

# MediaWiki & Cargo quick reference (1 of 2)

Covers MediaWiki 2.25 and MediaWiki extensions ParserFunctions 1.6, Cargo 1.3 and Page Forms 4.1

## MediaWiki syntax

```
"italics"
"bold"
""bold and italics""
==standard header==
===next-level header=== (...and so on)
[[Internal link]]
[[Internal link|alternate text]]
[http://example.com text of external link]
[[Category:Example]] (category tag)
[:Category:Example]] (link to category)
---- (horizontal line)
* bulleted item
# numbered item
:indentation
::double indentation (...and so on)
;term : definition
[[File:Image-name.jpg|thumb|frame|Caption text]]
{{Template name}} (call to template)
#REDIRECT [[Page name]]
```

## Parser functions

From *MediaWiki*: PAGENAME, FULLPAGENAME, localurl, urlencode, formatnum, #formatdate, lc, lfirst, uc, ucfirst, padleft, padright, #tag, !, etc.  
From *ParserFunctions*: #expr, #if, #ifeq, #iferror, #ifexpr, #ifexist, #rel2abs, #switch, #time, #timel, #titleparts;  
*originally from StringFunctions, now in ParserFunctions*: #len, #pos, #rpos, #sub, #pad, #replace, #explode

## Queries

**Special pages:** ViewData

**Syntax:** `{{#cargo_query:tables=|join on=|fields=|where=|group by=|having|order by=|limit=|default=|intro=|outro=|more results text=|format=|...format params...}}`

**The "where=" parameter:**

Holds a value similar to the SQL "WHERE" clause, but with extra allowed keywords:

**HOLDS** - used for a list of values

**HOLDS LIKE** - for a list of values; sample call is "Authors HOLDS LIKE '%Smith%'"

**NEAR** - used for coordinates; sample call is "CoordsField NEAR (45.434, 12.338, 10 km)" ("mi" can also be used)

**Sample query:** To create a table of all cities in Europe and their populations, sorted by population:

```
{{#cargo_query:tables=Cities,Countries
|join on=Cities._pageName=
Countries._pageName
|fields=Cities._pageName=City,
Cities.Population
|where=Countries.Continent="Europe"
|order by=Cities.Population}}
```

## Templates and data storage

**Special pages:** CargoTables, Drilldown, Templates, CreateTemplate

**In a template's <noinclude> tag:**

```
{{#cargo_declare:_table=tableName
|field1=fieldType1|field2= fieldType2|...}}
```

...or, to use a table declared elsewhere:

```
{{#cargo_attach:_table=tableName}}
```

**In a template's <includeonly> tag:**

```
{{#cargo_store:_table=tableName|field1=value1
|field2=value2|...}}
```

**Field types:** Page, String, Text, Integer, Float, Date, Datetime, Boolean, Coordinates, Wikitext, Searchtext, File, URL, Email

**Setting a field to display multiple values:**

```
{{#arraymap:value|delimiter|var|formula
new_delimiter}}
{{#arraymaptemplate:value|template-name|
delimiter|new_delimiter}}
```

**Sample template call:** For the page "Template:Restaurants":

```
<noinclude>This is the "Restaurants" template.
{{#cargo_declare:_table=Restaurants|
Address=Text|Cuisines=List (,) of Page}}
</noinclude><includeonly>
{{#cargo_store:_table=Restaurants|Address=
{{{Address}}}|Cuisines={{{Cuisines}}}}
""Address:"" {{{Address}}}}

""Cuisines:"" {{#arraymap:{{{Cuisines}}}|,|[[x]]}}
<includeonly>
```

**Running a compound query:**

```
{{#cargo_compound_query:query1|query2|...|
joint parameters}}
```

Each query's parameters should be separated by semicolons, instead of pipes.

