

Token Reference Guide

Tokens are the key to integrating dynamic content management functionality with your custom website designs. Tokens are small bits of code that our servers replace with dynamic content and functionality when the site is run on our system. The content that each token outputs can often be styled through your CSS files to meet your design objectives.

This guide is designed to help you understand and use tokens correctly and efficiently. For more information and a visual reference guide please visit <http://support.publishpath.com/token-reference-guide>.

Editable Region

Token: <\$region\$> Optional default content here <\$/region\$>

Purpose: This token defines an editable content region in your template, allowing logged-in users to manage the content in the region using our system's easy to use tools.

Attributes

- name= [required] each region must be given a unique name.

IMPORTANT:

- Every region **MUST HAVE** a unique name
- One region on every template must be named "MainContent". This should generally be the primary region and it will be used as the default region by the system for your template.

Example Usage: Create one or many editable regions on your template page. If you want default content to appear in the region in addition to any content added by the user, add that default content between the opening and closing tags. Default content is optional and is most often left blank:

```
<$region name="MainContent"$>
```

This is default content which will always appear in addition to any content added by the user. This content is optional and is most often left blank.

```
<$/region$>
```

Root Menu

Token: <\$rootmenu/\$>

Purpose: This token outputs an unordered list of hyperlinked menu items for each top-level page in your site. Top level pages are those pages whose parent page is "Home."

Attributes:

- `cssClass=` used to specify your own custom CSS class for use in styling the menu list. This class is attached to the `` tag.
- `showHome=` determines whether or not "Home" is output as the first item in the menu item list. This can be set as "true" or "false." Default is "true."

CSS:

- The CSS class, "selectedPageRoot," is output in both the list item `` and link `<a>` tags surrounding the menu item that represents the current page's root. This allows you to style the current page's root menu item differently from the rest of the menu items in the navigation list.
- The CSS class, "selectedPage," is output in both the list item `` and link `<a>` tags surrounding the menu item that represents the current page. This allows you to style the current page's menu item differently from the rest of the menu items in the navigation list.

Example Usage: Use this token to create a top-level navigation menu for your website. Use CSS to style the resulting unordered list of links as desired:

```
<$rootmenu cssClass="topNav" showHome="false"/$>
```

Section Menu

Token: <\$sectionmenu/\$>

Purpose: This token outputs an unordered list of hyperlinked menu items for each page in the current section only. This can have two possible results:

- If the current page does NOT have any child pages, the section menu outputs links to pages on the same level with the current page (its sibling pages).
- If the current page does have at least one child page, the section menu outputs links to the child pages of the current page.

Attributes:

- `cssClass=` used to specify your own custom CSS class for use in styling the menu list. This class is attached to the `` tag.

CSS:

- The CSS class, "selectedPage," is output in both the list item `` and link `<a>` tags surrounding the menu item that represents the current page. This allows you to style the current page's menu item differently from the rest of the menu items in the navigation list.
 - NOTE: if the current page has child pages, this styling will have no effect because the name of the current page will not appear on the list.

Example Usage: Use this token to create section menu on your site pages. Typically, the pages should also contain a root menu and a breadcrumb for maximum ease of navigation. Use CSS to style the resulting unordered list of links as desired:

```
<$sectionmenu cssClass="sectionNav"/$>
```

Breadcrumbs

Token: <\$breadcrumbs/\$>

Purpose: This token outputs a string of text links creating a hierarchical path from the home page to the current page. You can use a breadcrumb in conjunction with a menu system to provide users with an extra level of navigation through your site.

Attributes:

- `cssClass=` used to specify your own custom CSS class for use in styling the breadcrumb string. This class is attached to a `<div>` tag surrounding the string of links.
- `separator=` used to specify characters or symbols which will separate each link in the breadcrumb string. The default separator is "..."

CSS:

- The CSS class, "selectedPageRoot" is output in the link `<a>` tags surrounding the home page item.
- The CSS class, "selectedPageParent" is output in the link `<a>` tags surrounding all other items except for the home page and the current page..

