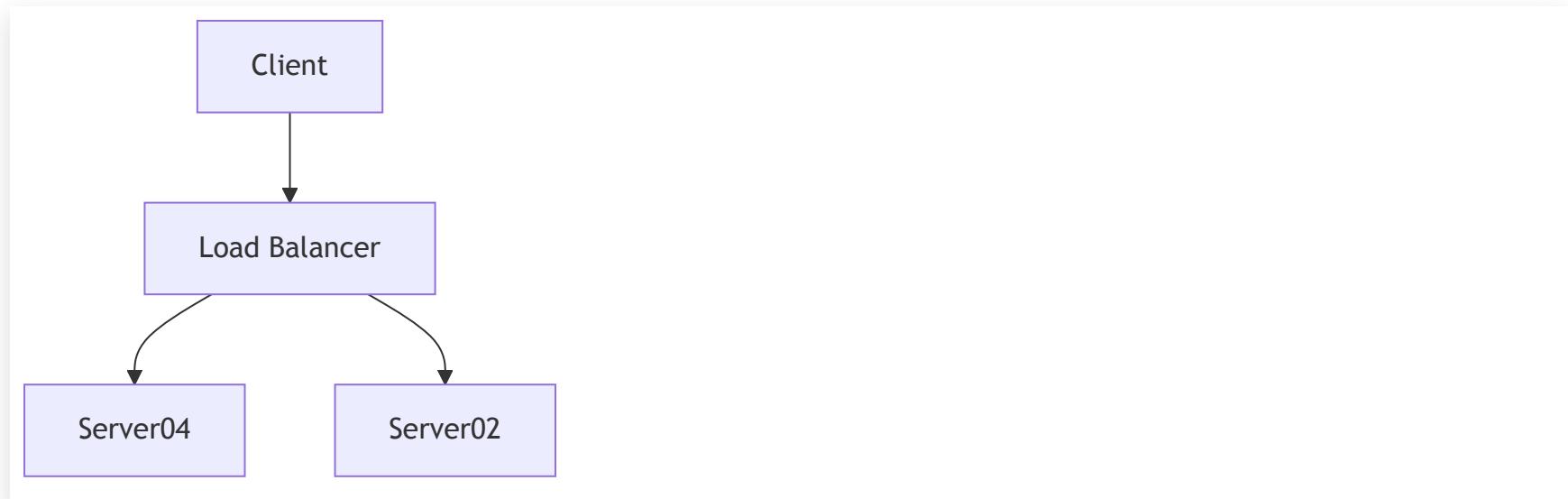


# THE HTML PRESENTATION FRAMEWORK

Created by [Hakim El Hattab](#) and [contributors](#)

# HELLO THERE

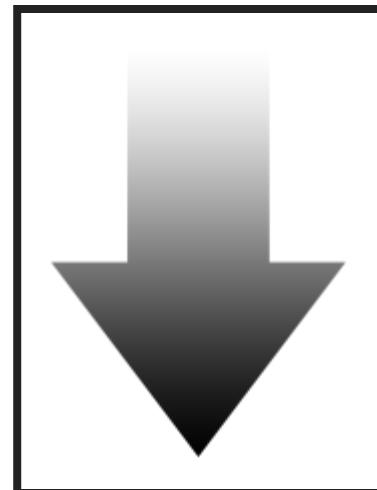
reveal.js enables you to create beautiful interactive slide decks using HTML. This presentation will show you examples of what it can do.



# VERTICAL SLIDES

Slides can be nested inside of each other.

Use the *Space* key to navigate through all slides.

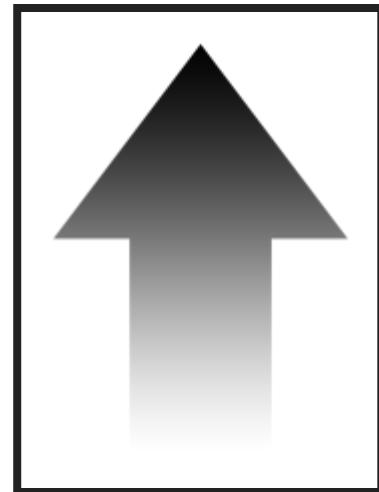


# **BASEMENT LEVEL 1**

Nested slides are useful for adding additional detail underneath a high level horizontal slide.

# BASEMENT LEVEL 2

That's it, time to go back up.



# SLIDES

Not a coder? Not a problem. There's a fully-featured visual editor for authoring these, try it out at  
<https://slides.com>.

# PRETTY CODE

```
1 import React, { useState } from 'react';
2
3 function Example() {
4   const [count, setCount] = useState(0);
5
6   return (
7     ...
8   );
9 }
```

Code syntax highlighting courtesy of [highlight.js](#).

