

Your Document Title

Document Subtitle

Author Name

None

Table of contents

1. Welcome to MkDocs Tutorial	4
1.1 Features Demonstrated	4
1.2 Quick Links	4
1.3 Getting Help	4
1.4 Commands	4
1.5 Project layout	4
2. User Guide	5
2.1 Getting Started with MkDocs	5
2.1.1 Installation	5
2.1.2 Creating a New Project	5
2.1.3 Project Layout	5
2.1.4 Basic Commands	5
2.2 Writing Content in MkDocs	6
2.2.1 Basic Markdown	6
2.2.2 Adding Images	6
2.2.3 Tables	6
2.2.4 Code Blocks	6
2.2.5 Best Practices	6
2.3 Customizing Your MkDocs Site	7
2.3.1 Theme Customization	7
2.3.2 Adding Extensions	7
2.3.3 Custom CSS	7
3. Features	8
3.1 Markdown Examples	8
3.1.1 Basic Formatting	8
3.1.2 Lists	8
3.1.3 Tables	8
3.1.4 Blockquotes	8
3.1.5 Code	9
3.1.6 Links and Images	9
3.1.7 Footnotes	9
3.1.8 Definition Lists	9
3.2 Code Blocks Examples	10
3.2.1 Basic Code Block	10
3.2.2 Code with Line Numbers	10

3.2.3	Code with Highlighting	10
3.2.4	Different Languages	10
3.2.5	Inline Code	11
3.2.6	Code with Title	11
3.3	Admonitions	12
3.3.1	Basic Admonitions	12
3.3.2	Collapsible Admonitions	12
3.3.3	Custom Titles	12
3.3.4	Nested Admonitions	13
3.3.5	Supported Types	13
4.	About This Documentation	14
4.1	Purpose	14
4.2	Features	14
4.3	Contact	14
4.4	Version Information	14
4.5	License	14
4.6	Acknowledgments	14

1. Welcome to MkDocs Tutorial

This is a comprehensive tutorial and example site for MkDocs with the Material theme. Here you'll find examples of various features and capabilities that MkDocs offers.

1.1 Features Demonstrated

- **Complete Documentation Structure** - See how to organize your documentation effectively
- **Material Theme Customization** - Examples of theme customization and features
- **Markdown Extensions** - Various markdown extensions and their usage
- **Search Functionality** - Built-in search feature
- **Responsive Design** - Works great on all devices

1.2 Quick Links

- [Getting Started](#) - Learn how to set up MkDocs
- [Markdown Examples](#) - See markdown formatting examples
- [Admonitions](#) - Learn about admonitions and callouts

1.3 Getting Help

If you need help with MkDocs, you can:

- Read the [official MkDocs documentation](#)
- Visit the [Material for MkDocs](#) website
- Check the [PyMdown Extensions](#) documentation

1.4 Commands

- `mkdocs new [dir-name]` - Create a new project.
- `mkdocs serve` - Start the live-reloading docs server.
- `mkdocs build` - Build the documentation site.
- `mkdocs -h` - Print help message and exit.

1.5 Project layout

```
mkdocs.yml    # The configuration file.
docs/
  index.md    # The documentation homepage.
  ...        # Other markdown pages, images and other files.
```

2. User Guide

2.1 Getting Started with MkDocs

2.1.1 Installation

To get started with MkDocs, you'll need Python installed on your system. Then you can install MkDocs using pip:

```
pip install mkdocs
pip install mkdocs-material
```

2.1.2 Creating a New Project

1. Create a new project:

```
mkdocs new my-project
cd my-project
```

2. Start the development server:

```
mkdocs serve
```

3. Open your browser and visit `http://127.0.0.1:8000`

2.1.3 Project Layout

```
mkdocs.yml    # The configuration file
docs/
  index.md    # The documentation homepage
  ...         # Other markdown pages, images and other files
```

2.1.4 Basic Commands

- `mkdocs serve` - Start the live-reloading docs server
- `mkdocs build` - Build the documentation site
- `mkdocs -h` - Print help message and exit



Pro Tip

Use `mkdocs serve` during development to preview your changes in real-time!

2.2 Writing Content in MkDocs

2.2.1 Basic Markdown

MkDocs uses Markdown for content creation. Here are some basic examples:

Headers

```
# H1 Header
## H2 Header
### H3 Header
```

Text Formatting

```
**Bold text**
*Italic text*
`Code text`
```

2.2.2 Adding Images

When adding images, use the following format to ensure proper PDF rendering:

```
![Alt text](image.jpg){ width="300" }
```

2.2.3 Tables

Create tables using the following syntax:

```
| Header 1 | Header 2 |
|-----|-----|
| Cell 1   | Cell 2   |
```

2.2.4 Code Blocks

Use triple backticks with language specification:

```
def hello_world():
    print("Hello, World!")
```

2.2.5 Best Practices

1. Use relative links for internal navigation
2. Keep image sizes consistent
3. Use proper heading hierarchy
4. Include alt text for images
5. Maintain consistent formatting

