

SLIDEV Docs - Ελληνικά



Sliddev

Table of contents

• EI - README	4
• EI - TRANSLATIONS	5
• Addons - Use	8
• Addons - Write an /addon	9
• Builtin - Components	11
• Builtin - Layouts	14
• Custom - Config katex	18
• Custom - Config mermaid	19
• Custom - Config monaco	20
• Custom - Config parser	22
• Custom - Config shortcuts	26
• Custom - Config vite	28
• Custom - Config vue	30
• Custom - Config windicss	31
• Custom - Directory structure	32
• Custom - Fonts	36
• Custom - Global layers	39
• Custom - Highlighters	41
• Custom - Index	43
• Custom - Vue context	45
• Guide - Animations	47
• Guide - Drawing	51
• Guide - Editors	53
• Guide - Exporting	56
• Guide - Faq	58
• Guide - Hosting	61
• Guide - Index	64
• Guide - Install	67
• Guide - Navigation	72
• Guide - Presenter mode	74
• Guide - Recording	75
• Guide - Syntax	76
• Guide - Why	85
• Resources - Covers	88

• Resources - Learning	89
• EI - Showcases	90
• Themes - Gallery	91
• Themes - Use	92
• Themes - Write a /theme	94

sli.dev

Οδηγίες χρήσης για το [Slidev](#)

Μεταφράσεις

	Repo	Site	Maintainers
English	docs	sli.dev	@antfu
简体中文	docs-cn	cn.sli.dev	@QC-L @Ivocin
Français	docs-fr	fr.sli.dev	@ArthurDanjou
Español	docs-es	es.sli.dev	@owlnai
Русский	docs-ru	ru.sli.dev	@xesjkeee
Việt Nam	docs-vn	vn.sli.dev	@bongudth
Deutsch	docs-de	de.sli.dev	@fabiankachlock
Português (BR)	docs-br	br.sli.dev	@luisfelipesdn12
Ελληνικά	docs-el	el.sli.dev	@GeopJr
日本語	docs-ja	ja.sli.dev	@IkumaTadokoro

Ξεκινήστε τον διακομιστή τοπικά

```
npm i -g npnpm
```

```
npnpm i
npnpm run dev
```

Και στη συνέχεια επισκεφθείτε <http://localhost:3000>

Ή εγκαταστήστε την [επέκταση Vite για το VS Code](#) για να επεξεργαστείτε δίπλα-δίπλα.

Βοηθήστε στη Μετάφραση

Δείτε [TRANSLATIONS.md](#)

[Go to TOC](#)

Help on Translating

First of all, thank you for being interested in contributing to translations!

You can find the repositories for each existing translation in [README.md](#). To help improve them, simply sending a Pull Request to their repo.

If the language you want to contribute isn't on the list, join [our Discord server](#), and find the `#translations` channel to see if someone is already working on the language you want, consider joining them and translate together. If not, you can start a new translation project with the following steps.

In case it's already been translated but you're wondering how to maintain it, skip to the end. ## Some tips before you get started

- It is recommended that you use your IDE of choice (e.g VSCode) paired with a development server running, so you can see your translation changes in real-time.
- You can mark these checkmarks as the translation progresses or use your own workflow. The translations don't need to be made in any particular order.
- Translations don't need to be literal, but they should convey the same message. In case you're not sure how to translate something, you can either leave it as it is or use online tools like WordReference or Linguee to aid you.
- Most translations will simply consist in editing Markdown files. Certain areas are buried under Vue components, which will be listed below. You can also use your IDE to find the string to translate.

Getting started

- Fork the main docs repo: [slidevjs/docs](#)
- Translate README.md, you can take one of the already translated repositories as an example.
- Share your repo's link to the `#translations` channel telling people you are working on it and find collaborators.

Translating Markdown files

- `showcases.md` - A gallery showcase of Slidev presentations.
- `index.md` - Mainpage content, note that some of it is buried under Vue components listed further below.

.vitepress/

- `config.js` - Sitemap
- `/theme/components/WorkingInProgress.vue` - WIP notice shown in mainpage
- `/theme/components/demo/Demo.vue` - Animated demo shown in mainpage
- `/theme/components/Environment.vue` - Describes the environment of a setting.

builtin/

- `components.md` - Use [Vue components](#) inside Slidev
- `layouts.md` - Use Vue layouts inside Slidev

custom/

- `config-katex.md` - Configuring Katex
- `config-mermaid.md` - Configuring Mermaid
- `config-monaco.md` - Configuring Monaco
- `config-shortcuts.md` - Configuring Shortcuts
- `config-vite.md` - Configuring Vite
- `config-vue.md` - Configuring Vue
- `config-windicss.md` - Configuring Windicss
- `directory-structure.md` - Configuring the directory structure
- `fonts.md` - Configuring fonts
- `global-layers.md` - Configuring the global layers
- `highlighters.md` - Configuring code highlighters
- `index.md` - Customizations index page
- `vue-context.md` - The Vue global context

guide/

- `animations.md` - Animations and transitions
- `editors.md` - Editor integrations
- `exporting.md` - Exporting your slides
- `faq.md` - Frequent Answered Questions
- `index.md` - Getting started with Slidev
- `navigation.md` - Navigation across slides
- `presenter-mode.md` - Toggling presenter mode
- `recording.md` - Recording your presentation
- `syntax.md` - Markdown syntax
- `why.md` - *Why Slidev?*

resources/

- `covers.md` - Curated covers for Slidev

themes/

- `gallery.md` - Theme gallery
- `use.md` - How to use Slidev themes
- `write-a-theme.md` - Write your own theme

addons/

- [use.md](#) - How to use Slidev addons
- [write-an-addon.md](#) - Write your own addon

Publishing your translations

- When you finish the translation (at least 90%), [@antfu](#) in the Discord and we will invite you to the org and make the translation official.
- Once the transferring is done, we will set up the subdomain, auto-deployment, and a daily sync-up bot to keep the translation up-to-date with the latest English docs.
- The site is live, and we will send a shout-out tweet on [our Twitter account](#).

Maintaining the translations up-to-date

- [docschina-bot](#) will periodically submit merge requests from the [slidev/docs](#) repository. Switch to the branch created in the pull request, make any changes necessary and merge it. [example](#).
- Sometimes it will occur that a merge request is made and you haven't merged the previous one. The latest PR always checks your main branch against the English one; so you can just close the previous PR(s), move your work to the latest one and merge it.

[Working-in-progress translation list](#)

Thanks again!

[Go to TOC](#)

Χρήση Πρόσθετου

Τα πρόσθετα είναι σύνολα επιπρόσθετων στοιχείων, διατάξεων, στυλ, ρυθμίσεων... κ.λπ. που μπορείτε να χρησιμοποιήσετε στην παρουσίασή σας.

Μοιάζουν αρκετά με [τα θέματα](#), αλλά γενικά:

- δεν επηρεάζουν το συνολικό στυλ των διαφανειών σας
- μπορείτε να χρησιμοποιήσετε πολλαπλά πρόσθετα σε μία παρουσίαση

Για να χρησιμοποιήσετε πρόσθετα, πρέπει να τα εγκαταστήσετε χειροκίνητα μέσω:

```
$ npm install [slidev-addon-package1] [slidev-addon-package2]
```

Και στη συνέχεια δηλώστε τα πρόσθετα είτε στο frontmatter σας:

```
---  
addons:  
  - slidev-addon-package1  
  - slidev-addon-package2  
---
```

Είτε στο αρχείο `package.json` σας:

```
// package.json  
{  
  "slidev": {  
    "addons": [  
      "slidev-addon-package1",  
      "slidev-addon-package2",  
    ]  
  }  
}
```

Γράψτε ένα Πρόσθετο

Διαθέσιμο από v0.32.1

Δυνατότητα

Ένα πρόσθετο μπορεί να συμβάλει στα ακόλουθα σημεία:

- Global στυλ (χρησιμοποιήστε τα με προσοχή, είναι περισσότερο ο ρόλος [των θεμάτων](#))
- Παροχή προσαρμοσμένων διατάξεων ή υπέρβαση της υπάρχουσας
- Παροχή προσαρμοσμένων στοιχείων ή παράκαμψη του υπάρχοντος
- Επέκταση των ρυθμίσεων του Windi CSS
- Ρύθμιση εργαλείων όπως το Monaco και το Prism

Συμβάσεις

Τα πρόσθετα δημοσιεύονται στο μητρώο npm και πρέπει να ακολουθούν τις παρακάτω συμβάσεις:

- Το όνομα του πακέτου πρέπει να αρχίζει με `slidev-addon-`, για παράδειγμα: `slidev-addon-awesome`
- Προσθέστε `slidev-addon` και `slidev` στο πεδίο `keywords` του `package.json` σας

Setup

Αρχικοποίηση

Για να δημιουργήσετε το πρόσθετό σας, ξεκινήστε δημιουργώντας έναν φάκελο με ένα αρχείο `package.json` (μπορείτε να χρησιμοποιήσετε `npm init`).

Στη συνέχεια, εγκαταστήστε τις τα dependencies του slidev:

```
$ npm install -D @slidev/cli
```

Έλεγχος

Για να δημιουργήσετε το πεδίο δοκιμών για το πρόσθετό σας, μπορείτε να δημιουργήσετε ένα αρχείο `example.md` με κάποιο περιεχόμενο.

Και προαιρετικά, μπορείτε επίσης να προσθέσετε κάποια scripts στο `package.json` σας

```
// package.json
{
  "scripts": {
    "dev": "slidev example.md",
    "build": "slidev build example.md",
    "export": "slidev export example.md",
```

```

    "screenshot": "slidev export example.md --format png"
}
}
```

Για να δημοσιεύσετε το πρόσθετό σας, απλά εκτελέστε `npm publish` και είστε έτοιμοι. Δεν απαιτείται διαδικασία κατασκευής (που σημαίνει ότι μπορείτε να δημοσιεύσετε απευθείας αρχεία `.vue` και `.ts`, το Slidev είναι αρκετά έξυπνο για να τα καταλάβει).

Τα σημεία συνεισφοράς του πρόσθετου ακολουθούν τις ίδιες συμβάσεις με την τοπική ρυθμίση, παρακαλούμε ανατρέξτε [στις οδηγίες για τις συμβάσεις ονομασίας](#).

Μεταδεδομένα πρόσθετου

Έκδοση Slidev

Εάν το πρόσθετο βασίζεται σε ένα συγκεκριμένο χαρακτηριστικό του Slidev που εισήχθη πρόσφατα, μπορείτε να ορίσετε την ελάχιστη έκδοση Slidev που απαιτείται για να λειτουργήσει σωστά το πρόσθετό σας:

```

// package.json
{
  "engines": {
    "slidev": ">=0.32.1"
  }
}
```

Εάν οι χρήστες χρησιμοποιούν παλαιότερες εκδόσεις του Slidev, θα εμφανιστεί σφάλμα.

Components

Ενσωματωμένα Components

Οι οδηγίες χρήσης αυτής της ενότητας βρίσκονται ακόμη σε εξέλιξη. Πριν από αυτό, μπορείτε να ρίξετε απευθείας μία ματιά στον [πηγαίο κώδικα](#).

Toc

Εισαγωγή πίνακα περιεχομένων (TOC).

Αν θέλετε μια διαφάνεια να μην εμφανίζεται στο στοιχείο `<Toc>`, μπορείτε να χρησιμοποιήσετε στο μπλοκ front matter της διαφάνειας:

```
---  
hideInToc: true  
---
```

Οι τίτλοι εμφανίζονται χρησιμοποιώντας το στοιχείο `<Titles>`

Χρήση

```
<Toc />
```

Παράμετροι:

- `columns` (`string | number`, προεπιλογή: `1`): Ο αριθμός των στηλών της απεικόνισης
- `listClass` (`string | string[]`, προεπιλογή: `''`): Κλάσεις που εφαρμόζονται στον πίνακα περιεχομένων
- `maxDepth` (`string | number`, προεπιλογή: `Infinity`): Το μέγιστο επίπεδο βάθους του τίτλου που θα εμφανιστεί
- `minDepth` (`string | number`, προεπιλογή: `1`): Το ελάχιστο επίπεδο βάθους του τίτλου που θα εμφανιστεί
- `mode` (`'all' | 'onlyCurrentTree' | 'onlySiblings'`, προεπιλογή: `'all'`):
 - `'all'`: Εμφάνιση όλων των στοιχείων
 - `'onlyCurrentTree'`: Εμφάνιση μόνο των στοιχείων που βρίσκονται στο τρέχον δέντρο (ενεργό στοιχείο, γονείς και παιδιά του ενεργού στοιχείου)
 - `'onlySiblings'`: Εμφάνιση μόνο των στοιχείων που βρίσκονται στο τρέχον δέντρο και των άμεσων αδελφών τους

Link

Εισάγετε έναν σύνδεσμο που μπορείτε να χρησιμοποιήσετε για να πλοηγηθείτε σε μια συγκεκριμένη διαφάνεια.

Χρήση

```
<Link to="42">Πηγαίνετε στη διαφάνεια 42</Link>
<Link to="42" title="Πηγαίνετε στη διαφάνεια 42"/>
```

Παράμετροι:

- `to (string | number)`: Το path της διαφάνειας στην οποία θέλετε να πλοηγηθείτε (οι διαφάνειες ξεκινούν από το 1)
- `title (string)`: Ο τίτλος που θα εμφανιστεί

Titles

Εισάγετε τον κύριο τίτλο από μια διαφάνεια που επεξεργάζεται ως HTML.

Οι τίτλοι και τα επίπεδα τίτλων ανακτώνται αυτόματα από το πρώτο στοιχείο τίτλου κάθε διαφάνειας.

Μπορείτε να παρακάμψετε αυτή την αυτόματη συμπεριφορά για μια διαφάνεια χρησιμοποιώντας τη σύνταξη front matter:

```
---  
title: Καταπληκτικός τίτλος διαφάνειας  
level: 2  
---
```

Χρήση

Το στοιχείο `<Titles>` είναι ένα εικονικό στοιχείο που μπορείτε να εισαγάγετε με:

```
import Titles from '@/slidev/titles.md'
```

Στη συνέχεια, μπορείτε να το χρησιμοποιήσετε με:

```
<Titles no="42" />
```

Παράμετροι:

- `no (string | number)`: Ο αριθμός της διαφάνειας από την οποία θα εμφανιστεί ο τίτλος (οι διαφάνειες ξεκινούν από το 1)

Προσαρμοσμένα Components

Δημιουργήστε ένα φάκελο `components/` στη ρίζα του έργου σας, και απλά βάλτε τα προσαρμοσμένα Vue components σας σε αυτόν, στη συνέχεια μπορείτε να το χρησιμοποιήσετε με το ίδιο όνομα στο αρχείο markdown σας!

Διαβάστε περισσότερα στην ενότητα [Προσαρμογές](#)

Components που παρέχονται από το θέμα

Τα θέματα επίσης μπορούν να παρέχουν components. Παρακαλούμε διαβάστε τις οδηγίες χρήσης τους για ό,τι προσφέρουν.

Δείτε περισσότερα στην ενότητα [δομή φακέλου](#).

Layouts

Ενσωματωμένα Layouts

Καθώς τα θέματα μπορούν να παρακάμψουν τη συμπεριφορά των layouts, ο καλύτερος τρόπος για να μάθετε ακριβώς τη χρήση, τις παραμέτρους και τα παραδείγματα είναι να ανατρέξετε στις οδηγίες χρήσης τους.

center

Εμφανίζει το περιεχόμενο στο κέντρο της οθόνης.

cover

Χρησιμοποιείται για την εμφάνιση του εξώφυλλου της παρουσίασης, το οποίο μπορεί να περιέχει τον τίτλο της παρουσίασης, τα συμφραζόμενα κλπ.

default

Το πιο βασικό layout, για την εμφάνιση οποιουδήποτε είδους περιεχομένου.

end

Η τελική σελίδα της παρουσίασης.

fact

Για να παρουσιάσετε κάποιο γεγονός ή δεδομένο με πολύ έμφαση στην οθόνη.

full

Χρησιμοποιήστε όλο το χώρο της οθόνης για την εμφάνιση του περιεχομένου.

image-left

Εμφανίζει μια εικόνα στην αριστερή πλευρά της οθόνης, το περιεχόμενο θα τοποθετηθεί στη δεξιά πλευρά.

Χρήση

```
---  
layout: image-left  
# η πηγή της εικόνας  
image: ./path/για/την/εικόνα
```

```
---  
# ένα προσαρμοσμένο όνομα class για το περιεχόμενο  
class: το-cool-περιεχόμενό-μου-στα-δεξιά
```

image-right

Εμφανίζει μια εικόνα στη δεξιά πλευρά της οθόνης, το περιεχόμενο θα τοποθετηθεί στην αριστερή πλευρά.

Χρήση

```
---  
layout: image-right  
  
# η πηγή της εικόνας  
image: ./path/για/την/εικόνα  
  
# ένα προσαρμοσμένο όνομα class για το περιεχόμενο  
class: το-cool-περιεχόμενό-μου-στα-αριστερά
```

image

Εμφανίζει μια εικόνα ως το κύριο περιεχόμενο της σελίδας.

Χρήση

```
---  
layout: image  
  
# η πηγή της εικόνας  
image: ./path/για/την/εικόνα
```

iframe-left

Shows a web page on the left side of the screen, the content will be placed on the right side.

Usage

```
---  
layout: iframe-left  
  
# the web page source  
url: https://github.com/slidesjs/slides  
  
# a custom class name to the content  
class: my-cool-content-on-the-right
```

iframe-right

Shows a web page on the right side of the screen, the content will be placed on the left side.

Usage

```
---  
layout: iframe-right  
  
# the web page source  
url: https://github.com/slidesjs/slides  
  
# a custom class name to the content  
class: my-cool-content-on-the-left  
---
```

iframe

Shows a web page as the main content of the page.

Usage

```
---  
layout: iframe  
  
# the web page source  
url: https://github.com/slidesjs/slides  
---
```

intro

Για την εισαγωγή της παρουσίασης, συνήθως με τον τίτλο της παρουσίασης, μια σύντομη περιγραφή, τον συγγραφέα κλπ.

none

'Ενα layout χωρίς κανένα υπάρχον styling.

quote

Για να εμφανίσετε ένα απόσπασμα με έμφαση.

section

Χρησιμοποιείται για να σηματοδοτήσει την έναρξη μιας νέας ενότητας παρουσίασης.

statement

Κάντε μια διαβεβαίωση/δήλωση ως το κύριο περιεχόμενο της σελίδας.

two-cols

Διαχωρίζει το περιεχόμενο της σελίδας σε δύο στήλες.

Χρήση

```
---  
layout: two-cols  
---  
  
# Αριστερά1  
Αυτό εμφανίζεται στα αριστερά2  
  
::right::  
  
# Δεξιά3  
Αυτό εμφανίζεται στα δεξιά4
```

Προσαρμοσμένα Layouts

Δημιουργήστε ένα φάκελο `layouts/` στη ρίζα του έργου σας, και απλά βάλτε τα προσαρμοσμένα Vue layout components σας σε αυτόν.

Διαβάστε περισσότερα στην ενότητα [Προσαρμογές](#)

Layouts που παρέχονται από το θέμα

Τα θέματα μπορούν να παρέχουν layouts ή να παρακάμψουν τα υπάρχοντα. Παρακαλούμε διαβάστε τις οδηγίες χρήσης τους για ό,τι προσφέρουν.

Ρυθμίστε KaTeX

Δημιουργήστε `./setup/katex.ts` με το εξής περιεχόμενο:

```
import { defineKatexSetup } from '@slidev/types'

export default defineKatexSetup(() => {
  return {
    /* ... */
  }
})
```

Με το `setup`, μπορείτε να δώσετε την προσαρμοσμένη ρύθμιση για τις [Επιλογές KaTeX](#). Ανατρέξτε στους ορισμούς των τύπων και στις οδηγίες χρήσης τους για περισσότερες λεπτομέρειες.

