>?metadata=true</i> to the request URL.</p>
binary	true/false	<p>Indicates whether the request retrieves binary data (e.g. the data of files uploaded into form fields). Binary data is retrieved in <i>Base64</i> format.</p> <p>To include the binary data in the response, append <i>?binary=true</i> to the request URL.</p>
children	true/false	<p>Indicates whether the result includes child objects.</p> <p>To load child objects, append <i>?children=true</i> to the request URL.</p>
maxrelative level	integer (all levels)	<p>If the <i>children</i> parameter is true, this parameters sets the maximum depth of the exported object tree structure.</p> <p>For example, to load all child objects down to the second level of the object tree, append <i>?children=true&maxrelativelevel=2</i> to the request URL.</p>
bindings	true/false	<p>Determines if the retrieved data includes bindings to child objects and sites. Requests only return bindings if the <i>objectdata</i> parameter is true.</p> <p>To load object data with its bindings, append <i>?bindings=true</i> to the request URL.</p>
otherbindings	true/false	<p>Determines if the retrieved data includes bindings to other objects (M:N relationships). Requests only return bindings if the <i>objectdata</i> parameter is true.</p> <p>To load object data with bindings to other objects, append <i>?otherbindings=true</i> to the request URL.</p>
metafiles	true/false	<p>Determines if the retrieved data includes metafiles attached to the object</p> <p>To load metafiles, append <i>?metafiles=true</i> to the request URL.</p>
relationships	true/false	<p>Determines if the retrieved data includes object relationships.</p> <p>To load relationships, append <i>?relationships=true</i> to the request URL.</p>
categories	true/false	<p>Determines if the retrieved data includes the object's category structure (if the object is stored in a hierarchical category structure).</p> <p>To load the category structure, append <i>?categories=true</i> to the request URL.</p>
translations	true/false	<p>Indicates whether the result includes a translation table of foreign keys.</p> <p>To load the translation table, append <i>?translations=true</i> to the request URL.</p>
hierarchy	true/false	<p>Only applies when loading a single object, with at least one of the following parameters enabled: <i>children</i>, <i>bindings</i>, <i>translations</i></p> <p>If true, the response data is exported in a hierarchical structure. If false, the children– bindings– parent structure is flat.</p> <p>To export the data in a hierarchical structure, append <i>?hierarchy=true</i> to the request URL.</p>