

# Building User-Friendly Guides for Wikimedia Tools

By: . Joris Darlington Quarshie  
. AfiMaame



wikiindaba



wikimediamorocco

# Introduction...



# What is technical documentation

**Technical documentation** is a type of communication that provides information about a technical product or service.

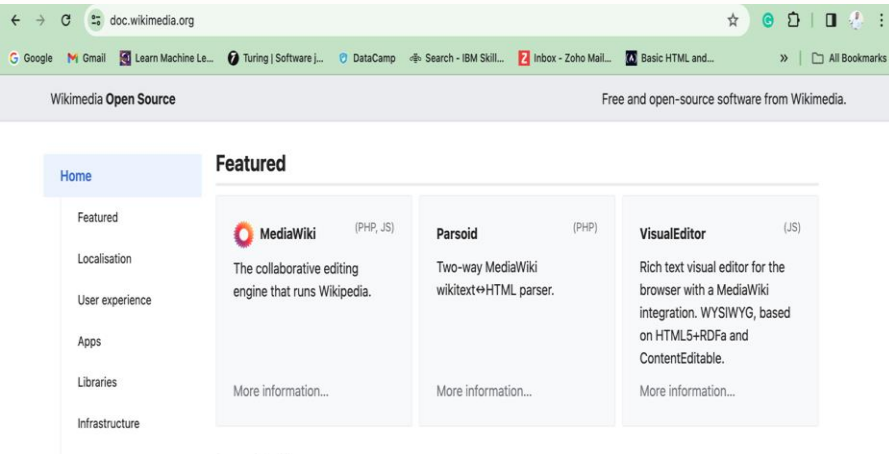
- It can be used to teach users how to use a product.
- Explain how a product works.
- Provide reference information for troubleshooting.

# Getting started with Wikimedia documentation



# Find Documentation...

- Visit the developer portal to find documentation for wikimedia projects.
- Read about documentation locations.
- [doc.wikimedia.org](https://doc.wikimedia.org) is automatically generated from the source code for a subset of wikimedia software (including Mediawiki,libraries,infrastructure and some apps).
- [https://www.mediawiki.org/wiki/Documentation/Find\\_docs](https://www.mediawiki.org/wiki/Documentation/Find_docs) outlines some of the locations here where documentation lives, like on wikis.

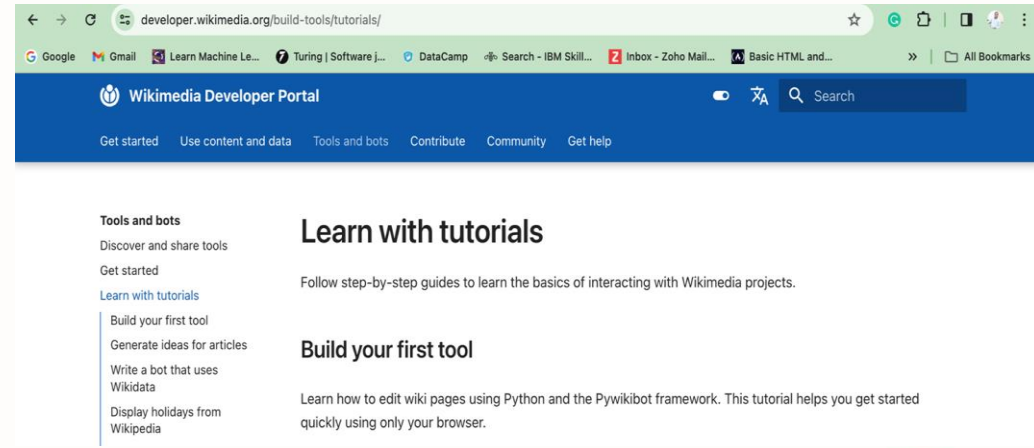


doc.wikimedia.org

Wikimedia Open Source Free and open-source software from Wikimedia.