**Sample compound query:** To show a map with one icon for clothing stores and another for fast food restaurants:

```
{{#cargo_compound_query:tables=Clothing
_stores;fields=Coordinates;icon=Shirt.png
|tables=Restaurants;fields=Coordinates
;where=Cuisines HOLDS "Fast food"
;icon=Hamburger.png|format=googlemaps}}
```

### Settings for LocalSettings.php

```
$wgCargoDefaultQueryLimit - limit of query
results when "limit" not set; default is 100
$wgCargoMaxQueryLimit - true limit of
query results; default is 500
$wgCargoDecimalMark - separator
character before decimals; default is "."
$wgCargoDigitGroupingCharacter -
numbers grouping character; default is ",",
```

**String value for a recurring event:**

```
{{#recurring_event:start=start date|end=end date
|period=number|unit={day|week|month|year}
|week number=number|include=[exclude=]}}
```

**Extra fields stored per table:**

```
_pageID, _pageTitle, _pageNamespace
For coordinates: fieldName__lat, fieldName__lon
```

### Notable settings for LocalSettings.php

```
$wgCargoDbType, $wgCargoDBserver,
$wgCargoDBname, $wgCargoDBuser,
$wgCargoDBpassword - set the Cargo DB
$wgCargo24HourTime - set to true to display
time values in 24-hour notation
```

## Display formats

**list** (default format if one field) - delimited list of values. *Parameter: delimiter*

**ul** - bulleted vertical list

**ol** - numbered vertical list

**category** - display in the manner of MediaWiki category pages

*ul, ol and category allow the "columns" param.*

**template** - like list, but "template" parameter required

**embedded** - full pages displayed

**outline** - hierarchical display of properties

*Parameter: outline fields*

**tree** - shows pages connected via a field relationship. *Parameter: parent field*

**table** - (default if multiple fields) table of values, with field names as column headers

**dynamic table** - filterable table of values, using the DataTables JavaScript library

**tag cloud** - tag cloud display, based on frequency of values

*Params: min size, max size, template*

**gallery** - display of images in a gallery

*Param: mode (values are traditional, packed, packed-overlay and packed-hover)*

**calendar** - JS-based monthly calendar

*Params: width, height, start date, view, color, text color*

**timeline** - scrolling JavaScript-based timeline

**bar chart** - bar chart, using the jqPlot JS library

**googlemaps** - maps, using Google Maps

**openlayers** - maps, using OpenLayers

*Params for maps: height, width, zoom, icon*

**csv** - CSV (comma-separated values) format

*Param: delimiter*

**json** - JSON format

**excel** - Excel/XLS file format

*Export formats also allow "link text", "filename" parameters.*

## Non-query maps

```
{{#cargo_display_map:point=|service=|width=
|height=|zoom=}}
```

# MediaWiki & Cargo quick reference (2 of 2)

Covers MediaWiki extensions Page Forms 4.1 and External Data 1.8

## Page Forms

**Special pages:** CreateForm, Forms, CreateClass, RunQuery

**Simple form definition:**

```
{{for template|template-name}}
field-label: {{{field|field-name}}}
{{{end template}}}
```

Free text:  
{{{standard input|free text}}}  
{{{standard input|save}}}

**Form tags:**

info field  
for template standard input  
end template

**'info' parameters:**

create title= partial form  
edit title= page name=  
query title= query form on top

**'for template' parameters:**

label= strict  
multiple add button text=  
embed in field= display=

**'field' parameters:**

input type=, class=, list, delimiter=,  
hidden, mandatory, restricted,  
default=, preload=, holds template,  
unique, unique for category=,  
unique for namespace=, cargo  
field=, cargo table=

*...for text inputs:*

size=, uploadable, image preview

*...for textarea inputs:*

autogrow, rows=, cols=, editor=

*...for 'tree':*

top category=, structure=, hideroot,  
depth=, height=, width=

*...for 'combobox' and 'tokens':*

existing values only

*...for enums and autocomplete:*

values=, values from category=,  
values from namespace=

*...for enums:*

show on select=*value1*=>*div-ID1*;*value2*=>*div-ID2*

*...for autocomplete:* remote

autocomplete, values dependent  
on=, values from url=

**'standard input' types:**

free text preview  
minor edit changes  
watch cancel  
summary run query  
save save and continue

