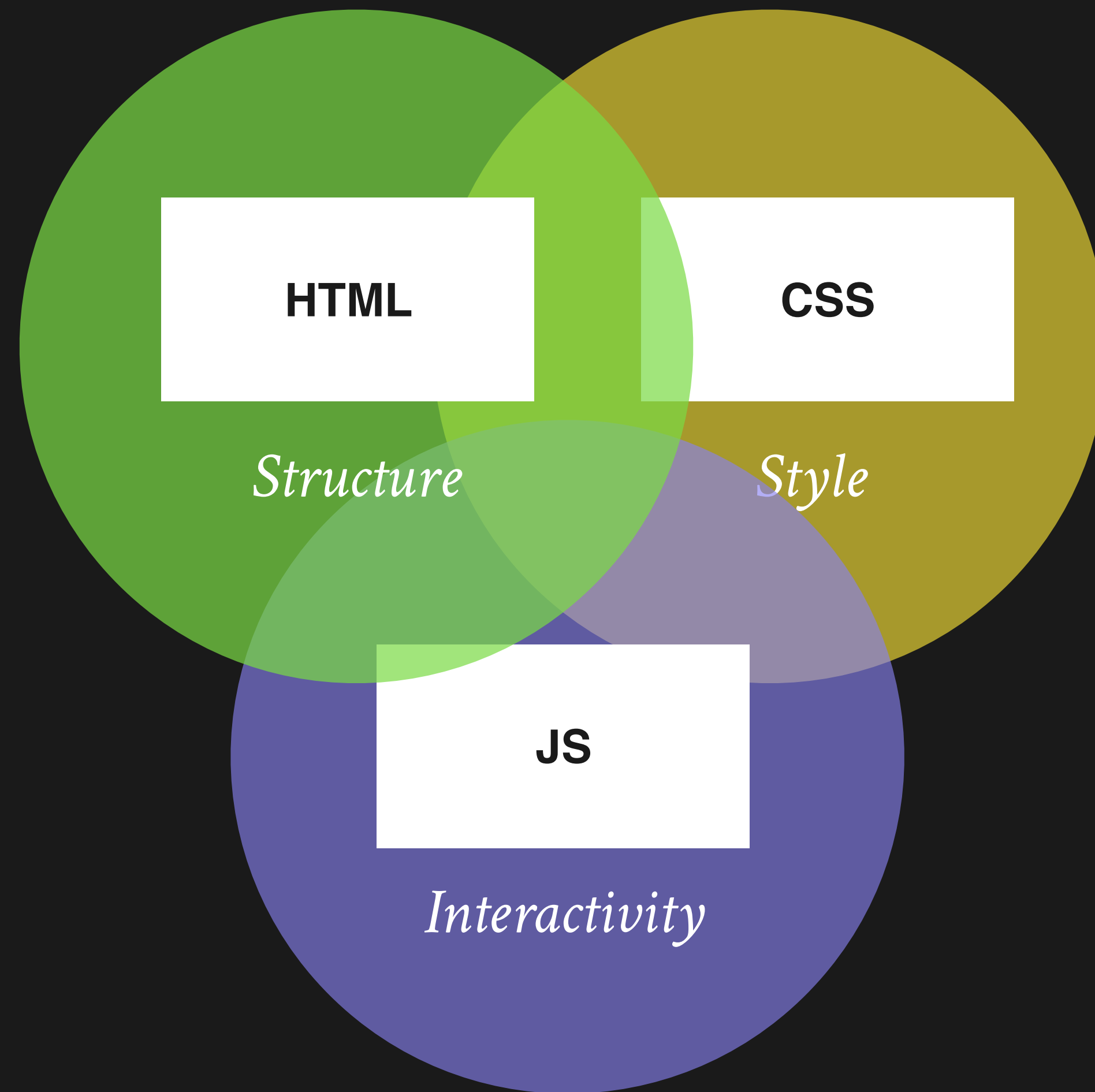




WEB TECHNOLOGIES PROGRAM

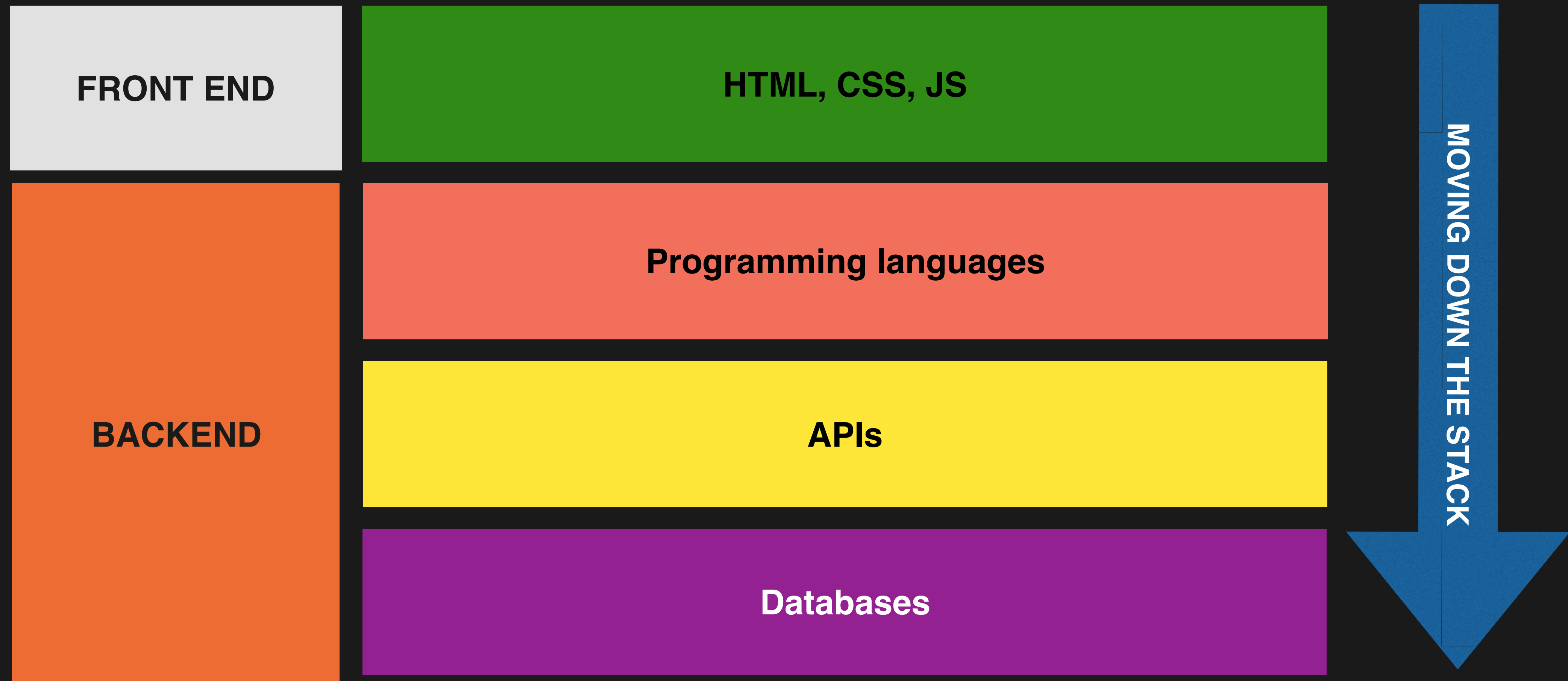
# An introduction to CSS

*How styles and structure work together*



# CSS in the full stack

*Where does it fit into the picture?*





C

Cascading

S

Style

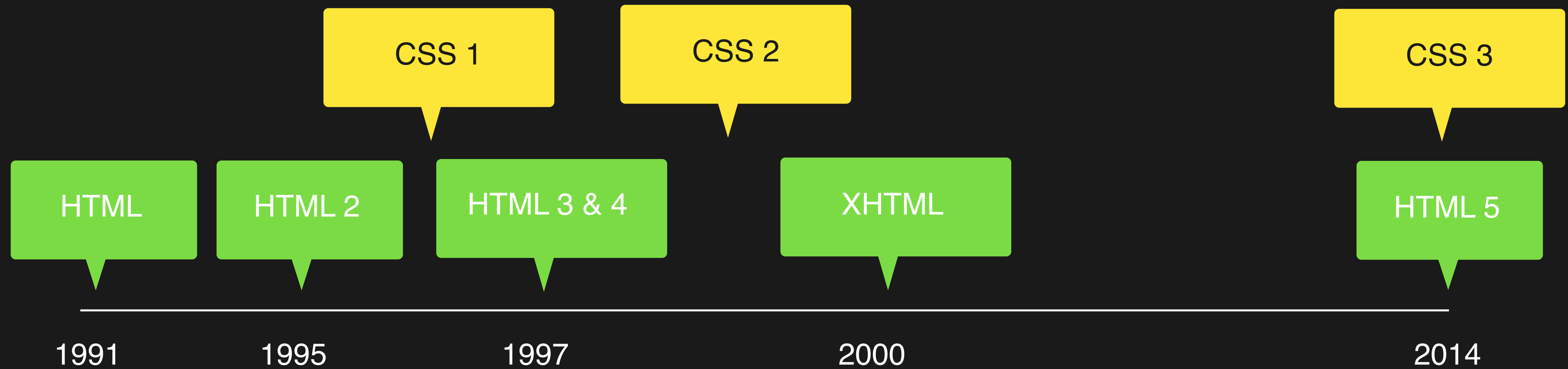
S

Sheets





# A brief history



**CSS**



Rounded corners!

A red shadow.