[Go to TOC](#)

Ρυθμίστε Mermaid

Δημιουργήστε `./setup/mermaid.ts` με το εξής περιεχόμενο:

```
import { defineMermaidSetup } from '@slidev/types'

export default defineMermaidSetup(() => {
  return {
    theme: 'forest',
  }
})
```

Με το `setup`, μπορείτε να δώσετε την προσαρμοσμένη ρύθμιση για το [Mermaid](#). Ανατρέξτε στους ορισμούς των τύπων και στις οδηγίες χρήσης τους για περισσότερες λεπτομέρειες.

Ρυθμίστε Monaco

Δημιουργήστε `./setup/monaco.ts` με το εξής περιεχόμενο:

```
import { defineMonacoSetup } from '@slidev/types'

export default defineMonacoSetup(async (monaco) => {
  // χρησιμοποιήστε `monaco` για να το ρυθμίσετε
})
```

Μάθετε περισσότερα για την [ρύθμιση του Monaco](#).

Χρήση

Για να χρησιμοποιήσετε το Monaco στις διαφάνειές σας, απλά προσθέστε `{monaco}` στα αποσπάσματα του κώδικα σας:

```
//``js
const count = ref(1)
const plusOne = computed(() => count.value + 1)

console.log(plusOne.value) // 2
plusOne.value++ // error
//``
```

Σε

```
//``js {monaco}
const count = ref(1)
const plusOne = computed(() => count.value + 1)

console.log(plusOne.value) // 2
plusOne.value++ // error
//``
```

Εξαγωγή

Από προεπιλογή, το Monaco θα λειτουργήσει ΜΟΝΟ σε λειτουργία `dev`. Αν θέλετε να το έχετε διαθέσιμο στο εξαγόμενο SPA, ρυθμίστε το στο frontmatter σας:

```
---
monaco: true # προεπιλεγμένο "dev"
---
```

Αυτόματη Εγκατάσταση Τύπων

Όταν χρησιμοποιείτε TypeScript με το Monaco, οι τύποι εξαρτήσεων θα εγκατασταθούν αυτόματα στο client-side.

```
//``ts {monaco}
import { ref } from 'vue'
import { useMouse } from '@vueuse/core'

const counter = ref(0)
//``
```

Στο παραπάνω παράδειγμα, βεβαιωθείτε ότι τα `vue` και `@vueuse/core` είναι εγκατεστημένα τοπικά ως `dependencies` / `devDependencies`, το Slides θα αναλάβει τα υπόλοιπα, ώστε οι τύποι να λειτουργούν αυτόματα για τον συντάκτη!

Ρύθμιση Θεμάτων

Το θέμα ελέγχεται από το Slides με βάση το φωτεινό/σκοτεινό θέμα. Αν θέλετε να το προσαρμόσετε, μπορείτε να περάσετε το id του θέματος στη συνάρτηση `setup`:

```
// ./setup/monaco.ts
import { defineMonacoSetup } from '@slidev/types'

export default defineMonacoSetup(() => {
  return {
    theme: {
      dark: 'vs-dark',
      light: 'vs',
    },
  }
})
```

Αν θέλετε να φορτώσετε προσαρμοσμένα θέματα:

```
import { defineMonacoSetup } from '@slidev/types'

// αλλάξετε στα θέματά σας
import dark from 'theme-vitesse/themes/vitesse-dark.json'
import light from 'theme-vitesse/themes/vitesse-light.json'

export default defineMonacoSetup((monaco) => {
  monaco.editor.defineTheme('vitesse-light', light as any)
  monaco.editor.defineTheme('vitesse-dark', dark as any)

  return {
    theme: {
      light: 'vitesse-light',
      dark: 'vitesse-dark',
    },
  }
})
```

Εάν δημιουργείτε ένα θέμα για το Slides, χρησιμοποιήστε δυναμικά `import()` μέσα στη συνάρτηση `setup` για να έχετε καλύτερα αποτελέσματα σε tree-shaking και code-splitting.

Ρυθμίστε και επεκτείνετε Parser

To Slidev κάνει parse το αρχείο παρουσίασης (π.χ. `slides.md`) σε τρία βήματα:

1. Πραγματοποιείται ένα βήμα "προετοιμασίας": το αρχείο χωρίζεται σε διαφάνειες χρησιμοποιώντας το διαχωριστικό `---` και λαμβάνοντας υπόψη τα πιθανά μπλοκ frontmatter.
2. Κάθε διαφάνεια γίνεται parsed με μια εξωτερική βιβλιοθήκη.
3. To Slidev επιλύει την ειδική ιδιότητα frontmatter `src:`, η οποία επιτρέπει τη συμπερίληψη άλλων αρχείων md.

Markdown Parser

Η ρύθμιση του markdown parser που χρησιμοποιείται στο βήμα 2 μπορεί να γίνει [ρυθμίζοντας τα εσωτερικά πρόσθετα του Vite](#)

Επεκτάσεις Preparser

Διαθέσιμο από v0.37.0

`:::warning` Σημαντικό: όταν τροποποιείτε τη ρύθμιση παραμέτρων του preparser, πρέπει να σταματήσετε και να εκκινήσετε ξανά το slidev (η επανεκκίνηση μπορεί να μην είναι αρκετή). `:::`

Ο preparser (βήμα 1 παραπάνω) είναι ιδιαίτερα επεκτάσιμος και επιτρέπει την εφαρμογή προσαρμοσμένων συντάξεων για τα αρχεία md σας. Η επέκταση του preparser θεωρείται **μια προηγμένη λειτουργία** και είναι πιθανό να χαλάσει τις [ενσωματώσεις επεξεργαστών](#) λόγω των έμμεσων αλλαγών στη σύνταξη.

Για να το προσαρμόσετε, δημιουργήστε ένα αρχείο `./setup/preparser.ts` με το ακόλουθο περιεχόμενο:

```
import { definePreparserSetup } from '@slidev/types'

export default definePreparserSetup((filepath) => {
  return [
    {
      transformRawLines(lines) {
        for (const i in lines) {
          if (lines[i] === '@@@')
            lines[i] = 'ΓΕΙΑ'
        }
      },
    },
  ]
})
```

Αυτό το παράδειγμα αντικαθιστά συστηματικά κάθε σειρά με την ένδειξη `@@@` με μια σειρά με την ένδειξη `γεια`. Παρουσιάζει τη δομή ενός αρχείου ρυθμίσεων του preparser και μερικές από τις κύριες έννοιες που περιλαμβάνει ο preparser:

- Το `definePreparserSetup` πρέπει να κληθεί με μια συνάρτηση ως παράμετρο.
- Η συνάρτηση λαμβάνει το `path` του αρχείου (του κεντρικού αρχείου παρουσίασης) και μπορεί να χρησιμοποιήσει αυτές τις πληροφορίες (π.χ. για να ενεργοποιήσει επεκτάσεις με βάση το αρχείο παρουσίασης).
- Η συνάρτηση πρέπει να επιστρέψει μια λίστα επεκτάσεων του `preparser`.
- Μια επέκταση μπορεί να περιέχει:
 - μια συνάρτηση `transformRawLines(lines)` που εκτελείται αμέσως μετά το parsing του frontmatter του αρχείου `md` και λαμβάνει μια λίστα όλων των σειρών (από το αρχείο `md`). Η συνάρτηση μπορεί να μεταλλάσσει τη λίστα αυθαίρετα.
 - μια συνάρτηση `transformSlide(content, frontmatter)` που καλείται για κάθε διαφάνεια, αμέσως μετά τη διάσπαση του αρχείου, και λαμβάνει το περιεχόμενο της διαφάνειας ως συμβολοσειρά και το frontmatter της διαφάνειας ως αντικείμενο. Η συνάρτηση μπορεί να μεταβάλλει το frontmatter και πρέπει να επιστρέψει τη συμβολοσειρά του περιεχομένου (ενδεχομένως τροποποιημένη, ενδεχομένως `undefined` αν δεν έχουν γίνει τροποποιήσεις).
 - ένα `name`

Παράδειγμα επεκτάσεων Preparser

Περίπτωση χρήσης 1: συμπαγής σύνταξη παρουσίασης σε ανώτατο επίπεδο

Φανταστείτε μια κατάσταση όπου (μέρος) της παρουσίασής σας παρουσιάζει κυρίως εικόνες εξωφύλλου και περιλαμβάνει άλλα αρχεία `md`. Μπορεί να θέλετε μια συμπαγή σημειογραφία όπου για παράδειγμα (μέρος του) `slides.md` έχει ως εξής:

```
@cover: /nice.jpg
# Καλώς ήρθατε
@src: page1.md
@src: page2.md
@cover: /break.jpg
@src: pages3-4.md
@cover: https://source.unsplash.com/collection/94734566/1920x1080
# Ερωτήσεις;
τα λέμε την επόμενη φορά
```

Για να επιτρέψετε αυτά τα συντακτικά `@src:` και `@cover:`, δημιουργήστε ένα αρχείο `./setup/preparser.ts` με το ακόλουθο περιεχόμενο:

```
import { definePreparserSetup } from '@slidev/types'

export default definePreparserSetup((filepath) => {
  return [
    {
      transformRawLines(lines) {
        let i = 0
        while (i < lines.length) {
          const l = lines[i]
          if (l.match(/^@cover:/i)) {
            lines.splice(i, 1,
              '----',
              'layout: cover',
            )
          }
        }
      }
    }
  ]
})
```

```

        `background: ${l.replace(/@cover: */i, '')}`,
        '---',
        '')
    continue
}
if (l.match(/^@src:/i)) {
    lines.splice(i, 1,
        '---',
        `src: ${l.replace(/^@src: */i, '')}`,
        '---',
        '')
    continue
}
i++
}
},
],
})
)

```

Και αυτό ήταν όλο.

Περίπτωση χρήσης 2: χρήση προσαρμοσμένου frontmatter για το περιτύλιγμα των διαφανειών

Φανταστείτε μια περίπτωση όπου συχνά θέλετε να μεγεθύνετε ορισμένες διαφάνειες σας, αλλά εξακολουθείτε να θέλετε να χρησιμοποιήσετε μια ποικιλία από υπάρχουσες διατάξεις, οπότε η δημιουργία μιας νέας διάταξης δεν θα ήταν κατάλληλη. Για παράδειγμα, μπορεί να θέλετε να γράψετε το `slides.md` σας ως εξής:

```

---
layout: quote
_scale: 0.75
---

# Καλώς ήρθατε

> υπέροχα!

---
_scale: 4
---
# Διάλειμμα

---
# Ok

---
layout: center
_scale: 2.5
---
# Ερωτήσεις;
τα λέμε την επόμενη φορά

```

Εδώ χρησιμοποιήσαμε μια κάτω παύλα στο `_scale` για να αποφύγουμε πιθανές συγκρούσεις με τις υπάρχουσες ιδιότητες του frontmatter (πράγματι, η περίπτωση του `scale`, χωρίς κάτω παύλα θα προκαλούσε πιθανά προβλήματα).

Για να χειριστείτε αυτή τη σύνταξη `_scale: ...` στο frontmatter, δημιουργήστε ένα αρχείο `./setup/preparser.ts` με το ακόλουθο περιεχόμενο:

```
import { definePreparserSetup } from '@slidev/types'

export default definePreparserSetup((filepath) => {
  return [
    {
      transformSlide(content, frontmatter) {
        if ('_scale' in frontmatter) {
          return [
            `<Transform :scale=${frontmatter['_scale']}>`,
            '',
            content,
            '',
            '</Transform>',
          ].join('\n')
        }
      },
    },
  ],
})
```

Και αυτό ήταν όλο.

Ρυθμίστε Συντομεύσεις

Διαθέσιμο από v0.20

Since v0.35.6 (excluded), you decide which base shortcuts to keep (see `...base`, below).

Ξεκινώντας

Δημιουργήστε `./setup/shortcuts.ts` με το εξής περιεχόμενο:

```
import type { NavOperations, ShortcutOptions } from '@slidev/types'
import { defineShortcutsSetup } from '@slidev/types'

export default defineShortcutsSetup((nav: NavOperations, base: ShortcutOptions[]) => {
  return [
    ...base, // keep the existing shortcuts
    {
      key: 'enter',
      fn: () => nav.next(),
      autoRepeat: true,
    },
    {
      key: 'backspace',
      fn: () => nav.prev(),
      autoRepeat: true,
    },
  ],
})
```

Με το setup, μπορείτε να παρέχετε την προσαρμοσμένη ρύθμιση για τις συντομεύσεις που αναφέρεται στην [Πλοήγηση](#). Η παραπάνω ρύθμιση δεσμεύει την επόμενη κίνηση ή διαφάνεια στο enter και την προηγούμενη κίνηση ή διαφάνεια στο backspace.

Η συνάρτηση ρύθμισης λαμβάνει ένα object με κάποιες μεθόδους πλοήγησης, και επιστρέφει ένα array που περιέχει κάποιες ρυθμίσεις συντόμευσης. Ανατρέξτε στους ορισμούς των τύπων για περισσότερες λεπτομέρειες.

Σύνθετη δέσμευση πλήκτρων

Ο τύπος `key` επιτρέπει μόνο συμβολοσειρές, αλλά μπορείτε να δεσμεύσετε πολλαπλά πλήκτρα χρησιμοποιώντας την ακόλουθη σύμβαση:

```
import type { NavOperations, ShortcutOptions } from '@slidev/types'
import { defineShortcutsSetup } from '@slidev/types'
```

```
export default defineShortcutsSetup((nav: NavOperations, base: ShortcutOptions[]) => {
  return [
    ...base,
    {
      key: 'ShiftLeft+ArrowRight',
      fn: () => nav.next(),
      autoRepeat: true,
    }
  ]
})
```

Σύνθετες λειτουργίες πλοήγησης

Οι λειτουργίες πλοήγησης `nav` σας επιτρέπουν να έχετε πρόσβαση σε ορισμένες λειτουργίες πέρα από τις βασικές επόμενη διαφάνεια ή προηγούμενη διαφάνεια. Δείτε τις ακόλουθες περιπτώσεις χρήσης:

```
import { defineShortcutsSetup, NavOperations } from '@slidev/types'

export default defineShortcutsSetup((nav: NavOperations) => {
  return [
    {
      key: 'e',
      // Ορίστε τη συντόμευση πληκτρολογίου `e` που θα χρησιμοποιείται ως σελιδοδείκτης
      // ή ένα είδος γρήγορης πρόσβασης, για να πλοηγηθείτε συγκεκριμένα στο
      // διαφάνεια με αριθμό 42
      fn: () => nav.go(42),
      autoRepeat: true,
    }
  ]
})
```

Ανατρέξτε στο [useMagicKeys | VueUse](#) για περισσότερες λεπτομέρειες σχετικά με το key pressed event.

Ρυθμίστε Vite

To Slidev λειτουργεί με τη βοήθεια του [Vite](#) εσωτερικά. Αυτό σημαίνει ότι μπορείτε να αξιοποιήσετε το εξαιρετικό σύστημα πρόσθετων του Vite για να προσαρμόσετε ακόμα περισσότερο τις διαφάνειές σας.

To `vite.config.ts` θα γίνει σεβαστό, αν έχετε ένα.

To Slidev έχει προρυθμισμένα τα ακόλουθα πρόσθετα:

- `@vitejs/plugin-vue`
- `unplugin-vue-components`
- `unplugin-icons`
- `vite-plugin-vue-markdown`
- `vite-plugin-remote-assets`
- `vite-plugin-windicss`
- `unocss/vite`

Μάθετε περισσότερα για τις [προρυθμίσεις](#) εδώ.

Ρύθμιση Εσωτερικών Πρόσθετων

Διαθέσιμο από v0.21

Για να ρυθμίσετε τα ενσωματωμένα πρόσθετα της παραπάνω λίστας, δημιουργήστε `vite.config.ts` με το εξής περιεχόμενο. Παρακαλώ σημειώστε ότι το Slidev έχει κάποιες προκαθορισμένες επιλογές για αυτά τα πρόσθετα, αυτή η χρήση θα παρακάμψει κάποιες από αυτές, το οποίο θα μπορούσε ενδεχομένως να προκαλέσει σφάλμα στην εφαρμογή. Παρακαλούμε αντιμετωπίστε το ως **μια λειτουργία για προχωρημένους**, βεβαιωθείτε ότι γνωρίζετε τι κάνετε πριν προχωρήσετε.

```
import { defineConfig } from 'vite'

export default defineConfig({
  slidev: {
    vue: {
      /* ρυθμίσεις vue */
    },
    markdown: {
      /* ρυθμίσεις markdown-it */
      markdownItSetup(md) {
        /* προσαρμοσμένα πρόσθετα markdown-it */
        md.use(/* ... */)
      },
      /* ρυθμίσεις για άλλα πρόσθετα */
    },
  },
})
```

Δείτε τους [ορισμούς των τύπων](#) για περισσότερες επιλογές.

[Go to TOC](#)

Ρυθμίστε Vue

Το Slidev χρησιμοποιεί το [Vue 3](#) για την απεικόνιση της εφαρμογής σε client side. Μπορείτε να επεκτείνετε την εφαρμογή για να προσθέσετε προσαρμοσμένα πρόσθετα ή ρυθμίσεις.

Δημιουργήστε `./setup/main.ts` με το εξής περιεχόμενο:

```
import { defineAppSetup } from '@slidev/types'

export default defineAppSetup(({ app, router }) => {
  // Vue App
  app.use(YourPlugin)
})
```

Αυτό θα μπορούσε επίσης να χρησιμοποιηθεί ως η κύρια είσοδος της εφαρμογής Slidev σας για να κάνετε κάποιες αρχικοποιήσεις πριν την εκκίνηση της.

Μάθετε περισσότερα: [Vue Application API](#).

Ρυθμίστε Windi CSS

Το markdown υποστηρίζει εκ φύσεως ενσωματωμένες επισημάνσεις HTML. Επομένως, μπορείτε να διαμορφώσετε το περιεχόμενό σας με τον τρόπο που θέλετε. Για λόγους ευκολίας, έχουμε ενσωματωμένο το [Windi CSS](#), ώστε να μπορείτε να μορφοποιήσετε τη σήμανση απευθείας χρησιμοποιώντας εργαλεία για class.

Για παράδειγμα:

```
<div class="grid pt-4 gap-4 grids-cols-[100px,1fr]>
  ### Όνομα
  - Στοιχείο 1
  - Στοιχείο 2
</div>
```

Η [Λειτουργία Attributify](#) του [Windi CSS v3.0](#) είναι ενεργοποιημένη από προεπιλογή.

Ρυθμίσεις

Για να ρυθμίσετε το Windi CSS, δημιουργήστε `setup/windicss.ts` με το εξής περιεχόμενο για να επεκτείνετε τις ενσωματωμένες ρυθμίσεις

```
// setup/windicss.ts

import { defineWindiSetup } from '@slidev/types'

// επεκτείνοντας τις ενσωματωμένες ρυθμίσεις του windicss
export default defineWindiSetup(() => ({
  shortcuts: {
    // ρυθμίστε το προεπιλεγμένο φόντο
    'bg-main': 'bg-white text-[#181818] dark:(bg-[#121212] text-[#ddd])',
  },
  theme: {
    extend: {
      // Οι γραμματοσειρές μπορούν να αντικατασταθούν εδώ, θυμηθείτε να
      // ενημερώσετε τους συνδέσμους των γραμματοσειρών στο `index.html`
      fontFamily: {
        sans: 'ui-sans-serif,system-ui,-apple-system,BlinkMacSystemFont,"Segoe UI",Roboto,"Helvetica Neue",Arial,"Noto Sans",sans-serif,"Apple Color Emoji","Segoe UI Emoji","Segoe UI Symbol","Noto Color Emoji",
        mono: 'Fira Code', monospace',
      },
    },
  },
}))
```

Μάθετε περισσότερα για τις [ρυθμίσεις Windi CSS](#)

[Go to TOC](#)

Δομή Φακέλου

To Slidev χρησιμοποιεί ορισμένες συμβάσεις δομής φακέλου για την ελαχιστοποίηση της επιφάνειας ρύθμισης και για να γίνουν οι επεκτάσεις λειτουργικότητας ευέλικτες και διαισθητικές.

