

Source Editor

Overview

By default, level 9 or level 10 administrators have the ability to edit the source code of a file via the Source Editor.

Source Editor privileges can be granted or restricted for users as follows:

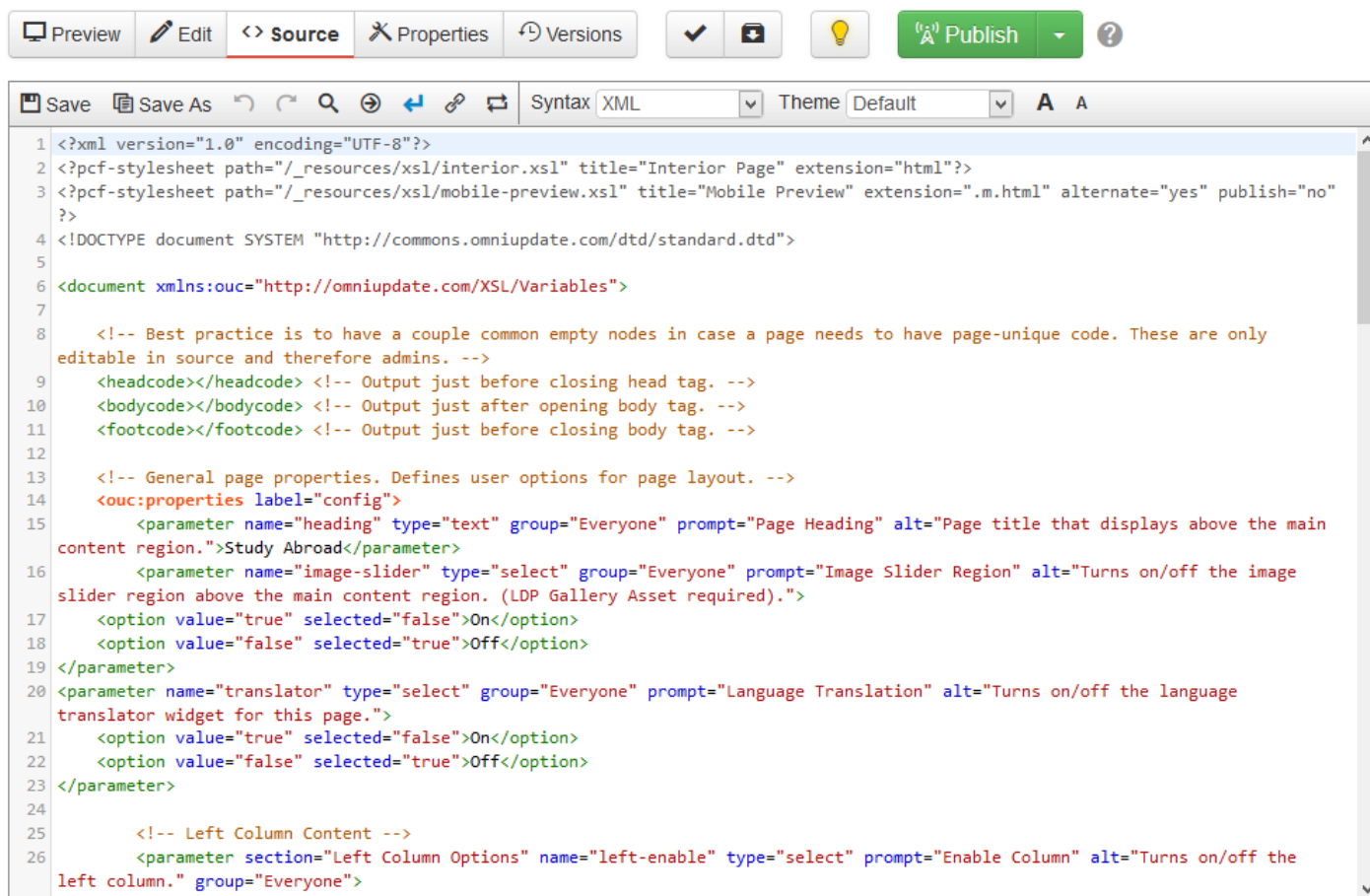
- A level 9 administrator can be restricted access to the Source Editor.
- User levels 1 through level 8 can be allowed access to the Source Editor.