2.3 Customizing Your MkDocs Site

2.3.1 Theme Customization

Colors and Fonts

You can customize the Material theme by modifying the `mkdocs.yml` file:

```
theme:
  name: material
  palette:
    primary: indigo
    accent: indigo
  font:
    text: Roboto
    code: Roboto Mono
```

Navigation

Configure the navigation structure:

```
nav:
  - Home: index.md
  - User Guide:
    - Getting Started: guide/getting-started.md
    - Customization: guide/customization.md
```

2.3.2 Adding Extensions

Enable Markdown extensions in `mkdocs.yml`:

```
markdown_extensions:
  - admonition
  - codehilite
  - toc:
    permalink: true
```

2.3.3 Custom CSS

Create a `docs/stylesheets/extra.css` file:

```
:root {
  --md-primary-fg-color: #007bff;
}
```

Then add it to `mkdocs.yml`:

```
extra_css:
  - stylesheets/extra.css
```

3. Features

3.1 Markdown Examples

This page demonstrates various Markdown features supported by MkDocs.

3.1.1 Basic Formatting

Bold text and *italic text*

Bold and italic text

~~Strikethrough text~~

3.1.2 Lists

Unordered list:

- Item 1
- Item 2
- Subitem 2.1
- Subitem 2.2
- Item 3

Ordered list:

1. First item
2. Second item
3. Third item

Task list:

- [x] Completed task
- [] Incomplete task
- [] Another task

3.1.3 Tables

Header 1	Header 2	Header 3
Cell 1	Cell 2	Cell 3
Cell 4	Cell 5	Cell 6

3.1.4 Blockquotes

This is a blockquote

It can span multiple lines

And can be nested

3.1.5 Code

Inline code: `print("Hello, World!")`

Code block with syntax highlighting:

```
def hello_world():
    print("Hello, World!")
    return True

# Function call
hello_world()
```

3.1.6 Links and Images

[Link to Google](#)

Image Alt Text

3.1.7 Footnotes

Here's a sentence with a footnote¹.

3.1.8 Definition Lists

Term 1

Definition 1

Term 2

Definition 2

Definition 2.1

1. This is the footnote content. ↩

3.2 Code Blocks Examples

3.2.1 Basic Code Block

```
def greet(name):  
    return f"Hello, {name}!"  
  
print(greet("World"))
```

3.2.2 Code with Line Numbers

```
1  class Calculator:  
2      def add(self, a, b):  
3          return a + b  
4  
5      def subtract(self, a, b):  
6          return a - b  
7  
8      def multiply(self, a, b):  
9          return a * b  
10  
11     def divide(self, a, b):  
12         if b == 0:  
13             raise ValueError("Cannot divide by zero")  
14         return a / b
```

3.2.3 Code with Highlighting

```
def process_data(data):  
    # This line is highlighted  
    result = data.process()  
    return result
```

3.2.4 Different Languages

JavaScript

```
function fetchData() {  
    return fetch('https://api.example.com/data')  
        .then(response => response.json())  
        .then(data => console.log(data));  
}
```

HTML

```
<div class="container">  
  <h1>Hello World</h1>
```

```
<p>This is a paragraph.</p>  
</div>
```

CSS

```
.container {  
  max-width: 1200px;  
  margin: 0 auto;  
  padding: 20px;  
}
```

3.2.5 Inline Code

You can use `inline code` within paragraphs.

3.2.6 Code with Title

example.py

```
def main():  
    print("This is a code block with a title")  
  
if __name__ == "__main__":  
    main()
```

3.3 Admonitions

Admonitions, also known as call-outs, are an excellent way to highlight important information.

3.3.1 Basic Admonitions

Note

This is a note admonition. Use it for general information.

Tip

This is a tip admonition. Use it for helpful tips and tricks.

Warning

This is a warning admonition. Use it to warn users about potential issues.

Danger

This is a danger admonition. Use it for critical information that needs attention.

Success

This is a success admonition. Use it to indicate successful operations.

3.3.2 Collapsible Admonitions

Click me to expand

This is a collapsible admonition. It starts collapsed.

Expanded by default


This is a collapsible admonition that starts expanded.

3.3.3 Custom Titles


Custom Title

You can customize the title of any admonition.

3.3.4 Nested Admonitions

 **Outer note**

This is an outer note.

 **Inner warning**

This is a nested warning inside a note.

3.3.5 Supported Types

- `note`
- `abstract`
- `info`
- `tip`
- `success`
- `question`
- `warning`
- `failure`
- `danger`
- `bug`
- `example`
- `quote`

4. About This Documentation

4.1 Purpose

This documentation serves as a comprehensive guide and example for MkDocs with the Material theme. It demonstrates various features and capabilities that you can use in your own documentation.

4.2 Features

- Material Design theme
- Responsive layout
- Dark mode support
- Search functionality
- PDF export capability
- Various Markdown extensions

4.3 Contact

If you have any questions or suggestions about this documentation, please feel free to reach out.

4.4 Version Information

- MkDocs version: 1.5+
- Material theme version: 9.0+
- Python version: 3.8+

4.5 License

This documentation is licensed under the MIT License.

4.6 Acknowledgments

Special thanks to: - The MkDocs team - Material for MkDocs creators - All contributors