Η βασική δομή έχει ως εξής:

```
your-slidev/
  └── components/      # προσαρμοσμένα components
  └── layouts/         # προσαρμοσμένα layouts
  └── public/          # στατικά assets
  └── setup/           # προσαρμοσμένα setup / hooks
  └── styles/          # προσαρμοσμένα style
  └── index.html        # injections στο index.html
  └── slides.md         # η κύρια είσοδος διαφανειών
  └── vite.config.ts    # επεκτείνοντας το vite config
```

Όλα είναι προαιρετικά.

Components

Συμβάσεις: `./components/*.{vue,js,ts,jsx,tsx,md}`

Τα components μέσα σε αυτόν το φάκελο μπορούν να χρησιμοποιηθούν άμεσα στις διαφάνειες Markdown με το ίδιο όνομα component όπως το όνομα του αρχείου.

Για παράδειγμα:

```
your-slidev/
  ...
  └── components/
      └── MyComponent.vue
      └── HelloWorld.ts
```

```
<!-- slides.md -->

# Η διαφάνεια μου

<MyComponent :count="4"/>

<!-- κατ' οι δύο ονομασίες λειτουργούν -->

<hello-world foo="bar">
  Slot
</hello-world>
```

Αυτή η δυνατότητα λειτουργεί με τη βοήθεια του `unplugin-vue-components`, μάθετε περισσότερα εκεί.

To Slidev παρέχει επίσης κάποια [ενσωματωμένα components](#) για να χρησιμοποιήσετε.

Layouts

Συμβάσεις: `./layouts/*.{vue,js,ts,jsx,tsx}`

```
your-slidev/
  ...
  └── layouts/
    ├── cover.vue
    └── my-cool-theme.vue
```

Μπορείτε να χρησιμοποιήσετε οποιοδήποτε όνομα αρχείου για το layout σας. Στη συνέχεια, αναφέρετε το layout σας στην επικεφαλίδα YAML σας χρησιμοποιώντας το όνομα αρχείου.

```
---  
layout: my-cool-theme  
---
```

Αν το layout που παρέχετε έχει το ίδιο όνομα με ένα ενσωματωμένο layout ή ένα layout θέματος, το προσαρμοσμένο layout σας θα έχει προτεραιότητα έναντι του ενσωματωμένου/θεματικού layout. Η σειρά προτεραιότητας είναι `τοπικό > θεματικό > ενσωματωμένο`.

Στο layout component, χρησιμοποιήστε `<slot/>` για το περιεχόμενο της διαφάνειας. Για παράδειγμα:

```
<!-- default.vue -->
<template>
  <div class="slidev-layout default">
    <slot />
  </div>
</template>
```

Public

Συμβάσεις: `./public/*`

Τα assets σε αυτό το φάκελο θα σερβίρεται στο path της ρίζας `/` κατά dev, και αντιγράφεται στη ρίζα του φακέλου dist ως έχει. Διαβάστε περισσότερα για τον [φάκελο public tou Vite](#).

Style

Συμβάσεις: `./style.css | ./styles/index.{css,js,ts}`

Τα αρχεία που ακολουθούν αυτή τη σύμβαση θα γίνονται injected στη ρίζα της εφαρμογής. Εάν πρέπει να εισαγάγετε πολλαπλές εισόδους css, μπορείτε να δημιουργήσετε την ακόλουθη δομή και να διαχειριστείτε μόνοι σας τη σειρά εισαγωγής.

```
your-slidev/
  ...
  └── styles/
    ├── index.ts
    ├── base.css
    ├── code.css
    └── layouts.css
```

```
// styles/index.ts

import './base.css'
import './code.css'
import './layouts.css'
```

Ta styles θα επεξεργαστούν από το [Windi CSS](#) και το [PostCSS](#), ώστε να μπορείτε να χρησιμοποιήσετε css nesting και [at-directives](#) κατευθείαν. Για παράδειγμα:

```
.slidev-layout {
  @apply px-14 py-10 text-[1.1rem];

  h1, h2, h3, h4, p, div {
    @apply select-none;
  }

  pre, code {
    @apply select-text;
  }

  a {
    color: theme('colors.primary');
  }
}
```

Μάθετε περισσότερα για τη σύνταξη.

index.html

Συμβάσεις: `index.html`

To `index.html` παρέχει τη δυνατότητα να κάνετε inject meta tags και/ή scripts στο κύριο `index.html`

Για παράδειγμα, για το ακόλουθο προσαρμοσμένο `index.html`:

```
<!-- ./index.html -->
<head>
  <link rel="preconnect" href="https://fonts.gstatic.com">
  <link href="https://fonts.googleapis.com/css2?family=Fira+Code:wght@400;600&family=Nunito+Sans:wght@200;400;600&display=swap" rel="stylesheet">
</head>

<body>
  <script src=".//your-scripts"></script>
</body>
```

To τελικό `index.html` που θα φιλοξενηθεί θα είναι:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <link rel="icon" type="image/png"
  href="https://cdn.jsdelivr.net/gh/slidevjs/slidev/assets/favicon.png">
  <!-- injected head -->
```

```
<link rel="preconnect" href="https://fonts.gstatic.com">
<link href="https://fonts.googleapis.com/css2?
family=Fira+Code:wght@400;600&family=Nunito+Sans:wght@200;400;600&display=swap"
rel="stylesheet">
</head>
<body>
  <div id="app"></div>
  <script type="module" src="__ENTRY__"></script>
  <!-- injected body -->
  <script src=".//your-scripts"></script>
</body>
</html>
```

Global Layers

Συμβάσεις: [global-top.vue](#) | [global-bottom.vue](#)

Μάθετε περισσότερα: [Global Layers](#)

[Go to TOC](#)

Γραμματοσειρές

Διαθέσιμο από v0.20

Ενώ μπορείτε να χρησιμοποιήσετε HTML και CSS για να προσαρμόσετε τις γραμματοσειρές και το style για τις διαφάνειές σας όπως θέλετε, το Slidev παρέχει επίσης έναν βολικό τρόπο για την αβίαστη χρήση τους.

Στο frontmatter σας, ρυθμίστε τα εξής

```
---  
  fonts:  
    # ουσιαστικά το κείμενο  
    sans: 'Robot'  
    # χρήση με το `font-serif` css class από το windicss  
    serif: 'Robot Slab'  
    # για μπλοκ κώδικα, γραμμικό κώδικα κτλ.  
    mono: 'Fira Code'  
---
```

Και αυτό είναι όλο.

Οι γραμματοσειρές θα **εισαχθούν αυτόματα από το Google Fonts**. Αυτό σημαίνει ότι μπορείτε να χρησιμοποιήσετε απευθείας οποιεσδήποτε γραμματοσειρές είναι διαθέσιμες στο Google Fonts.

Τοπικές Γραμματοσειρές

Εξ ορισμού, το Slidev υποθέτει ότι όλες οι γραμματοσειρές που καθορίζονται μέσω των ρυθμίσεων `fonts` προέρχονται από το Google Fonts. Αν θέλετε να χρησιμοποιήσετε τοπικές γραμματοσειρές, καθορίστε το `fonts.local` για να εξαιρεθείτε από την αυτόματη εισαγωγή.

```
---  
  fonts:  
    # όπως το font-family σε css, μπορείτε να χρησιμοποιήσετε `` ` για να διαχωρίσετε  
    # πολλαπλές γραμματοσειρές ως εφεδρικές  
    sans: 'Helvetica Neue, Robot'  
    # σημαδέψτε 'Helvetica Neue' ως τοπική γραμματοσειρά  
    local: 'Helvetica Neue'  
---
```

Βαριά & Πλάγια

Εξ ορισμού, το Slidev εισάγει τρία βάρη `200, 400, 600` για κάθε γραμματοσειρά. Μπορείτε να τα ρυθμίσετε με:

```
---  
  fonts:  
    sans: 'Robot'  
    # προεπιλογή
```

```

weights: '200,400,600'
# εισαγωγή πλάγιων γραμματοσειρών, προεπιλογή `false`
italic: false
---
```

Αυτή η ρύθμιση ισχύει για όλες τις γραμματοσειρές ιστού. Για πιο λεπτομερή έλεγχο των βαρών κάθε γραμματοσειράς, θα πρέπει να τα εισαγάγετε χειροκίνητα με [HTML](#) και CSS.

Εφεδρικές Γραμματοσειρές

Στα περισσότερα σενάρια, χρειάζεται να καθορίσετε μόνο την "ειδική γραμματοσειρά" και το Slides θα προσθέσει τις εφεδρικές γραμματοσειρές για εσάς, για παράδειγμα:

```

---  

fonts:  

  sans: 'Robot'  

  serif: 'Robot Slab'  

  mono: 'Fira Code'  

---
```

Θα έχει ως αποτέλεσμα

```

.font-sans {
  font-family: "Robot", ui-sans-serif, system-ui, -apple-
system, BlinkMacSystemFont, "Segoe UI", Roboto, "Helvetica Neue", Arial, "Noto
Sans", sans-serif, "Apple Color Emoji", "Segoe UI Emoji", "Segoe UI Symbol", "Noto
Color Emoji";
}
.font-serif {
  font-family: "Robot Slab", ui-serif, Georgia, Cambria, "Times New
Roman", Times, serif;
}
.font-mono {
  font-family: "Fira Code", ui-monospace, SFMono-
Regular, Menlo, Monaco, Consolas, "Liberation Mono", "Courier New", monospace;
}
```

Εάν θέλετε να απενεργοποιήσετε τις εφεδρικές γραμματοσειρές, ρυθμίστε ως εξής

```

---  

fonts:  

  mono: 'Fira Code, monospace'  

  fallback: false  

---
```

Πάροχοι

- Επιλογές: `google` | `none`
- Προεπιλογή: `google`

Επί του παρόντος, υποστηρίζεται μόνο το Google Fonts, σχεδιάζουμε να προσθέσουμε περισσότερους παρόχους στο μέλλον. Αν ορίσετε την τιμή `none`, θα απενεργοποιήσετε εντελώς τη δυνατότητα αυτόματης εισαγωγής και θα αντιμετωπίσετε όλες τις γραμματοσειρές ως τοπικές.

```
---  
  fonts:  
    provider: 'none'  
---
```

[Go to TOC](#)

Global Layers

Διαθέσιμο από v0.17

Τα global layers σας επιτρέπουν να έχετε προσαρμοσμένα στοιχεία που είναι **μόνιμα** σε όλες τις διαφάνειες. Αυτό θα μπορούσε να είναι χρήσιμο για footers, κινήσεις διαφανειών, global effects, κτλ.

To Sliddev παρέχει τρία layers για αυτή τη χρήση, δημιουργήστε `global-top.vue`, `global-bottom.vue` ή `custom-nav-controls.vue` στη ρίζα του έργου σας και θα το αναλάβει αυτόματα.

Σχέση layers:

- Global Top (`global-top.vue`)
- Διαφάνειες
- Global Bottom (`global-bottom.vue`)
- NavControls
 - Προσαρμοσμένοι έλεγχοι πλοήγησης (`custom-nav-controls.vue`)

Παράδειγμα

```
<!-- global-bottom.vue -->
<template>
  <footer class="absolute bottom-0 left-0 right-0 p-2">Το ονομά σας</footer>
</template>
```

Το κείμενο `Το ονομά σας` θα εμφανιστεί σε όλες τις διαφάνειές σας.

```
<!-- custom-nav-controls -->
<template>
  <button class="icon-btn" title="Next" @click="$slidev.nav.next">
    <carbon:arrow-right />
  </button>
</template>
```

Το κουμπί `Next` θα εμφανιστεί στο NavControls.

Για να το ενεργοποιήσετε υπό όρους, μπορείτε να το εφαρμόσετε με το [Vue Global Context](#).

```
<!-- κρύψτε το footer στην Σελίδα 4 -->
<template>
  <footer
    v-if="$slidev.nav.currentPage !== 4"
    class="absolute bottom-0 left-0 right-0 p-2">
    Το ονομά σας
  </footer>
</template>
```

```
<!-- κρύψτε το footer στο "cover" layout -->
<template>
  <footer>
    <!-- Το ονομά σας -->
  </footer>
</template>
```

```
<!-- ένα παράδειγμα footer για τις σελίδες -->
<template>
  <footer>
    <!-- Το ονομά σας -->
  </footer>
</template>
```

```
<!-- custom-nav-controls -->
<!-- κρύψτε το κουμπί στη λειτουργία παρουσιαστή -->
<template>
  <button v-if="!$slidev.nav.isPresenter" class="icon-btn" title="Next"
    @click="$slidev.nav.next">
    <carbon:arrow-right />
  </button>
</template>
```

Επισημαντήρες

To Slidev περιλαμβάνει δύο επισημαντήρες σύνταξης για να επιλέξετε:

- [Prism](#)
- [Shiki](#)

Το **Prism** είναι ένας από τους πιο δημοφιλής επισημαντήρες σύνταξης. Η επισήμανση γίνεται με την προσθήκη token classes στον κώδικα και χρωματίζεται με τη χρήση CSS. Μπορείτε να περιηγηθείτε στα [επίσημα θέματά](#) τους, ή δημιουργήστε/προσαρμόστε ένα μόνοι σας πολύ εύκολα χρησιμοποιώντας το `prism-theme-vars`.

Το **Shiki**, από την άλλη πλευρά, είναι ένας TextMate grammar-powered επισημαντήρας σύνταξης. Δημιουργεί χρωματιστά tokens, ώστε να μην απαιτείται επιπλέον CSS. Δεδομένου ότι έχει εξαιρετική υποστήριξη γραμματικής, τα παραγόμενα χρώματα είναι πολύ ακριβή, ακριβώς όπως αυτά που θα δείτε στο VS Code. Το Shiki επιπλέον περιλαμβάνει [ένα σωρό από ενσωματωμένα θέματα](#). Το μειονέκτημα του Shiki είναι ότι απαιτεί επίσης θέματα TextMate (συμβατά με θέματα VS Code) για να κάνει επισήμανση, τα οποία μπορεί να είναι λίγο πιο δύσκολο να προσαρμοστούν.

Τα θέματα Slidev συνήθως υποστηρίζουν και Prism και Shiki, αλλά ανάλογα με το θέμα που χρησιμοποιείτε, μπορεί να υποστηρίζει μόνο ένα από αυτά.

Όταν έχετε τη δυνατότητα επιλογής, ο συμβιβασμός είναι βασικά:

- **Prism** για ευκολότερη προσαρμογή
- **Shiki** για πιο ακριβή επισήμανση

Εξ ορισμού, το Slidev χρησιμοποιεί το Prism. Μπορείτε να το αλλάξετε τροποποιώντας το frontmatter σας:

```
---  
highlighter: shiki  
---
```

Ρυθμίστε Prism

Για να ρυθμίσετε το Prism σας, μπορείτε απλά να εισαγάγετε το `css` του θέματος ή να χρησιμοποιήσετε το `prism-theme-vars` για να ρυθμίσετε τα θέματα τόσο για τη φωτεινή όσο και για τη σκοτεινή λειτουργία. Ανατρέξτε στις οδηγίες του για περισσότερες λεπτομέρειες.

Ρυθμίστε Shiki

Δημιουργήστε `./setup/shiki.ts` με το εξής περιεχόμενο

```
/* ./setup/shiki.ts */  
import { defineShikiSetup } from '@slidev/types'  
  
export default defineShikiSetup(() => {  
    return {
```

```
        theme: {
          dark: 'min-dark',
          light: 'min-light',
        },
      })
    })
```

Ανατρέξτε στις [οδηγίες του Shiki](#) για τα ονόματα διαθέσιμων θεμάτων.

Ή αν θέλετε να χρησιμοποιήσετε το δικό σας θέμα:

```
/* ./setup/shiki.ts */

import { defineShikiSetup } from '@slidev/types'

export default defineShikiSetup(async({ loadTheme }) => {
  return {
    theme: {
      dark: await loadTheme('path/to/theme.json'),
      light: await loadTheme('path/to/theme.json'),
    },
  }
})
```

[Go to TOC](#)

Προσαρμογές

Το Slidelv είναι πλήρως προσαρμόσιμο, από το style έως τις ρυθμίσεις των εργαλείων. Σας επιτρέπει να ρυθμίσετε τα εσωτερικά εργαλεία ([Vite](#), [Windi CSS](#), [Monaco](#), κτλ.)

Ρυθμίσεις Frontmatter

Μπορείτε να ρυθμίσετε το Slidelv στο frontmatter της πρώτης σας διαφάνειας, τα παρακάτω δείχνουν την προεπιλεγμένη τιμή για κάθε επιλογή.

```
---
# id ή όνομα πακέτου θέματος
# Μάθετε περισσότερα: https://el.sli.dev/themes/use.html
theme: 'default'
# τίτλος της διαφάνειάς σας, αν δεν καθοριστεί θα προκύψει αυτόματα από την πρώτη επικεφαλίδα
title: 'Slidelv'
# titleTemplate της εποικείας, `%s` θα αντικατασταθεί από τον τίτλο της σελίδας
titleTemplate: '%s - Slidelv'
# πληροφορίες για τις διαφάνειές σας, μπορεί να είναι μια συμβολοσειρά markdown
info: false

# ενεργοποιήστε τη λήψη pdf στην κατασκευή SPA, μπορεί επίσης να είναι ένα προσαρμοσμένο url
download: false
# όνομα αρχείου του αρχείου εξαγωγής
exportFilename: 'slidelv-exported'
# επισημαντής σύνταξης, μπορεί να είναι 'prism' ή 'shiki'
highlighter: 'prism'
# εμφάνιση αριθμών γραμμών σε blocks
lineNumbers: false
# ενεργοποίηση του επεξεργαστή monaco, μπορεί να είναι boolean, 'dev' ή 'build'
monaco: 'dev'
# λήψη απομακρυσμένων assets τοπικά χρησιμοποιώντας το vite-plugin-remote-assets, μπορεί να είναι boolean, 'dev' ή 'build'
remoteAssets: false
# ελέγχει αν τα κείμενα στις διαφάνειες είναι επιλέξιμα
selectable: true
# ενεργοποίηση της εγγραφής διαφανειών, μπορεί να είναι boolean, 'dev' ή 'build'
record: 'dev'

# αναγκάστε χρωματικό σχήμα για τις διαφάνειες, μπορεί να είναι 'auto', 'light', ή 'dark'
colorSchema: 'auto'
# λειτουργία router για το vue-router, μπορεί να είναι "history" ή "hash"
routerMode: 'history'
# αναλογία διαστάσεων για τις διαφάνειες
aspectRatio: '16/9'
# πραγματικό πλάτος του καμβά, μονάδα σε px
canvasWidth: 980
# χρησιμοποιείται για την προσαρμογή του θέματος, θα εισάγει τα στυλ της ρίζας ως `--slidelv-theme-x` για το χαρακτηριστικό `x`
themeConfig:
  primary: '#5d8392'

# favicon, μπορεί να είναι θέση τοπικού αρχείου ή διεύθυνση URL
```

```

favicon: 'https://cdn.jsdelivr.net/gh/slidevjs/slidev/assets/favicon.png'
# URL του διακομιστή PlantUML που χρησιμοποιείται για την απεικόνιση των
διαγραμμάτων
plantUmlServer: 'https://www.plantuml.com/plantuml'
# οι γραμματοσειρές θα εισαχθούν αυτόματα από το Google fonts
# Μάθετε περισσότερα: https://elsli.dev/custom/fonts
fonts:
  sans: 'Roboto'
  serif: 'Roboto Slab'
  mono: 'Fira Code'

# προεπιλεγμένο frontmatter ισχύει για όλες τις διαφάνειες
defaults:
  layout: 'default'
  # ...

# επιλογές σχεδίασης
# Μάθετε περισσότερα: https://elsli.dev/guide/drawing.html
drawings:
  enabled: true
  persist: false
  presenterOnly: false
  syncAll: true
---

```

Δείτε τους [ορισμούς των τύπων](#) για περισσότερες επιλογές.

Δομή Φακέλου

Το Slidev χρησιμοποιεί συμβάσεις δομής φακέλου για την ελαχιστοποίηση της επιφάνειας ρύθμισης και κάνει τις επεκτάσεις ευέλικτες και διαισθητικές στη λειτουργικότητα.

Ανατρέξτε στην ενότητα [Δομή Φακέλου](#).

Ρυθμίστε Εργαλεία

- [Επισημαντήρες](#)
- [Ρυθμίστε Vue](#)
- [Ρυθμίστε Vite](#)
- [Ρυθμίστε Windi CSS](#)
- [Ρυθμίστε Monaco](#)
- [Ρυθμίστε KaTeX](#)
- [Ρυθμίστε Mermaid](#)

[Go to TOC](#)

Vue Global Context

To Slidev κάνει inject ένα global Vue context `$slidev` για προηγμένες συνθήκες ή χειριστήρια πλοήγησης.

Χρήση

Μπορείτε να έχετε πρόσβαση σε αυτό οπουδήποτε στα markdown και Vue template, με την σύνταξη "Mustache".

