

Demos

VH, VW units

```
font-size: 5vw;  
height: 50vh;
```

SVG sprites

- Combines single **HTTP request** advantage of bitmap sprites, with resolution-independent advantages of SVG.
- Use Modernizr to test for SVG support. If found, adds SVG class to the HTML element.

```
<html class="svg">
```

- This rule changes the background image to SVG.
- Edit SVG class in web inspector to see the change.

```
.svg .actions-svg a.sprite {  
    background-image: url('sprite.svg');
```

- Uses **background-position** for vertical offsetting.
- David Bushell <http://dbushell.com>

Responsive Calendar

- **Vertical** media queries used.
- Hide/show inline **spans**

PatternLab

- Brad Frost
- PHP implementation

- Identify hierarchy of things that make up a page.
- Trying to avoid tight coupling between elements of page and layout too early in responsive design.
- **ATOMS** : a text input field in a form
- **MOLECULE** : combination of text input and magnify icon
- **ORGANISM** : Header combining search box with Home/About/Blog/Contact
- **TEMPLATE** : abstract layout for a type of page
- **PAGE** : actual instance of a template with content.
- View this hierarchy of things at different viewport widths

Mobile First

- BBC multi column layout
- Use **jQuery slide** left or right to add content in **mobile first** layout.

LocalStorage.

```
var raceTime = JSON.parse( localStorage.time );
localStorage.time = JSON.stringify( raceTime );
```

FlexBox

The **display order** of child elements can be changed by CSS, independent of their **source code** order in the HTML.

```
.contentOrder {
    -webkit-order: 2;
}
```

jQuery to add/remove this class from a child element, and so change its display position.

```
$("#stairs").toggleClass( "contentOrder" )
```

Retina displays

Media query to pull in double-size image on retina displays, and then use

background-size property to shrink it to lower quality image size.

```
@media screen and (-webkit-device-pixel-ratio: 2) {  
  .desert {  
    background: url(../images/large.jpeg);  
    background-size: 400px 250px;  
  }  
}
```

In Javascript we can also test for **window.devicePixelRatio** which is 1 on desktop screens, and 2 or higher, on retina screens.

This is the ratio between physical pixels on the screen, and **device independent pixels DIPS**, the abstract unit used by CSS in media queries and HTML in meta viewport statements.

```
window.devicePixelRatio
```