Both of these settings can be configured by a level 10 administrator in the Restrictions panel of User Settings, found in **Setup > Users**. For more information on user permissions, visit the [Users](#) page.

The Source Editor is different than the

HTML Source Editor

available through the WYSIWYG Editor; the HTML Source Editor only allows a user to edit the source code of the region which is currently being edited.



The Source Editor will, by default, highlight the code based upon the syntax for the type of file detected. The **Syntax** drop-down menu in the Source Editor toolbar allows users to change

the highlighting. When HTML or XML is selected in the **Syntax** drop-down menu, the Source Editor automatically completes the closing tag of an HTML or XML element. Selecting **None** from the **Syntax** drop-down disables this feature as well as removes any highlighting.

Users can perform a regex find and replace from the Source Editor to edit source code. The regex behavior within the Source Editor is slightly different than the regex behavior in the Find and Replace tool, in that the Source Editor will perform a line-by-line search while Find and Replace searches the whole page. Find and Replace matches white space throughout the document.

Source Editor will also provide auto-complete suggestions as you are typing, for ease and efficiency. The suggestions provided will vary depending on the option selected in the **Syntax** drop-down menu.

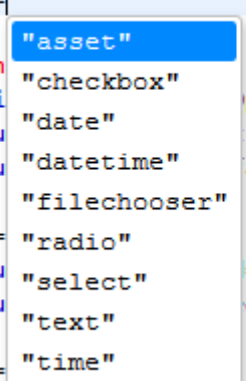
The following syntax options will trigger auto-complete functionality:

- **Javascript files:** JS-related methods/syntax
- **HTML files:** HTML nodes/attributes
- **XML files (PCF, TCF, TMPL and XSL):** Proprietary OU nodes/attributes such as `<ouc:div>` and `<ouc:editor>`, HTML nodes/attributes, and common XSL nodes/attributes.
- **CSS files:** CSS properties/pseudo classes

```

:er>
:parameter type=
!-- Left Column
:parameter secti
<option valu
<option valu
:/parameter>
:parameter name=
<option valu
<option valu
:/parameter>
:parameter name=
<option value="true" selected="false">

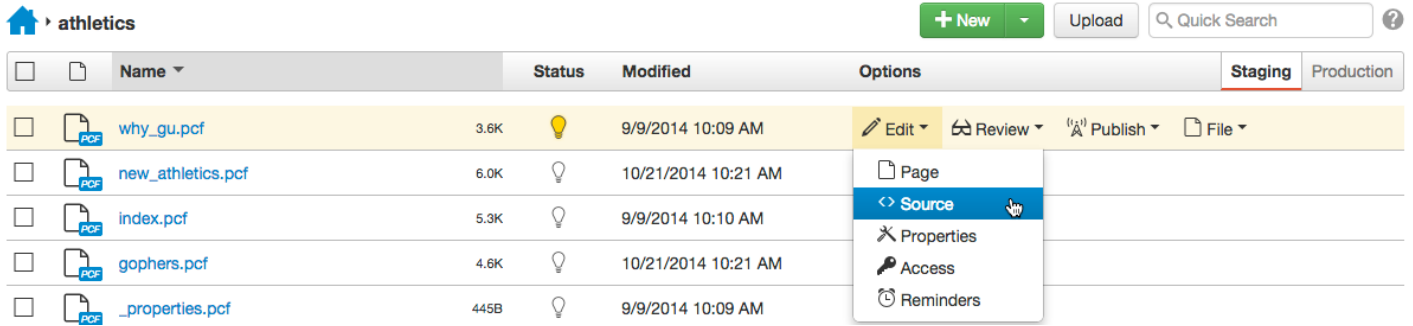
```



Accessing the Source Editor

The Source Editor can be accessed through any of the following ways:

- From the **Content > Pages** view, check out the desired page. Select the **Edit** menu and choose **Source**.



- Open the page from the Pages list view and click the **Source** button in the **Page Actions** toolbar.

