# {CSSS}

## CSS-based SlideShow System

By Lea Verou

# Introduction

# What is it?

**A simple framework for building presentations with modern web standards**

# What is it?

**A simple framework for building presentations with modern web standards**

- ✔ **An HTML file contains the whole presentation**

# What is it?

**A simple framework for building presentations with modern web standards**

- ✔ An HTML file contains the whole presentation
- ✔ Themes as CSS files

# What is it?

**A simple framework for building presentations with modern web standards**

- ✔ An HTML file contains the whole presentation
- ✔ Themes as CSS files
- ✔ JavaScript handles what CSS can't (keyboard shortcuts etc)

# History

# History

- ✔ I had to create a presentation for my talk at Front Trends 2010

# History

- ✔ I had to create a presentation for my talk at Front Trends 2010
- ✔ **The only web-based presentation systems I knew of didn't fit my needs**

# History

- ✔ I had to create a presentation for my talk at Front Trends 2010
- ✔ The only web-based presentation systems I knew of didn't fit my needs
- ✔ …so I rolled up my own

# History

- ✔ I had to create a presentation for my talk at Front Trends 2010
- ✔ The only web-based presentation systems I knew of didn't fit my needs
- ✔ …so I rolled up my own
- ✔ **FT2010 attendees asked me to release it**

# History

- ✔ I had to create a presentation for my talk at Front Trends 2010
- ✔ The only web-based presentation systems I knew of didn't fit my needs
- ✔ …so I rolled up my own
- ✔ FT2010 attendees asked me to release it
- ✔ and here it is! ;-)

# Navigation

Only through keyboard for now :(

- ✔ → or ↓ to advance to the next slide or incrementally displayed item
- ✔ ← or ↑ to go to the previous slide or incrementally displayed item
- ✔ Ctrl* + → or Ctrl* + ↓ to jump to the next slide
- ✔ Ctrl* + ← or Ctrl* + ↑ to jump to the previous slide
- ✔ Home to go to the first slide, End to go to the last
- ✔ Ctrl* + G to jump to an arbitrary slide (by slide number or identifier)

* Ctrl or Shift actually. Only Shift works in Opera.

# Features

# Thumbnail view

**Press Ctrl+H (or Shift+H if you're in Opera) now.**

# Thumbnail view

**Press Ctrl+H (or Shift+H if you're in Opera) now.**

**Cool, huh? You can press Ctrl+Shift+H to see all slides (warning: can be slow!)**

# Timer

- ✔ Timer in the form of a progress bar on the top
- ✔ Define the talk duration with the attribute `data-duration` on the `body`
- ✔ Style the timer and the end state with the selectors `#timer` and `#timer.end` respectively.

# Presenter view

**This slide has presenter notes. They are only visible in presenter view (Ctrl+P or Shift+P).**

# <style scoped />

You can use the HTML5 `scoped` attribute on style elements in slides, to only style the current slide or inside sections that include multiple slides to style all of them and nothing else.

For example, inspect this slide and then run `$$('style[scoped]')[0].sheet.cssRules[0].selectorText` in the console to see how the original selector gets changed.

# More features

# More features

- ✔ IDs are dynamically assigned by JavaScript…

# More features

- ✔ IDs are dynamically assigned by JavaScript…
- ✔ **…but you can also assign your own, bringing the best of both worlds**

# More features

- ✔ IDs are dynamically assigned by JavaScript…
- ✔ …but you can also assign your own, bringing the best of both worlds
- ✔ **Incremental display of slide contents (just add `class="delayed"`)**

# More features

- ✔ IDs are dynamically assigned by JavaScript…
- ✔ …but you can also assign your own, bringing the best of both worlds
- ✔ Incremental display of slide contents (just add `class="delayed"`)
- ✔ **Slide scaling based on window size (size everything you want to be scalable in ems)**

# More features

✔ IDs are dynamically assigned by JavaScript…

✔ …but you can also assign your own, bringing the best of both worlds

✔ Incremental display of slide contents (just add `class="delayed"`)

✔ Slide scaling based on window size (size everything you want to be scalable in ems)

✔ **Document.title changing according to slide title (fetched either from the `title` attribute or the slide's heading)**

# Drawbacks

- ✔ Only supports modern browsers. Why?
  - ➜ More lightweight
  - ➜ Easier to understand code
  - ➜ It's a presentation, so the environment is controlled anyway
- ✔ No mouse click to advance to the next slide. I consider it annoying.
- ✔ Doesn't work in mobile :(

# Plugins

# Plugin: CSS Snippets

```
new CSSSnippet(document.getElementById('snippet'));
```

- ✔ Dynamically edit CSS and reflect the results in any element's style
  - ➜ Which one? Specify a selector with the `data-subject` attribute or leave blank for the slide to be the subject
- ✔ Automatically prefixes CSS3 properties/values, when needed
  - ➜ Also does it with the style attribute (statically) if the element is not a text field (doesn't work in IE and Firefox 3.6)
- ✔ Textarea automatically adjusts rows/font-size on line break (use class `dont-adjust` to turn off)

# CSS Snippets demo

**Edit the following CSS code:**

```
background-color:red;
background-image: linear-gradient(red, #600)
```

# CSS Snippets: raw CSS

```
@keyframes rainbow {
        from, to { background: red }
        16% { background: yellow }
        33% { background: green }
        50% { background: aqua }
        66% { background: blue }
        83% { background: fuchsia }
}

#animation-demo { animation: 30s rainbow infinite; }
```

# Plugin: CSS Controls

- ✔ Use sliders and other controls to adjust CSS values
  - ➜ Which one? Specify a selector with the `data-subject` attribute or leave blank for the slide to be the subject
  - ➜ Specify style with `data-style` (use `{value}` where the result will be)
- ✔ Automatically prefixes CSS3 properties/values, when needed

# CSS Controls demo

Slide lightness:

border-width: 30

font-family: sans-serif

# Plugin: Code highlight

✔ Syntax highlighting for every `<code>` element with a `lang` attribute.

✔ Only highlights when the slide becomes active, for performance

✔ Currently only supports JavaScript, but it's easy to add any language

✔ Styling done through easy to understand classes like `.string` or `.comment`

# Code highlight example

```
// Comment
document.addEventListener('DOMAttrModified',function(e) {
  var node = e.target, attr = e.attrName;
  if(/^input$/i.test(node.nodeName)
    && (attr === 'placeholder' || attr === 'value')
    || foo > 5)
      Placeholder.update(node);
}, false);
```

# Thank you!

Get it or contribute at http://github.com/LeaVerou/CSSS

Credits:

✔ Colored wood background by Luke Roberts

✔ The name CSSS is actually inspired by S5, as you could probably guess ;-)

✔ A big thanks to the community at Github for various contributions to this project