Example Usage: Use this token to create a breadcrumb of links to help the user navigate through the site. The breadcrumb is often found at the top of the main content area.

```
<$breadcrumbs cssClass="bc" separator="/" />
```

Search

Token: <\$search/\$>

Purpose: This token outputs a search form that allows the user to enter a search term and click a button to search your site for content. Matches are displayed on an automatically generated search results page using the default template selected for your website. Simply add this token to your templates and full search functionality is implemented on your site.

Attributes:

- `buttonText=` specifies the text that appears on the submit button for your search form.
- `labelText=` specifies the text that appears as a label for the input field on your search form.

CSS: The search form output is wrapped in a `<div>` tag with the class of "search". The `labelText` is rendered as a ``, the text `<input>` is classed as "searchInput", and the submit button `<input>` is classed as "searchButton".

Example Usage: Use this token to create a search form on your site.

```
<$search buttonText="Click to Search" labelText="Search this Website"/$>
```

Website Name

Token: <\$websitename/\$>

Purpose: This token displays the name of the website as saved in the website settings.

Attributes: None

Example Usage: Create a button that displays the name of the website and links it back to your home page.

```
<a href="/"><$websitename/$></a>
```

Page Title

Token: <\$pagetitle/\$>

Purpose: This token displays the name of the current page.

Attributes: None

Example Usage: Display the name of each page automatically as a header at the top.

```
<h1><$pagetitle/$></h1>
```

Date

Token: <\$date/\$>

Purpose: This token displays the current date in the following format: Thursday, August 17th, 2006

Attributes: None

Example Usage: Add the current date anywhere on the page.

```
<p><$date/$></p>
```

Global Menu

Token: <\$globalmenu/\$>

Purpose: This token outputs a set of nested unordered lists of hyperlinked menu items displaying the site's root menu and expanding down through the page structure to the current page plus the current page's children, if any. This allows you to create a single, nested menu structure for the entire site. As the user clicks a page in the menu, the menu expands to display that page's child pages, while keeping the rest of the site's root menu in view and available for navigation.

Attributes:

- `cssClass=` used to specify your own custom CSS class for use in styling the menu list. This class is attached to the top-level `` tag.
- `showHome=` determines whether or not "Home" is output as the first item in the menu item list. This can be set as "true" or "false." Default is "true."
- `alwaysShowChildren=` if set to "true," the token always outputs a complete set of nested, unordered lists showing every page in the site's structure at all times. This is mainly used to create CSS-styled drop-down or fly-out menus. Default is "false".

CSS:

- The CSS class, "selectedPage" is output in both the list item `` and link `<a>` tags surrounding the menu item that represents the current page.
- The CSS class, "selectedPageRoot" is output in both the list item `` and link `<a>` tags surrounding the menu item that represents the current page's root page (the page on the root menu under which the current page is found).
- The CSS class, "selectedPageParent" is output in both the list item `` and link `<a>` tags surrounding the menu item that represents any parent page of the current page IF the parent page is not the same page as the root page.

Example Usage: Use this token to create a single navigation menu on your site that expands and contracts in a nested manner as the user navigates through the pages. It is not recommended to use this token on the same template with the Root Menu token or the Section Menu token as the pages displayed by the Global Menu token will replicate pages on the other menu tokens. Generally, a template should either employ a combination of Section and Root Menus, or a single instance of Global Menu. Create a global menu as follows:

```
<$globalmenu cssClass="globalNav" showHome="false"/$>
```

You may also use the Global Menu token to generate a complete list of all pages in the site for use in creating a CSS-styled drop down or fly-out menu system, by employing the alwaysShowChildren attribute as follows:

```
<$globalmenu cssClass="globalNav" showHome="true" alwaysShowChildren="true"/>
```

This document was created with Win2PDF available at <http://www.win2pdf.com>.
The unregistered version of Win2PDF is for evaluation or non-commercial use only.
This page will not be added after purchasing Win2PDF.