Xicu Marí

How to create beautiful presentations with Marp

Tutorial by xicu.info

Alicia Marí / **Marp Tutorial /** Table of contents

1. First of all... what is Marp?
2. How to install and use Marp
3. Create a Marp presentation
    - i. Create the Markdown file
    - ii. Enable Marp and start writing
    - iii. Preview the content
    - iv. Export the presentation
4. Working with images
    - i. Adding an image
    - ii. Customizing the image
        - a. Change the size of the image
        - b. Set image as background
        - c. Apply a filter
5. Customize with a CSS theme
    - i. Use a built-in theme
    - ii. Use an external theme
    - iii. Use my theme
        - a. Download and activate the theme
        - b. Add a logo
        - c. Custom classes
6. Conclusion

/ **Marp Tutorial /** First of all... what is Marp?

Marp is a tool for creating visually appealing presentations. It is developed by Yuki Hattori. As a differential trait, Marp uses Markdown—a lightweight, plain text markup language known for being easy to read and write.

Markdown's straightforward syntax has made it popular among writers and developers for producing content that is both structured and readable. Marp builds on this simplicity, transforming Markdown into professional-looking slides with minimal effort.

With Marp, the focus moves towards the content, writing it in plain text with basic Markdown commands, without the distraction of complex design tools.

/ **Marp Tutorial /** How to install and use Marp

There are several ways to install and use Marp. The one I use is the Marp extension for Visual Studio Code. This extension allows you to create, preview and export Marp presentations directly without leaving the VS Code editor.

To install the Marp extension, follow these steps:

1. Open Visual Studio Code (if you don't have it installed, you can download it from here).
2. Go to the Extensions view and earch for "Marp" in the Extensions view search box.
3. Click on the "Install" button to install the extension.

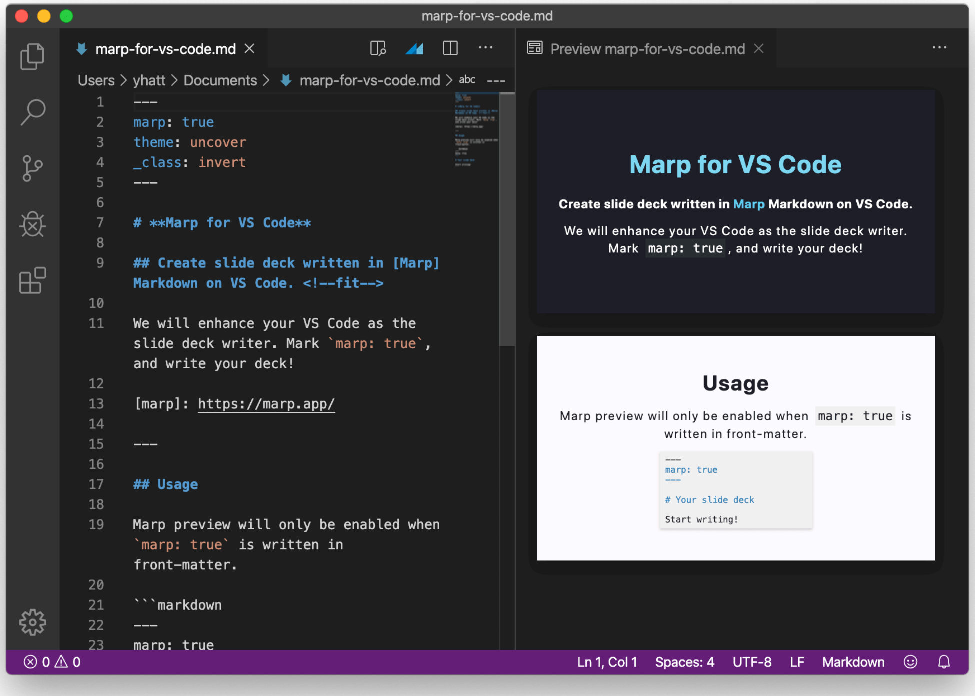
/ Marp Tutorial / Create a Marp presentation

Let's start with a simple example to show you how easy it is to create a Marp presentation, focusing for the moment only on the content.

1. Create the Markdown file

Once the extension is installed, you can jump up into creating Marp presentations by initializing a new Markdown file. You can do this by creating a new file with the `.md` extension.

If it is the first time you are using Visual Studio Code, it will ask you to install the recommended extensions for Markdown highlighting and previewing, so you can easy identify titles, links and other features of the Markdown syntax.


marp-for-vs-code.md
marp-for-vs-code.md ×
Users > yhatt > Documents > marp-for-vs-code.md > abc ---
Preview marp-for-vs-code.md ×
⊗ 0 ⚠ 0
Ln 1, Col 1
Spaces: 4
UTF-8
LF
Markdown

~~~
---
marp: true
theme: uncover
_class: invert
---

# **Marp for VS Code**

## Create slide deck written in [Marp]
Markdown on VS Code. <!--fit-->

We will enhance your VS Code as the
slide deck writer. Mark `marp: true`,
and write your deck!

[marp]: https://marp.app/

---

## Usage

Marp preview will only be enabled when
`marp: true` is written in
front-matter.