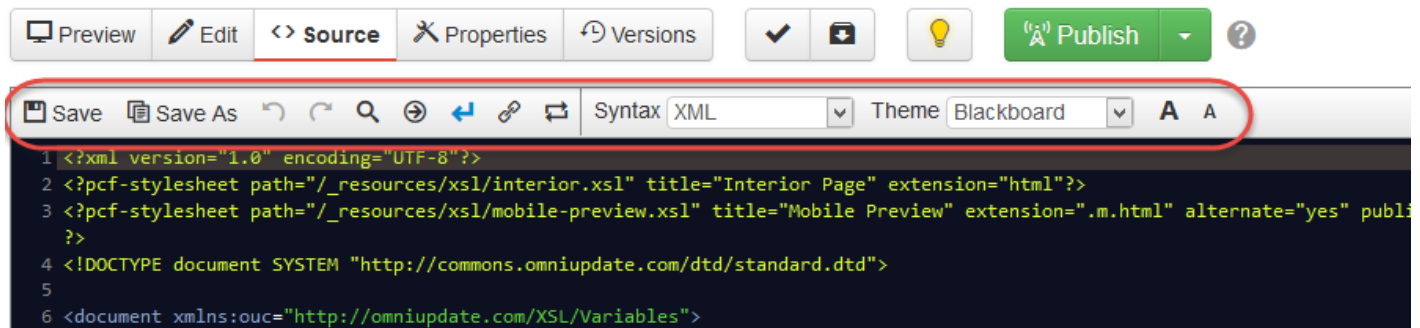


- Check out the page or click the **Edit** button, and then click the blue **Edit Source** button at the bottom of the page.



Using the Source Editor

The Source Editor contains a modified toolbar, which has different functions than those within the WYSIWYG Editor.



The functions for the Source Editor toolbar are as follows:

- **Save:** This saves all changes made to source and exits the user from the Source Editor. The user will be taken back to the Preview view of the page. To save in place without exiting the Source Editor, use keyboard shortcuts (**Ctrl+S** on PC, **Cmd+S** on Mac).
- **Save As:** This allows the user to change the name, path, and extension of the file in order to save a copy. This is an alternative way to create a new file in the OU Campus system.
- **Undo/Redo:** This allows a user to undo a change that has been made in the source code, or to redo a step that has been undone.
- **Find and Replace:** This allows the user to search for (and, if they desire, replace) text or code within the source. Searches can use a regular expression (regex) or simple text string. For more information about regex, see: [Regular Expressions \(RegEx\)](#)
- **Go To Line:** This will open up a box beneath the toolbar where a line number can be inputted. Inputting a line number will automatically direct the Source Editor cursor to the start of that line.
- **Line Wrap:** This button enables or disables wrap-around text. When enabled, text will be bound to the width of the Source Editor window. Lines of code longer than this width will continue on the next line (though a new line number will not be assigned). When disabled, lines of code longer the width of the window will be accessible via a horizontal scroll bar. Line wrap does not alter the code in any way, only how the user views it.
- **Insert File Path:** A user may insert a dependency tag for a file directly into the source code, if Dependency Manager is enabled. This button will open a modal that allows the user to select a file from any configured servers (e.g., staging, production, auxiliary sites, and alternative publish targets). The dependency tag will be inserted at the cursor's location in the Source Editor, so be sure to set the cursor at the desired location. Note that this function only inserts the dependency tag or link, and no other pieces of code.
- **Insert Asset:** A user may also insert an asset directly into the source code. This button will open a modal that allows a user to browse for a specific asset. The dependency tag will be inserted at the cursor's location in the Source Editor, so be sure to set the cursor at the desired location.
- **Syntax:** This drop-down allows the user to select the code syntax based on code type. A few of the available options at HTML, XML (default), CSS, and JavaScript.
- **Theme:** A user may choose between a large variety of themes, which will change the colors of the text and background.
- **Increase/Decrease Font Size:** Clicking each icon increases or decreases the font size of the Source Editor.