Gradient Background

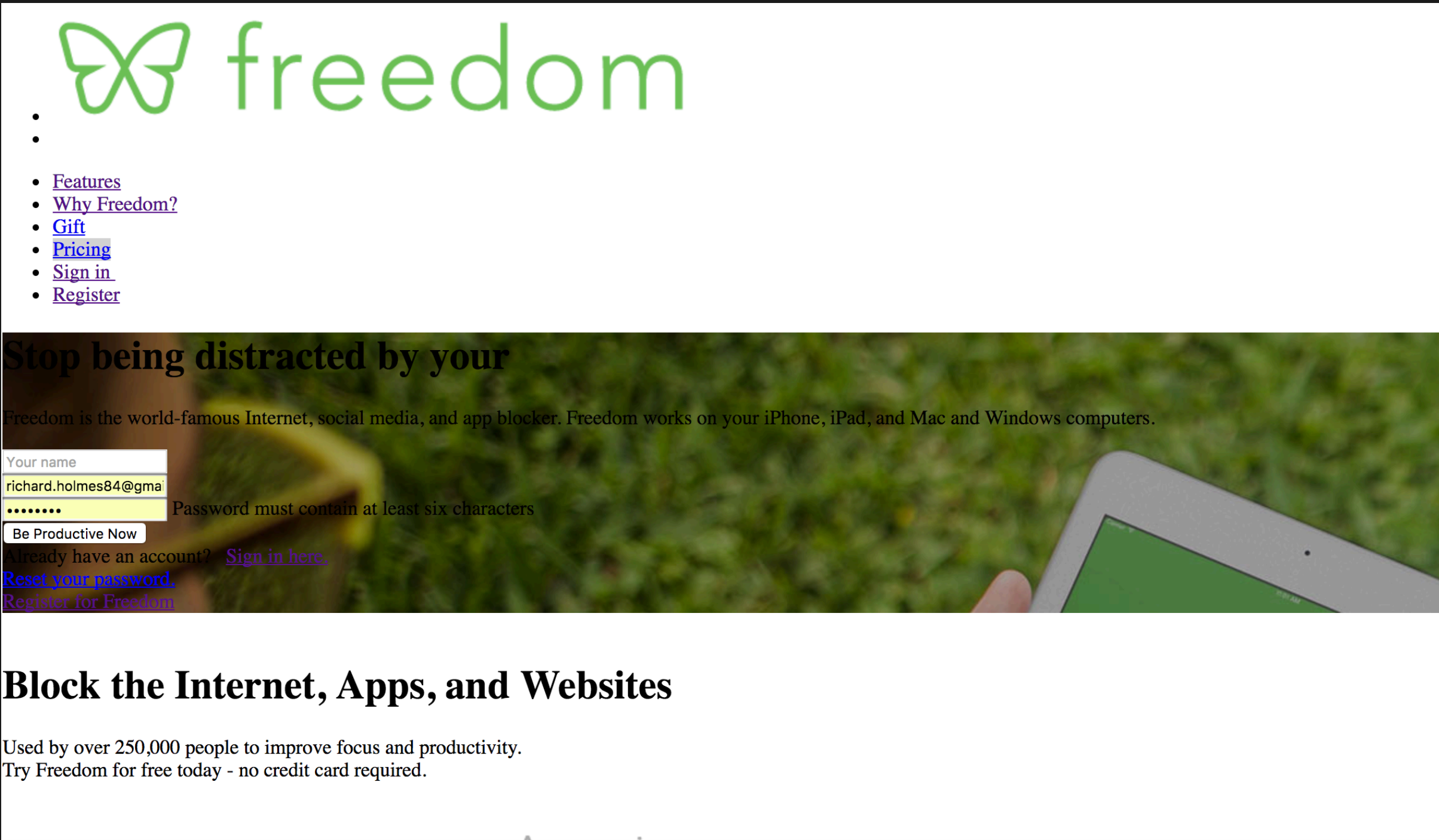




# What can CSS do?



# HTML only



# HTML + CSS



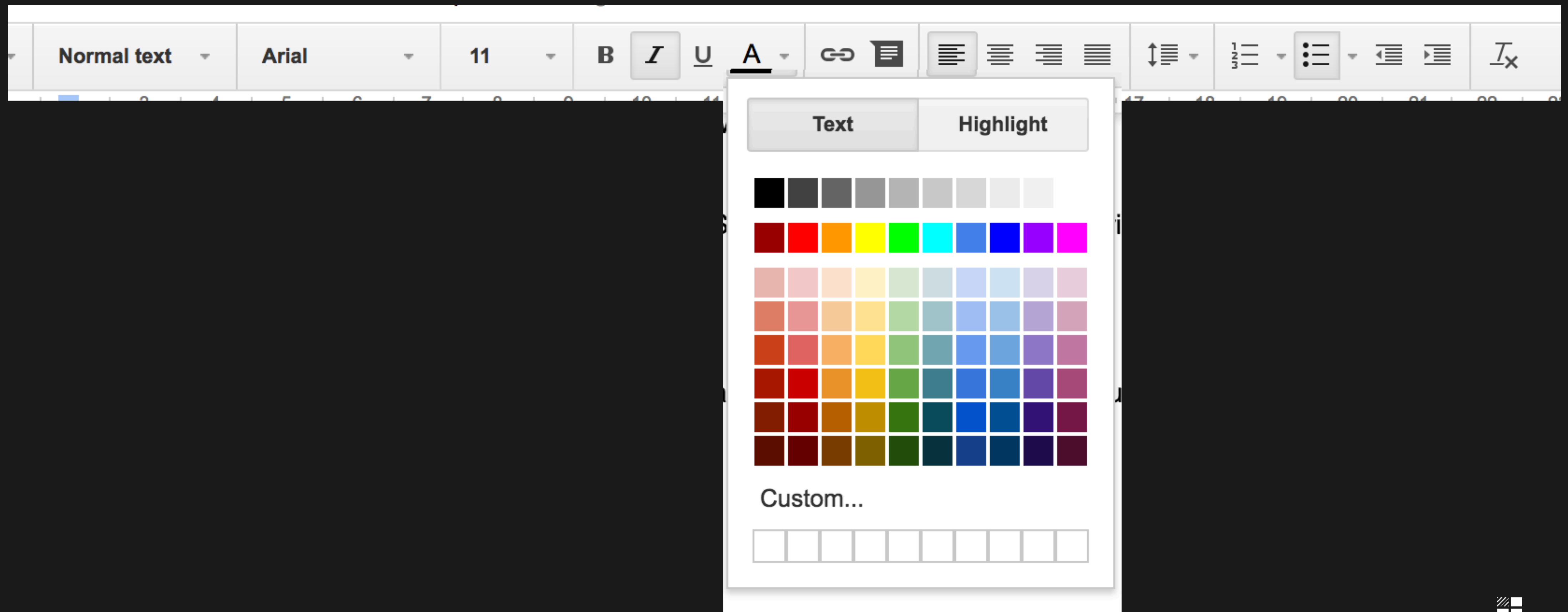
# The 2 functions of CSS

**Applying styles**

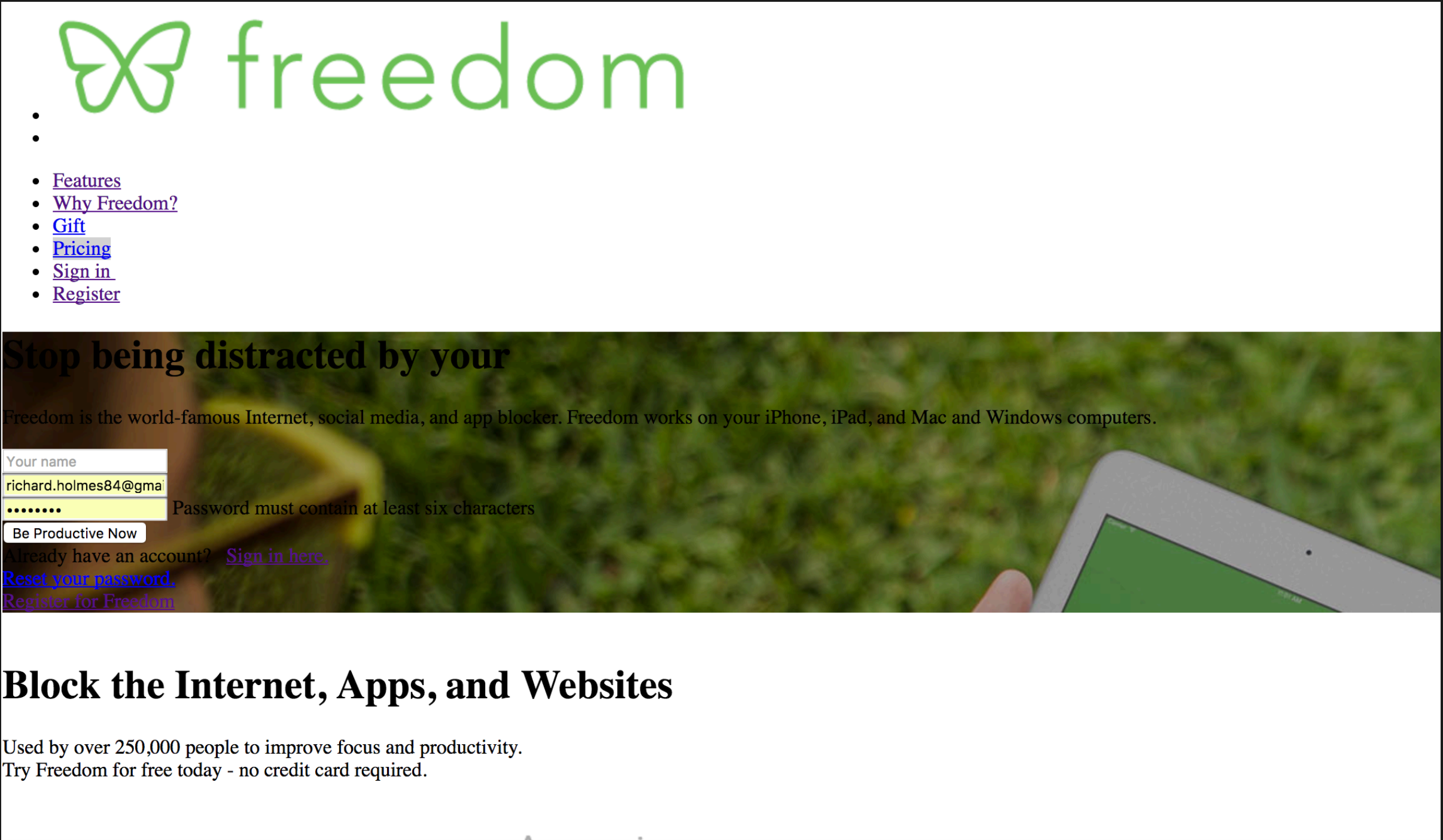
**Positioning  
elements**











CSS positioning

CSS background

CSS buttons



CSS fonts



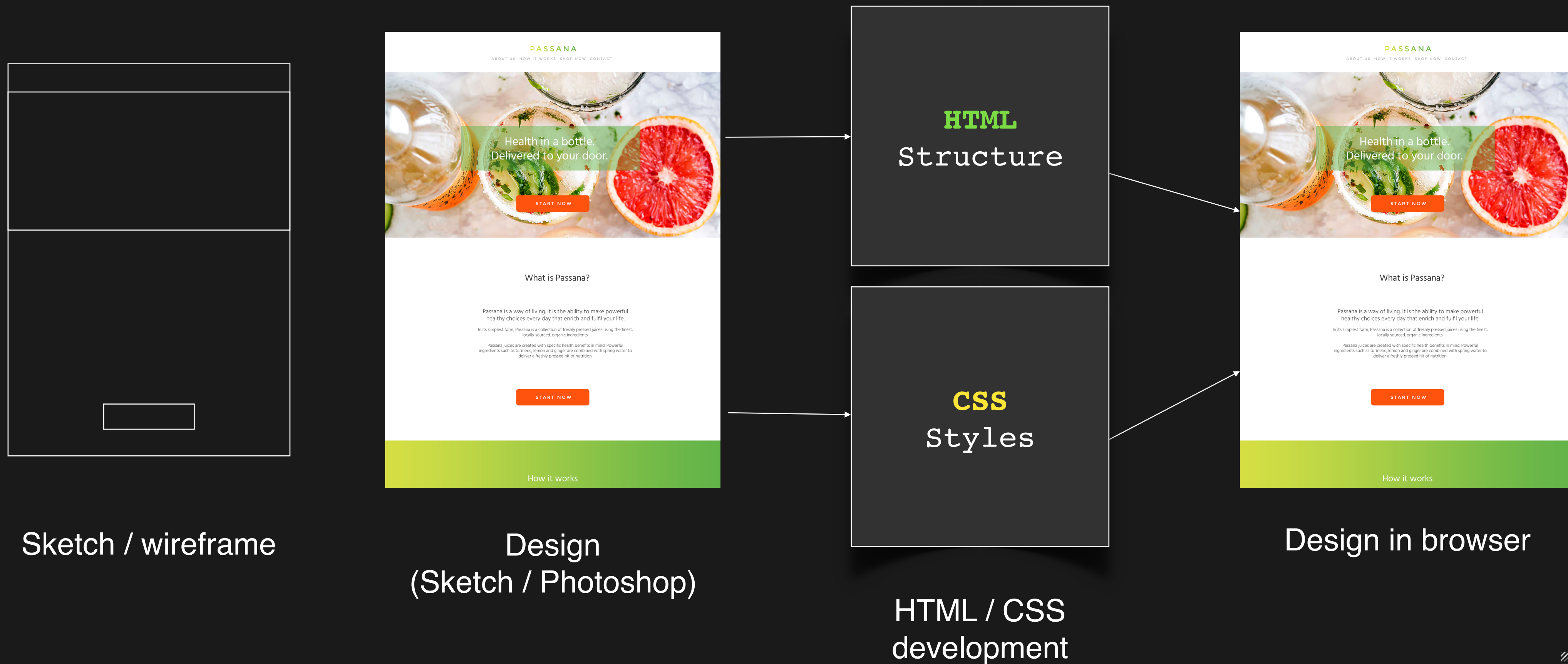
# Who uses CSS?