```markdown
---
marp: true
~~~

Marp for VS Code

Create slide deck written in Marp Markdown on VS Code.

We will enhance your VS Code as the slide deck writer. Mark `marp: true`, and write your deck!

Usage

Marp preview will only be enabled when `marp: true` is written in front-matter.

```
---
marp: true
---

# Your slide deck

Start writing!
```

2. Enable Marp and start writing

To enable Marp in the Markdown file, you need to add `marp: true` at the beginning of the file, wrapped in three dashes.

Then, you can start writing your presentation using Markdown. This guide provides a detailed explanation of the basic syntax.

You could start, for example, with the following code:

```
---
marp: true
headingDivider: 1
---

# Slide 1
Now you can start writing your presentation using Markdown...

# Slide 2
...and create as many slides as you need.
```

- `headingDivider` **directive**: allows you to separate the slides by the number of `#` characters in the headings.
- Otherwise you could use the `---` separator to separate the slides.

3. Preview the content

You can preview what you are writing at every moment. To do so, just click on the `Open Preview to the Side` button in the top right corner of the editor.

4. Export the presentation

Finally, to export the presentation, click on the `Show Quick Pick of Marp Commands` button in the top right corner of the editor and select "Export slide deck". You can choose between several formats, such as PDF, PNG, PowerPoint or HTML.

/ **Marp Tutorial /** Working with images

So far, we have only used text to create the slides. However, we all know that a picture is worth a thousand words. Therefore, we will learn now how to add images to our presentation, as well as how to customize them to our needs.

/ **Marp Tutorial** / Working with images

Adding an image

To add an image to a slide, you can use the following syntax:

```
![Alt text](/path/to/image.jpg)
```

The path to the image can be either an absolute path or a relative path to the Markdown file. On the other hand, the `Alt text` label is the text that will be displayed if the image cannot be loaded.

If you don't want to write paths, you are lucky because Visual Studio Code allows you to drag and drop the image into the editor, which will automatically copy the image in your working directory.

Finally, you could also use a URL to an image hosted on the internet, like that:

```
![Alt text](https://example.com/image.jpg)
```

In Marp, you can include keywords in the `Alt text` label to customize the image. The complete list of directives can be found here. Here are the most common ones:

Change the size of the image

With the following keywords, you can change the size of the image, specifying the width and/or height in pixels. Here are some examples:

- Width: `![w:250](/path/to/image.jpg)`
- Height: `![h:120](/path/to/image.jpg)`
- Width and height: `![w:32 h:32](/path/to/image.jpg)`

The aspect ratio is maintained if only one of the dimensions is specified.

/ Marp Tutorial / Working with images

Set image as background

The following structure is used:

```
![bg](/path/to/image.jpg)
```

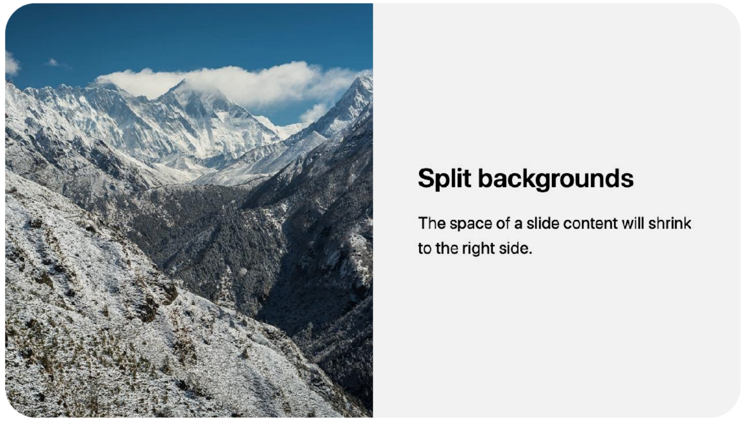
Additionally, you can resize the image with the following keywords:

- `bg fit`: Scale to fit the slide.
- `bg auto`: Use the original size.
- `bg 150%`: Scale by the percentage.

On the other hand, you can create a split background with the following syntax:

```
![bg left](/path/to/image.jpg)
```

You can also use `bg right` to set the image as the background of the right half of the slide. To specify the size, add the percentage after the keywords, for instance, `bg rifht:33%`.



Split backgrounds

The space of a slide content will shrink to the right side.

Apply a filter

You can apply CSS filters to modify the appearance of the image. The following filters are available: `blur`, `brightness`, `contrast`, `drop-shadow`. `grayscale`, `hue-rotate`, `invert`, `opacity`, `saturate` and `sepia`. Multiple filters can be applyed as well:

```
![brightness:.8 sepia:50%](https://example.com/image.jpg)
```

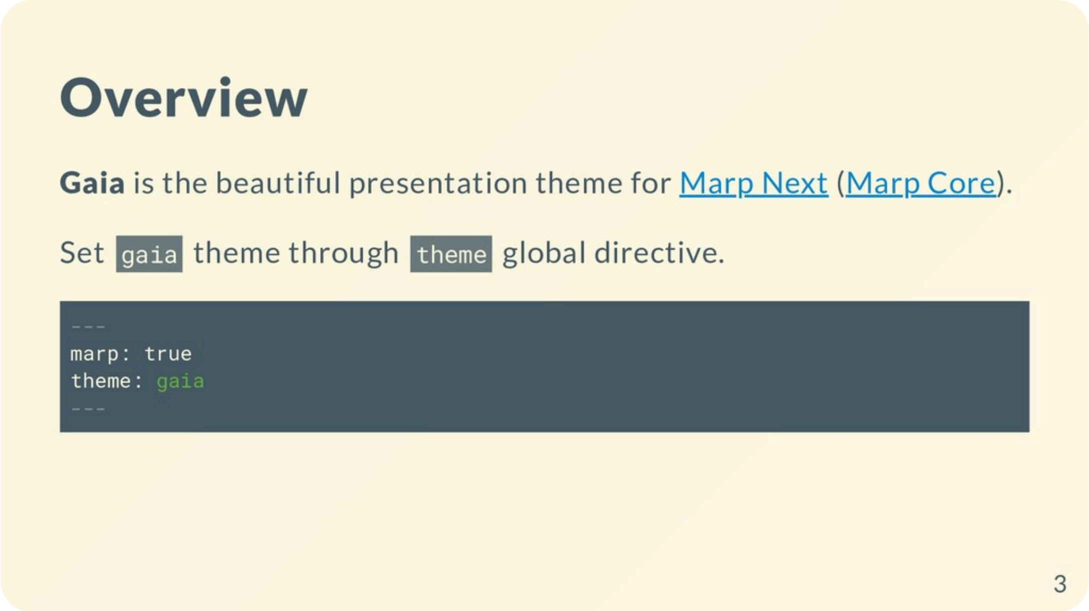
/ Marp Tutorial / Customize with a CSS theme

On the previous sections, the examples were based on the default Marp theme. However, Marp allows you to customize the appearance of your presentation by using CSS themes, similarly to how you would do it with a website.

Use a built-in theme

Marp comes with a set of built-in themes. To use one of these, you need to add the `theme` directive, specifying the name of the theme. For instance:

```
marp: true
theme: gaia
```



Overview

Gaia is the beautiful presentation theme for Marp Next (Marp Core).

Set `gaia` theme through `theme` global directive.

```
---
marp: true
theme: gaia
---
```

3

Use an external theme

To use external themes, you must configure Visual Studio Code to recognize the CSS files:

1. Go to the settings by clicking on the gear icon in the lower left corner of the window.
2. Search for `marp themes` in the search box.
3. In the `Marp: Themes` section, add the path to the CSS file you want to use. For example, if you have a CSS file named `style.css` in the same directory as the Markdown file, you should add `./style.css` for Visual Studio Code to recognize it.

Use an external theme

Once you have configured Visual Studio Code to recognize the CSS file, you can download the CSS file from the internet. Here you have a collection of themes created by the Marp community.

Then, you should add the `theme` directive in the preamble, followed by the name of the theme, to activate it the theme. You can find the theme name at the start of the CSS file, indicated by the `@theme theme-name` label.

Download and activate the theme

I have created a minimal theme that you can use in your presentations. You can download the CSS file from here and add the following preamble to your Markdown file:

```
marp: true
theme: minimal
paginate: true
headingDivider: 2
style: |
   h2:before {
      /* Add the title of the presentation here */
      content: '/ Title / ';
   }
```

This preamble indicates that the pages have page number indicators and a slide is created each time a H2 heading (`##`) is declared. Additionally, the title of the presentation is shown in each slide.

Alicia Marí / **Marp Tutorial /** Use my custom theme

The explanation of how to create a custom theme is out of the scope of this post, but you can find more information about it here. If you are really interested in creating your own theme, please let me know in the comments and I will write a post about it.

Add a logo

The theme also admits the use of a logo. It is added to the presentation by pasting a PNG named `logo.png` in the same directory as the Markdown file. The default logo can be downloaded from here.

The logo will be displayed in the top left corner of the slides. You may need to adjust the position of the header if the logo aspect ratio is different from the default one.

Custom classes

In addition to the built-in classes, you can create custom classes to apply to the slides. If you want to apply a custom class globally, you can add the `class` directive in the preamble, followed by the name of the class. Otherwise, you can add the `_class` directive in the slide you want to apply the class to.

Alicia Marí / **Marp Tutorial /** Customize with a CSS theme

In my theme's CSS file, I have defined some custom classes to invert the color scheme and to create columns. Here you can find two examples:

- To **invert** the color scheme, use the `_class: invert` directive.

```
## Slide 1
<!-- _class: invert -->
This slide's background is black
and the text is white.
```

- To create **two columns**, use the `_class: col2` directive. You can create up to 4 columns by using `col3` and `col4`. Each column is separated by a H3 heading (`###`).

```
## Slide 2
<!-- _class: col2 -->
### Column 1
This slide...
### Column 2
...is divided into two columns.
```

/ **Marp Tutorial /** Conclusion

If you have reached this point, you are ready to start creating beautiful presentations with Marp. Enjoy! 🚀