Home Featured

Featured	MediaWiki (PHP, JS)	Parsoid (PHP)	VisualEditor (JS)
Localisation	The collaborative editing engine that runs Wikipedia.	Two-way MediaWiki wikitext+HTML parser.	Rich text visual editor for the browser with a MediaWiki integration. WYSIWYG, based on HTML5+RDFa and ContentEditable.
User experience	More information...	More information...	More information...
Apps			
Libraries			
Infrastructure			



developer.wikimedia.org/build-tools/tutorials/

Wikimedia Developer Portal

Get started Use content and data Tools and bots Contribute Community Get help

Tools and bots Discover and share tools

Get started

Learn with tutorials

- Build your first tool
- Generate ideas for articles
- Write a bot that uses Wikidata
- Display holidays from Wikipedia

Learn with tutorials Follow step-by-step guides to learn the basics of interacting with Wikimedia projects.

Build your first tool Learn how to edit wiki pages using Python and the Pywikibot framework. This tutorial helps you get started quickly using only your browser.

# Improve your documentation...

- Use templates, best practices, to audit and improve your documentation.
- Visit the documentation toolkit.
- For more complex tools or software that have more (6+) pages of documentation
- [https://www.mediawiki.org/wiki/Documentation/How-to\\_template](https://www.mediawiki.org/wiki/Documentation/How-to_template)
- <https://www.mediawiki.org/wiki/Documentation/Toolkit>
- [https://www.mediawiki.org/wiki/Documentation/Landing\\_pages](https://www.mediawiki.org/wiki/Documentation/Landing_pages)

# Learn about Writing...

- Find style guides and tips to help you write clearly and effectively.
- Browse style guides and writing resources.
- [https://www.mediawiki.org/wiki/Documentation/Technical\\_style\\_guides\\_and\\_templates](https://www.mediawiki.org/wiki/Documentation/Technical_style_guides_and_templates)

## **Creating New content:**

- Pretend you're explaining it to someone who's a member of one of your key audiences.
- Start with prerequisites, end with next steps.
- Create an outline and then fill it in.
- Write, then edit.
- Include real examples, like working API calls with explanations of the request and sample responses.

Style exercise...

# Style exercise!

Rewrite this sentence to use active voice:

"This code can be shared with other users."



# Style exercise...

## Possible solutions:

"You can share this code with other users"

OR

"Other users can find and use this code".

Style exercise...

# Style exercise!

Rewrite this sentence to use plain language:

“There is no hard and fast limit on API requests, but be considerate and try not to take a site down. Most system administrators reserve the right to unceremoniously block you if you do endanger the stability of their site.”

# Style exercise...

## Possible solutions:

"There is no official limit on API requests, but administrators may block your client if it impacts the stability of a site."

# Get involved in documentation work

- Help improve wikimedia technical documentation
- Connect with the community and learn about outreach programs.
- Get involved...
- <https://www.mediawiki.org/wiki/Documentation/Contribute>
- <https://toolhub.wikimedia.org/>
- [https://wikitech.wikimedia.org/wiki/Portal:Toolforge/Tool\\_sweep](https://wikitech.wikimedia.org/wiki/Portal:Toolforge/Tool_sweep)

# Building user-friendly guides for Wikimedia tools

When building user-friendly guides for Wikimedia tools, there are a few key things to keep in mind:

- **What is software X?** What does the software/ tool do? Why would the readers want to use it? What does the rest of the doc cover? What would the reader be able to do after reading this documentation?

eg. This guide covers how to install and run X....

By the end of this guide you will be able to do X in Y ways using Z

- **Installing, Setting up and Configuring X...**

Prerequisite: Knowledge requirement, Software requirement, Hardware requirements and Access requirement (Accounts or permission needed to proceed).

# Building user-friendly guides for Wikimedia tools

When building user-friendly guides for Wikimedia tools, there are a few key things to keep in mind:

- **Do a thing...** this is where you write your instructions for how to complete key tasks for how to complete “the thing”.