# How is CSS applied?



# The front end development process





# How is CSS applied?

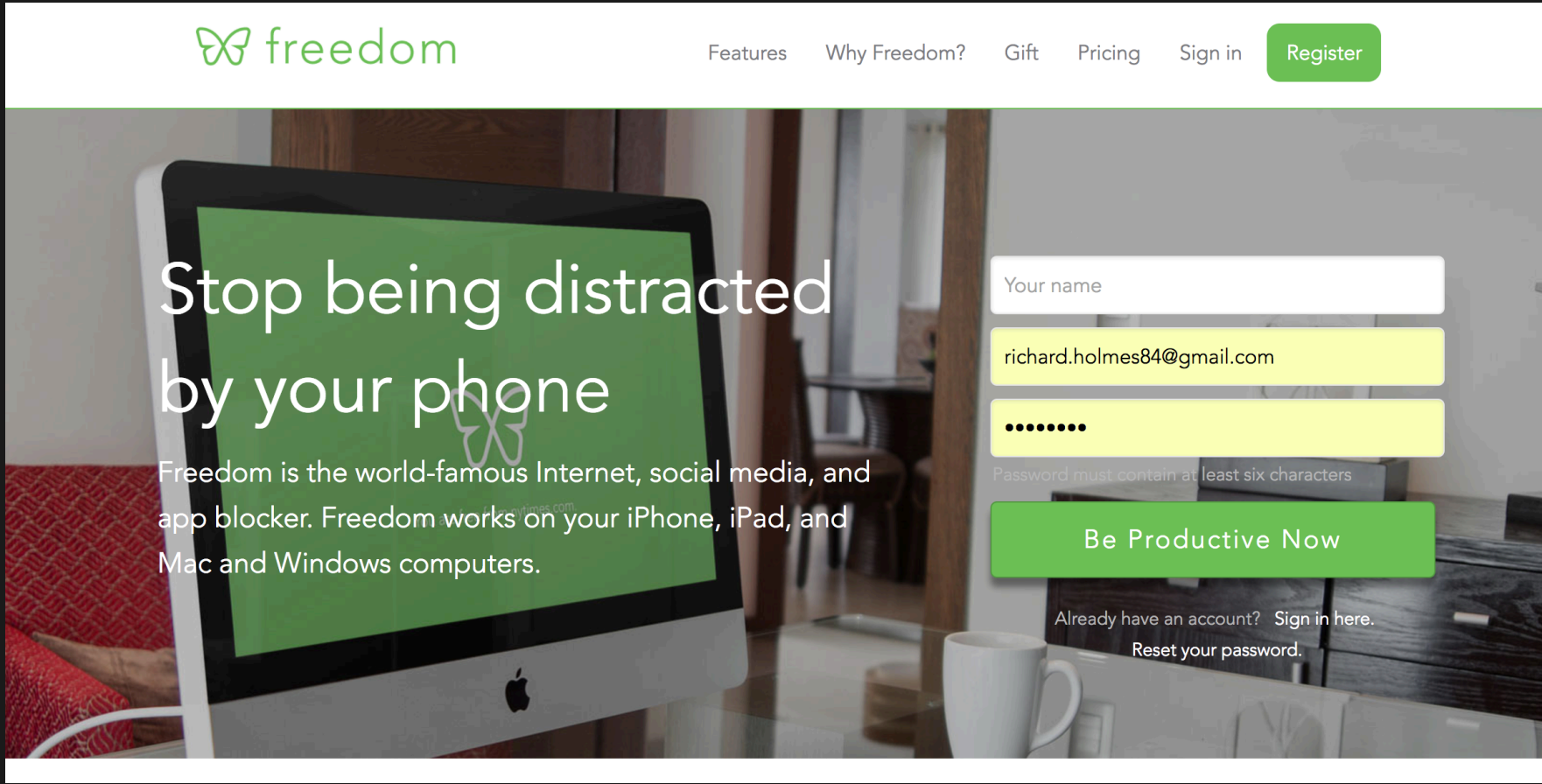
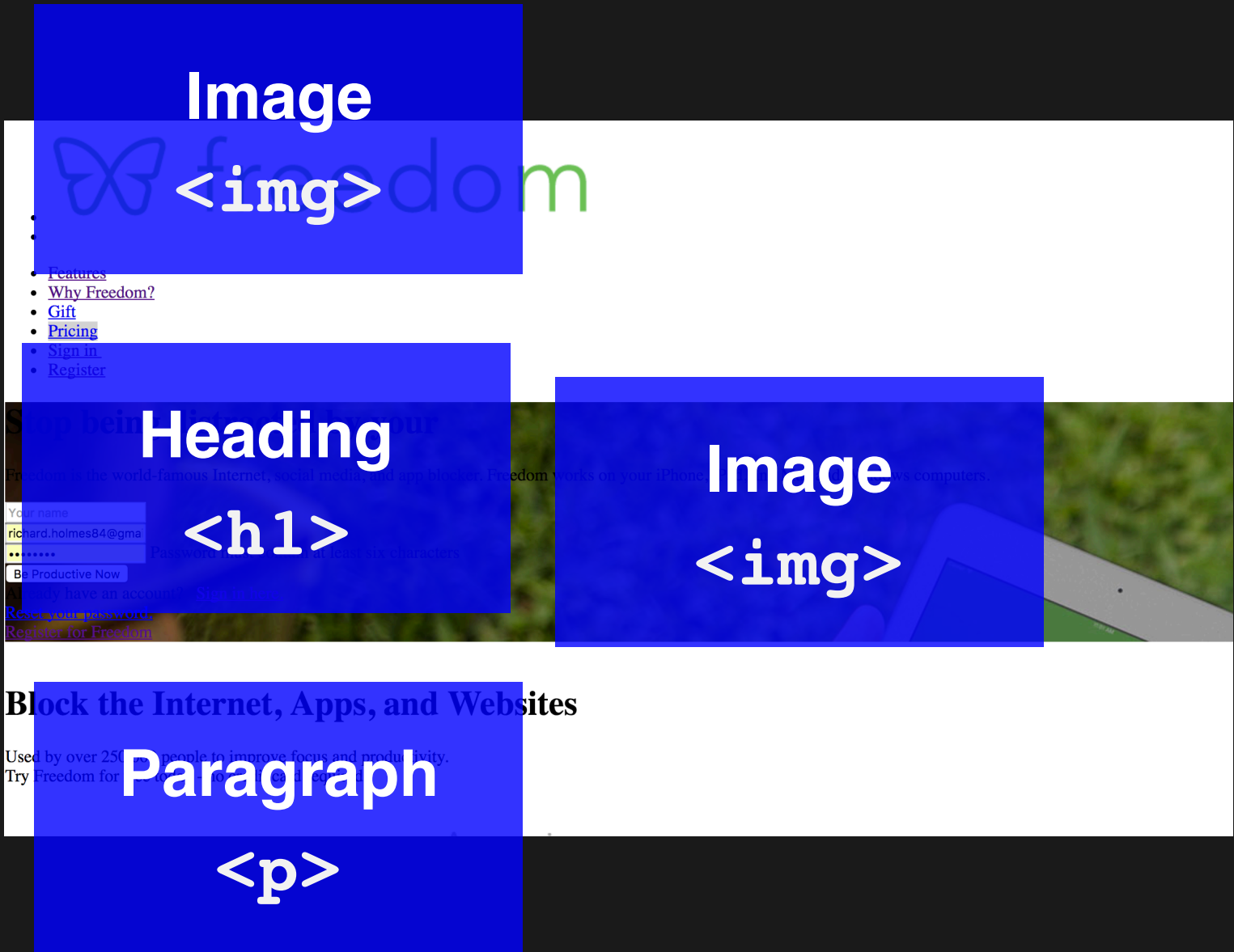
**CSS  
Stylesheets**