Users can also use a variety of keyboard shortcuts in the Source Editor. The available keyboard shortcuts include:

- **Ctrl (Windows)/Cmd (Mac) + S:** Save-in-Place. Saves the content onto the staging server, but allows users to remain in the editor and make further changes.
- **Ctrl/Cmd + A:** Select All. Selects all content in the Source Editor.
- **Ctrl/Cmd + D:** Deletes the entire line of code on which the cursor is currently placed.
- **Ctrl/Cmd + Z:** Undo the last change.
- **Ctrl/Cmd + Y:** Redo the last undone change.
- **Ctrl/Cmd + F:** Brings up the Find and Replace tool for the Source Editor.
- **Ctrl/Cmd + /:** Comments out the currently-selected line(s) of code. This function is syntax-specific; the format of the comment will depend on the language selected in the Syntax drop-down menu.
- **Shift + Tab:** Auto-indent. Takes all selected text and indents it according to the nesting of elements.
- **Ctrl/Cmd + [:** Indents all selected code by one tab less.
- **Ctrl/Cmd +]:** Indents all selected code by one tab more.
- **Ctrl + Q:** Folds the element on the current line, hiding child elements and other content, making the code easier to read.

<pre> 97 <ouc:div label="left-quicklinks" group=" 98 <h4>Quick Links</h4> 99 100 Link 1 101 Link 2 102 Link 3 103 104 </ouc:div> </pre>	<pre> 97 <ouc:div label="left-quicklinks" 98 <h4>Quick Links</h4> 99 104 </ouc:div> </pre>
Before Folding	After Folding

Using Gadgets in the Source Editor

Users are able to drag and drop images and videos from the Images Gadget and YouTube Gadget, respectively, into the Source Editor just as they would in the WYSIWYG Editor.

When an image or video is added to the Source Editor from a gadget, the Insert/Edit Image or Insert/Edit Video modal will not appear; instead, a simple `` or `<iframe>` tag will be added into the source with only the essential information included. A user must add other information (e.g., image description, dimensions, alignment) manually after the image or video has been inserted.

```

<ouc:div label="maincontent" group="Everyone" button-text="Main Content"><ouc:editor
csspath="/_resources/ou/editor/content.css" cssmenu="/_resources/ou/editor/styles.txt" width="1020"
wysiwyg-class="maincontent"/><p>The International Programs Office
offers study abroad programs in over 20 countries throughout the world, in every academic
discipline at GU! Study abroad for a year or a semester while earning credit towards your major,
minor, or general education requirements. Instruction is in a wide range of languages, including
many options in English. Financial aid and scholarships are available. Fill out the following form
to receive more information in your chosen field:</p>

```

Syntax Highlighting

The Source Editor in OU Campus automatically determines the correct type of document based upon the file extension. However, the need may arise to override the automatic behavior and display a document in a different style. When a user selects a different code syntax, the displayed highlighting is specific for that code base.

The list of code syntax types available in the drop-down menu are as follows:

1. **None** (.txt): The first option, when selected, displays the the code without syntax highlight and without code complete.
2. **HTML** (.html, .tmpl, .jsp)
3. **XML** (.xml, .pcf, .tcf, .xsl, .rss, .xslt, .xsd, .dtd)
4. **CSS** (.css)
5. **JavaScript** (.js)
6. **PHP** (.php, .php3)
7. **Python** (.py)
8. **VBScript** (.vbs)
9. **Perl** (.pl, .cgi)
10. **MySQL** (.sql)
11. **C#** (.cs)

Dependency Tags in the Source Editor

Dependency tags are inserted into the Source Editor via the **Insert File Path** tool (for files, folders, or servers) or the **Insert Asset** tool (for assets). Once inserted, the Dependency tag appears as a tag surrounded by two curly braces, such as below:

```
81
82 <ouc:div label="left-quicklinks" group="Everyone" button-text="Quick Links">
83 <ouc:editor csspath="/_resources/ou/editor/content.css" cssmenu="/_resour
84 <h4>Quick Links</h4>
85 <ul>
86 <li><a href="{{f:5932499}}">Link 1</a></li>
87 <li><a href="#">Link 2</a></li>
88 <li><a href="#">Link 3</a></li>
89 </ul>
90 </ouc:div>
91
```

Dependency Manager also includes tracking and reporting broken links within the Source Editor. If a dependency tag is identified as broken, it is highlighted in red in-context so it can be easily identified and fixed.

```
82 <ouc:div label="left-quicklinks" group="Everyone" button-text="
83 <ouc:editor csspath="/_resources/ou/editor/content.css" css
84 <h4>Quick Links</h4>
85 <ul>
86 <li><a href="#">Link 1</a></li>
87 <li><a href="{{f:6040148}}">Link 2</a></li>
88 <li><a href="#">Link 3</a></li>
89 </ul>
90 </ouc:div>
91
```