

Automated Menu Creation

This tutorial shows you how to use the [AtlassianConfluence:Macros] to automatically generate self-updating menus...

There are three scenarios where you might want to automate the structure of your menus:

- You have a big menu structure that matches the structure of content in your [space]
- You want to allow other people to add content and have the menus automatically updated
- You're lazy (a bit like me) and don't want to type loads of [AtlassianConfluence:Wiki Markup Editor] to create your menus

If any of those describe you, read on 😊

But first, there are some scenarios where you might want to *avoid* automating menus:

- Automated menus can't have icons in them (yet... we're working on that feature)
- Automated menus have limited ordering options (alphabetically, date, etc) - if you want a very specific order to items, you'll have to manually add them.
- You can't define custom on-screen text or tool tips when using macros (unless they specifically support such a feature)

So, assuming that you still want to automate your menus, let's begin...

Macros to the Rescue

You can use **any** macro that outputs an unordered bullet list of items within a menu. 😊 Let's take a look at two of them...

{children} macro

This is by far the most common macro to make it's way in to the menu settings:

```
{menu}[Builder]
  {children}
{menu}
```

The [AtlassianConfluence:children macro] basically creates a bulleted list of child pages from a given starting point.

To ensure that children are listed from a specific starting point, use the *page* parameter as follows:

```
{menu}[Builder]
  {children:page=Builder}
{menu}
```

Voila! That's exactly

Unknown macro: {footnote}

Well, not quite. We use the *depth=2* parameter of the children macro to automatically create sub-menus for any pages that themselves have children.

how the *Builder* menu and most other menus are created at the top of pages in this user guide.

Unknown macro: {display-footnotes}

{incoming-links} macro

The [AtlassianConfluence:incoming-links macro] lists pages that link to the current page. As such, if you're on a page that nothing links to, it won't output anything

Unknown macro: {footnote}

The incoming-links macro actually allows you to specify what to output if there are no incoming links:

```
{incoming-links:style=disc}* No incoming links{incoming-links}
```

. However, as something does link to this page, we will get some output:

```
{menu}[Automated Menu Creation]
  {incoming-links:style=disc}{incoming-links}
{menu}
```

As you can see, this macro needs to be told to output as a bulleted list (by default it outputs with icons, but not the sort that are usable in the menu) and because the macro can have a body we've had to include the closing tag as well.

On screen, for this specific page, the macro would display:

Unknown macro: {menu}

[Automated Menu Creation](#)

Unknown macro: {incoming-links}

Which produces the menu:

Unknown macro: {incoming-links}

Unknown macro: {menu}

[Automated Menu Creation](#)

Unknown macro: {incoming-links}

Our Navigate Menu

While writing this tutorial, we got a bit carried away playing around with other macros and came up with the idea of a navigate menu:

```
{menu}Navigate
  {submenu}Incoming Links
    {incoming-links:style=disc}* No incoming links{incoming-links}
  {submenu}
  {menuseparator}
  {submenu}Page Headings
    {wikimenu}{toc:maxLevel=5}{wikimenu}
  {submenu}
  {submenu}Child Pages
    {children}
  {submenu}
{menu}
```

Be warned, if you add this menu, it can get very big! We've included it here as an example of the different types of macros that can be used.

Have fun!