```
<!-- slides.md -->
# Σελίδα 1
Η τρέχουσα σελίδα είναι: {{ $slidev.nav.currentPage }}
```

```
<!-- Foo.vue -->
<template>
  <div>Τίτλος: {{ $slidev.configs.title }}</div>
  <button @click="$slidev.nav.next">Επόμενη Σελίδα</button>
</template>
```

Ιδιότητες

`$slidev.nav`

Ένα αντιδραστικό object που περιέχει τις ιδιότητες και τα χειριστήρια της πλοήγησης των διαφανειών. Για παραδείγματα:

```
$slidev.nav.next() // πήγαινε στο επόμενο βήμα
$slidev.nav.nextSlide() // πήγαινε στην επόμενη διαφάνεια (παραλειψετε v-clicks)
$slidev.nav.go(10) // πήγαινε στην διαφάνεια #10
```

```
$slidev.nav.currentPage // αριθμός τρέχουσας διαφάνειας
$slidev.nav.currentLayout // id τρέχοντος layout
$slidev.nav.clicks // σύνολο τρεχόντων clicks
```

Για περισσότερες διαθέσιμες ιδιότητες, ανατρέξτε στις εξαγωγές του `nav.ts`.

`$slidev.configs`

Ένα αντιδραστικό object κρατάει τις αναλυμένες [ρυθμίσεις στο πρώτο frontmatter](#) του `slides.md` σας. Για παράδειγμα

```
---  
title: Το πρώτο μου Slidev!  
---
```

```
  {{ $slidev.configs.title }} // 'To πρώτο μου Slidev!'
```

\$slidev.themeConfigs

Ένα αντιδραστικό object κρατάει τις αναλυμένες ρυθμίσεις του θέματος.

```
---
```

```
  title: Το πρώτο μου Slidev!
  themeConfig:
    primary: #213435
---
```

```
  {{ $slidev.themeConfigs.primary }} // '#213435'
```

[Go to TOC](#)

Κινήσεις

Κινήσεις Click

v-click

Για να εφαρμόσετε "κινήσεις click" σε στοιχεία, μπορείτε να χρησιμοποιήσετε την οδηγία `v-click` ή τα `<v-click>` components

```
# Γεια
<! -- Χρήση component: αυτό θα είναι αόρατο μέχρι να πατήσετε "next" -->
<v-click>

Γεια σου Κόσμε
</v-click>

<! -- Χρήση οδηγιας: αυτό θα είναι αόρατο μέχρι να πατήσετε "next" για δεύτερη φορά
-->
<div v-click class="text-xl p-2">
    Ει!
</div>
```

v-after

Η χρήση του `v-after` είναι παρόμοια με το `v-click` αλλά θα κάνει το στοιχείο ορατό όταν ενεργοποιηθεί το προηγούμενο `v-click`.

```
<div v-click>Γεια σου</div>
<div v-after>Κόσμε</div>
```

Όταν κάνετε click στο κουμπί "next", τόσο το `Γεια σου` όσο και το `Κόσμε` θα εμφανιστούν μαζί.

v-click-hide

Το ίδιο με το `v-click`, αλλά αντί να κάνει το στοιχείο να εμφανιστεί, κάνει το στοιχείο αόρατο μετά το click.

```
<div v-click-hide>Γεια</div>
```

v-clicks

Το `v-clicks` παρέχεται μόνο ως component. Είναι μια συντομογραφία για την εφαρμογή της οδηγίας `v-click` σε όλα τα στοιχεία-παιδιά του. Είναι ιδιαίτερα χρήσιμο όταν δουλεύετε με λίστες.

```
<v-clicks>
  - Στοιχείο 1
  - Στοιχείο 2
```

```
- Στοιχείο 3
- Στοιχείο 4

</v-clicks>
```

Ένα στοιχείο θα γίνεται ορατό κάθε φορά που κάνετε click στο "next".

Προσαρμοσμένος Αριθμός Clicks

Εξ ορισμού, το Slidev μετράει πόσα βήματα απαιτούνται πριν από τη μετάβαση στην επόμενη διαφάνεια. Μπορείτε να παρακάμψετε αυτή τη ρύθμιση περνώντας την επιλογή `clicks` frontmatter:

```
---  
# 10 click σε αυτή τη διαφάνεια, πριν προχωρήσετε στην επόμενη  
clicks: 10  
---
```

Διάταξη

Περνώντας το δείκτη των click στις οδηγίες σας, μπορείτε να προσαρμόσετε τη σειρά της αποκάλυψης

```
<div v-click>1</div>
<div v-click>2</div>
<div v-click>3</div>
```

```
<!-- η σειρά είναι αντίστροφη -->
<div v-click="3">1</div>
<div v-click="2">2</div>
<div v-click="1">3</div>
```

```
---  
clicks: 3  
---  
  
<!-- ορατό μετά από 3 clicks -->  
<v-clicks at="3">  
  <div>Γεια</div>  
</v-clicks>
```

Μεταβάσεις Στοιχείων

Όταν εφαρμόζετε την οδηγία `v-click` στα στοιχεία σας, θα τους επισυνάψει το όνομα του class `slidev-vclick-target`. Όταν τα στοιχεία είναι κρυμμένα, θα επισυνάπτεται επίσης το όνομα του class `slidev-vclick-hidden`. Για παράδειγμα:

```
<div class="slidev-vclick-target slidev-vclick-hidden">Κείμενο</div>
```

Μετά από ένα click, θα γίνει

```
<div class="slidev-vclick-target">Κείμενο</div>
```

Εξ ορισμού, μια διακριτική μετάβαση αδιαφάνειας εφαρμόζεται σε αυτά τα classes:

```
// η προεπιλογή

.slidev-vclick-target {
  transition: opacity 100ms ease;
}

.slidev-vclick-hidden {
  opacity: 0;
  pointer-events: none;
}
```

Μπορείτε να τα παρακάμψετε για να προσαρμόσετε τα effects μετάβασης στα προσαρμοσμένα stylesheets σας.

Για παράδειγμα, μπορείτε να επιτύχετε τις μεταβάσεις κλιμάκωσης προς τα πάνω με:

```
// styles.css

.slidev-vclick-target {
  transition: all 500ms ease;
}

.slidev-vclick-hidden {
  transform: scale(0);
}
```

Για να καθορίσετε τις κινήσεις μόνο για ορισμένες διαφάνειες ή layouts

```
.slidev-page-7,
.slidev-layout.my-custom-layout {
  .slidev-vclick-target {
    transition: all 500ms ease;
  }

  .slidev-vclick-hidden {
    transform: scale(0);
  }
}
```

Μάθετε περισσότερα για την [προσαρμογή των styles](#)

Κίνηση

To Slidev έχει ενσωματωμένο το [@vueuse/motion](#). Μπορείτε να χρησιμοποιήσετε την οδηγία `v-motion` σε οποιαδήποτε στοιχεία για να εφαρμόσετε κίνηση σε αυτά. Για παράδειγμα

```
<div
  v-motion
  :initial="{ x: -80 }"
  :enter="{ x: 0 }">
  Slidev
</div>
```

Το κείμενο `Slidev` θα μετακινηθεί από το `-80px` στην αρχική του θέση κατά την αρχικοποίηση.

Σημείωση: Το Slidev φορτώνει προκαταβολικά την επόμενη διαφάνεια για λόγους απόδοσης, πράγμα που σημαίνει ότι οι κινήσεις μπορεί να ξεκινήσουν πριν πλοηγηθείτε στη σελίδα. Για να λειτουργήσει σωστά, μπορείτε να απενεργοποιήσετε την προφόρτωση για τη συγκεκριμένη διαφάνεια

```
---  
preload: false  
---
```

Ή ελέγξτε τον κύκλο ζωής του στοιχείου με το `v-if` για να έχετε λεπτομερείς χειρισμούς

```
<div  
  v-if="$slidev.nav.currentPage === 7"  
  v-motion  
  :initial="{ x: -80 }"  
  :enter="{ x: 0 }">  
  Slidev  
</div>
```

Μάθετε περισσότερα: [Demo](#) | [@vueuse/motion](#) | [v-motion](#) | [Presets](#)

Μεταβάσεις Σελίδων

Ενσωματωμένη υποστήριξη για τις διαφάνειες ΔΕΝ παρέχεται ΑΚΟΜΑ στην τρέχουσα έκδοση. Σχεδιάζουμε να προσθέσουμε την υποστήριξή τους στην επόμενη κύρια έκδοση. Πριν γίνει αυτό, μπορείτε ακόμα να χρησιμοποιήσετε τα προσαρμοσμένα `styles` και τα `libraries` σας για να το κάνετε αυτό.

[Go to TOC](#)

Ζωγραφική & Σημειώσεις

Διαθέσιμο από v0.23

Έχουμε ενσωματωμένο το [drauu](#) για ζωγραφική και σημειώσεις που θα μπορούσαν να βελτιώσουν περαιτέρω την παρουσίασή σας.

Για να ξεκινήσετε, κάντε κλικ στο εικονίδιο στη γραμμή εργαλείων και αρχίστε να ζωγραφίζετε. Είναι επίσης διαθέσιμο στη [Λειτουργία Παρουσιαστή](#). Οι ζωγραφιές και οι σημειώσεις που δημιουργήσατε θα **συγχρονιστούν** αυτόματα σε όλες τις εφαρμογές σε πραγματικό χρόνο.

Χρήση με Stylus Pen

Όταν χρησιμοποιείτε ένα stylus pen σε ένα tablet (για παράδειγμα, iPad με Apple Pencil), το Slidew μπορεί να ανιχνεύσει έξυπνα τον τύπο εισόδου. Μπορείτε να ζωγραφίσετε απευθείας στις διαφάνειές σας με το στυλό χωρίς να ενεργοποιήσετε τη λειτουργία ζωγραφικής, ενώ τα δάχτυλα ή το ποντίκι σας ελέγχουν την πλοήγηση.

Επίμονες Ζωγραφιές

Η ακόλουθη ρύθμιση frontmatter σας επιτρέπει να διατηρήσετε τα σχέδιά σας ως SVGs στον φάκελο `.slidev/drawings` και να τα έχετε μέσα στο εξαγόμενο pdf σας ή στο φιλοξενούμενο ιστότοπο.

```
---  
drawings:  
  persist: true  
---
```

Απενεργοποίηση Ζωγραφιών

Εντελώς:

```
---  
drawings:  
  enabled: false  
---
```

Μόνο σε Development:

```
---  
drawings:  
  enabled: dev  
---
```

Μόνο σε Λειτουργία Παρουσιαστή:

```
---  
drawings:  
  presenterOnly: true  
---
```

Συγχρονισμός Ζωγραφιών

Από προεπιλογή, το Slidev συγχρονίζει τις ζωγραφιές σας σε όλες τις εφαρμογές. Αν μοιράζεστε τις διαφάνειές σας με άλλους, ίσως θελήσετε να απενεργοποιήσετε το συγχρονισμό μέσω:

```
---  
drawings:  
  syncAll: false  
---
```

Με αυτή τη ρύθμιση, μόνο οι ζωγραφιές από την εφαρμογή του παρουσιαστή θα μπορούν να συγχρονιστούν με άλλες.

[Go to TOC](#)

Υποστήριξη Συντάκτη

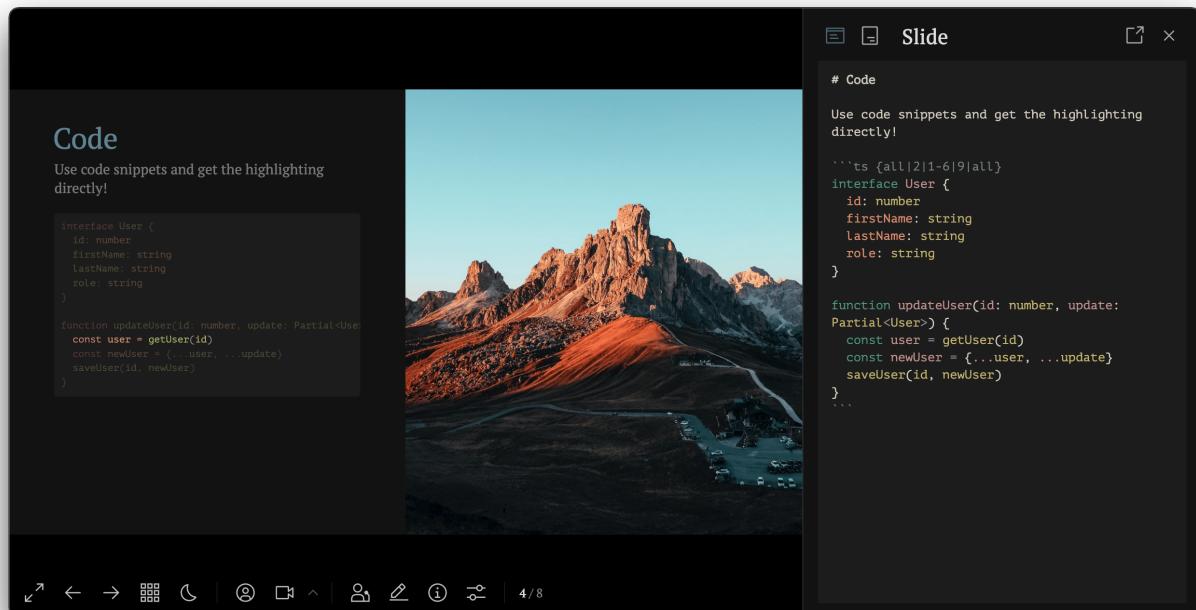
Δεδομένου ότι το Slidev χρησιμοποιεί Markdown ως πηγή εισόδου, μπορείτε να χρησιμοποιήσετε ΟΠΟΙΟΝΔΗ-ΠΟΤΕ συντάκτη αγαπάτε για να το γράψετε.

Αν θέλετε κάποια υψηλού επιπέδου διαχείριση των διαφανειών σας, σας παρέχουμε τις ακόλουθες ενσωματώσεις επεξεργαστή!

Ενσωματωμένος Συντάκτης

Το Slidev περιλαμβάνει ένα ενσωματωμένο συντάκτη [CodeMirror](#) που θα επαναφορτώσει και θα αποθηκεύσει αμέσως τις αλλαγές στο αρχείο σας.

Κάντε κλικ στο κουμπί για να το ανοίξετε.



Επέκταση VS Code



Slidev for VS Code

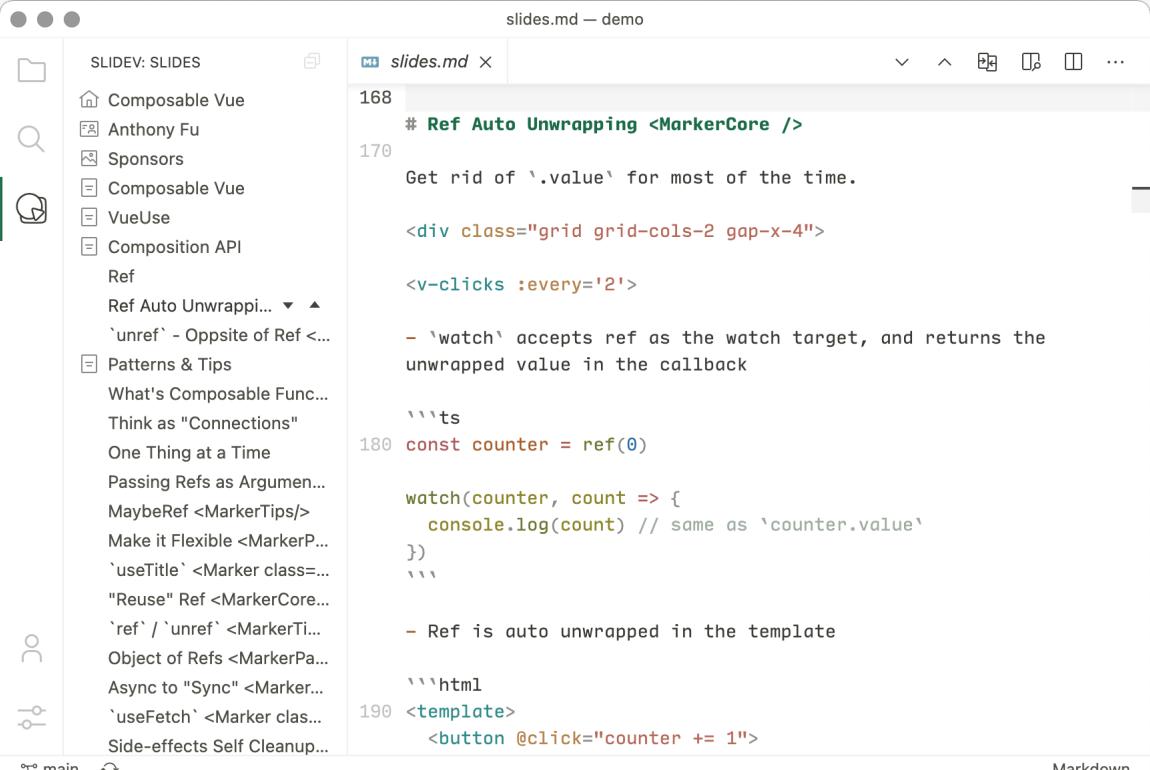
VS Code Marketplace v0.4.1

downloads 26k

Η επέκταση VS Code παρέχει ορισμένες δυνατότητες που σας βοηθούν να οργανώσετε καλύτερα τις διαφάνειές σας και να έχετε μια γρήγορη επισκόπηση αυτών.

Δυνατότητες

- Προβολή διαφανειών στον πλευρικό πίνακα
- Κουμπιά μετάβασης στην επόμενη / προηγούμενη διαφάνεια
- Αναδιάταξη διαφανειών
- Αναδίπλωση για block ολίσθησης
- Μετατροπή Markdown σε HTML



The screenshot shows a code editor window with the file `slides.md` open. The left sidebar displays a tree view of files under the folder `SLIDEV: SLIDES`, including `Composable Vue`, `Anthony Fu`, `Sponsors`, `Composable Vue`, `VueUse`, `Composition API`, `Ref`, `Ref Auto Unwrapping`, `'unref' - Oppsite of Ref`, `Patterns & Tips`, `What's Composable Func...`, `Think as "Connections"`, `One Thing at a Time`, `Passing Refs as Argument...`, `MaybeRef <MarkerTips/>`, `Make it Flexible <MarkerP...`, `'useTitle' <Marker class=...`, `"Reuse" Ref <MarkerCore...`, `'ref' / 'unref' <MarkerTi...`, `Object of Refs <MarkerPa...`, `Async to "Sync" <Marker...`, `'useFetch' <Marker clas...`, and `Side-effects Self Cleanup...`. The main editor area contains the following code:

```

168 # Ref Auto Unwrapping <MarkerCore />
170 Get rid of `value` for most of the time.
<div class="grid grid-cols-2 gap-x-4">
<v-clicks :every='2'>
  - `watch` accepts ref as the watch target, and returns the
    unwrapped value in the callback
``ts
180 const counter = ref(0)

watch(counter, count => {
  console.log(count) // same as `counter.value`
})
```
 - Ref is auto unwrapped in the template
``html
190 <template>
 <button @click="counter += 1">

```

---

[Go to TOC](#)

# Εξαγωγή

## Διαφάνειες

### PDF

Η εξαγωγή σε PDF ή PNG βασίζεται στο [Playwright](#) για την απεικόνιση. Επομένως, θα πρέπει να εγκαταστήσετε το `playwright-chromium` για να χρησιμοποιήσετε αυτή τη λειτουργία. Αν κάνετε εξαγωγή σε περιβάλλον CI, μπορεί να σας φανεί χρήσιμος [ο οδηγός CI του playwright](#).