**'standard input' parameters:**

label= class=

### Input types

Field type	Allowed input types (default is bolded)
Page	<b>combobox</b> , text with autocomplete, text, tree
String, File, Email, URL, etc.	<b>text</b> , text with autocomplete, combobox, textarea (with autocomplete), regexp
Integer	<b>text</b> , rating
Text, Wikitext	<b>textarea</b> , etc.
Date	<b>date</b> , year, datepicker
Datetime	<b>datetime</b> , datetimepicker
Enumeration	<b>dropdown</b> , radiobutton
Boolean	<b>checkbox</b>
Coordinates	<b>openlayers</b> , googlemaps
List of Page	<b>tokens</b> , text (with auto.), textarea (with auto.), tree
List of String, etc.	<b>text</b> , text with autocomplete, textarea (with auto.), tokens
List of Text, etc.	Same as above, but default is <b>textarea</b>
List of enums	<b>checkboxes</b> , listbox

**Form input:**

```
{{#forminput:form=|size=|default value=|button  
text=|query string=|popup|autocomplete on  
category=|autocomplete on namespace=  
|remote autocomplete|query string params}}
```

**Form input query string options:**

(values are separated by '&' if they're part of  
"query string=", or '|' if they're regular params)  
*template\_name*[*field\_name*]=  
namespace= super\_page=

**Setting the 'edit with form' tab:**

In a namespace page (Project:*namespace-name*),  
category page, regular page or template:  
{{#default\_form:*form-name*}}

### Forms that set an automatic page name

**In form definition:**

```
{{{info|page name=...<template-name[field-  
name]|> ... <unique number>}}}
```

**Linking to the form:**

```
{{#formlink:form=|link text=|link type=|query  
string=|target=|popup|query string params}}
```

**Linking to Special:RunQuery:**

```
{{#queryformlink:form=|link text=|existing page  
link text=|link type=|query string=|popup  
|query string params}}
```

**Pointing red links to forms:**

```
{{#formredlink:form=|link text=|link type=|existing  
page link text=|query string=|target=|popup  
|query string params}}
```

**A link to edit a page automatically:**

```
{{{#autoedit:form=|target=|link text=|link type=|  
query string=|reload}}}
```

### Notable settings for LocalSettings.php

\$wgPageFormsRenameEditTabs - set to true to  
change 'edit with form' to 'edit', 'edit' to 'edit source'  
\$wgPageFormsRenameMainEditTab - set to true  
to change 'edit' to 'edit source'  
\$wgAmericanDates - set to true to show month  
first in date input  
\$wgPageForms24HourTime - set to true to display  
time input in 24-hour notation

## External Data

**Getting data from a URL:**

```
{{#get_web_data:url=URL|format=format|data=data  
mappings|filters=filters|delimiter=delimiter|use  
xpath|post data=additional data|cache seconds=  
num}}
```

Allowed formats: CSV, JSON, XML, GFF

```
{{#get_soap_data:url=URL|request=request function  
|request data=param1=values1,...|response=data  
retrieval function|data=data mappings}}
```

**Getting data from a file:**

```
{{#get_file_data:file=file ID|directory=directory ID|file  
name=file name|format=format|data=data  
mappings|filters=filters|use xpath|cache  
seconds=num}}
```

Either "file", or the combination of "directory" and "file  
name", should be specified.

Allowed formats: CSV, JSON, XML, GFF

**Getting data from a database:**

```
{{#get_db_data:server=server ID|from=from clause  
|where=where clause|data=data mappings}}
```

For MongoDB, there are also "find query",  
"aggregate" params.

**Getting data from an LDAP server:**

```
{{#get_ldap_data:domain=domain ID|filter=LDAP  
filter|data=data mappings}}
```

"data", "filters" values in all of these should be  
comma-separated.

**Displaying external data:**

```
{{#external_value:variable_name}}  
{{#for_external_table:expression}}
```

The expression value in #for\_external\_table should  
contain variables of the form {{{*variable\_name*}}}.

**Displaying and storing a table of values:**

```
{#display_external_table:template=template|  
data=data mappings|delimiter=delimiter}}
```

If template is Cargo-enabled, it will store the data.

### Notable settings for LocalSettings.php

\$edgStringReplacements - array for hiding API keys  
\$edgAllowExternalDataFrom - array for an API  
"whitelist" - necessary if API keys are being hidden  
\$edgCacheTable - database table for caching data  
DB and LDAP access have other required settings.