**Inline CSS**



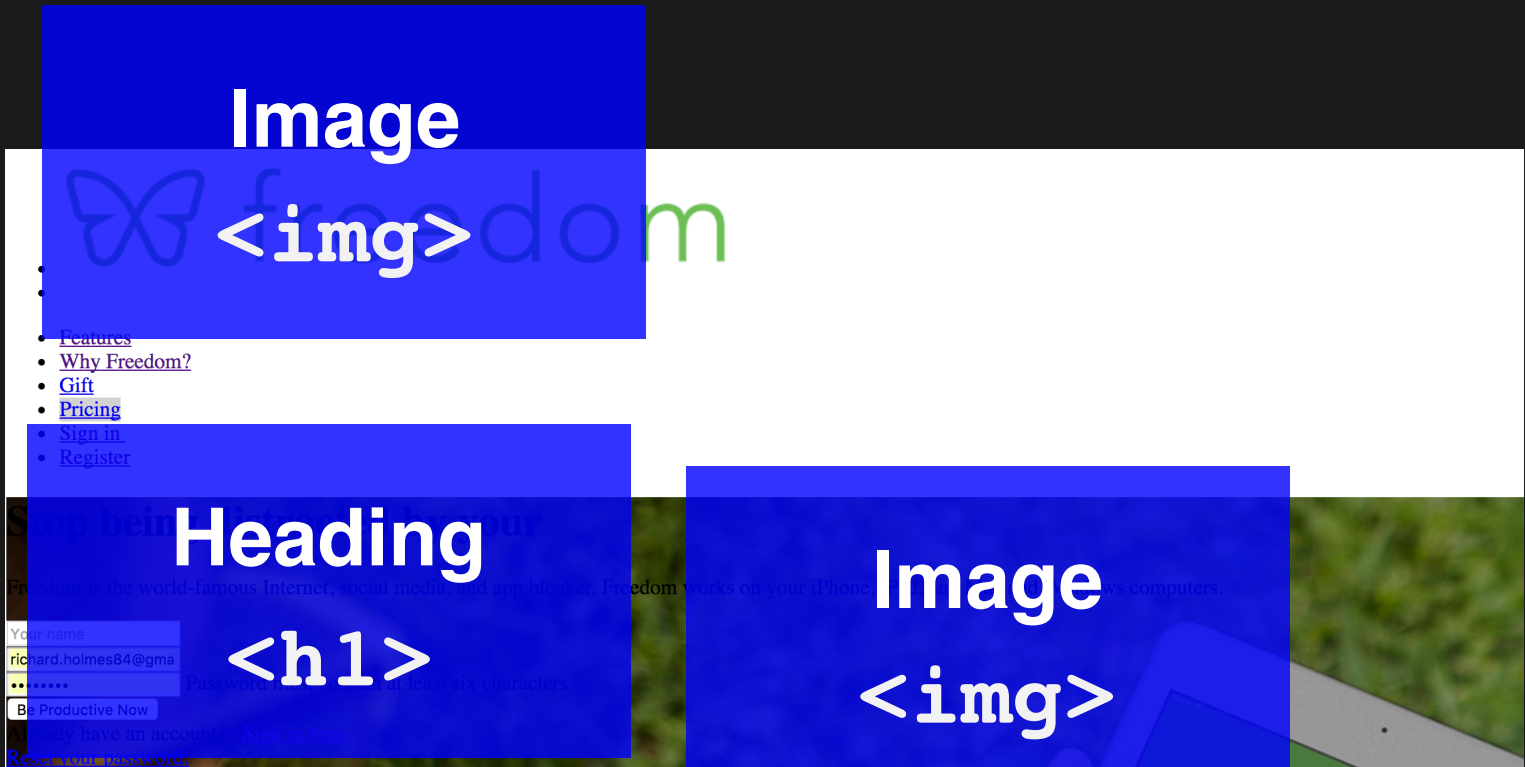
# CSS stylesheets







# HTML Structure



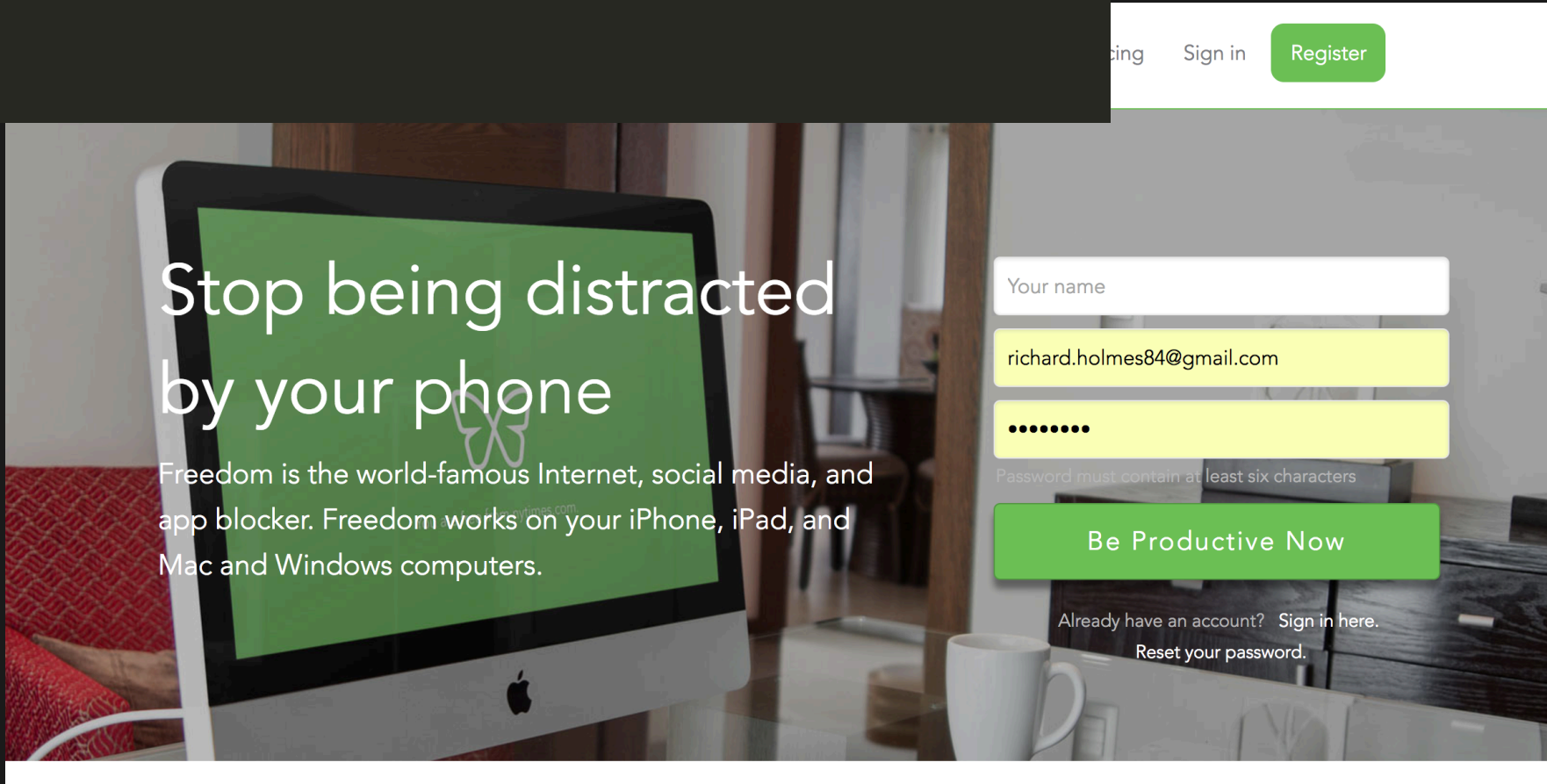
```
<!DOCTYPE html> - <!-- appears once, before any HTML tags. It helps the browser to display content correctly. -->
<html>
<head>

<title>Title of website</title>
<!-- CSS styles are put in the header-->
<link rel="stylesheet" type="text/css" href="styles.css">
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>

</head>
<body>
<!-- the elements of the web page are put inside the body tags. -->
</body>
</html>
```



# CSS Stylesheet





# CSS stylesheet

```
/*-This is the finished stylesheet. This is how you add comments in CSS-*/
```

```
#passana-container h1 {  
  font-size: 48px;  
  font-weight: 400;  
}
```

```
#passana-container h2 {  
  font-size: 36px;  
  font-weight: 400;  
}
```

```
.center {  
  text-align: center;  
}
```

```
h1, p, body {  
  font-family: 'Hind';  
}
```

```
.underline {  
  text-decoration: underline;  
}
```

```
p {  
  line-height: normal;  
}
```

```
/*-----Header section-----*/
```

```
.main-logo {  
  display: block;  
  margin: 0 auto;  
}
```

```
nav {  
  padding: 0 5% 0 5%;  
  margin-top: 3%;  
}
```

```
nav li {  
  display: inline;  
  list-style-type: none;  
  padding-right: 3%;  
  font-size: 14px;  
}
```

Make the font size 36px

Center the text

Add space above the navigation



# Advantages of stylesheets

1. Increased control
2. SEO benefits
3. Maintainability



# HTML Structure

```
<h1 style="color:blue;margin: 0 auto; width: 50%; text-align: center;">HTML Structure</h1>

<h2>What does HTML do? </h2>

<h3>Structure - </h3>
<p>HTML enables you to describe the structure of a document so it can be interpreted and displayed correctly by the browser and styled CSS.
It tells the browser what the page consists of. It tells the browser 'this is an image, this is a table, this is a paragraph and so on'. These are known as elements.</p>

<ul>
<li>Elements - the 'things' on a page are known as elements. </li>
<li>Linking - one of the most important elements in HTML is the anchor tag - this allows documents to be linked together. This was one of the primary reasons for HTML when it was introduced; linking documents together to create a 'network' of documents that scientists could use</li>
</ul>

<h3>Who uses HTML?</h3>
<ul>
<li>Your front end developers work with HTML, primarily to build the structure of web pages. </li>
<li>Backend programmers will sometimes use it but </li>
<li>Marketing folks may using HTML for writing blog posts </li>
<li>Example - tools like WordPress have a 'Visual' editor or a 'Text' editor - https://www.evernote.com/l/AHVsuBQpKP1Hu6lo5x0uErrXWxkr1mFbq-w </li>
<a href="http://www.link.com">Click</a>
</ul>

 |
```

# CSS Stylesheet

```
/*-This is the finished stylesheet for the HTML document - CSS-*/

#passana-container h1 {
  font-size: 48px;
  font-weight: 400;
}

#passana-container h2 {
  font-size: 36px;
  font-weight: 400;
}

.center {
  text-align: center;
}

h1, p, body {
  font-family: 'Hind';
}

.underline {
  text-decoration: underline;
}

p {
  line-height: normal;
}

/*-----Header section-----*/

.main-logo {
  display: block;
  margin: 0 auto;
}

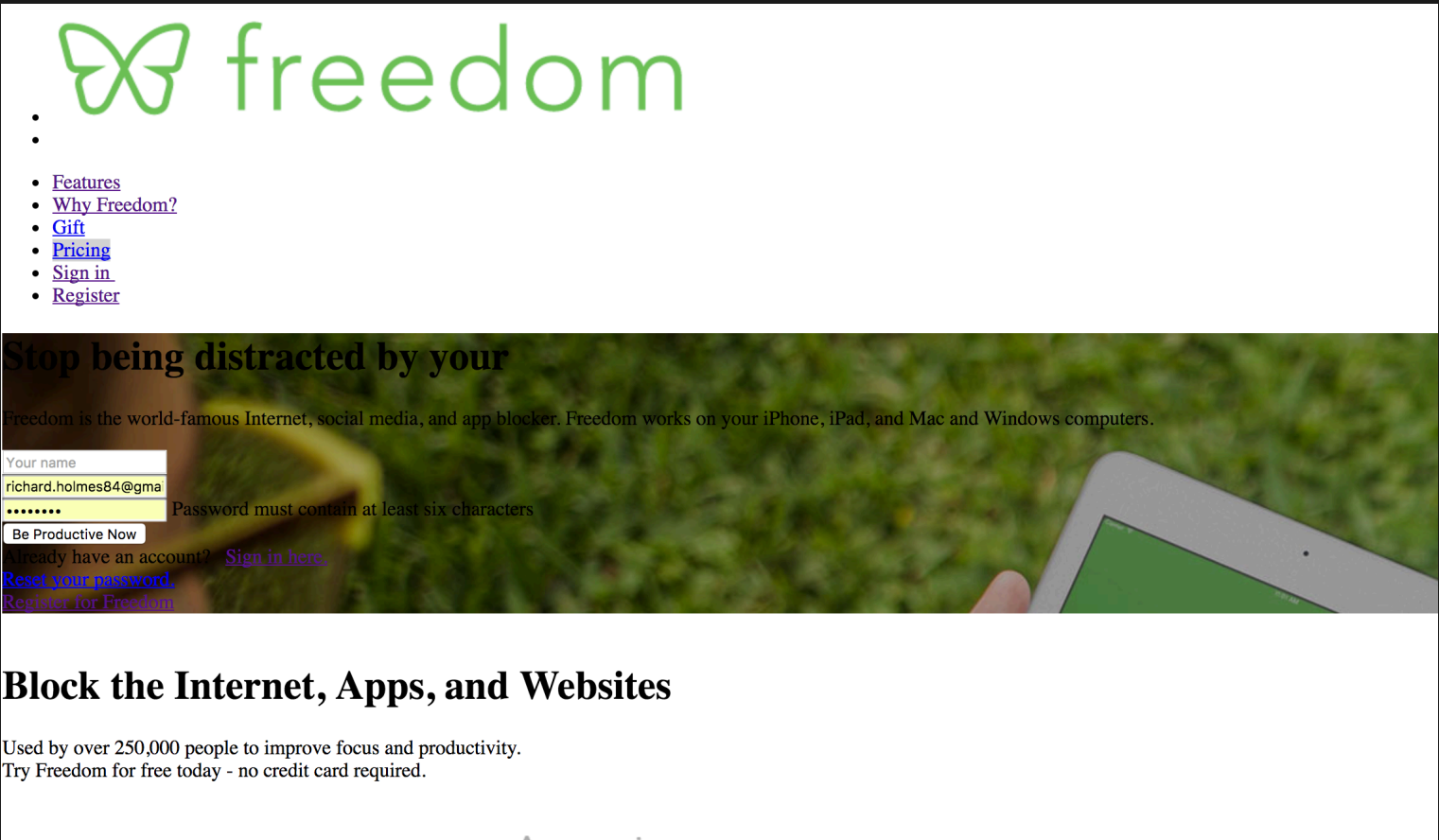
nav {
  padding: 0 5% 0 5%;
  margin-top: 3%;
}

nav li {
  display: inline;
  list-style-type: none;
  padding-right: 3%;
  font-size: 14px;
}
```



# Inline CSS









# HTML

## Structure

# CSS

## Styles



```
1 <h1 style="color:blue;margin-left:30px;">This is a heading</h1>
2
3
4
5 <h2>What does HTML do? </h2>
6
7
8 <h3>Structure - </h3>
9 <p>HTML enables you to describe the structure of a document so it can be interpreted and displayed correctly by the browser and styled CSS.
10 It tells the browser what the page consists of. It tells the browser 'this is an image, this is a table, this is a paragraph and so on'.
11 These are known as elements.</p>
12
13 <ul>
14 <li>Elements - the 'things' on a page are known as elements. </li>
15 <li>Linking - one of the most important elements in HTML is the anchor tag - this allows documents to be linked together. This was one of the
16 primary reasons for HTML when it was introduced; linking documents together to create a 'network' of documents that scientists could use</li>
17 </ul>
18
19 <h3>Who uses HTML?</h3>
20 <ul>
21 <li>Your front end developers work with HTML, primarily to build the structure of web pages. </li>
22 <li>Backend programmers will sometimes use it but </li>
23 <li>Marketing folks may using HTML for writing blog posts </li>
24 <li>Example - tools like WordPress have a 'Visual' editor or a 'Text' editor - https://www.evernote.com/l/AHVsuBQpKP1Hu6lo5x0uErrXWxkr1mFbq-w
25 </li>
26 <a href="http://www.link.com">Click</a>
27 </ul>
28
29 
```





# Why not to use inline CSS

1. Separation of structure and styles
2. *Maintenance*
3. However...



# Where inline may be acceptable

# THE VERGE

TWEET
SHARE

...y saw nothing revolutionary about [4K on the Xperia Z5](#) has a partnership with Amazon for 4K HDR content, and the HDR element truly does wonderful things for colour reproduction. In one of Sony's demos against the Z5 Premium, the XZ showed off warmer, more natural yellows and a broadly more realistic and inviting reproduction of the streets of Lisbon. A Portuguese Sony rep was on hand to confirm, effusively, that the XZ Premium display was more faithful to the real Lisbon.