Εγκαταστήστε `playwright-chromium`

```
$ npm i -D playwright-chromium
```

Τώρα εξάγετε τις διαφάνειές σας σε PDF χρησιμοποιώντας την ακόλουθη εντολή

```
$ slidev export
```

Μετά από λίγα δευτερόλεπτα, οι διαφάνειές σας θα είναι έτοιμες στο `./slides-export.pdf`.

Σε περίπτωση που θέλετε να εξάγετε τις διαφάνειές σας χρησιμοποιώντας τη σκοτεινή έκδοση του θέματος, χρησιμοποιήστε την επιλογή `--dark`:

```
$ slidev export --dark
```

### Εξαγωγή Βημάτων Clicks

Διαθέσιμο από v0.21

Εξ ορισμού, το Slidev εξάγει μία σελίδα ανά διαφάνεια με απενεργοποιημένες τις κινήσεις click. Εάν θέλετε να εξάγετε διαφάνειες με πολλαπλά βήματα σε πολλαπλές σελίδες, περάστε την επιλογή `--with-clicks`.

```
$ slidev export --with-clicks
```

### PNGs

Όταν περνάτε την επιλογή `--format png`, το Slidev θα εξάγει εικόνες PNG για κάθε διαφάνεια αντί για PDF.

```
$ slidev export --format png
```

## Single-Page Application (SPA)

Δείτε την [Στατική Φιλοξενία](#).

# Σημειώσεις παρουσιαστή

Διαθέσιμο από v0.36.8

Εξάγετε μόνο τις σημειώσεις του παρουσιαστή (το τελευταίο μπλοκ σχολίων για κάθε διαφάνεια) σε ένα έγγραφο κειμένου σε PDF.

 \$ slidev export-notes

---

[Go to TOC](#)

# Συχνές Ερωτήσεις

## Πλέγματα

Από τη στιγμή που το Slides βασίζεται στο Διαδίκτυο, μπορείτε να εφαρμόσετε οποιοδήποτε layout πλέγματος όπως θέλετε. [CSS Grids](#), [flexboxes](#), ή ακόμα και [Masonry](#), έχετε τον πλήρη έλεγχο.

Από τη στιγμή που έχουμε ενσωματωμένο το [Windi CSS](#), εδώ είναι ένας απλός τρόπος για να αναφερθείτε:

```
<div class="grid grid-cols-2 gap-4">
<div>
 Η πρώτη στήλη
</div>
<div>
 Η δεύτερη στήλη
</div>
```

Για να προχωρήσετε περαιτέρω, μπορείτε να προσαρμόσετε το μέγεθος κάθε στήλης όπως:

```
<div class="grid grid-cols-[200px,1fr,10%] gap-4">
<div>
 Η πρώτη στήλη (200px)
</div>
<div>
 Η δεύτερη στήλη (auto fit)
</div>
<div>
 Η τρίτη στήλη (10% πλάτος στο parent container)
</div>
```

Μάθετε περισσότερα για το [Windi CSS Grids](#).

## Τοποθέτηση

Οι διαφάνειες ορίζονται σε σταθερά μεγέθη (προεπιλογή `980x552px`) και να προσαρμόζεται στην οθόνη του χρήστη. Μπορείτε να χρησιμοποιήσετε με ασφάλεια απόλυτη θέση στις διαφάνειές σας, καθώς αυτές θα κλιμακώνονται μαζί με την οθόνη.

Για παράδειγμα:

```
<div class="absolute left-30px bottom-30px">
Αυτό είναι ένα footer ευθυγραμμισμένο αριστερά προς τα κάτω
</div>
```

Για να αλλάξετε το πραγματικό μέγεθος του καμβά, μπορείτε να περάσετε την επιλογή `canvasWidth` στο πρώτο σας frontmatter:

```

canvasWidth: 800

```

## Μέγεθος Γραμματοσειράς

Αν αισθάνεστε ότι το μέγεθος της γραμματοσειράς στις διαφάνειές σας είναι πολύ μικρό, μπορείτε να το προσαρμόσετε με διάφορους τρόπους:

### Παράκαμψη Τοπικού Style

Μπορείτε να παρακάμψετε τα στυλ για κάθε διαφάνεια με την ενσωματωμένη ετικέτα `<style>`.

```
Σελίδα 1
<style>
h1 {
 font-size: 10em;
}
</style>

Σελίδα 2
Αυτό δεν θα επηρεαστεί.
```

Μάθετε περισσότερα: [Ενσωματωμένα Styles](#)

### Παράκαμψη Global Style

Μπορείτε να παρέχετε προσαρμοσμένα global styles δημιουργώντας το `./style.css`, για παράδειγμα

```
/* style.css */
h1 {
 font-size: 10em !important;
}
```

Μάθετε περισσότερα: [Global Style](#)

## Κλιμάκωση του Καμβά

Η αλλαγή του πραγματικού μεγέθους του καμβά θα κλιμακώσει όλα τα περιεχόμενά σας (κείμενο, εικόνες, στοιχεία κτλ.) και τις διαφάνειες

```

προεπιλογή: 980
δεδομένου ότι ο καμβάς μικραίνει, το οπτικό μέγεθος θα γίνει μεγαλύτερο
canvasWidth: 800

```

## Χρήση Transform

Παρέχουμε ένα ενσωματωμένο στοιχείο `<Transform />` component, το οποίο είναι ένα λεπτό περιτύλιγμα της ιδιότητας CSS transform.

```
<Transform :scale="1.4">
- Στοιχείο 1
- Στοιχείο 2
</Transform>
```

---

[Go to TOC](#)

# Στατική Φιλοξενία

## Build Single Page Applications (SPA)

Μπορείτε επίσης να μετατρέψετε τις διαφάνειες σε ένα αυτο-φιλοξενούμενο SPA:

```
$ slidev build
```

Η παραγόμενη εφαρμογή θα είναι διαθέσιμη στο `dist/` και στη συνέχεια μπορείτε να την φιλοξενήσετε στα [GitHub Pages](#), [Netlify](#), [Vercel](#), ή όπου αλλού θέλετε. Τώρα μπορείτε να μοιραστείτε τις διαφάνειές σας με τον υπόλοιπο κόσμο με έναν μόνο σύνδεσμο.

### Βασικό Path

Για να εγκαταστήσετε τις διαφάνειές σας κάτω από υποδιαδρομές, θα πρέπει να περάσετε την επιλογή `--base`. Για παράδειγμα:

```
$ slidev build --base /ομιλίες/η-cool-ομιλία-μου/
```

Ανατρέξτε στις [οδηγίες χρήσης του Vite](#) για περισσότερες πληροφορίες.

### Παροχή PDF με δυνατότητα λήψης

Μπορείτε να παρέχετε ένα PDF με δυνατότητα λήψης στους θεατές του SPA σας με την ακόλουθη ρύθμιση:

```

download: true

```

To Slidev θα δημιουργήσει ένα αρχείο PDF μαζί με την κατασκευή και ένα κουμπί λήψης θα εμφανιστεί στο SPA.

Μπορείτε επίσης να δώσετε μια προσαρμοσμένη διεύθυνση URL για το PDF. Σε αυτή την περίπτωση, η διαδικασία απεικόνισης θα παραλειφθεί.

```

download: 'https://myside.com/my-talk.pdf'

```

### Παραδείγματα

Ακολουθούν μερικά παραδείγματα εξαγομένων SPA:

- [Πρότυπο Εκκίνησης](#)
- [Composable Vue του Anthony Fu](#)

Για περισσότερα, δείτε τις [Παρουσιάσεις](#).

## Φιλοξενία

Συνιστούμε να χρησιμοποιήσετε το `npm init slidev@latest` για την δημιουργία του έργου σας, το οποίο περιέχει τα απαραίτητα αρχεία ρυθμίσεων για τις υπηρεσίες φιλοξενίας.

### Netlify

- [Netlify](#)

Δημιουργήστε το αρχείο `netlify.toml` στη ρίζα του έργου σας με το ακόλουθο περιεχόμενο.

```
[build.environment]
 NODE_VERSION = "14"

[build]
 publish = "dist"
 command = "npm run build"

[[redirects]]
 from = "/"
 to = "/index.html"
 status = 200
```

Στη συνέχεια, μεταβείτε στο πίνακα ελέγχου του Netlify και δημιουργήστε ένα νέο ιστότοπο με το repository.

### Vercel

- [Vercel](#)

Δημιουργήστε το αρχείο `vercel.json` στη ρίζα του έργου σας με το ακόλουθο περιεχόμενο.

```
{
 "rewrites": [
 { "source": "/(.*)", "destination": "/index.html" }
]
}
```

Στη συνέχεια, μεταβείτε στο πίνακα ελέγχου του Vercel και δημιουργήστε ένα νέο ιστότοπο με το repository.

## GitHub Pages

- [GitHub Pages](#)

Για να εγκαταστήσετε τις διαφάνειές σας στο GitHub Pages:

- ανεβάστε όλα τα αρχεία του έργου στο repo σας (π.χ. με το όνομα `name_of_repo`)
- δημιουργήστε το αρχείο `.github/workflows/deploy.yml` με το ακόλουθο περιεχόμενο για να εγκαταστήσετε τις διαφάνειές σας στο GitHub Pages μέσω των GitHub Actions. Σε αυτό το αρχείο, αντικαταστήστε το `<name_of_repo>` με το `name_of_repo`.

```
name: Deploy pages
on: push
jobs:
```

```
deploy:
 runs-on: ubuntu-latest
 steps:
 - uses: actions/checkout@v2
 - uses: actions/setup-node@v2
 with:
 node-version: '14'
 - name: Install dependencies
 run: npm install
 - name: Install slidev
 run: npm i -g @slidev/cli
 - name: Build
 run: slidev build --base <name_of_repo>
 - name: Deploy pages
 uses: crazy-max/ghaction-github-pages@v2
 with:
 build_dir: dist
 env:
 GITHUB_TOKEN: ${{ secrets.GITHUB_TOKEN }}
```

- Στο repository σας, μεταβείτε στην επιλογή Settings>Pages. Στην ενότητα "Build and deployment", επιλέξτε "Deploy from a branch", επιλέξτε "gh-pages" και "root". Κάντε κλικ στο save.
- Τέλος, αφού εκτελεστούν όλες οι διαδικασίες εργασιών, θα πρέπει να εμφανιστεί ένας σύνδεσμος προς τις διαφάνειες στο Settings>Pages.

# Ξεκινώντας

## Σύνοψη

Το Sliddev (slide + dev, [/sliddev/](#)) είναι ένας διαδικτυακός δημιουργός και παρουσιαστής διαφανειών. Είναι σχεδιασμένο για προγραμματιστές ώστε να εστιάζουν στη συγγραφή υλικού σε Markdown έχοντας παράλληλα τη δύναμη του HTML και των Vue components για να παρέχουν τέλεια layouts και σχεδιασμούς με ενσωματωμένες διαδραστικές επιδείξεις στις παρουσιάσεις τους.

Χρησιμοποιεί ένα πλούσιο σε δυνατότητες αρχείο markdown για τη δημιουργία όμορφων διαφανειών με μια άμεση εμπειρία επαναφόρτωσης, μαζί με πολλές ενσωματωμένες λειτουργίες, όπως live coding, εξαγωγή σε PDF, καταγραφή παρουσιάσεων κ.ο.κ. Εφόσον λειτουργεί με τη βοήθεια του διαδικτύου, μπορείτε να κάνετε οτιδήποτε με το Sliddev - οι δυνατότητες είναι απεριόριστες.

Μπορείτε να μάθετε περισσότερα για το σκεπτικό πίσω από το έργο στην ενότητα [Γιατί Sliddev](#).

## Δυνατότητες

- 📝 **Βασισμένο στο Markdown** - χρησιμοποιήστε τους αγαπημένους σας συντάκτες και ροή εργασίας
- 💻 **Φιλικό προς Προγραμματιστές** - ενσωματωμένοι επισημαντήρες σύνταξης, live coding, κτλ.
- 🎭 **Θεματοποιήσιμο** - θέματα μπορούν να μοιραστούν και να χρησιμοποιηθούν με πακέτα npx
- 🌈 **Stylish** - on-demand εργαλεία μέσω των [Windi CSS](#) ή [UnoCSS](#).
- 👾 **Διαδραστικό** - απρόσκοπτη ενσωμάτωση Vue components
- 🖨 **Λειτουργία Παρουσιαστή** - χρησιμοποιήστε ένα άλλο παράθυρο ή ακόμα και το τηλέφωνό σας για να ελέγξετε τις διαφάνειές σας
- 🖼 **Ζωγραφική** - ζωγραφίστε και σημειώστε πάνω στις διαφάνειές σας
- TeX **LaTeX** - ενσωματωμένη υποστήριξη μαθηματικών εξισώσεων LaTeX
- 💻 **Διαγράμματα** - δημιουργεί διαγράμματα με περιγραφές κειμένου
- 🌟 **Εικονίδια** - απευθείας πρόσβαση σε εικονίδια από οποιοδήποτε σύνολο εικονιδίων
- 💻 **Συντάκτες** - ενσωματωμένος συντάκτης, ή [επέκταση για το VS Code](#)
- 🎥 **Καταγραφή Παρουσίασης** - ενσωματωμένη καταγραφή παρουσίασης και προβολή κάμερας
- 👉 **Φορητό** - εξαγωγή σε PDF, PNGs, ή ακόμα και σε ένα φιλοξενήσιμο SPA
- ⚡ **Γρήγορο** - άμεση επαναφόρτωση με τη βοήθεια του [Vite](#)
- 🛠 **Hackable** - χρησιμοποιώντας πρόσθετα Vite, Vue components, ή οποιαδήποτε πακέτα npx

## Στοίβα Τεχνολογίας

Το Sliddev λειτουργεί με το συνδυασμό αυτών των εργαλείων και τεχνολογιών.

- [Vite](#) - Ένα εξαιρετικά γρήγορο frontend tooling
- [Vue 3 με Markdown](#) - Συγκεντρωθείτε στο υλικό έχοντας παράλληλα τη δύναμη του HTML και των Vue components όποτε χρειάζεται.
- [Windi CSS](#) ή [UnoCSS](#) - On-demand utility-first CSS framework, σχεδιάστε τις διαφάνειές σας με άνεση
- [Prism](#), [Shiki](#), [Monaco Editor](#) - Πρώτης κατηγορίας υποστήριξη αποσπασμάτων κώδικα με δυνατότητα live coding

- [RecordRTC](#) - Ενσωματωμένη καταγραφή παρουσίασης και προβολή κάμερας
- [VueUse](#) οικογένεια - `@vueuse/core`, `@vueuse/head`, `@vueuse/motion`, κτλ.
- [Iconify](#) - Συλλογή εικονιδίων.
- [Draau](#) - Υποστήριξη ζωγραφικής και σημειώσεων
- [KaTeX](#) - Μαθηματική απεικόνιση LaTeX.
- [Mermaid](#) - Κειμενικά Διαγράμματα.

## Δημιουργία σκελετού για την Πρώτη Σας Παρουσίαση

### Δοκιμάστε το Online

[sli.dev/new](https://sli.dev/new)



[Open in StackBlitz](#)

### Δημιουργήστε Τοπικά

Με NPM:

```
$ npm init slidev
```

Με Yarn:

```
$ yarn create slidev
```

Ακολουθήστε τις οδηγίες και αρχίστε να δημιουργείτε τις διαφάνειές σας τώρα! Για περισσότερες λεπτομέρειες σχετικά με τη σύνταξη markdown, διαβάστε τον [οδηγό σύνταξης](#).

## Command Line Interface

Σε ένα έργο όπου το Slidev είναι εγκατεστημένο, μπορείτε να χρησιμοποιήσετε το `slidev` binary στα npm scripts σας.

```
{
 "scripts": {
 "dev": "slidev", // εκκίνηση του διακομιστή dev
 "build": "slidev build", // κατασκευή για production SPA
 "export": "slidev export" // εξαγωγή διαφανειών σε pdf
 }
}
```

Διαφορετικά, μπορείτε να το χρησιμοποιήσετε με `npx`

```
$ npx slidev
```

Τρέξτε `slidev --help` για περισσότερες διαθέσιμες επιλογές.

## Σύνταξη Markdown

To Slidev διαβάζει το αρχείο `slides.md` στη ρίζα του έργου σας και τα μετατρέπει σε διαφάνειες. Κάθε φορά που κάνετε αλλαγές σε αυτό, το περιεχόμενο των διαφανειών θα ενημερώνεται αμέσως. Για παράδειγμα:

```
Slidev

Γεια σου κόσμε

Σελίδα 2

Άμεση χρήση code blocks για επισήμανση

//```ts
console.log('Γεια σου, κόσμε!')
//```

Σελίδα 3
```

Διαβάστε περισσότερα για τη σύνταξη Markdown του Slidev στο [οδηγός σύνταξης](#).

---

[Go to TOC](#)

# Εγκατάσταση

## Πρότυπο Εκκίνησης

To Slidev απαιτεί **Node.js >=14.0**

Ο καλύτερος τρόπος για να ξεκινήσετε είναι να χρησιμοποιήσετε το επίσημο πρότυπο εκκίνησης μας.

Με NPM:

```
$ npm init slidev@latest
```

Με Yarn:

```
$ yarn create slidev
```

Ακολουθήστε τις οδηγίες και θα ανοίξει αυτόματα η προβολή διαφανειών στη διεύθυνση <http://localhost:3030/>.

Περιέχει επίσης το βασικό setup και ένα σύντομο demo με οδηγίες για το πώς να ξεκινήσετε με το Slidev.

## Εγκαταστήστε Χειροκίνητα

Αν εξακολουθείτε να προτιμάτε να εγκαταστήσετε το Slidev χειροκίνητα ή θα θέλατε να το ενσωματώσετε στα υπάρχοντα έργα σας, μπορείτε να το κάνετε:

```
$ npm install @slidev/cli @slidev/theme-default
```

```
$ touch slides.md
```

```
$ npx slidev
```

Παρακαλείστε να σημειώσετε ότι εάν χρησιμοποιείτε **pnpm**, θα πρέπει να ενεργοποιήσετε την επιλογή **shamefully-hoist** για να λειτουργήσει σωστά το Slidev:

```
$ echo 'shamefully-hoist=true' >> .npmrc
```

## Εγκαταστήστε Globally

Διαθέσιμο από v0.14

Μπορείτε να εγκαταστήσετε το Slidev globally με την ακόλουθη εντολή

```
$ npm i -g @slidev/cli
```

Και στη συνέχεια χρησιμοποιήστε `slidev` παντού χωρίς να δημιουργείτε ένα έργο κάθε φορά.

```
$ slidev
```

Αυτή η εντολή θα προσπαθήσει επίσης να χρησιμοποιήσει το τοπικό `@slidev/cli` εάν έχει βρεθεί στο `node_modules`.