## Step 1

## Step 2

When your reader completes all the tasks they should have successfully done “the thing”

# Building user-friendly guides for Wikimedia tools

When building user-friendly guides for Wikimedia tools, there are a few key things to keep in mind:

- **Adding code samples:** Provide code samples in each step on how to do it. If possible, provide code snippets in multiple languages.

```
JavaScript_Example.js
/*
  hello.js

  MediaWiki API Demos
  Demo of `...` module

  MIT License
*/

var url = "https://en.wikipedia.org/w/api.php";

var params = {
  action: "query",
  format: "json"
};

url = url + "?origin=";
Object.keys(params).forEach(function(key){url += "&" + key + "=" + params[key]});

fetch(url)
  .then(function(response){return response.json();})
  .then(function(response) {
    console.log(response);
  })
  .catch(function(error){console.log(error)});
```

```
PHP_Example.php
<?php
/*
  hello.php

  MediaWiki API Demos
  Demo of `...` module

  MIT License
*/

$endPoint = "https://en.wikipedia.org/w/api.php";
$params = [
  "action" => "query",
  "format" => "json"
];

$url = $endPoint . "?" . http_build_query( $params );

$ch = curl_init( $url );
curl_setopt( $ch, CURLOPT_RETURNTRANSFER, true );
$output = curl_exec( $ch );
curl_close( $ch );

$result = json_decode( $output, true );

echo( $result );
```

# Building user-friendly guides for Wikimedia tools

When building user-friendly guides for Wikimedia tools, there are a few key things to keep in mind:

- **Next Steps:** If there are common steps that are likely to come next like troubleshooting, link to those resources.
- **Get help:** How can a reader get support when they get stuck during the process.

Who maintains the page.

Who to contact for help, how and where to do that.



# Writing documentation for a new tool

- **Identify your audience.** Who are you writing documentation for? What are their needs and knowledge levels?
- **Identify the features of the tool.** What are the most important features of the tool? What features do you need to document?
- **Gather information about the tool.** How does the tool work? What are the different ways to use the tool?
- **Organize the information.** Once you have gathered all of the information, think about what your users need to do, what the subtasks are for them to accomplish that goal. structure your documentation site to align with the steps in the user's journey and the order in which they will need to do tasks.
- **Write the documentation.** Use clear and concise language. Avoid using jargon and technical terms that your audience may not understand.
- **Test the documentation with users.** Get feedback from users and make sure the documentation is easy to follow and understand.

# Writing documentation for a new tool

- **Template:Tool** on wikitech. Please make sure to have a toolinfo.json record to include your tool's information in toolhub

<https://wikitech.wikimedia.org/wiki/Template:Tool>

- Adding it manually through the toolhub web interface

[https://meta.wikimedia.org/wiki/Toolhub#Adding\\_tools\\_to\\_the\\_catalog](https://meta.wikimedia.org/wiki/Toolhub#Adding_tools_to_the_catalog)

- Publishing a README.md file. Ideally tools should have a readme.md and contributing.md files in their directory and linked to a documentation page where more in-depth documentation is published.

# Summary...

- Identify your audience
- Choose the right tools
- Write clear and concise language
- Test your documentation with users.

# Resources

- <https://www.mediawiki.org/wiki/Documentation>
- [https://www.mediawiki.org/wiki/Documentation/Find\\_docs](https://www.mediawiki.org/wiki/Documentation/Find_docs)
- <https://www.mediawiki.org/wiki/Documentation/Toolkit>
- [https://www.mediawiki.org/wiki/Documentation/Technical\\_style\\_guides\\_and\\_templates](https://www.mediawiki.org/wiki/Documentation/Technical_style_guides_and_templates)
- [https://www.mediawiki.org/wiki/Documentation/Technical\\_documentation\\_templates\\_and\\_suggestions](https://www.mediawiki.org/wiki/Documentation/Technical_documentation_templates_and_suggestions)
- [https://meta.wikimedia.org/wiki/Toolhub#Adding\\_tools\\_to\\_the\\_catalog](https://meta.wikimedia.org/wiki/Toolhub#Adding_tools_to_the_catalog)
- <https://wikitech.wikimedia.org/wiki/Template:Tool>

# Credits

- [Wikimedia Technical Documentation Team](#)
- [Mediawiki Documentation Page](#)
- [Mediawiki.org](#)
- [Wikimedia Developer Portal](#)
- [Wikimedia Documentation Project](#)

# Contact

Name: Joris Darlington Quarshie

Email: [joris.quarshie@gmail.com](mailto:joris.quarshie@gmail.com)

Community Lead, African Wikimedia Technical Community



# Thank you