Sony Xperia XZ Premium

Oscars 2017: how to watch the Academy Awards online

Nokia's 3310 returns to life as a modern classic

AT&T is making its unlimited plan way more competitive



## Quick mockups / demos



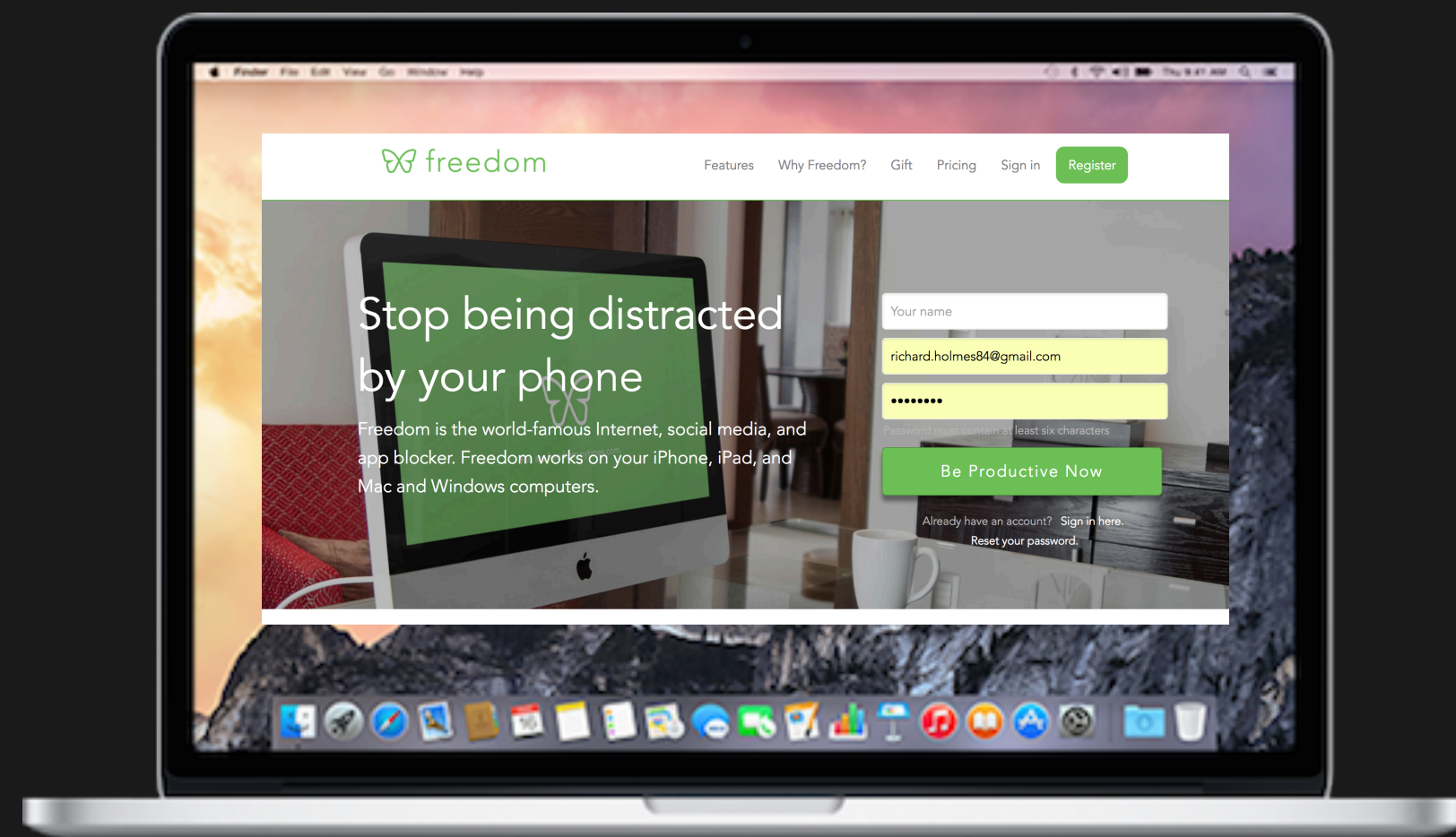
## Emails



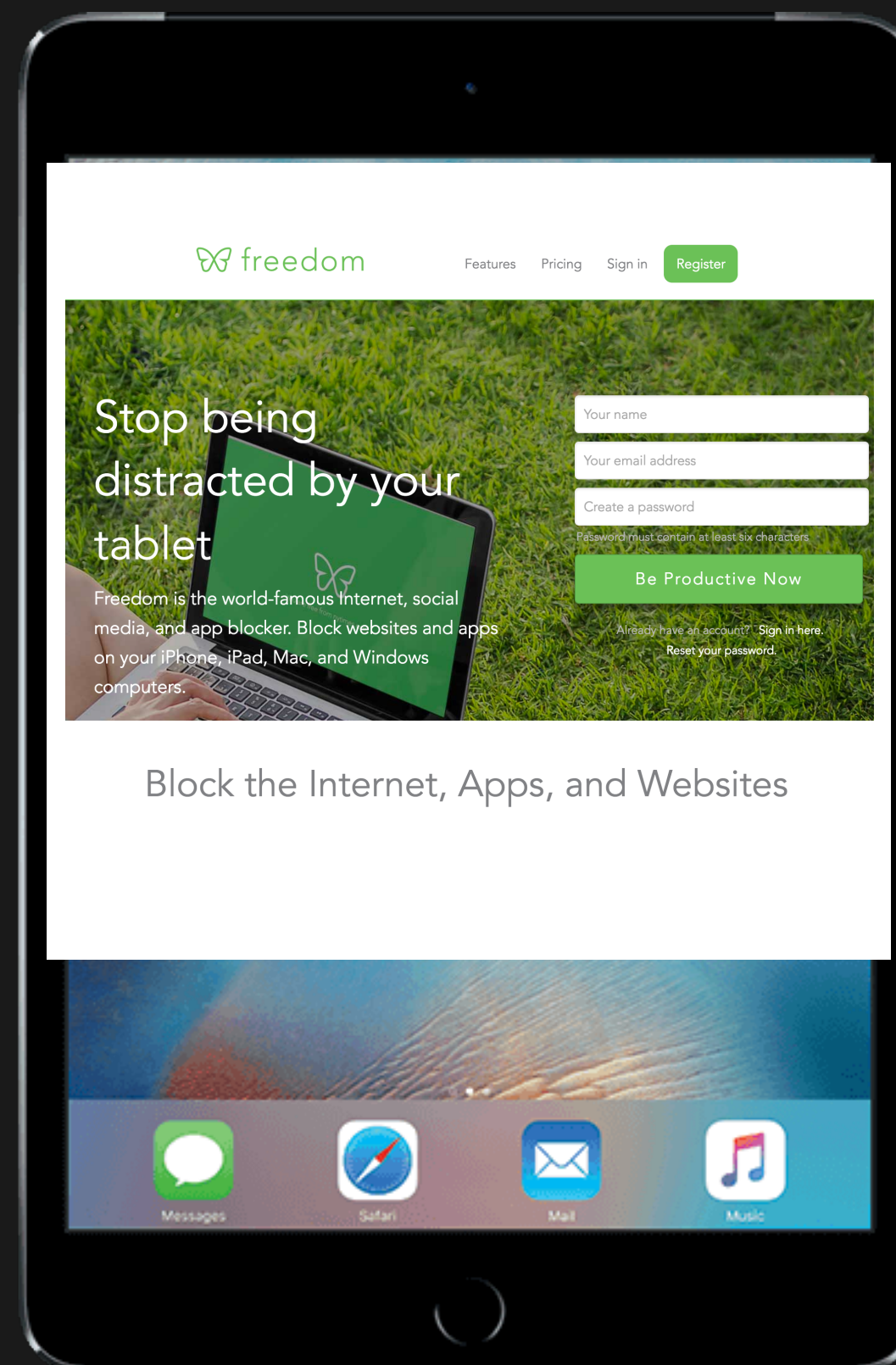
# CSS & mobile



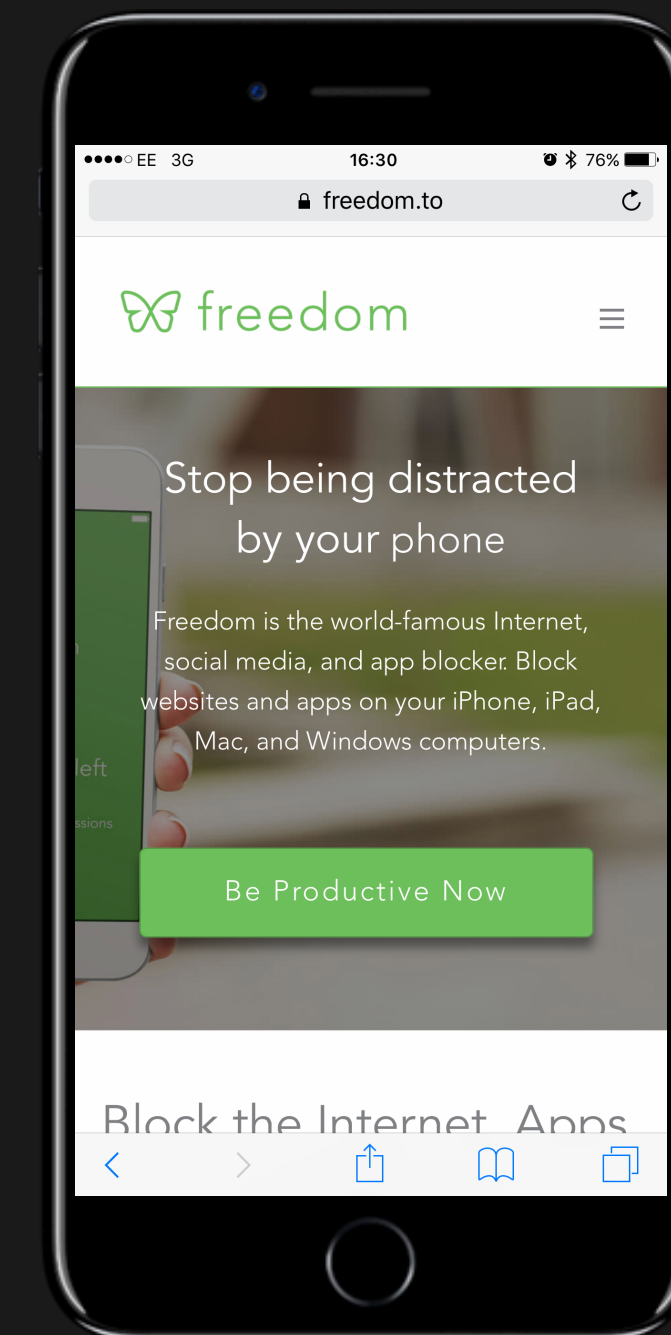




Desktop



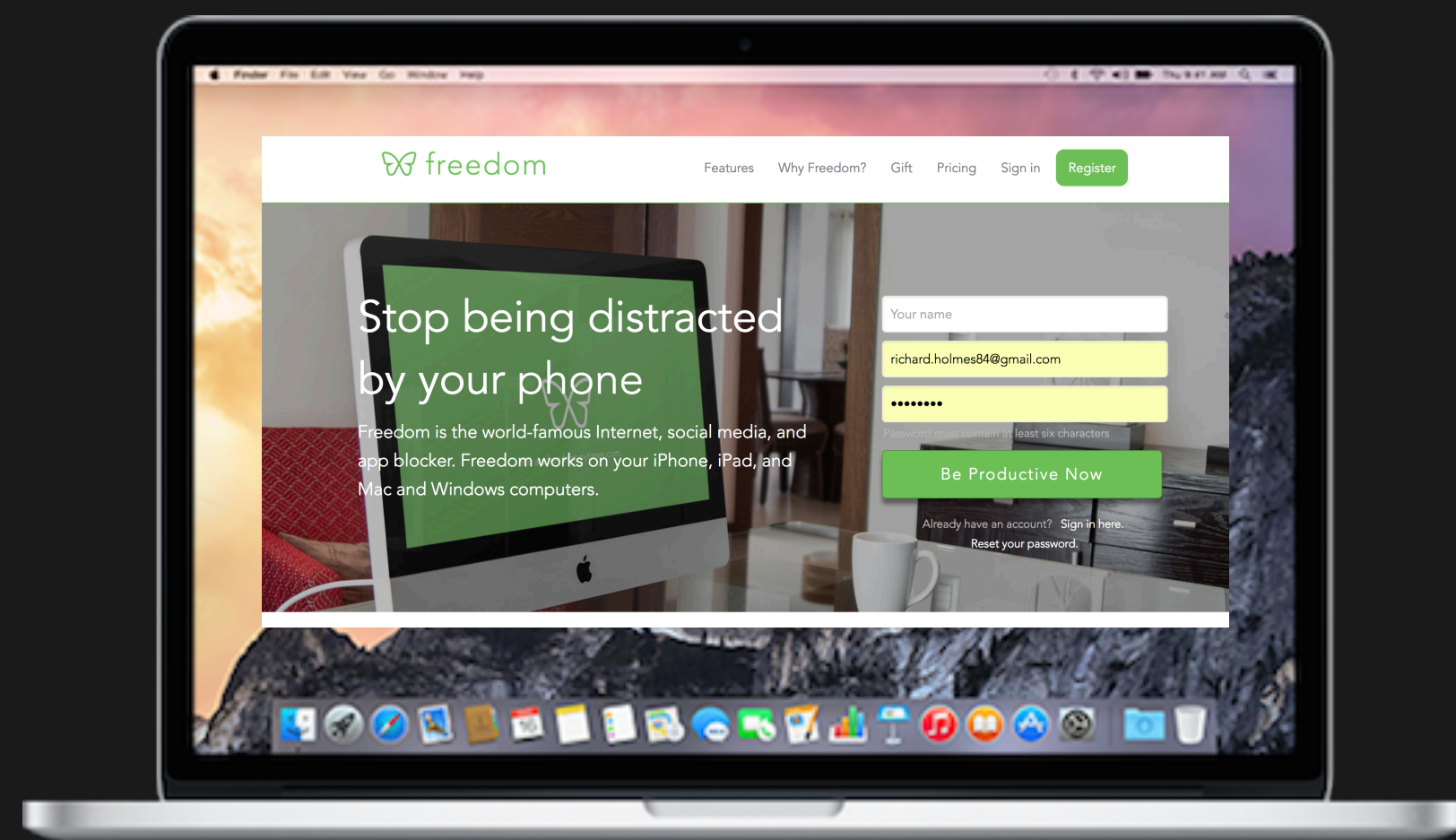
Tablet



Mobile



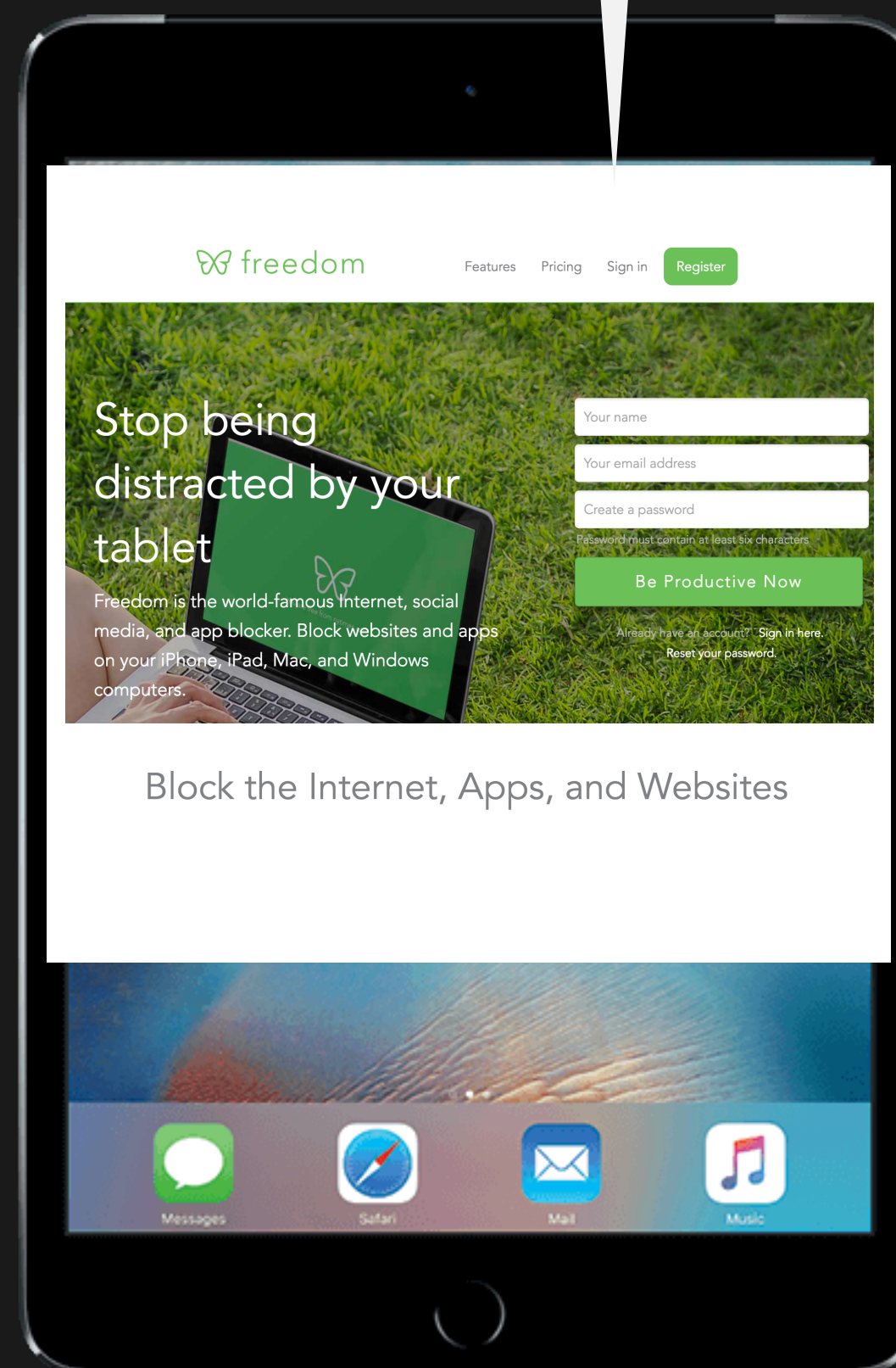




Desktop

If device is desktop,  
apply styles 1

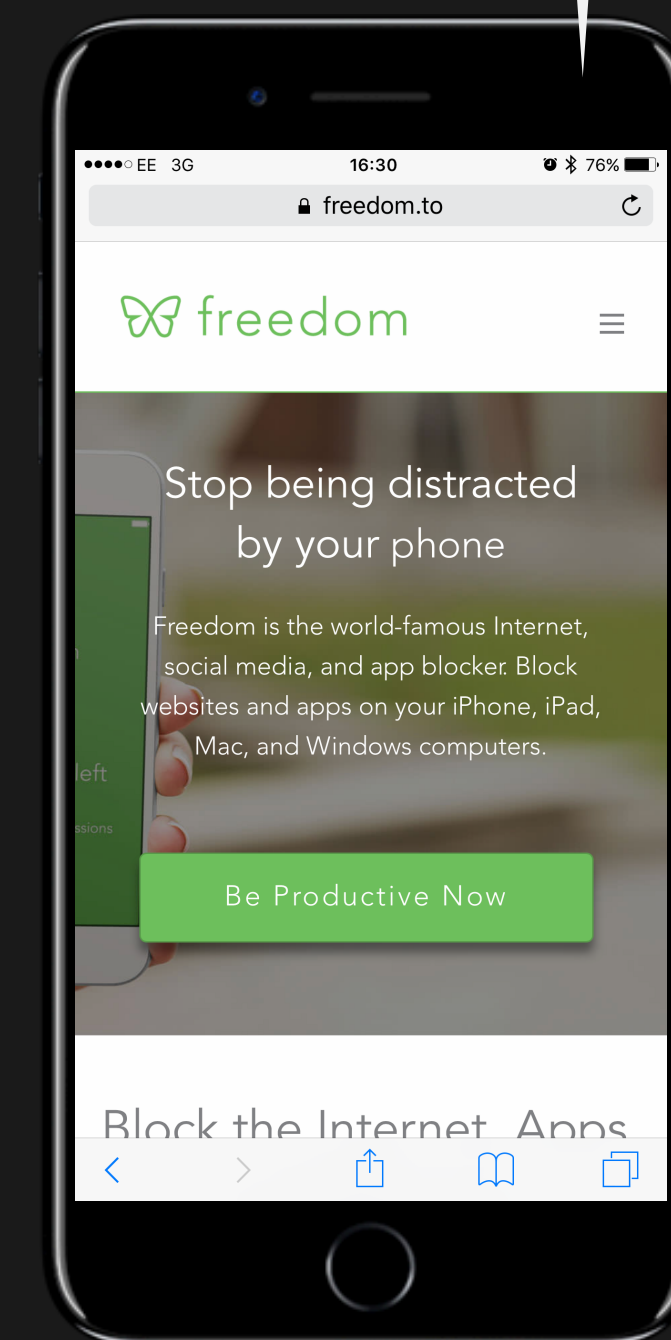
Remove navigation elements



Tablet

If device is tablet,  
apply styles 2

Use navigation hamburger

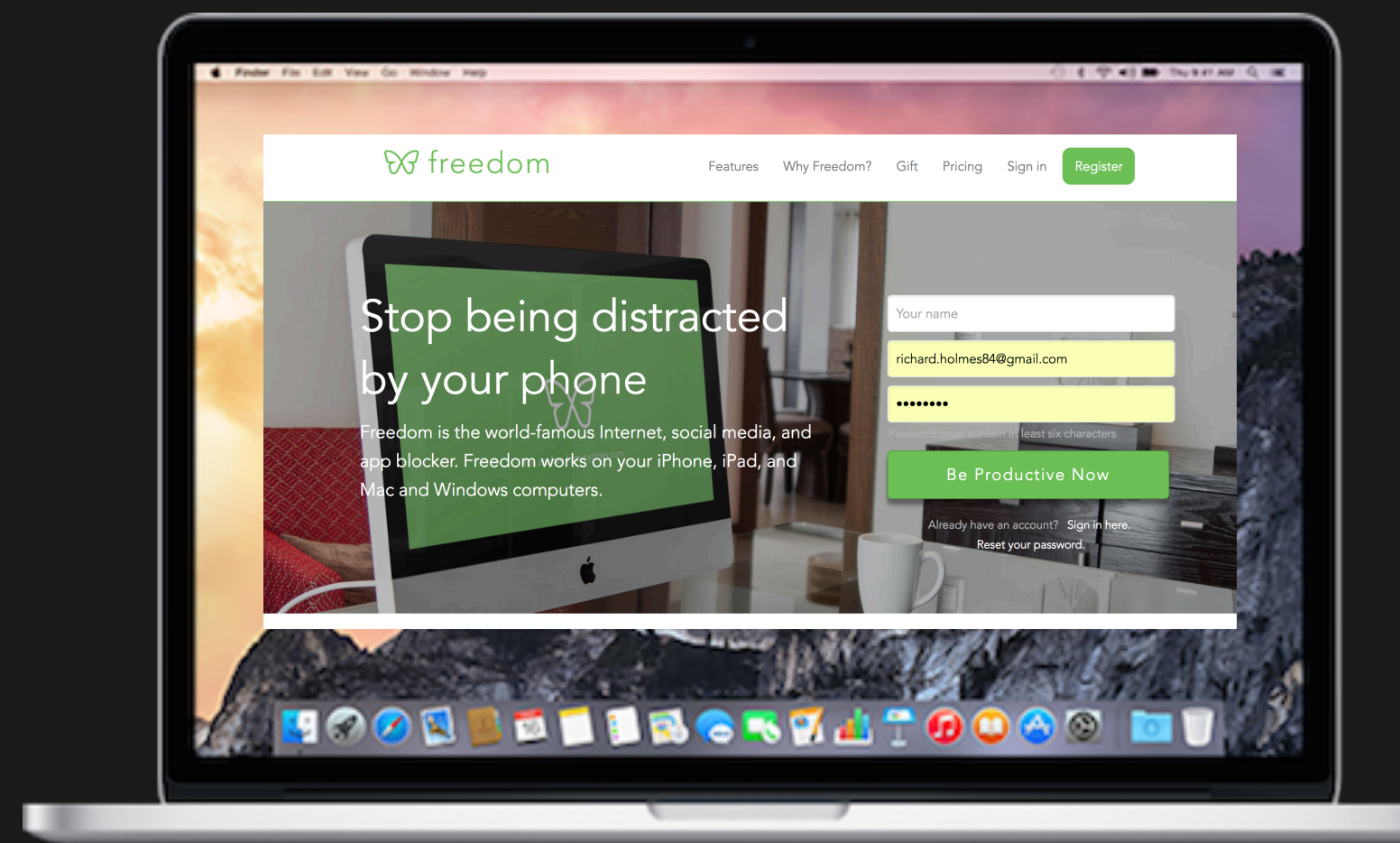


Mobile

If device is phone,  
apply styles 3



## Examples of media queries



Desktop

```
/* ----- iPad 3 and 4 ----- */
/* Portrait and Landscape */
@media only screen
  and (min-device-width: 768px)
  and (max-device-width: 1024px)
  and (-webkit-min-device-pixel-ratio: 2) {

}

/* Portrait */
@media only screen
  and (min-device-width: 768px)
  and (max-device-width: 1024px)
  and (orientation: portrait)
  and (-webkit-min-device-pixel-ratio: 2) {

}

/* Landscape */
@media only screen
  and (min-device-width: 768px)
  and (max-device-width: 1024px)
  and (orientation: landscape)
  and (-webkit-min-device-pixel-ratio: 2) {

}
```

Tablet

```
/* ----- iPhone 6 ----- */

/* Portrait and Landscape */
@media only screen
  and (min-device-width: 375px)