## Εγκατάσταση σε Docker

Αν χρειάζεστε έναν γρήγορο τρόπο για να εκτελέσετε μια παρουσίαση με containers, μπορείτε να χρησιμοποιήσετε την προκατασκευασμένη εικόνα `docker` που συντηρείται από τον `tangramor`, ή φτιάξτε το δικό σας.

Απλά εκτελέστε την ακόλουθη εντολή στο φάκελο εργασίας σας:

```
docker run --name slidev --rm -it \
--user node \
-v ${PWD}:/slidev \
-p 3030:3030 \
tangramor/slidev:latest
```

Εάν ο φάκελος εργασίας σας είναι άδειος, θα δημιουργήσει ένα πρότυπο `slides.md` και άλλα σχετικά αρχεία, και θα ξεκινήσει τον διακομιστή στη θύρα `3030`.

Μπορείτε να έχετε πρόσβαση στις διαφάνειές σας από το `http://localhost:3030/`

## Δημιουργία εικόνας που μπορεί να γίνει deploy

Ή μπορείτε να δημιουργήσετε το δικό σας slidev project σε μια εικόνα docker με ένα Dockerfile:

```
FROM tangramor/slidev:latest
ADD . /slidev
```

Δημιουργήσετε την εικόνα docker: `docker build -t myppt .`

Και εκτελέστε το container: `docker run --name myslides --rm --user node -p 3030:3030 myppt`

Μπορείτε να επισκεφθείτε τις διαφάνειές σας από το `http://localhost:3030/`

## Κατασκευή φιλοξενήσιμης εφαρμογής SPA (Single Page Application)

Εκτελέστε την εντολή `docker exec -i slidev npx slidev build` στο τρέχον container `slidev`. Θα δημιουργήσει στατικά αρχεία HTML στο φάκελο `dist`.

### Φιλοξενήστε στο Github Pages

Μπορείτε να φιλοξενήσετε το `dist` σε μια στατική ιστοσελίδα όπως το [Github Pages](#) ή το Gitlab Pages.

Επειδή στο Github pages το url μπορεί να περιέχει υποφακέλους, πρέπει να τροποποιήσετε το παραγόμενο `index.html` για να αλλάξετε το `href="/assets/xxx` σε `href=". /assets/xxx`. Ή μπορείτε να χρησιμοποιήσετε την επιλογή `--base=/<subfolder>/` κατά τη διάρκεια της διαδικασίας κατασκευής, όπως: `docker exec -i slidev npx slidev build --base=/slidev_docker/`.

Και για να αποφύγετε τη διαδικασία κατασκευής του Jekyll, πρέπει να προσθέσετε ένα κενό αρχείο `.nojekyll`.

### Φιλοξενήστε σε docker

Μπορείτε επίσης να το φιλοξενήσετε μόνοι σας με docker:

```
docker run --name myslides --rm -p 80:80 -v ${PWD}/dist:/usr/share/nginx/html
nginx:alpine
```

Ή δημιουργήστε μια στατική εικόνα με ένα Dockerfile:

```
FROM nginx:alpine
COPY dist /usr/share/nginx/html
```

Δημιουργήστε την εικόνα docker: `docker build -t mystaticppt .`

Και εκτελέστε το container: `docker run --name myslides --rm -p 80:80 mystaticppt`

Μπορείτε να επισκεφθείτε τις διαφάνειές σας από το `http://localhost/`

Ανατρέξτε στο [tangramor/slidev\\_docker](#) για περισσότερες πληροφορίες.

## Περιβάλλον Γραμμής Εντολών (CLI)

`@slidev/cli` Εκθέστε μερικές εντολές που μπορείτε να χρησιμοποιήσετε με το `npx slidev ...` ή προσθέτοντας scripts στο `package.json`:

```
{
 "script": {
 "dev": "slidev"
 }
}
```

Σε αυτή την περίπτωση θα μπορείτε να εκτελέσετε `npm run dev`.

Μπορείτε να περάσετε επιλογές σε οποιεσδήποτε εντολές:

- η επιλογή boolean είναι `true` αν είναι παρούσα, αλλιώς είναι `false` (για παράδειγμα: `slidev --open`)
- ορισμένες επιλογές μπορούν να έχουν τιμές που μπορείτε να προσθέσετε ακριβώς μετά την επιλογή ή χρησιμοποιώντας τον χαρακτήρα `=` (για παράδειγμα: `slidev --port 8080` ή `slidev --port=8080`)

Αν χρησιμοποιείτε npm scripts, μην ξεχάσετε να προσθέσετε `--` μετά την εντολή npm:

```
npm run slidev -- --open
```

## slidev [entry]

Ξεκινήστε έναν τοπικό διακομιστή για το Slidev.

- `[entry] (string, προεπιλογή: slides.md):` το path για την είσοδο διαφανειών markdown.

Επιλογές:

- `--port, -p (number, προεπιλογή: 3030):` αριθμός port.
- `--open, -o (boolean, προεπιλογή: false):` άνοιγμα στο πρόγραμμα περιήγησης.
- `--remote [password] (string):` ακούει τον public host και ενεργοποιεί τον απομακρυσμένο έλεγχο, εάν περαστεί μια τιμή, τότε η λειτουργία του παρουσιαστή είναι ιδιωτική και προσβάσιμη μόνο με τη διαβίβαση του συγκεκριμένου κωδικού πρόσβασης στην παράμετρο `password` του URL query.
- `--log ('error', 'warn', 'info', 'silent', προεπιλογή: 'warn'):` επίπεδο καταγραφής δεδομένων.
- `--force, -f (boolean, προεπιλογή false):` αναγκάζει τον βελτιστοποιητή να αγνοήσει την προσωρινή μνήμη και να κάνει re-bundle.
- `--theme, -t (string):` παρακάμπτει το θέμα.

## slidev build [entry]

Κατασκευή φιλοξενήσιμου SPA.

- `[entry] (string, προεπιλογή: slides.md):` το path για την είσοδο διαφανειών markdown.

Επιλογές:

- `--watch, -w (boolean, προεπιλογή: false):` παρακολούθηση κατασκευής.
- `--out, -o (string, προεπιλογή: dist):` φάκελος εξόδου.
- `--base (string, προεπιλογή: /):` base URL (δείτε <https://cli.vuejs.org/config/#publicpath>)
- `--download (boolean, προεπιλογή: false):` επιτρέπει τη λήψη των διαφανειών σε μορφή PDF μέσα στο SPA.
- `--theme, -t (string):` παρακάμπτει το θέμα.

## slidev export [entry]

Εξαγωγή διαφανειών σε PDF (ή άλλη μορφή).

- `[entry] (string, προεπιλογή: slides.md):` το path για την είσοδο διαφανειών markdown.

Επιλογές:

- `--output (string, προεπιλογή: χρησιμοποιήστε το exportFilename (δείτε https://el.sli.dev/custom/#%CF%81%CF%85%CE%B8%CE%BC%CE%AF%CF%83%CE%B5%CE%B9%CF%82-frontmatter) ή χρησιμοποιήστε [entry]-export):` το path για την εξαγωγή.
- `--base ('pdf', 'png', 'md', προεπιλογή: 'pdf'):` μορφή εξόδου.
- `--timeout (number, προεπιλογή: 30000):` χρονικό όριο για την απεικόνιση της σελίδας εκτύπωσης (δείτε <https://playwright.dev/docs/api/class-page#page-goto>).

- `--range` (`string`): περιοχές σελίδων για εξαγωγή (για παράδειγμα: `'1,4-5,6'`).
- `--dark` (`boolean`, προεπιλογή: `false`): εξαγωγή με σκοτεινό θέμα.
- `--with-clicks`, `-c` (`boolean`, προεπιλογή: `false`): εξαγωγή σελίδων για κάθε κλικ (δείτε <https://el.sli.dev/guide/animations.html#κινήσεις-click>).
- `--theme`, `-t` (`string`): παρακάμπτει το θέμα.

## slidev format [entry]

Διαμορφώστε το αρχείο markdown.

- `[entry]` (`string`, προεπιλογή: `slides.md`): το path για την είσοδο διαφανειών markdown.

## slidev theme [subcommand]

Λειτουργίες που σχετίζονται με το θέμα.

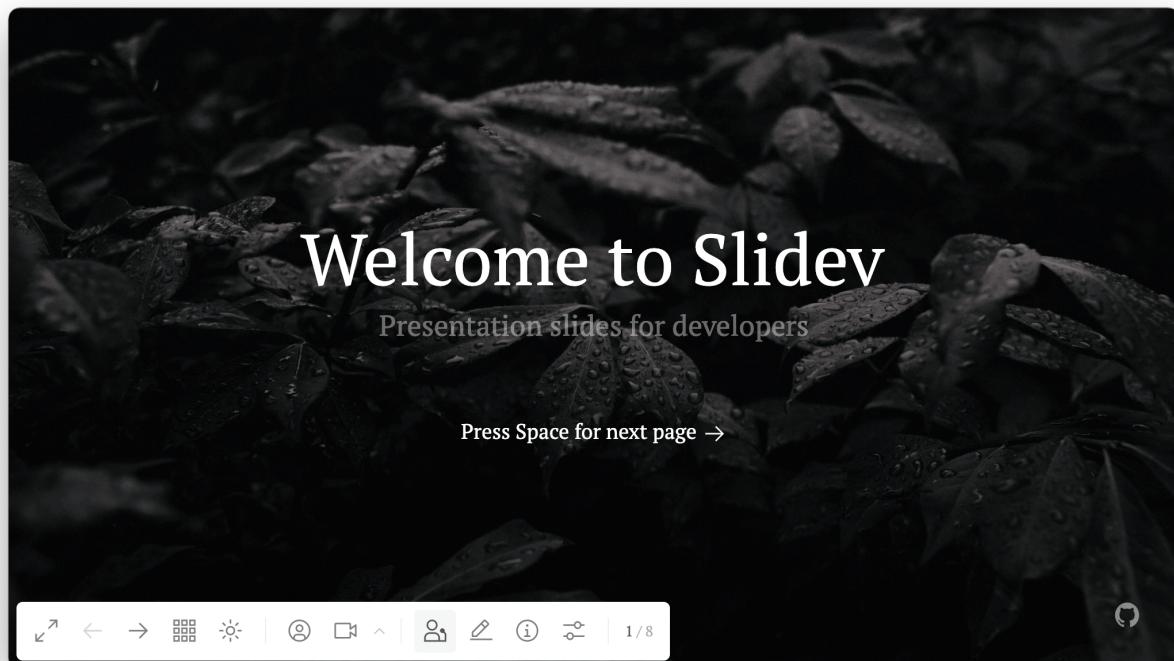
Υποεντολές:

- `eject` `[entry]`: Εξαγωγή του τρέχοντος θέματος στο τοπικό σύστημα αρχείων
  - `[entry]` (`string`, προεπιλογή: `slides.md`): το path για την είσοδο διαφανειών markdown.
  - Επιλογές: `--dir` (`string`, προεπιλογή: `theme`): φάκελος εξόδου. `--theme`, `-t` (`string`): παρακάμπτει το θέμα.

# Πλοήγηση

## Μπάρα Πλοήγησης

Μετακινήστε το ποντίκι σας στην αριστερή κάτω γωνία της σελίδας Slidev για να εμφανιστεί η μπάρα πλοήγησης.

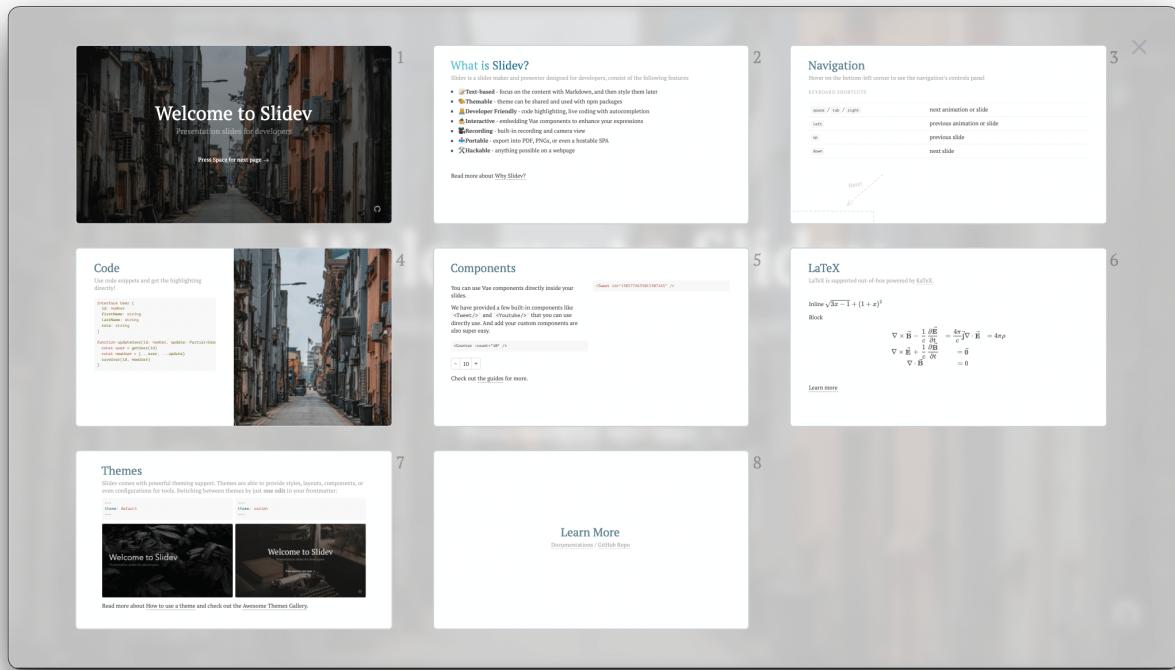


Συντομεύσεις Κουμπί	Περιγραφή
f	εναλλαγή πλήρους οθόνης
δεξιά / κενό	επόμενη κίνηση ή διαφάνεια
αριστερά	προηγούμενη κίνηση ή διαφάνεια
πάνω	- προηγούμενη διαφάνεια
κάτω	- επόμενη διαφάνεια
o	εναλλαγή εμφάνισης της <a href="#">επισκόπησης διαφανειών</a>
d	εναλλαγή σκοτεινής λειτουργίας
-	εναλλαγή <a href="#">προβολής κάμερας</a>
-	<a href="#">καταγραφή παρουσίασης</a>
-	<a href="#">ενεργοποίηση της λειτουργίας παρουσιαστή</a>

Συντομεύσεις Κουμπί	Περιγραφή
-	εναλλαγή εμφάνισης του <b>ενσωματωμένου συντάκτη</b>
-	λήψη διαφανειών (εμφανίζεται μόνο σε <b>κατασκευή SPA</b> )
-	εμφάνιση πληροφοριών σχετικά με τις διαφάνειες
-	εμφάνιση μενού ρυθμίσεων
g	- εμφάνιση goto...

## Επισκόπηση Διαφανειών

Πατώντας το πλήκτρο o ή κάνοντας click στο κουμπί στη μπάρα πλοήγησης, μπορείτε να έχετε την επισκόπηση των διαφανειών σας, ώστε να μπορείτε να μεταπηδάτε εύκολα μεταξύ τους.



[Go to TOC](#)

# Λειτουργία Παρουσιαστή

Κάντε click στο κουμπί στον πίνακα πλοήγησης, ή επισκεφθείτε το <http://localhost:3030/presenter> χειροκίνητα, για να ενεργοποιήσετε τη λειτουργία παρουσιαστή. Κάθε φορά που εισέρχεστε στη λειτουργία παρουσιαστή, οι άλλες σελίδες θα παραμένουν αυτόματα συγχρονισμένες με τον παρουσιαστή.

The screenshot shows a presentation slide titled "Welcome to Sliddev" with the subtitle "Presentation slides for developers". It includes a "Press Space for next page →" instruction and a navigation bar at the bottom. To the right of the slide is a sidebar titled "What is Sliddev?" which lists features like "Developer focused", "Developer friendly", "Code highlighting", "Live coding", "Autocompletion", "Keyboard navigation", "Recording", "Screenshot", and "Shareable". Below the sidebar is a note about slide notes and a link to the docs.

What is Sliddev?  
Sliddev is a slide deck generator that makes it easy to implement one or more of the following features:

- **Developer focused**: Focus on the content and mechanics, and their style doesn't limit you.
- **Developer friendly**: Code highlighting, live coding with autocompletion, keyboard navigation, and recording.
- **Code highlighting**: Code highlighting with syntax detection.
- **Autocompletion**: Autocompletion for code snippets and other user experiences.
- **Keyboard navigation**: Built-in recording and camera view.
- **Screenshot**: Export slide PPTX, PDF, or even a executable IPA.
- **Shareable**: Getting people on a conference

Read more about Why Sliddev?

The last comment block of each slide will be treated as slide notes. It will be visible and editable in Presenter Mode along with the slide. [Read more in the docs](#)

Navigation icons: back, forward, search, grid, lightbulb, etc. | 1 / 7

---

[Go to TOC](#)

# Καταγραφή Παρουσίασης

Το Slidew εχει ενσωματωμένη καταγραφή παρουσίασης και προβολή κάμερας. Μπορείτε να τα χρησιμοποιήσετε για να καταγράψετε την παρουσίασή σας εύκολα σε ένα μέρος.

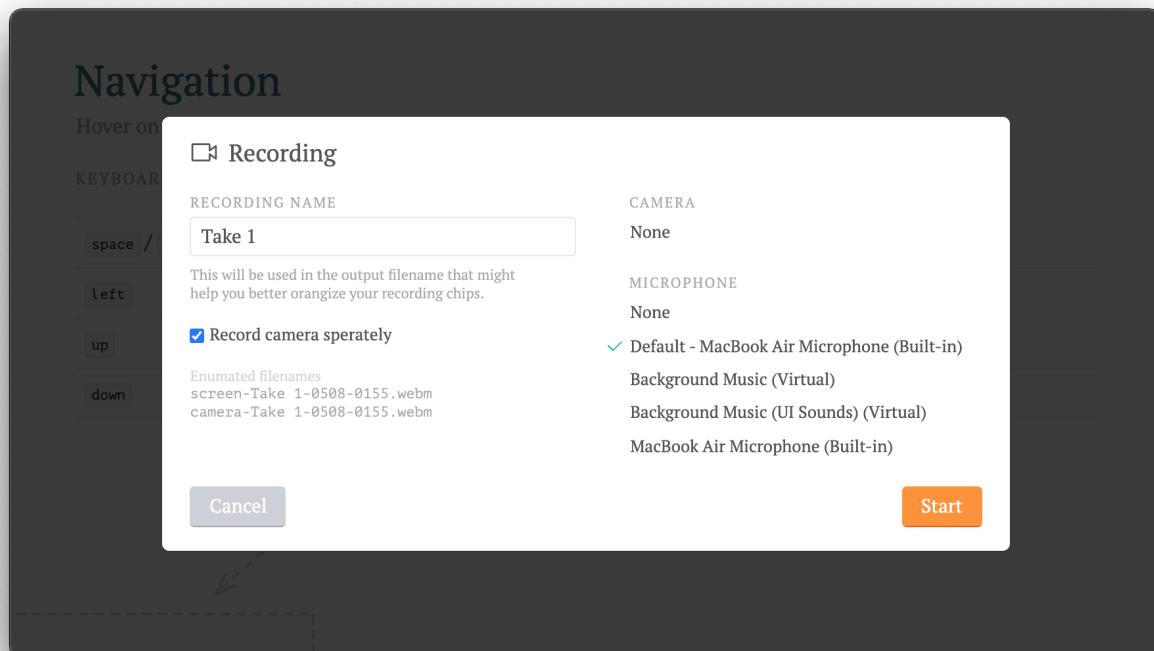
## Προβολή Κάμερας

Κάντε click στο κουμπί στον πίνακα πλοήγησης για να εμφανίσετε την προβολή της κάμεράς σας στην παρουσίαση. Μπορείτε να τη σύρετε για να τη μετακινήσετε και να χρησιμοποιήσετε το χειριστήριο στη δεξιά κάτω γωνία για να αλλάξετε το μέγεθός της. Το μέγεθος και η θέση θα παραμείνουν στο `localStorage` και επομένως θα είναι συνεπή ανάμεσα σε πολλαπλές ανανεώσεις, οπότε δεν χρειάζεται να ανησυχείτε γι' αυτό.

