

[Management Tools](https://cloud.google.com/products/management/) (https://cloud.google.com/products/management/)

[Cloud Shell](https://cloud.google.com/shell/) (https://cloud.google.com/shell/) [Documentation](#)

Walkthrough Markdown Reference

Cloud Shell tutorials, also known as Walkthroughs, help users familiarize themselves with your project quickly and effectively, while being easy to set up and launch. A Cloud Shell Walkthrough is a set of instructions written in [CommonMark](http://commonmark.org/) (http://commonmark.org/) Markdown. It can also include directives which are Walkthrough specific Markdown expressions to add advanced functionality like spotlighting and project selection.

Refer to the [Writing Walkthroughs guide](https://cloud.google.com/shell/docs/walkthroughs) (https://cloud.google.com/shell/docs/walkthroughs) for how to set up, compose, and launch a Walkthrough in Cloud Shell.

Directive format

To add a directive, use the following format:

```
<walkthrough-directive-name param-name="param-value">  
</walkthrough-directive-name>
```



In this format, 'directive-name' and 'param' are placeholders. For example, if you'd like to use the directive, `editor-open-file`, and the parameter, `filePath`, use this format:

```
<walkthrough-editor-open-file filePath="test/hello.md">  
</walkthrough-editor-open-file>
```



Note: Parameters are strings separated by whitespace. Quotes around the parameters are optional. If the parameter does contain whitespace, such as the label text for a spotlight pointer, quotes are required.

Cloud Shell directives

The tables below list the directives, and their respective parameters, available for use when authoring a Cloud Shell Walkthrough.

Walkthrough metadata and layout

Directive: Author

```
<walkthrough-author></walkthrough-author>
```



The author tag defines metadata about the author. All parameters are optional.

Parameters	Description
<code>name</code>	Author's name.
<code>repositoryUrl</code>	Link to the repository which is used for feedback.
<code>tutorialName</code>	Walkthrough id

Directive: Walkthrough Duration

```
<walkthrough-tutorial-duration></walkthrough-tutorial-duration>
```



The Walkthrough duration tag renders a clock icon and message estimating how long the Walkthrough will take.

Parameters	Description
<code>duration</code>	Estimated Walkthrough duration in minutes.

Directive: Suggested Walkthrough card

```
<walkthrough-tutorial-card></walkthrough-tutorial-card>
```



A Walkthrough card tag links to a suggested Walkthrough.

Parameters	Description
<code>url</code>	URL to the Walkthrough.
<code>icon</code>	Icon name.

Parameters	Description
<code>label</code>	Label for analytics for click events.

Directive: Footnote

```
<walkthrough-footnote>Footnote text here</walkthrough-footnote>
```



A footnote directive styles the text as a footnote at the bottom of the step. Footnote text is written as the inner text.

Project selection

Directive: Project setup

```
<walkthrough-project-setup></walkthrough-project-setup>
```



The project setup component helps the user create or select a project to use. This step will block the user from continuing until a valid project has been selected.

The project permissions tag sets required project permissions for the Walkthrough. If the user lacks the permissions on the selected project, the project setup will not consider it a valid project.

See the [documentation](https://cloud.google.com/iam/docs/granting-changing-revoking-access) (https://cloud.google.com/iam/docs/granting-changing-revoking-access) for a description of how project permissions work.

Parameters	Description
<code>permissions</code>	(<i>Optional</i>) Comma separated list of required permissions.

Directive: Project and billing setup

```
<walkthrough-project-billing-setup></walkthrough-project-billing-setup>
```




The project and billing setup component helps the user create or select a project to use. This step will block the user from continuing until a valid project with billing enabled has been selected.

Parameters	Description
<code>permissions</code>	(Optional) Comma separated list of required permissions.

Spotlights and buttons

Directive: Spotlight pointer

```
<walkthrough-spotlight-pointer>Label text here</walkthrough-spotlight-pointer>
```



The spotlight pointer directive creates a clickable label that will spotlight the specified UI element on the screen. **Only one** of the parameters is needed. If both are provided, the `spotlightId` will be used.

The pointer can select an element by instrumentation id, which is a property on the DOM element, or by using a CSS selector. Both types of selectors allow comma separated selectors to match elements. All elements matching the selector will be highlighted.

Label text is written as the inner text.

Parameters	Description
<code>spotlightId</code>	Instrumentation ID, can use commas to separate multiple IDs.
<code>cssSelector</code>	A CSS selector


Spotlight targets/instrumentation IDs	Description
<code>console-nav-menu</code>	Console navigation menu, hamburger in top left.
<code>devshell-activate-button</code>	Button to open Cloud Shell, near top right.
<code>devshell-web-editor-button</code>	Opens Orion editor, on Cloud Shell's top bar.

Spotlight targets/instrumentation IDs	Description
<code>devshell-web-preview-button</code>	Opens a web preview window for Cloud Shell.

For spotlights within the editor frame, see the section of [Editor directives](https://cloud.google.com/shell/docs/walkthrough-markdown-reference#editor_directives) (https://cloud.google.com/shell/docs/walkthrough-markdown-reference#editor_directives).