  and (max-device-width: 667px)
  and (-webkit-min-device-pixel-ratio: 2) {

}

/* Portrait */
@media only screen
  and (min-device-width: 375px)
  and (max-device-width: 667px)
  and (-webkit-min-device-pixel-ratio: 2)
  and (orientation: portrait) {

}

/* Landscape */
@media only screen
  and (min-device-width: 375px)
  and (max-device-width: 667px)
  and (-webkit-min-device-pixel-ratio: 2)
  and (orientation: landscape) {

}
```

Mobile





# CSS syntax



# CSS syntax

## 1. Selector

*What are you selecting? What element do you want to apply CSS to e.g. H1.*

h1 {

color: red;

## 2. Property

*What's the property?  
What are you trying to change?  
E.g. 'color'*

## 3. Value

*What's the value?  
Properties have specific values that are applicable e.g. red.*

}



# 1. Selectors





# Selectors

| Selector | Example                                              |
|----------|------------------------------------------------------|
| Element  | <pre>h1 {<br/>  property: value<br/>}</pre>          |
| ID       | <pre>#box {<br/>  property: value<br/>}</pre>        |
| Class    | <pre>.center {<br/>  text-align: center;<br/>}</pre> |



# Live example

*Using element selectors to make elements red*

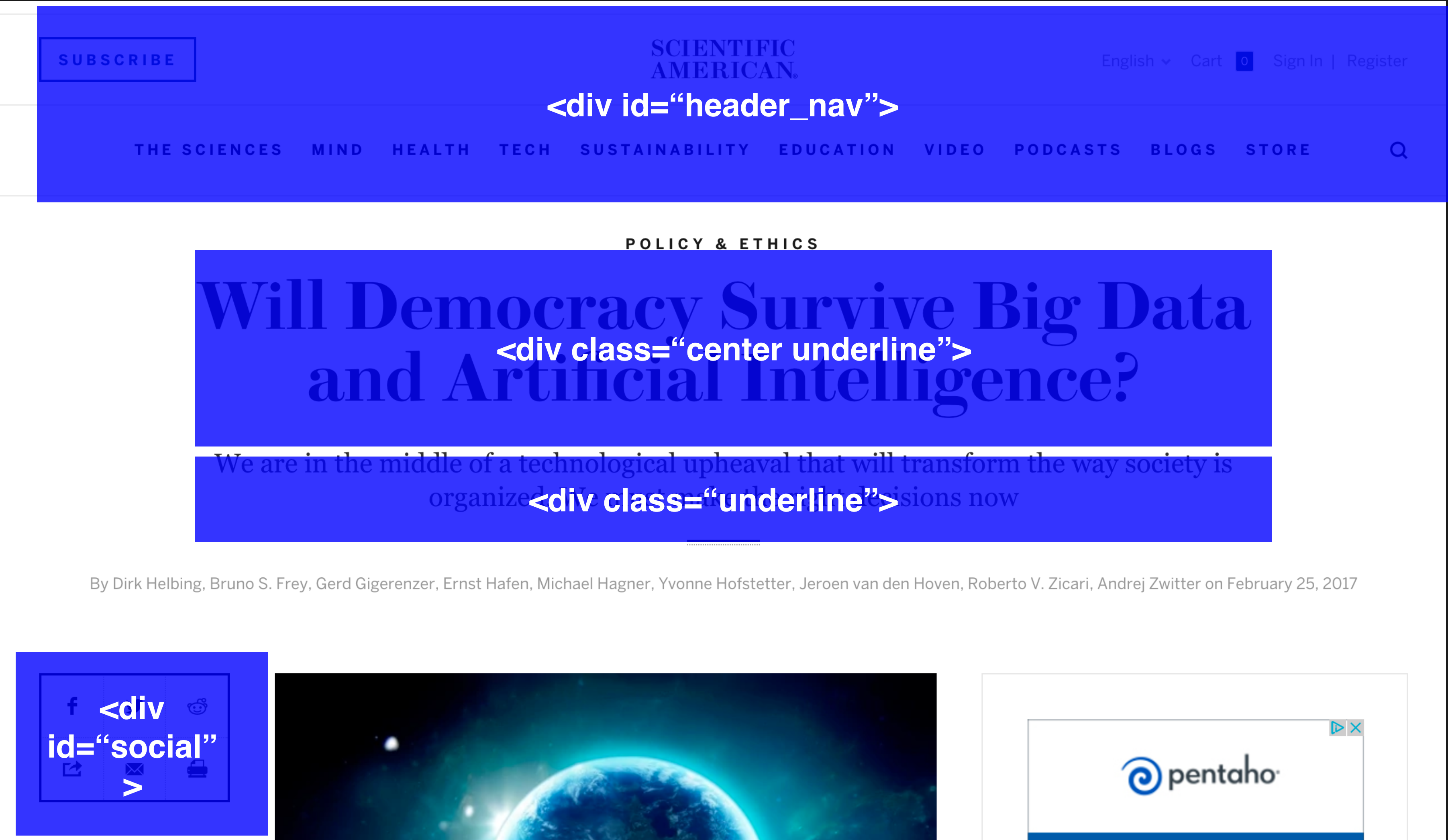




# Divs & CSS



# Divs & CSS



# ID selectors



The diagram consists of a large white square representing an HTML document. Inside this square, centered, is a smaller blue rectangle. Within the blue rectangle, the text `<div id =“box”>` is written in white.