## Καταγραφή Παρουσίασης

Κάνοντας click στο κουμπί στον πίνακα πλοήγησης θα σας εμφανίσει ένα παράθυρο διαλόγου. Εδώ μπορείτε να επιλέξετε είτε να καταγράψετε την κάμερά σας ενσωματωμένη στις διαφάνειές σας είτε να τις διαχωρίσετε σε δύο αρχεία βίντεο.

Αυτή η δυνατότητα λειτουργεί με τη βοήθεια του [RecordRTC](#) και χρησιμοποιεί το [WebRTC API](#).



[Go to TOC](#)

# Σύνταξη Markdown

Οι διαφάνειες γράφονται μέσα σε **ένα ενιαίο αρχείο markdown** (εξ ορισμού `./slides.md`).

Μπορείτε να χρησιμοποιήσετε [τις δυνατότητες του Markdown](#) όπως θα κάνατε κανονικά, με την πρόσθετη υποστήριξη του ενσωματωμένου HTML και των Vue Components. Επίσης υποστηρίζεται styling χρησιμοποιώντας [Windi CSS](#). Χρησιμοποιήστε το `---` συνοδευμένο από μια νέα γραμμή για να διαχωρίσετε τις διαφάνειές σας.

```
Slides
```

Γεια σου, Κόσμε!

---

```
Σελίδα 2
```

Άμεση χρήση code blocks για επισήμανση

```
//```ts
console.log('Γεια σου, Κόσμε!')
//```
```

---

```
Σελίδα 3
```

Μπορείτε να χρησιμοποιήσετε απευθείας τα Windi CSS και Vue components για να διαμορφώσετε καλ να εμπλουτίσετε τις διαφάνειές σας.

```
<div class="p-3">
 <Tweet id="20" />
</div>
```

## Front Matter & Layouts

Καθορίστε layouts και άλλα μεταδεδομένα για κάθε διαφάνεια μετατρέποντας τα διαχωριστικά σε [front matter blocks](#). Κάθε frontmatter αρχίζει με μια τριπλή παύλα και τελειώνει με μια άλλη. Τα κείμενα μεταξύ τους είναι objects δεδομένων σε μορφή [YAML](#). Για παράδειγμα:

```

layout: cover

Slides

Αυτό είναι το εξώφυλλο.

layout: center
background: './images/background-1.png'
class: 'text-white'

```

## # Σελίδα 2

Αυτή είναι μια σελίδα με το layout `center` και μια εικόνα φόντου.

---

## # Σελίδα 3

Πρόκειται για μια προεπιλεγμένη σελίδα χωρίς πρόσθετα μεταδεδομένα.

Ανατρέξτε στις [προσαρμογές](#) για περισσότερες πληροφορίες.

## Code Blocks

Ένας μεγάλος λόγος που φτιάχνω το Slides είναι ότι πρέπει να κάνω τον κώδικά μου να φαίνεται σωστά στις διαφάνειες. Έτσι, όπως ακριβώς περιμένατε, μπορείτε να χρησιμοποιήσετε code block με μορφή Markdown για να επισημάνετε τον κώδικά σας.

```
//``ts
console.log('Γεια σου, Κόσμε!')
//``
```

Υποστηρίζουμε και [Prism](#) και [Shiki](#) ως επισημαντήρες σύνταξης. Ανατρέξτε στην ενότητα [επισημαντήρες](#) για περισσότερες πληροφορίες.

## Επισήμανση Γραμμής

Για να επισημάνετε συγκεκριμένες γραμμές, απλά προσθέστε τους αριθμούς γραμμών μέσα σε μία αγκύλη `{ }`. Οι αριθμοί γραμμών αρχίζουν να μετρούν από το 1.

```
//``ts {2,3}
function add(
 a: Ref<number> | number,
 b: Ref<number> | number
) {
 return computed(() => unref(a) + unref(b))
}
//``
```

Για να αλλάξετε την επισήμανση σε πολλαπλά βήματα, μπορείτε να χρησιμοποιήσετε `|` για να τα διαχωρίσετε. Για παράδειγμα

```
//``ts {2-3|5|all}
function add(
 a: Ref<number> | number,
 b: Ref<number> | number
) {
 return computed(() => unref(a) + unref(b))
}
//``
```

Αυτό θα επισημάνει πρώτα τα `a: Ref<number> | number` και `b: Ref<number> | number`, και μετά `return computed(() => unref(a) + unref(b))` μετά από ένα click, και τέλος, ολόκληρο το block. Μάθετε περισσότερα στον [οδηγό κινήσεων clicks](#).

Για να παραλείψετε την επισήμανση οποιασδήποτε γραμμής, μπορείτε να ορίσετε τον αριθμό γραμμής σε `0`.  
Για παράδειγμα

```
//``ts {0}
function add(
 a: Ref<number> | number,
 b: Ref<number> | number
) {
 return computed(() => unref(a) + unref(b))
}
//``
```

Αν ο κώδικας δεν χωράει σε μία διαφάνεια, μπορείτε να περάσετε μια επιπλέον επιλογή `maxHeight` που θα ορίσει σταθερό ύψος και θα ενεργοποιήσει το scrolling

```
//``ts {2|3|7|12} {maxHeight:'100'}
function add(
 a: Ref<number> | number,
 b: Ref<number> | number
) {
 return computed(() => unref(a) + unref(b))
}
/// ...as many lines as you want
const c = add(1, 2)
//``
```

## Monaco Editor

Κάθε φορά που θέλετε να κάνετε κάποια τροποποίηση στην παρουσίαση, απλά προσθέστε `{monaco}` μετά το αναγνωριστικό γλώσσας - αυτό μετατρέπει το block σε έναν πλήρως εξοπλισμένο Monaco editor!

```
//``ts {monaco}
console.log('ΓειαΣουΚόσμε')
//``
```

Μάθετε περισσότερα για την [ρύθμιση Monaco](#).

## Ενσωματωμένα Styles

Μπορείτε να χρησιμοποιήσετε την ετικέτα `<style>` στο Markdown σας απευθείας για να παρακάμψετε τα styles για την **τρέχουσα διαφάνεια**.

```
Αυτό είναι Κόκκινο

<style>
h1 {
 color: red
}
</style>

Η επόμενη διαφάνεια δεν επηρεάζεται
```

Η ετικέτα `<style>` στο Markdown είναι πάντα [περιορισμένη](#). Για να έχετε global style παρακάμψεις, δείτε την [ενότητα προσαρμογής](#)

Με την βοήθεια του [Windi CSS](#), μπορείτε να χρησιμοποιήσετε άμεσα εμφωλευμένα css και [օδηγίες](#) (π.χ. `@apply`)

```
Slides

> Γεια σου `κόσμε`

<style>
blockquote {
 code {
 @apply text-teal-500 dark:text-teal-400;
 }
}
</style>
```

## Στατικά Assets

Ακριβώς όπως θα κάνατε σε markdown, μπορείτε να χρησιμοποιήσετε εικόνες από μια απομακρυσμένη ή τοπική διεύθυνση URL.

Για τα απομακρυσμένα assets, το ενσωματωμένο `vite-plugin-remote-assets` θα τα αποθηκεύσει στη μνήμη cache στο δίσκο κατά την πρώτη εκτέλεση, ώστε να μπορείτε να έχετε άμεση φόρτωση ακόμα και για μεγάλες εικόνες αργότερα.

 ! [Remote Image]([https://sli.dev/favicon.png\)](https://sli.dev/favicon.png)

Για τοπικά assets, τοποθετήστε τα στο `φάκελο public` και αναφερθείτε σε αυτά με **slash από μπροστά**.

 ! [Local Image](pic.png)

Αν θέλετε να εφαρμόσετε προσαρμοσμένα μεγέθη ή styles, μπορείτε να τα μετατρέψετε σε ετικέτα `<img>`

## Σημειώσεις

Μπορείτε επίσης να κρατάτε σημειώσεις για κάθε διαφάνεια. Θα εμφανίζονται στη [Λειτουργία Παρουσιαστή](#) για να μπορείτε να ανατρέχετε σε αυτές κατά τη διάρκεια των παρουσιάσεων.

Σε Markdown, το τελευταίο block σχολίων σε κάθε διαφάνεια θα αντιμετωπίζεται ως σημείωση.

```

layout: cover

Σελίδα 1

Αυτό είναι το εξώφυλλο.

<!-- Αυτή είναι μια σημείωση --></pre>

```

```

Σελιδα 2
<!-- Αυτή ΔΕΝ είναι σημείωση, διότι προηγείται του περιεχομένου της διαφάνειας. -->
Αυτή είναι η δεύτερη σελιδα
<!--
Αυτή είναι μια άλλη σημείωση
-->
```

## Εικονίδια

To Slidev σας επιτρέπει να έχετε πρόσβαση σε σχεδόν όλα τα δημοφιλή σύνολα εικονιδίων ανοιχτού κώδικα **απευθείας** στο markdown σας. Με την βοήθεια του `unplugin-icons` και του [Iconify](#).

Η ονομασία ακολουθεί τη μετατροπή του [Iconify](#) `{collection-name}-{icon-name}`. Για παράδειγμα:

- `<mdi-account-circle />` - από τα [Material Design Icons](#)
- `<carbon-badge />` - από το [Carbon](#)
- `<uim-rocket />` - από τα [Unicons Monochrome](#)
- `<twemoji-cat-with-tears-of-joy />` - από τα [Twemoji](#)
- `<logos-vue />` - από τα [SVG Logos](#)
- Και πολλά άλλα...

Περιηγηθείτε και αναζητήστε όλα τα εικονίδια που είναι διαθέσιμα με το [Icônes](#).

## Μορφοποίηση Εικονιδίων

Μπορείτε να διαμορφώσετε τα εικονίδια όπως και τα άλλα στοιχεία HTML. Για παράδειγμα:

```
<uim-rocket />
<uim-rocket class="text-3xl text-red-400 mx-2" />
<uim-rocket class="text-3xl text-orange-400 animate-ping" />
```

## Slots

Διαθέσιμο από v0.18

Ορισμένα layouts μπορούν να παρέχουν πολλαπλά σημεία συμβολής χρησιμοποιώντας [τα named slots](#) του [Vue](#).

Για παράδειγμα, στο `two-cols` [layout](#), μπορείτε να έχετε δύο στήλες αριστερά (`default` slot) και δεξιά (`right` slot) διπλα-διπλα.

```

layout: two-cols

```

```
<template v-slot:default>
Αριστερά
Αυτό εμφανίζεται στα αριστερά
</template>
<template v-slot:right>
Δεξιά
Αυτό εμφανίζεται στα δεξιά
</template>
```

## Αριστερά

Αυτό εμφανίζεται στα αριστερά

## Δεξιά

Αυτό εμφανίζεται στα δεξιά

Παρέχουμε επίσης μια συντόμευση σύνταξης `::name::` για το όνομα του slot. Το ακόλουθο παράδειγμα λειτουργεί ακριβώς όπως και το προηγούμενο.

```

layout: two-cols

Αριστερά
Αυτό εμφανίζεται στα αριστερά
::right::

Δεξιά
Αυτό εμφανίζεται στα δεξιά
```

Μπορείτε επίσης να καθορίσετε ρητά το προεπιλεγμένο slot και να το παρέχετε σε προσαρμοσμένη σειρά

```

layout: two-cols

::right::

Δεξιά
Αυτό εμφανίζεται στα δεξιά
::default::

Αριστερά
Αυτό εμφανίζεται στα αριστερά
```

# Ρυθμίσεις

Όλες οι ρυθμίσεις που απαιτούνται μπορούν να οριστούν στο αρχείο Markdown. Για παράδειγμα:

```

theme: serif
layout: cover
background: 'https://source.unsplash.com/1600x900/?nature,water'

Slidelv

Αυτό είναι το εξώφυλλο.
```

Μάθετε περισσότερα για [τις ρυθμίσεις frontmatter](#)

# LaTeX

To Slidev έρχεται με ενσωματωμένη υποστήριξη για LaTeX, με τη βοήθεια του [KaTeX](#).

## Γραμμική

Περιβάλλετε το LaTeX σας με ένα απλό `$` σε κάθε πλευρά για γραμμική απεικόνιση.

```
$\sqrt{3x-1}+(1+x)^2$
```

## Block

Χρησιμοποιήστε δύο (`$$`) για απεικόνιση block. Αυτή η λειτουργία χρησιμοποιεί μεγαλύτερα σύμβολα και κεντράρει το αποτέλεσμα.

```
$$
\begin{array}{c}
\nabla \times \vec{\mathbf{B}} - \frac{1}{c} \nabla \frac{\partial \vec{\mathbf{E}}}{\partial t} + \frac{4\pi}{c} \vec{\mathbf{c}} \times \vec{\mathbf{E}} & = 4\pi \rho \\
\\
\nabla \times \vec{\mathbf{E}} + \frac{1}{c} \nabla \frac{\partial \vec{\mathbf{B}}}{\partial t} & = \vec{\mathbf{0}} \\
\\
\nabla \cdot \vec{\mathbf{B}} & = 0
\end{array}
$$
```

Μάθετε περισσότερα: [Demo](#) | [KaTeX](#) | [markdown-it-katex](#)

# Διαγράμματα

Μπορείτε επίσης να δημιουργήσετε διαγράμματα / γραφήματα από περιγραφές κειμένου στο Markdown σας, με τη βοήθεια του [Mermaid](#).

Ta code blocks που χαρακτηρίζονται ως `mermaid` θα μετατρέπονται σε διαγράμματα, για παράδειγμα:

```
//```mermaid
sequenceDiagram
 AlIkη->Γιάννης: Γεια σου Γιάννη, τι κάνεις;
 Σημείωση για την Αλίκη, Γιάννη: Μια τυπική επικοινωνία
//```

```

Μπορείτε επίσης να του περάσετε ένα object επιλογών για να καθορίσετε την κλιμάκωση και τη θεματοποίηση. Η σύνταξη του object είναι ένα JavaScript object literal, θα πρέπει να προσθέσετε εισαγωγικά ( ` ) για τα strings και να χρησιμοποιήσετε κόμμα ( , ) μεταξύ κλειδιών.

```
//```mermaid {theme: 'neutral', scale: 0.8}
graph TD
B[Κείμενο] --> C{Απόφαση}
C -->|Ένα| D[Αποτέλεσμα 1]
C -->|Δύο| E[Αποτέλεσμα 2]
//```

```

Μάθετε περισσότερα: [Demo](#) | [Mermaid](#)

## Πολλαπλές Είσοδοι

Διαθέσιμο από v0.15

Μπορείτε να χωρίσετε το `slides.md` σας σε πολλαπλά αρχεία και να τα οργανώσετε όπως θέλετε.

`slides.md` :

```
Σελίδα 1

Αυτή είναι μια κανονική σελίδα

src: ./subpage2.md

<!-- αυτή η σελίδα θα φορτωθεί από το './subpage2.md' -->
Το γραμμικό υλικό θα αγνοηθεί
```

`subpage2.md` :

```
Σελίδα 2

Αυτή η σελίδα είναι από ένα άλλο αρχείο
```

## Συγχώνευση Frontmatter

Μπορείτε να παρέχετε frontmatters τόσο από την κύρια είσοδό σας όσο και από εξωτερικές σελίδες markdown. Εάν υπάρχουν τα ίδια κλειδιά σε αυτές, αυτά από την **κύρια είσοδο έχουν μεγαλύτερη προτεραιότητα**. Για παράδειγμα

```
slides.md :
```

```

src: ./cover.md
background: https://sli.dev/bar.png
class: text-center

```

```
cover.md :
```

```

layout: cover
background: https://sli.dev/foo.png

```

# Εξώφυλλο

Σελίδα Εξωφύλλου

Θα καταλήξουν να είναι ισοδύναμα της επόμενης σελίδας:

```

layout: cover
background: https://sli.dev/bar.png
class: text-center

Εξώφυλλο

Σελίδα Εξωφύλλου
```

## Επαναχρησιμοποίηση Σελίδων

Με την υποστήριξη πολλαπλών εισόδων, η επαναχρησιμοποίηση σελίδων είναι απλή. Για παράδειγμα:

```

src: ./cover.md

src: ./intro.md

src: ./content.md

επαναχρησιμοποίηση
src: ./content.md

```

---

[Go to TOC](#)

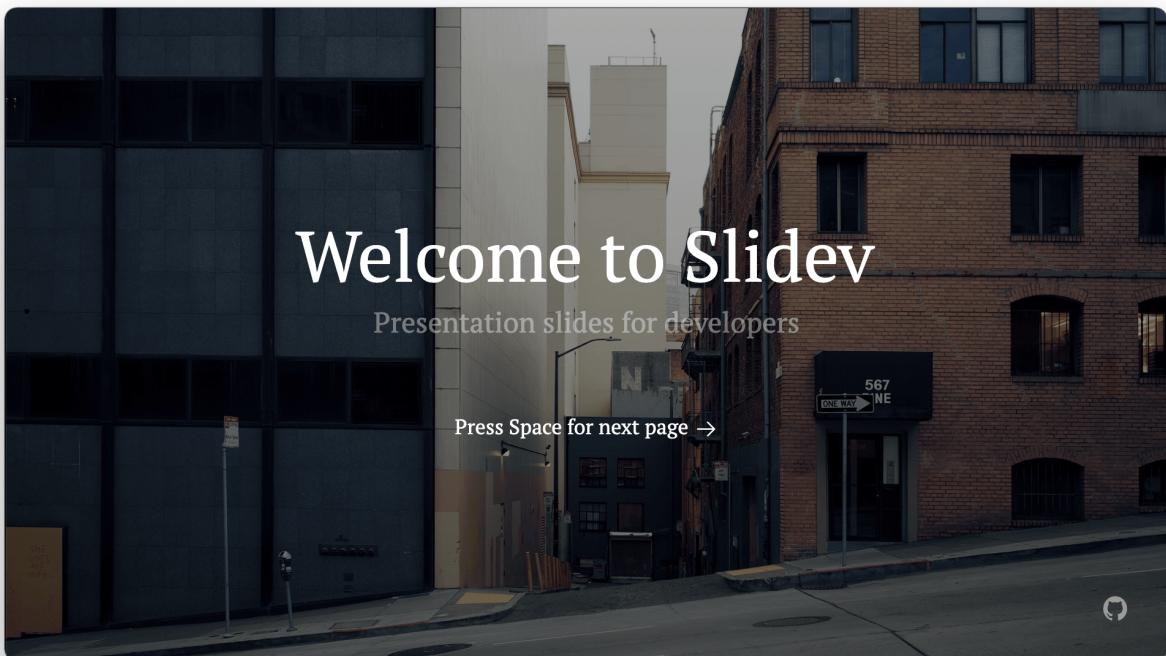
# Γιατί Slidev

Υπάρχουν πολλοί πλούσιοι σε δυνατότητες, γενικής χρήσης, κατασκευαστές διαφανειών WYSIWYG όπως το [Microsoft PowerPoint](#) και το [Apple Keynote](#). Λειτουργούν πολύ καλά στη δημιουργία ωραίων διαφανειών με κινήσεις, διαγράμματα και πολλά άλλα πράγματα, ενώ είναι πολύ διαισθητικοί και εύκολοι στην εκμάθηση. Οπότε γιατί να μπούμε στον κόπο να δημιουργήσουμε το Slidev?