Directive: Enable APIs

```
<walkthrough-enable-apis></walkthrough-enable-apis>
```




The enable APIs component creates a button that enables APIs when clicked. This will allow your project to access the GCP APIs.

Refer to the [documentation](https://cloud.google.com/apis/) (<https://cloud.google.com/apis/>) for which APIs you can enable.

Parameters	Description
<code>apis</code>	Comma separated list of APIs to enable.

Directive: Open Cloud Shell

```
<walkthrough-open-cloud-shell-button></walkthrough-open-cloud-shell-button>
```




The open Cloud Shell component creates a button that opens Cloud Shell when clicked (with the current project selected).

Parameters	Description
<code>open-cloud-shell</code>	Opens Cloud Shell when the parent step is viewed.

Directive: Open Cloud Shell in the background

```
<walkthrough-devshell-precreate></walkthrough-devshell-precreate>
```



The `devshell-precreate` directive provisions a Cloud Shell instance in the background so the load is faster. Runs when the step is shown.

Dynamic variables

Directive: Data binding

```
<walkthrough-watcher-constant></walkthrough-watcher-constant>
```



Walkthroughs allow data binding for key-value pairs. The values are set by directives or by the Walkthrough framework. They can be defined within the Walkthrough session using the directive `watcher-constant`.

Afterwards, the string value can be referenced with:

```
{{<key>}}
```

Example usage in Markdown:

```
# My Walkthrough
```



```
## Step 1
```

```
<walkthrough-watcher-constant key="my-key" value="Hello Google">
</walkthrough-watcher-constant>
```

```
My message: {{my-key}}
```

There are built-in keys available:

Key	Value description
<code>project-id</code>	The user's current project id.
<code>project-name</code>	The user's project name.

Directive: Watcher constant

```
<walkthrough-watcher-constant></walkthrough-watcher-constant>
```



A Watcher is a dynamic string that can be set for a Walkthrough. A watcher constant defines the value at the beginning of the Walkthrough. See the [data binding \(#data-binding\)](#) section for more information.

Parameters	Description
<code>key</code>	Key to the watcher.
<code>value</code>	Value of the watcher.







Directive: Inline icons

```
<walkthrough-inline-icon-name></walkthrough-inline-icon-name>
```




'inline-icon-name' is a placeholder for the icon you'd like to specify (like `nav-menu-icon` and `conclusion-trophy`).

The following icons are styled as inline icons.

Inline icon name	Icon
<code>cloud-shell-icon</code>	
<code>web-preview-icon</code>	
<code>cloud-shell-editor-icon</code>	
<code>nav-menu-icon</code>	
<code>notification-menu-icon</code>	
<code>pin-section-icon</code>	

The following icons are styled as images.

Inline icon name	Icon
<code>conclusion-trophy</code>	

Cloud Shell Editor directives

Editor directives are supported by the [Cloud Shell Editor](https://console.cloud.google.com/cloudshell/editor) (<https://console.cloud.google.com/cloudshell/editor>) page.

Directive: Open file

```
<walkthrough-editor-open-file></walkthrough-editor-open-file>
```



The open file directive creates a clickable label to open a file from the Cloud Shell disk in the editor.

Label text is written as the inner text.

Parameters	Description
<code>filePath</code>	Relative file path

Directive: Open selected text in editor

```
<walkthrough-editor-select-line></walkthrough-editor-select-line>
```



The `editor-select-line` directive creates a clickable label to open a file in the editor and select a line of text within it.

Parameters	Description
<code>filePath</code>	Relative file path
<code>startLine</code>	Start line
<code>startCharacterOffset</code>	Start index
<code>endLine</code>	End line
<code>endCharacterOffset</code>	End index

Directive: Open regex specified text in editor


```
<walkthrough-editor-select-regex>Label text here</walkthrough-editor-select-regex>
```

The `editor-select-regex` directive creates a clickable label to open a file in the editor and selects text within it, based on the provided regex.

Parameters	Description
<code>filePath</code>	Relative file path
<code>regex</code>	Regex to match

Label text is written as the inner text.

Directive: Spotlight pointer

```
<walkthrough-editor-spotlight>Label text here</walkthrough-editor-spotlight>
```

The spotlight pointer directive creates a clickable label that will spotlight the specified UI element within the editor on the screen.

The pointer can only select predefined targets.

Parameters	Description
<code>spotlightId</code>	Editor ID

Label text is written as the inner text.

Editor spotlight targets	Description
<code>fileMenu</code>	Highlights the file menu button.
<code>editMenu</code>	Highlights the edit menu button.
<code>navigator</code>	Highlight a file specified in the item field.

Except as otherwise noted, the content of this page is licensed under the [Creative Commons Attribution 4.0 License](https://creativecommons.org/licenses/by/4.0/) (<https://creativecommons.org/licenses/by/4.0/>), and code samples are licensed under the [Apache 2.0 License](https://www.apache.org/licenses/LICENSE-2.0).

(<https://www.apache.org/licenses/LICENSE-2.0>). For details, see our [Site Policies](#)
(<https://developers.google.com/terms/site-policies>). Java is a registered trademark of Oracle and/or its affiliates.

Last updated December 3, 2019.