# WITH ANIMATIONS

```
1 import React, { useState } from 'react';
2
3 function Example() {
4   const [count, setCount] = useState(0);
5
6   return (
7     <div>
8       <p>You clicked {count} times</p>
9       <button onClick={() => setCount(count + 1)}>
10         Click me
11       </button>
12     </div>
13   );
14 }
15
```

# WITH ANIMATIONS

```
1 import React, { useState } from 'react';
2
3 function Example() {
4   const [count, setCount] = useState(0);
5
6   return (
7     <div>
8       <p>You clicked {count} times</p>
9       <button onClick={() => setCount(count + 1)}>
10         Click me
11       </button>
12     </div>
13   );
14 }
15
```

# WITH ANIMATIONS

```
10      Click me
11      </button>
12    </div>
13  );
14 }
15
16 function SecondExample() {
17   const [count, setCount] = useState(0);
18
19   return (
20     <div>
21       <p>You clicked {count} times</p>
22       <button onClick={() => setCount(count + 1)}>
23         Click me
24       </button>
```

# WITH ANIMATIONS

```
15  ';
14 }
15
16 function SecondExample() {
17   const [count, setCount] = useState(0);
18
19   return (
20     <div>
21       <p>You clicked {count} times</p>
22       <button onClick={() => setCount(count + 1)}>
23         Click me
24       </button>
25     </div>
26   );
27 }
```

# POINT OF VIEW

Press **ESC** to enter the slide overview.

Hold down the **alt** key (**ctrl** in Linux) and click on any element to zoom towards it using [zoom.js](#). Click again to zoom back out.

(NOTE: Use **ctrl + click** in Linux.)

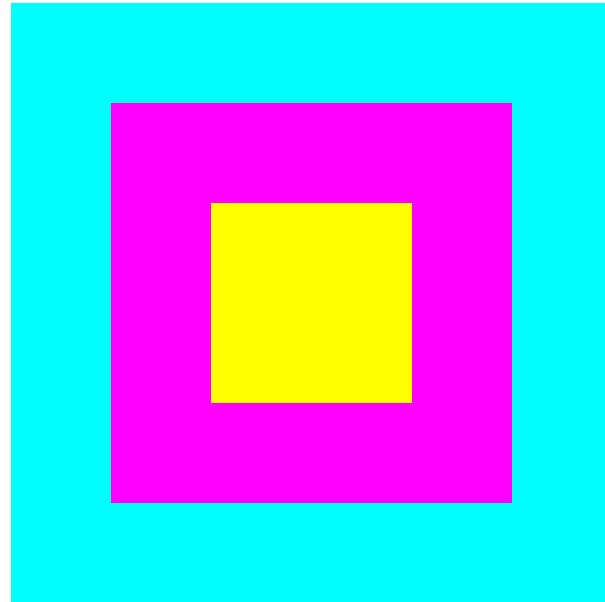
# AUTO-ANIMATE

Automatically animate matching elements across  
slides with [Auto-Animate](#).





**AUTO-ANIMATE**



**AUTO-ANIMATE**

# **TOUCH OPTIMIZED**

Presentations look great on touch devices, like mobile phones and tablets. Simply swipe through your slides.

# MARKDOWN SUPPORT

Write content using inline or external Markdown.  
Instructions and more info available in the [docs](#).

```
1 <section data-markdown>
2   ## Markdown Support
3
4   Write content using inline or external Markdown.
5   Instructions and more info available in the [docs](https://)
6 </section>
```

# LIGHTBOX

Turn any element into a **lightbox** using  
**data-preview-image** & **data-preview-video**.

```

```



```

```



Add the `r-fit-text` class to auto-size text

F

I

T

T

I

# FRAGMENTS

Hit the next arrow...

# FRAGMENTS

Hit the next arrow...

... to step through ...

# FRAGMENTS

Hit the next arrow...

... to step through ...

... a

# FRAGMENTS

Hit the next arrow...

... to step through ...

... a fragmented

# FRAGMENTS

Hit the next arrow...

... to step through ...

... a fragmented slide.

# FRAGMENT STYLES

There's different types of fragments, like:

grow

shrink

fade-out

Highlight red blue green

# FRAGMENT STYLES

There's different types of fragments, like:

grow

shrink

fade-out

Highlight red blue green

# FRAGMENT STYLES

There's different types of fragments, like:

grow

shrink

fade-out

Highlight red blue green

# FRAGMENT STYLES

There's different types of fragments, like:

grow

shrink

Highlight red blue green

# FRAGMENT STYLES

There's different types of fragments, like:

grow

shrink

fade-right,

Highlight red blue green

# FRAGMENT STYLES

There's different types of fragments, like:

grow

shrink

fade-right, up,

Highlight red blue green

# FRAGMENT STYLES

There's different types of fragments, like:

grow

shrink

fade-right, up, down,

Highlight red blue green

# FRAGMENT STYLES

There's different types of fragments, like:

grow

shrink

fade-right, up, down, left

Highlight red blue green

# FRAGMENT STYLES

There's different types of fragments, like:

grow

shrink

fade-right, up, down, left

fade-in-then-out

Highlight red blue green

# FRAGMENT STYLES

There's different types of fragments, like:

grow

shrink

fade-right, up, down, left

fade-in-then-semi-out

Highlight red blue green

# FRAGMENT STYLES

There's different types of fragments, like:

grow

shrink

fade-right, up, down, left

fade-in-then-semi-out

Highlight **red** blue green

# FRAGMENT STYLES

There's different types of fragments, like:

grow

shrink

fade-right, up, down, left

fade-in-then-semi-out

Highlight red blue green

# FRAGMENT STYLES

There's different types of fragments, like:

grow

shrink

fade-right, up, down, left

fade-in-then-semi-out

Highlight red blue green

# TRANSITION STYLES

You can select from different transitions, like:

None - Fade - Slide - Convex - Concave - Zoom

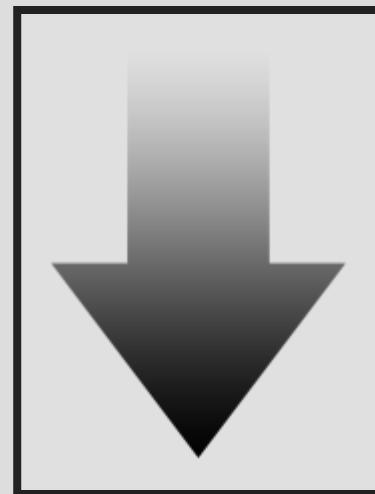
# THEMES

reveal.js comes with a few themes built in:

Black (default) - White - League - Sky - Beige - Simple  
Serif - Blood - Night - Moon - Solarized

# SLIDE BACKGROUNDS

Set `data-background="#dddddd"` on a slide to change the background color. All CSS color formats are supported.



# GRADIENT BACKGROUNDS

```
<section data-background-gradient= "linear-gradient(to bottom, #ddd, #191919)">
```

# IMAGE BACKGROUNDS

```
<section data-background="image.png">
```

# TILED BACKGROUNDS

```
<section data-background="image.png" data-background-repeat="r
```

# VIDEO BACKGROUNDS

```
<section data-background-video="video.mp4,video.webm">
```

A photograph of a man with short brown hair, wearing a white button-down shirt and a dark blue and grey striped tie. He is seated at a light-colored wooden desk, looking upwards and to the right with a wide-eyed, surprised expression. His mouth is slightly open. On the desk in front of him is a small, round, metallic object, possibly a paperweight or a clock. To his left, the back of another person's head and shoulders are visible; this person has dark hair and is wearing a dark jacket over a light-colored shirt. The background is a plain, light-colored wall.

... AND GIFS.

# BACKGROUND TRANSITIONS

Different background transitions are available via the `backgroundTransition` option. This one's called "zoom".

```
Reveal.configure({ backgroundTransition: 'zoom' })
```

# BACKGROUND TRANSITIONS

You can override background transitions per-slide.

```
<section data-background-transition="zoom">
```

## IFRAME BACKGROUNDS

Since reveal.js runs on the web, you can easily embed other web content. Try interacting with the page in the background.

# MARVELOUS LIST

- No order here
- Or here
- Or here
- Or here

# FANTASTIC ORDERED LIST

1. One is smaller than...
2. Two is smaller than...
3. Three!

# TABULAR TABLES

| Item     | Value | Quantity |
|----------|-------|----------|
| Apples   | \$1   | 7        |
| Lemonade | \$2   | 18       |
| Bread    | \$3   | 2        |

# CLEVER QUOTES

These guys come in two forms, inline: “*The nice thing about standards is that there are so many to choose from*” and block:

“*For years there has been a theory that millions of monkeys typing at random on millions of typewriters would reproduce the entire works of Shakespeare. The Internet has proven this theory to be untrue.*”

# INTERGALACTIC INTERCONNECTIONS

You can link between slides internally, [like this](#).

# SPEAKER VIEW

There's a [speaker view](#). It includes a timer, preview of the upcoming slide as well as your speaker notes.

Press the S key to try it out.

# EXPORT TO PDF

Presentations can be [exported to PDF](#), here's an example:



# GLOBAL STATE

Set `data-state="something"` on a slide and "`something`" will be added as a class to the document element when the slide is open. This lets you apply broader style changes, like switching the page background.

# STATE EVENTS

Additionally custom events can be triggered on a per slide basis by binding to the data-state name.

```
Reveal.on( 'customevent', function() {  
  console.log( '"customevent" has fired' );  
});
```

# TAKE A MOMENT

Press B or . on your keyboard to pause the presentation. This is helpful when you're on stage and want to take distracting slides off the screen.

# MUCH MORE

- Right-to-left support
- Extensive JavaScript API
- Auto-progression
- Parallax backgrounds
- Custom keyboard bindings

# THE END

- Try the online editor
- Source code & documentation