



Inhaltsverzeichnis

Modul:Check for unknown parameters/Doku

Dies ist die Dokumentationsseite für **Modul:Check for unknown parameters**

Vorlage:Lua

This module may be appended to a template to check for uses of unknown parameters.

Inhaltsverzeichnis

1 Usage	2
1.1 Basic usage	2
1.2 Lua patterns	3
2 Example	3
3 See also	3

Usage

Basic usage

```
{{#invoke:check for unknown parameters|check
|unknown=[[Category:Some tracking category]]
|arg1|arg2|arg3|argN}}
```

or to sort the entries in the tracking category by parameter with a preview error message

```
{{#invoke:check for unknown parameters|check
|unknown=[[Category:Some tracking category|_VALUE_]]
|preview=unknown parameter "_VALUE_"
|arg1|arg2|...|argN}}
```

or for an explicit red error message

```
{{#invoke:check for unknown parameters|check
|unknown=<span class="error">Sorry, I don't recognize _VALUE_</span>
|arg1|arg2|...|argN}}
```

Here, `arg1`, `arg2`, ..., `argN`, are the known parameters. Unnamed (positional) parameters can be added too: `|1|2|argname1|argname2|...`. Any parameter which is used, but not on this list, will cause the module to return whatever is passed with the unknown parameter. The `_VALUE_` keyword, if used, will be changed to the name of the parameter. This is useful for either sorting the entries in a tracking category, or for provide more explicit information.

By default, the module makes no distinction between a defined-but-blank parameter and a non-blank parameter. That is, both unlisted `Vorlage:Para` and `Vorlage:Para` are reported. To only track non-blank parameters use `Vorlage:Para`.

By default, the module ignores blank positional parameters. That is, an unlisted **Vorlage:Para** is ignored. To *include* blank positional parameters in the tracking use **Vorlage:Para**.

Lua patterns

This module supports **Lua patterns** (similar to **regular expressions**), which are useful when there are many known parameters which use a systematic pattern. For example, **template:infobox3cols** uses

```
| regexp1 = header[%d][%d]*  
| regexp2 = label[%d][%d]*  
| regexp3 = data[%d][%d]*[abc]?  
| regexp4 = class[%d][%d]*[abc]?  
| regexp5 = rowclass[%d][%d]*  
| regexp6 = rowstyle[%d][%d]*  
| regexp7 = rowcellstyle[%d][%d]*
```

to match all parameters of the form headerNUM, labelNUM, dataNUM, dataNUMa, dataNUMb, dataNUMc, ..., rowcellstyleNUM, where NUM is a string of digits.

Example

```
{{Infobox  
| above = {{{name|}}}  
  
| label1 = Height  
| data1 = {{{height|}}}  
  
| label2 = Weight  
| data2 = {{{weight|}}}  
  
| label3 = Website  
| data3 = {{{website|}}}  
}}<!--  
end infobox, start tracking  
-->{{#invoke:Check for unknown parameters|check  
| unknown = {{main other|[[Category:Some tracking category|_VALUE_]]}}  
| preview = unknown parameter "_VALUE_"  
| name  
| height | weight  
| website  
}}
```

See also

- **Vorlage:Clc** (category page can have header **Vorlage:TI**)
- **Module:Check for deprecated parameters** – similar module that checks for deprecated parameters
- **Module:Check for clobbered parameters** – module that checks for conflicting parameters
- **Module:TemplatePar** – similar function (originally from dewiki)
- **Template:Parameters** and **Module:Parameters** – generates a list of parameter names for a given template



- **Project:TemplateData** based template parameter validation
- **Module:Parameter validation** checks a lot more
- **User:Bamyers99/TemplateParametersTool** - A tool for checking usage of template parameters