```
<div id =“box”>
```

HTML



The diagram consists of a large dark gray square representing a CSS stylesheet. Inside this square, the text `#box` is written in a reddish-orange color.

```
#box
```

CSS stylesheet





# Class selectors

*For re-usable styles*



# Class selectors

```
<div class =“center”>
```

```
<h2 class =“center  
underline blue”>
```

HTML

## Stylesheet

```
.center {  
text-align: center;  
}  
  
.underline {  
text-decoration: underline;  
}  
  
.blue {  
color: blue;  
}
```

CSS stylesheet





# Live examples

*Divs & CSS working together*



## 2. Property



# CSS syntax

## 1. Selector

*What are you selecting? What element do you want to apply CSS to e.g. H1.*

h1 {

color: red;

## 2. Property

*What's the property?  
What are you trying to change?  
E.g. 'color'*

## 3. Value

*What's the value?  
Properties have specific values that are applicable e.g. red.*

}





# CSS properties

*What's useful to learn?*

`Font-family` (*chooses the font*)

`Font-size` (*chooses the font size*)

`Text-align` (*chooses text orientation*)

`Background-color` (*sets the background color*)



# 3. Value



# CSS syntax

## 1. Selector

*What are you selecting? What element do you want to apply CSS to e.g. H1.*

h1 {

color: red;

## 2. Property

*What's the property?  
What are you trying to change?  
E.g. 'color'*

## 3. Value

*What's the value?  
Properties have specific values that are applicable e.g. red.*