Το Slidev έχει ως στόχο να παρέχει την ευελιξία και τη διαδραστικότητα στους προγραμματιστές ώστε να κάνουν τις παρουσιάσεις τους ακόμη πιο ενδιαφέρουσες, εκφραστικές και ελκυστικές, χρησιμοποιώντας τα εργαλεία και τις τεχνολογίες με τις οποίες είναι ήδη εξοικειωμένοι.

Όταν εργάζεστε με επεξεργαστές WYSIWYG, είναι εύκολο να σας αποσπάσουν την προσοχή οι επιλογές μορφοποίησης. Το Slidev διορθώνει αυτό με το διαχωρισμό του περιεχομένου και των γραφικών. Αυτό σας επιτρέπει να επικεντρωθείτε σε ένα πράγμα τη φορά, ενώ παράλληλα μπορείτε να επαναχρησιμοποιήσετε τα θέματα της κοινότητας. Το Slidev δεν επιδιώκει να αντικαταστήσει εξ ολοκλήρου άλλους κατασκευαστές διαφανειών. Αντίθετα, επικεντρώνεται στην εξυπηρέτηση της κοινότητας των προγραμματιστών.

## Slidev



Ακολουθούν μερικά από τα πιο cool χαρακτηριστικά του Slidev:

## Markdown-based

Το Slidev χρησιμοποιεί μια εκτεταμένη μορφή Markdown για να αποθηκεύει και να οργανώνει τις διαφάνειές σας σε ένα ενιαίο αρχείο απλού κειμένου. Αυτό σας επιτρέπει να επικεντρωθείτε στη δημιουργία του υλικού. Και δεδομένου ότι το υλικό και τα styles είναι διαχωρισμένα, αυτό καθιστά επίσης δυνατή την αβίαστη εναλλαγή μεταξύ διαφορετικών θεμάτων.

Μάθετε περισσότερα για την [Σύνταξη Markdown του Slidev](#).

## Θεματοποιήσιμο

Τα θέματα για το Slidev μπορούν να μοιραστούν και να εγκατασταθούν χρησιμοποιώντας πακέτα πρ. Στη συνέχεια τα εφαρμόζετε με μία μόνο γραμμή ρύθμισης.

Δείτε την [συλλογή θεμάτων](#) ή [μάθετε πώς να δημιουργήσετε ένα θέμα](#).

## Φιλικό προς Προγραμματιστές

Το Slidev παρέχει πρώτης τάξεως υποστήριξη για αποσπάσματα κώδικα για προγραμματιστές. Υποστηρίζει και [Prism](#) και [Shiki](#) για να έχετε τέλεια επισήμανση σύνταξης, ενώ παράλληλα να μπορείτε να τροποποιείτε τον κώδικα ανά πάσα στιγμή. Με ενσωματωμένο το [Monaco editor](#), σας δίνει επίσης τη δυνατότητα να κάνετε live coding / επίδειξη στην παρουσίασή σας με αυτόματη συμπλήρωση, type hovering ακόμη και υποστήριξη ελέγχου τύπων TypeScript.

Μάθετε περισσότερα για τους [επισημαντήρες](#) και για τη [ρύθμιση του Monaco](#).

## Γρήγορο

Το Slidev λειτουργεί με τη βοήθεια των [Vite](#), [Vue 3](#) και [Windi CSS](#), που σας δίνουν την πιο υπέροχη εμπειρία συγγραφής. Κάθε αλλαγή που κάνατε θα αντικατοπτρίζεται στις διαφάνειές σας **αμέσως**.

Μάθετε περισσότερα για την [στοιβα τεχνολογίας μας](#)

## Διαδραστικό & Εκφραστικό

Μπορείτε να γράψετε προσαρμοσμένα Vue components και να τα χρησιμοποιήσετε απευθείας μέσα στο αρχείο markdown σας. Μπορείτε επίσης να αλληλεπιδράσετε με αυτά μέσα στην παρουσίαση για να εκφράσετε την ιδέα σας με έναν πιο ενδιαφέρων και διαισθητικό τρόπο.

## Υποστήριξη Καταγραφής Παρουσίασης

Το Slidev έχει ενσωματωμένη καταγραφή παρουσίασης και προβολή κάμερας. Μπορείτε να μοιραστείτε την παρουσίασή σας με την προβολή της κάμεράς σας μέσα ή να τις καταγράψετε και να τις αποθηκεύσετε ξεχωριστά για την οθόνη και την κάμερά σας. Όλα με μία κίνηση, χωρίς να απαιτούνται πρόσθετα εργαλεία.

Μάθετε περισσότερα για την [καταγραφή παρουσίασης εδώ](#).

## Φορητό

Εξαγωγή σε PDF, PNGs, ή ακόμα και σε ένα φιλοξενήσιμο Single-page Application (SPA) με μία μόνο εντολή, και μοιραστείτε τα οπουδήποτε.

Διαβάστε περισσότερα γι' αυτό στις [οδηγίες εξαγωγής](#).

## Hackable

Όντας βασισμένο σε τεχνολογίες ιστού, οτιδήποτε μπορεί να γίνει σε μια εφαρμογή ιστού είναι επίσης δυνατό με το Slidev. Για παράδειγμα, WebGL, αιτήσεις API, iframes ή ακόμη και live sharing. Εξαρτάται από τη φαντασία σας!

## Δοκιμάστε το

Παίζοντας με το Slidev θα σας πει περισσότερα από χίλιες λέξεις. Είστε μόνο μια εντολή μακριά:

```
$ npm init slidev
```

Ή για να έχετε μια γρήγορη προεπισκόπηση:

Slidev First Preview Demo

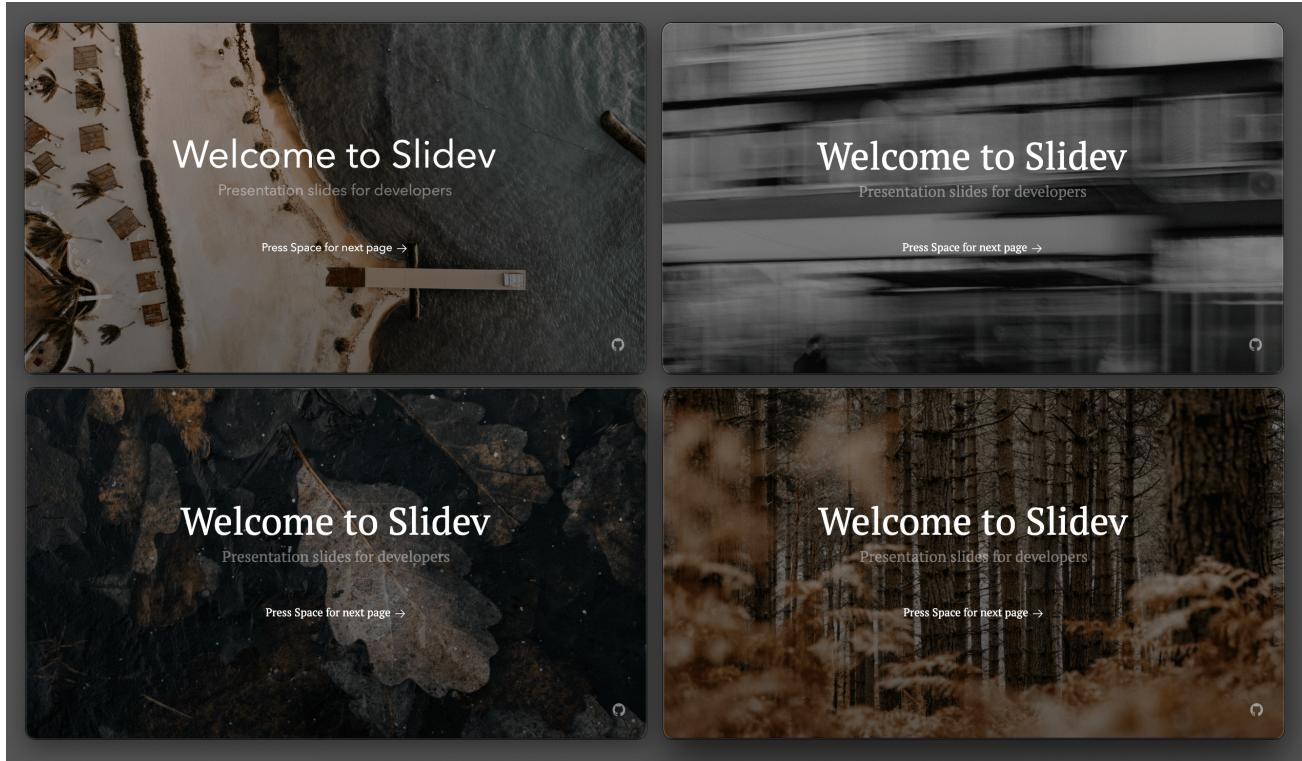


---

[Go to TOC](#)

# Επιμελημένα Εξώφυλλα

Επιμεληθήκαμε μερικά εξώφυλλα για να επιδείξουμε το πρότυπο εκκίνησης μας.



---  
# τυχαία εικόνα από την επιμελημένη συλλογή  
**background:** <https://source.unsplash.com/collection/94734566/1920x1080>  
---

Αν σας αρέσει κάποιο από αυτά, δείτε την [συλλογή Unsplash](#) μας και μάθετε τους συγγραφείς τους.

---

[Go to TOC](#)

# Μαθησιακοί Πόροι

## English

### Videos

Slidev - one of the best presentation software and it is free!



### Articles

- [Tips To Turn R Markdown Into Slidev Presentation](#) by Hiroaki Yutani

## 中文

- [Slidev：一个用Markdown写slides的神器](#) by 梦里风林
- [神器！这款开源项目可以让你使用 Markdown 来做 PPT！](#) by [Github掘金计划](#)
- [【用 markdown 写 Slide!】神器 Slidev 的安装及 bug 解决](#) by [HaloHoohoo](#)

## 日本語

- [開発者のためのスライド作成ツール Slidev がすごい](#) by [ryo\\_kawamata](#)
- [Markdownでオシャレなスライドを作るSli.dev](#) by [Nobuko YAMADA](#)

---

[Go to TOC](#)

# Παρουσιάσεις

Ομιλίες / Παρουσιάσεις που χρησιμοποιούν Slidev.

---

[Go to TOC](#)

# Συλλογή Θεμάτων

Περιηγηθείτε σε φοβερά θέματα διαθέσιμα για το Slidev εδώ.

Διαβάστε περισσότερα για το [πώς να χρησιμοποίησετε ένα Θέμα](#), ή για το [πώς να δημιουργήσετε το δικό σας](#) και μοιραστείτε το με την κοινότητα!

## Επίσημα Θέματα

### Θέματα της Κοινότητας

Εδώ είναι τα επιμελημένα θέματα από την κοινότητα.

### Περισσότερα Θέματα

Βρείτε όλα τα [διαθέσιμα θέματα στο NPM](#).

# Χρησιμοποίηστε Θέμα

Η αλλαγή του θέματος στο Slidev είναι εκπληκτικά εύκολη. Το μόνο που χρειάζεται να κάνετε είναι να προσθέστε το πεδίο `theme:` στο frontmatter σας.

```

theme: serif

```

Μπορείτε να ξεκινήστε τον διακομιστή, ο οποίος αυτόματα θα σας ζητήσει να εγκαταστήσετε το θέμα

? The theme "@slidev/theme-serif" was not found in your project, do you want to install it now? › (Y/n)

ή εγκαταστήστε το θέμα χειροκίνητα μέσω

```
$ npm install @slidev/theme-serif
```

Και αυτό είναι όλο, απολαύστε το νέο θέμα! Για περισσότερες λεπτομέρειες σχετικά με τη χρήση, μπορείτε να ανατρέξετε στο README του θέματος.

Θέλετε να μοιραστείτε το θέμα σας? Μάθετε για το [πώς να δημιουργήσετε ένα θέμα](#).

## Εξαγωγή Θέματος

Αν θέλετε να αποκτήσετε τον πλήρη έλεγχο του τρέχοντος θέματος, μπορείτε να **εξάγετε** στο τοπικό σας σύστημα αρχείων και να το τροποποιήσετε όπως θέλετε. Εκτελώντας την ακόλουθη εντολή

```
$ slidev theme eject
```

Θα εξαγάγει το θέμα που χρησιμοποιείτε αυτή τη στιγμή στο `./theme`, και θα αλλάξει το frontmatter σε

```

theme: ./theme

```

Αυτό θα μπορούσε επίσης να είναι χρήσιμο αν θέλετε να κάνετε ένα θέμα βασισμένο σε ένα υπάρχον. Εάν το κάνετε, θυμηθείτε να αναφέρετε το αρχικό θέμα και τον συγγραφέα :)

## Τοπικό Θέμα

'Οπως πιθανώς ανακαλύψατε από την προηγούμενη ενότητα, μπορείτε να έχετε ένα τοπικό θέμα για το έργο σας. Έχοντας ένα **σχετικό path** στο πεδίο του θέματος σας.

```

theme: ./path/για/το/θέμα

```

Ανατρέξτε στο [πως να δημιουργήσετε ένα θέμα](#) για περισσότερες πληροφορίες.

---

[Go to TOC](#)

# Δημιουργήστε ένα Θέμα

Για να ξεκινήσετε, σας προτείνουμε να χρησιμοποιήσετε τη γεννήτριά μας για να δημιουργήσετε τον σκελετό του πρώτου σας θέματος

```
$ npm init slidev-theme
```

Στη συνέχεια, μπορείτε να το τροποποιήσετε και να παίξετε μαζί του. Μπορείτε επίσης να ανατρέξετε στα [επι-σημα θέματα](#) για παραδείγματα.

## Δυνατότητες

Ένα θέμα μπορεί να συμβάλλει στα ακόλουθα σημεία:

- Global styles
- Παροχή προεπιλεγμένων ρυθμίσεων (γραμματοσειρές, χρωματικό σχήμα, επισημαντήρες, κτλ.)
- Παροχή προσαρμοσμένων layouts ή παράκαμψη των υπαρχόντων
- Παροχή προσαρμοσμένων components ή παράκαμψη των υπαρχόντων
- Επέκταση ρυθμίσεων του Windi CSS
- Ρύθμιση εργαλείων όπως Monaco και Prism

## Συμβάσεις

Τα θέματα δημοσιεύονται στο μητρώο npm, και θα πρέπει να ακολουθούν τις παρακάτω συμβάσεις:

- Το όνομα του πακέτου πρέπει να αρχίζει με `slidev-theme-`, για παράδειγμα: `slidev-theme-awesome`
- Προσθέστε `slidev-theme` και `slidev` στο πεδίο `keywords` του `package.json` σας

## Στήσιμο

Για να στήσετε το πεδίο δοκιμών για το θέμα σας, μπορείτε να δημιουργήσετε `example.md` με το ακόλουθο frontmatter, για να πείτε στο Slidev ότι χρησιμοποιείτε τον τρέχοντα φάκελο ως θέμα.

```

theme: ./

```

Προαιρετικά, μπορείτε επίσης να προσθέσετε κάποια scripts στο `packages.json` σας

```
// package.json
{
 "scripts": {
 "dev": "slidev example.md",
 "build": "slidev build example.md",
 "export": "slidev export example.md",
 "screenshot": "slidev export example.md --format png"
 }
}
```

Για να δημοσιεύσετε το θέμα σας, απλά εκτελέστε `npm publish` και είστε έτοιμοι. Δεν απαιτείται διαδικασία κατασκευής (που σημαίνει ότι μπορείτε να δημοσιεύσετε απευθείας αρχεία `.vue` και `.ts`, το Slidev είναι αρκετά έξυπνο για να τα καταλάβει).

Τα σημεία συνεισφοράς του θέματος ακολουθούν τις ίδιες συμβάσεις με την τοπική ρυθμίση, παρακαλούμε ανατρέξτε [στις οδηγίες για τις συμβάσεις ονομασίας](#).

## Προεπιλεγμένες Ρυθμίσεις

Διαθέσιμο από v0.19

Ένα θέμα μπορεί να παρέχει προεπιλεγμένες ρυθμίσεις μέσω του `package.json`.

```
// package.json
{
 "slidev": {
 "default": {
 "aspectRatio": "16/9",
 "canvasWidth": 980,
 "fonts": {
 "sans": "Robot",
 "mono": "Fira Code"
 }
 }
 }
}
```

Οι γραμματοσειρές θα εισαχθούν αυτόματα από το [Google Fonts](#).

Μάθετε περισσότερα για τις [γραμματοσειρές](#) και τις [ρυθμίσεις frontmatter](#)

## Μεταδεδομένα Θέματος

### Χρωματικό Σχήμα

Εξ ορισμού, το Slidev υποθέτει ότι τα θέματα υποστηρίζουν τόσο τη φωτεινή όσο και τη σκοτεινή λειτουργία. Αν θέλετε το θέμα σας να παρουσιάζεται μόνο σε ένα σχεδιασμένο χρωματικό σχήμα, θα πρέπει να το καθορίσετε ρητά στο `package.json`

```
// package.json
{
 "name": "slidev-theme-my-cool-theme",
 "keywords": [
 "slidev-theme",
 "slidev"
],
 "slidev": {
 "colorSchema": "light" // ή "dark" ή "both"
 }
}
```

Για να αποκτήσετε πρόσβαση στη σκοτεινή λειτουργία κατά τη δημιουργία των στυλ του θέματός σας, μπορείτε να τυλίξετε το CSS ειδικά για τη σκοτεινή λειτουργία μέσα σε ένα `dark` class:

```
/* γενικό CSS εδώ */

html:not(.dark) {
 /* φωτεινή λειτουργία CSS εδώ */
}

html.dark {
 /* σκοτεινή λειτουργία CSS εδώ */
}
```

To Slidev εναλλάσσει ένα `dark` class στο στοιχείο `html` της σελίδας για εναλλαγή χρωματικού σχήματος.

## Επισημαντήρας

Τα χρώματα επισήμανσης σύνταξης παρέχονται επίσης στο θέμα. Υποστηρίζουμε και [Prism](#) και [Shiki](#). Για περισσότερες πληροφορίες ανατρέξτε [στις οδηγίες επισήμανσης σύνταξης](#).

Μπορείτε να υποστηρίξετε ένα από τα δύο ή και τα δύο. Ανατρέξτε στο προεπιλεγμένο θέμα για παραδείγματα ρυθμίσεων `./styles/code.css` / `./setup/shiki.ts`.

Επίσης, θυμηθείτε να καθορίσετε τους υποστηριζόμενους επισημαντήρες στο `package.json` σας

```
// package.json
{
 "slidev": {
 "highlighter": "shiki" // ή "prism" ή "all"
 }
}
```

## Έκδοση Slidev

Εάν το θέμα βασίζεται σε ένα συγκεκριμένο χαρακτηριστικό του Slidev που εισάχθηκε πρόσφατα, μπορείτε να ορίσετε την ελάχιστη έκδοση του Slidev που απαιτείται για να λειτουργεί σωστά το θέμα σας:

```
// package.json
{
 "engines": {
 "slidev": ">=0.19.3"
 }
}
```

Εάν οι χρήστες χρησιμοποιούν παλαιότερες εκδόσεις του Slidev, θα εμφανιστεί ένα σφάλμα.

---

[Go to TOC](#)

# Colophon

This book is created by using the following sources:

- Slidev - Ελληνικά
- GitHub source: [slidevjs/docs-el](https://github.com/slidesjs/docs-el)
- Created: 2022-11-27
- Bash v5.2.2
- Vivliostyle, <https://vivliostyle.org/>
- By: @shinokada
- GitHub repo: <https://github.com/shinokada/markdown-docs-as-pdf>