reveal js

THE HTML PRESENTATION FRAMEWORK

Created by Hakim El Hattab and contributors

Sponsored by codeless s slides

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

Slides can be nested inside of each other.

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

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

That's it, time to go back up.

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

```
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.

```jsx
import React, { useState } from 'react';

function Example() {
  const [count, setCount] = useState(0);

  return (
    <div>
      <p>You clicked {count} times</p>
      <button onClick={() => setCount(count + 1)}>
        Click me
      </button>
    </div>
  );
}

function SecondExample() {
```

```
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
16  function SecondExample() {
```

```jsx
import React, { useState } from 'react';

function Example() {
  const [count, setCount] = useState(0);

  return (
    <div>
      <p>You clicked {count} times</p>
      <button onClick={() => setCount(count + 1)}>
        Click me
      </button>
    </div>
  );
}

function SecondExample() {
```

```
 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
16  function SecondExample() {
```

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

Hold down the **alt** key (**ctrl** in Linux) and click on a element to zoom towards it using zoom.js. Click agai zoom back out.
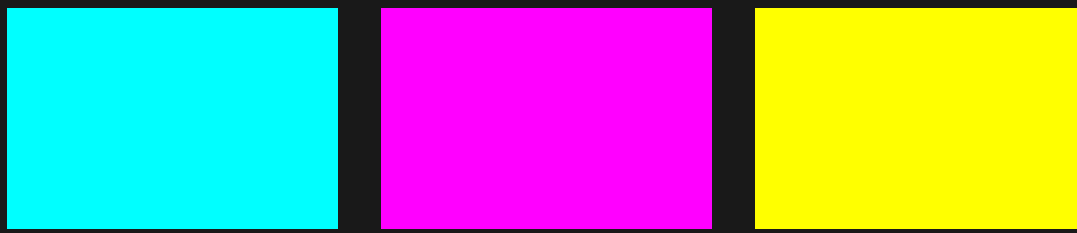
(NOTE: Use ctrl + click in Linux.)

Automatically animate matching elements across sli
with Auto-Animate.

AUTO-ANIMATE

AUTO-ANIMATE

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

Write content using inline or external Markdown. Instructions and more info available in the readme

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

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

FIT TEXT

Hit the next arrow...

Hit the next arrow...

... to step through ...

Hit the next arrow...

... to step through ...

... a

Hit the next arrow...

... to step through ...

... a fragmented

Hit the next arrow...

... to step through ...

... a fragmented slide.

There's different types of fragments, like:

grow

shrink

fade-out

Highlight red blue green

There's different types of fragments, like:

grow

shrink

fade-out

Highlight red blue green

There's different types of fragments, like:

grow

shrink

fade-out

Highlight red blue green

There's different types of fragments, like:

grow

shrink

Highlight red blue green

There's different types of fragments, like:

grow

shrink

fade-right,

Highlight red blue green

There's different types of fragments, like:

grow

shrink

fade-right,up,

Highlight red blue green

There's different types of fragments, like:

grow

shrink

fade-right,up,down,

Highlight red blue green

There's different types of fragments, like:

grow

shrink

fade-right,up,down,left

Highlight red blue green

There's different types of fragments, like:

grow

shrink

fade-right,up,down,left

fade-in-then-out

Highlight red blue green

There's different types of fragments, like:

grow

shrink

fade-right,up,down,left

fade-in-then-semi-out

Highlight red blue green

There's different types of fragments, like:

grow

shrink

fade-right,up,down,left

fade-in-then-semi-out

Highlight red blue green

There's different types of fragments, like:

grow

shrink

fade-right,up,down,left

fade-in-then-semi-out

Highlight red blue green

There's different types of fragments, like:

grow

shrink

fade-right,up,down,left

fade-in-then-semi-out

Highlight red blue green

You can select from different transitions, like:
None - Fade - Slide - Convex - Concave - Zoom

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

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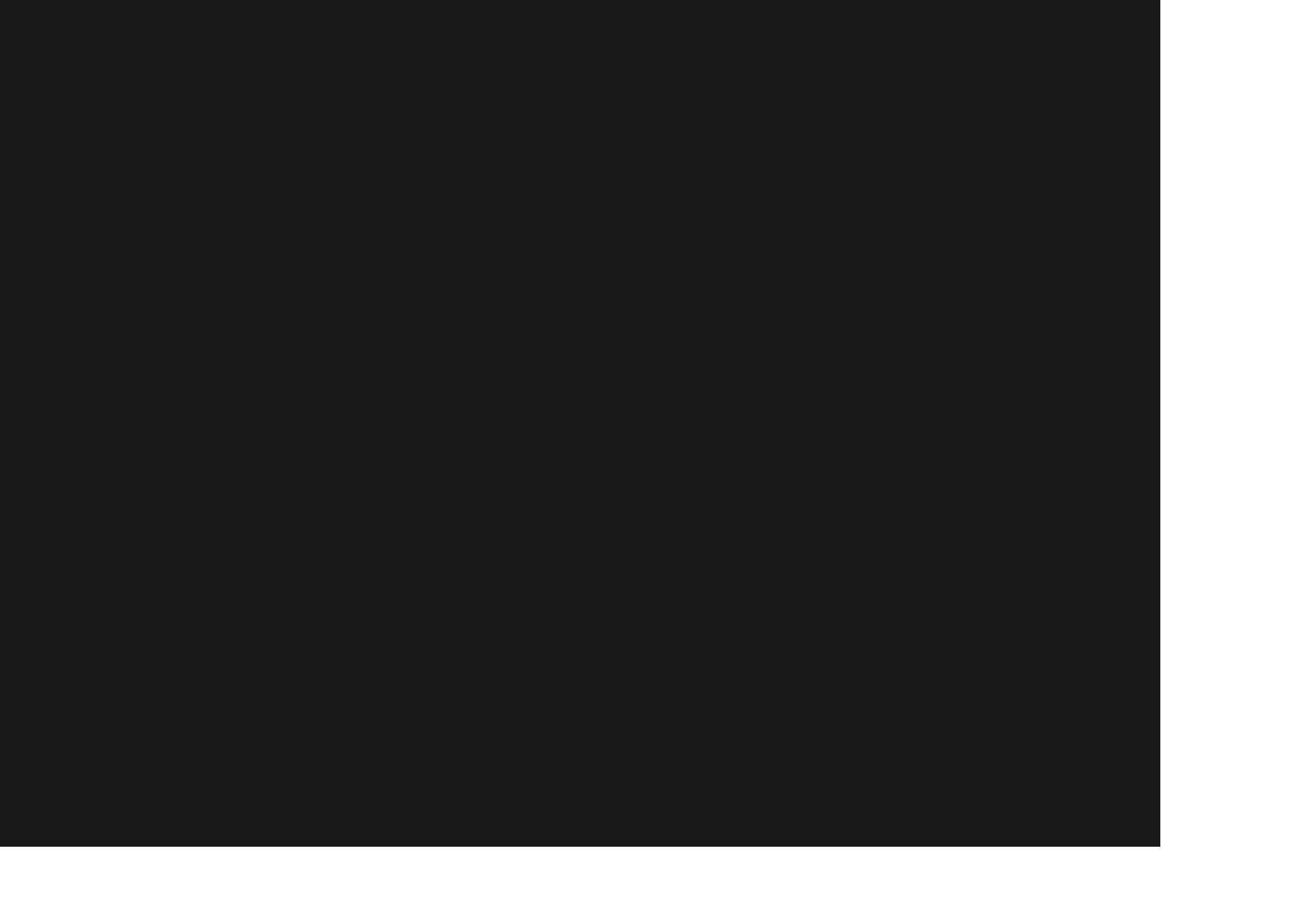
```
<section data-background="image.png" data-background-repeat="repeat"
```

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

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

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

You can override background transitions per-slide

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

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

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

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

These guys come in two forms, inline: *"The nice thi about standards is that there are so many to choo 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."*

INTERCONNECTIONS
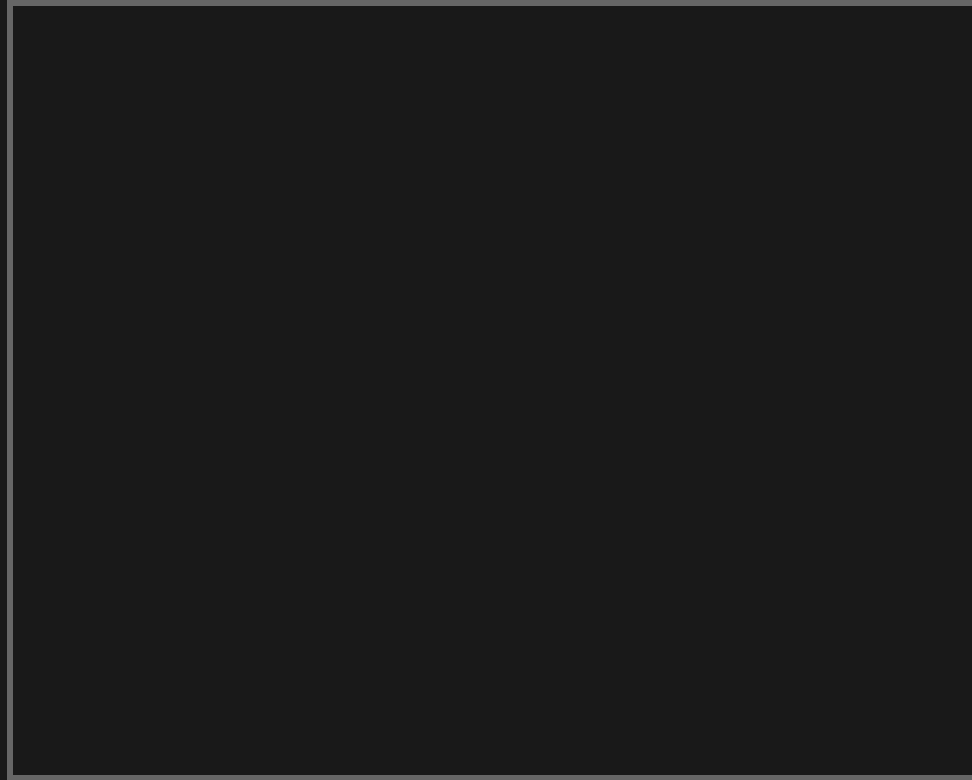
You can link between slides internally, like this.

There's a speaker view. It includes a timer, preview the upcoming slide as well as your speaker notes

Press the *S* key to try it out.

Presentations can be exported to PDF, here's an
example:

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.

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

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

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.

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

- Try the online editor
- Source code & documentation