}





# Resources

| Property                     | Description                                                                       | Values                                                       | Example                                                         |
|------------------------------|-----------------------------------------------------------------------------------|--------------------------------------------------------------|-----------------------------------------------------------------|
| <code>color</code>           | The color of the element                                                          | #RRGGBB (Red, Green, Blue hex values)                        | <pre>h1 {   color: red; }</pre>                                 |
| <code>text-align</code>      | The alignment of an element, usually text                                         | left   right   center   justify                              | <pre>p {   text-align: center; }</pre>                          |
| <code>text-decoration</code> | Decorates the text with a variety of funky decorations. Most common is underline. | none   underline   overline   line-through   blink   inherit | <pre>p {   text-decoration: underline; }</pre>                  |
| <code>text-transform</code>  | Transforms text - often into uppercase.                                           | none   capitalize   uppercase   lowercase                    | <pre>h2 {   text-transform: uppercase; }</pre>                  |
| <code>line-height</code>     | The height between lines in text                                                  | px, %                                                        | <pre>p {   line-height: 1%; }</pre>                             |
| <code>letter-spacing</code>  | Defines the space between letters in text                                         | normal or px value                                           | <pre>p {   letter-spacing: 2px; }</pre>                         |
| <code>font-family</code>     | Defines the font family for text                                                  | font or font-family [, font or font-family ...]              | <pre>p {   font-family: Arial, Helvetica, Sans   Serif; }</pre> |
| <code>font-size</code>       | Defines the size of a font for text                                               | px or em value                                               | <pre>p {   font-size: 12px; }</pre>                             |
|                              |                                                                                   |                                                              | <pre>p {   font-weight: normal; }</pre>                         |

CSS cheat sheet



# Live examples

*CSS properties and values*



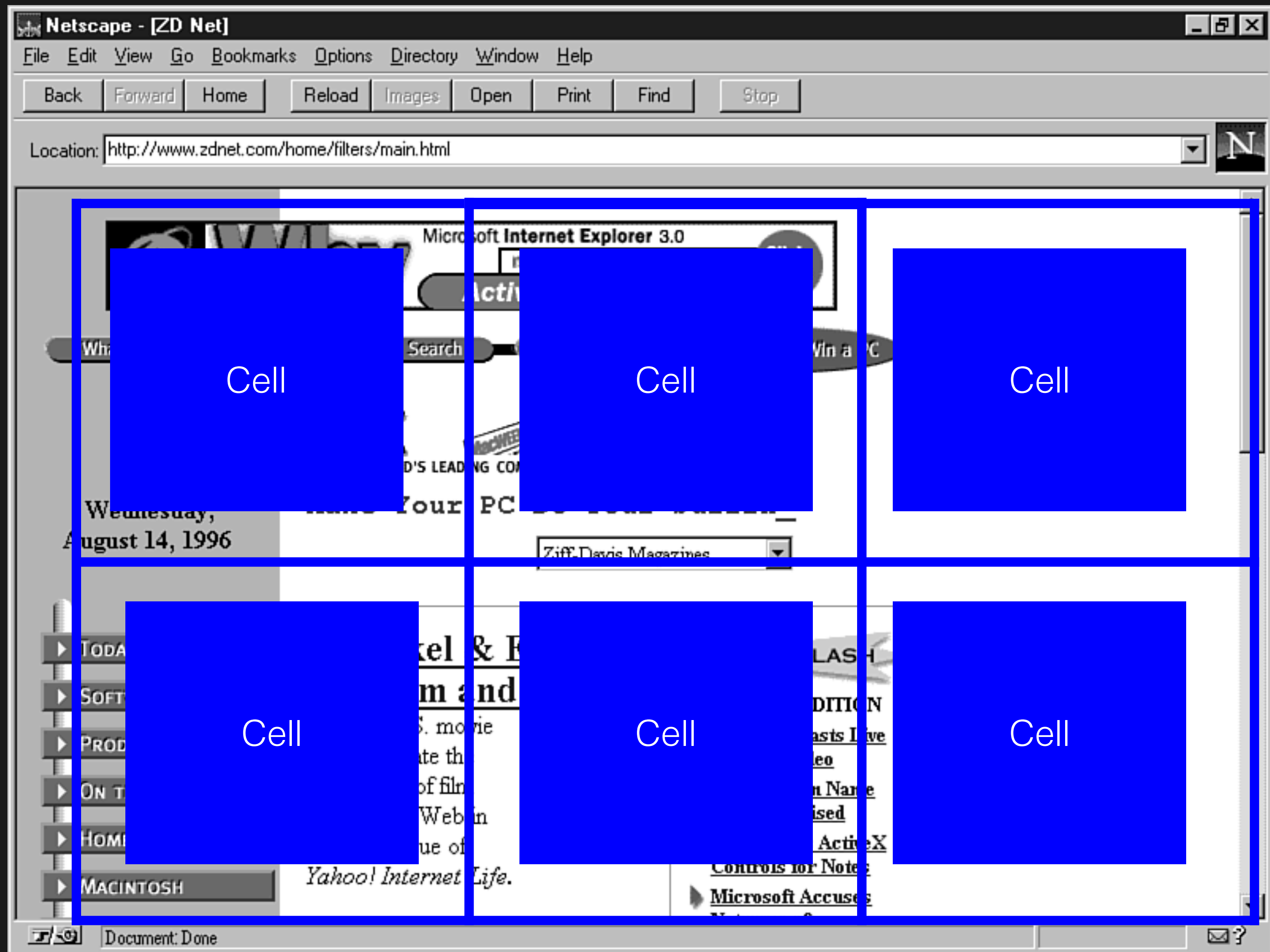
# The 2 functions of CSS

**Applying styles**

**Positioning  
elements**







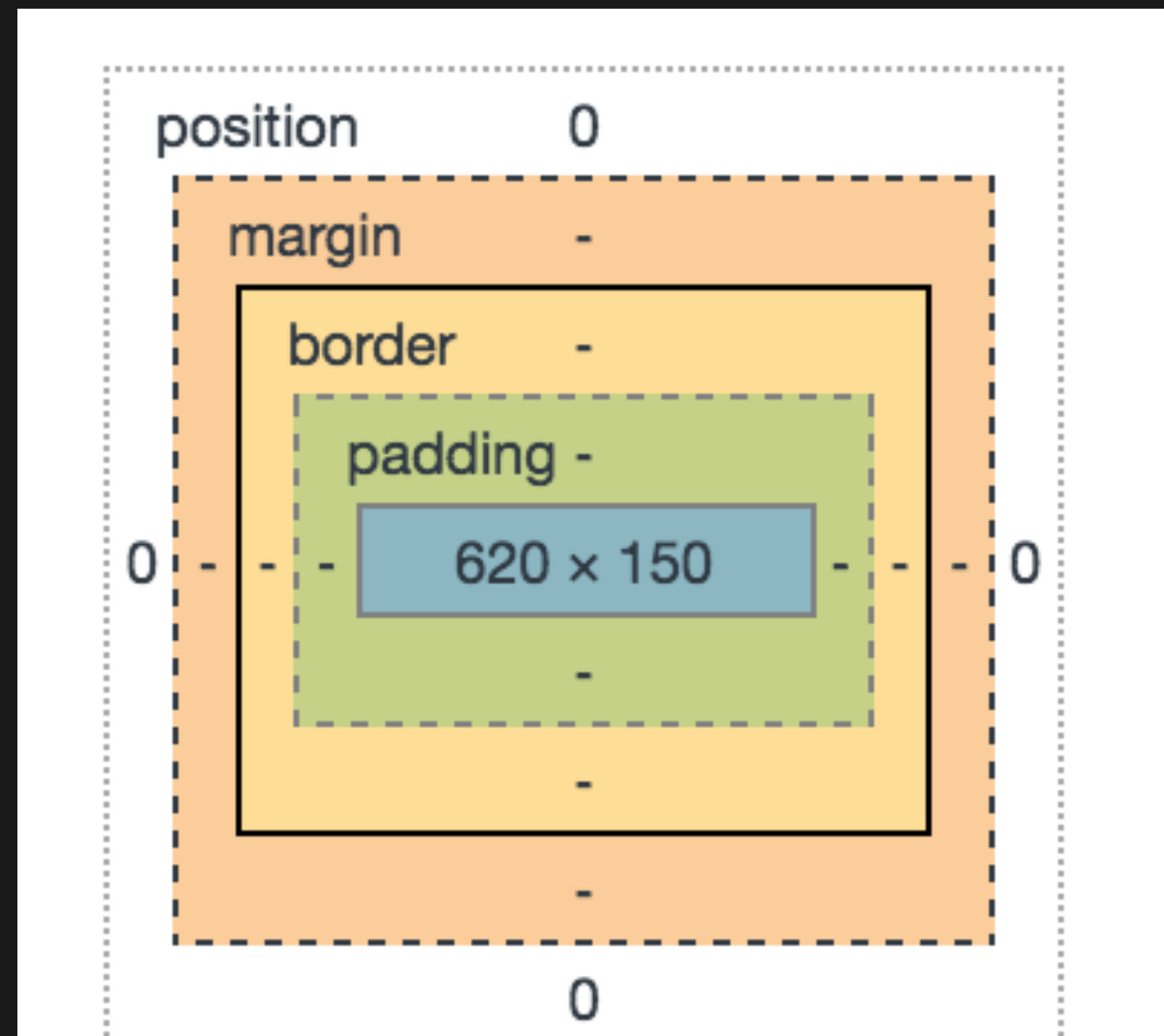


# The box model

[illegible]



# The box model



# Live examples

*The box model in action*



# CSS grids and frameworks

*Shiny, new toys = shiny, happy developers*









## Bootstrap Grid System

Bootstrap's grid system allows up to 12 columns across the page.

If you do not want to use all 12 columns individually, you can group the columns together to create wider columns:

|         |        |        |        |        |        |        |        |        |        |        |        |
|---------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| span 1  | span 1 | span 1 | span 1 | span 1 | span 1 | span 1 | span 1 | span 1 | span 1 | span 1 | span 1 |
| span 4  |        |        |        | span 4 |        |        |        | span 4 |        |        |        |
| span 4  |        |        |        | span 8 |        |        |        |        |        |        |        |
| span 6  |        |        |        |        |        | span 6 |        |        |        |        |        |
| span 12 |        |        |        |        |        |        |        |        |        |        |        |

Bootstrap's grid system is responsive, and the columns will re-arrange depending on the screen size: On a big screen it might look better with the content organized in three columns, but on a small screen it would be better if the content items were stacked on top of each other.







## Contextual backgrounds

Similar to the contextual text color classes, easily set the background of an element to any contextual class. Anchor components will darken on hover, just like the text classes.

### EXAMPLE

Nullam id dolor id nibh ultricies vehicula ut id elit.

Duis mollis, est non commodo luctus, nisi erat porttitor ligula.

Maecenas sed diam eget risus varius blandit sit amet non magna.

Etiam porta sem malesuada magna mollis euismod.

Donec ullamcorper nulla non metus auctor fringilla.

```
<p class="bg-primary">...</p>
<p class="bg-success">...</p>
<p class="bg-info">...</p>
<p class="bg-warning">...</p>
<p class="bg-danger">...</p>
```

## Single button dropdowns

Turn a button into a dropdown toggle with some basic markup changes.

### EXAMPLE

Default ▾

Primary ▾

Success ▾

Info ▾

Warning ▾

Danger ▾

```
<!-- Single button -->
<div class="btn-group">
  <button type="button" class="btn btn-default dropdown-toggle" data-toggle="dropdown" aria-
haspopup="true" aria-expanded="false">
    Action <span class="caret"></span>
  </button>
  <ul class="dropdown-menu">
    <li><a href="#">Action</a></li>
    <li><a href="#">Another action</a></li>
    ... <a href="#">Something else here</a></li>
    <li role="separator" class="divider"></li>
    <a href="#">Separated link</a></li>
```

Copy





# A brief recap

1. CSS syntax specifies 3 parts: the *selector*, the *property* and the *value*
2. Over 40 CSS properties are available; each has a specific value
3. CSS positioning uses the *box model* and *floats* to achieve table-free layouts

