



Au Mau Duong
Technical Trainers

CSS Selectors and Rules

Primary Selectors, Nested Selectors,
Attribute Selectors, Pseudo Selectors



Table of Contents

1. Primary Selectors
2. Nested Selectors
3. Attribute Selectors
4. Pseudo Selectors
5. Default Styles and Style Precedence

```
body { ... }  
font-family: Arial;  
  
margin: 2px; width: 150px;  
  
border: 1px solid blue;  
  
border: 0;  
background: #4ae871;  
  
position: absolute; left: 0;  
  
color: #2aa0bd;  
display: inline;
```

CSS Selectors

- CSS Selectors determine which element the rules apply to:
 - All elements of specific type (**tag**)
 - Those that match a specific attribute (**id, class**)
 - Elements may be matched depending on how they are nested in the document tree (HTML)
- Examples:

```
.header a { color: green }
```

```
#menu>li { padding-top: 8px }
```

Primary Selectors

- By tag (element selector):

```
h1 { font-family: Verdana, sans-serif; }  
p { margin: 0; color: #EEE; }
```

- By element ID:

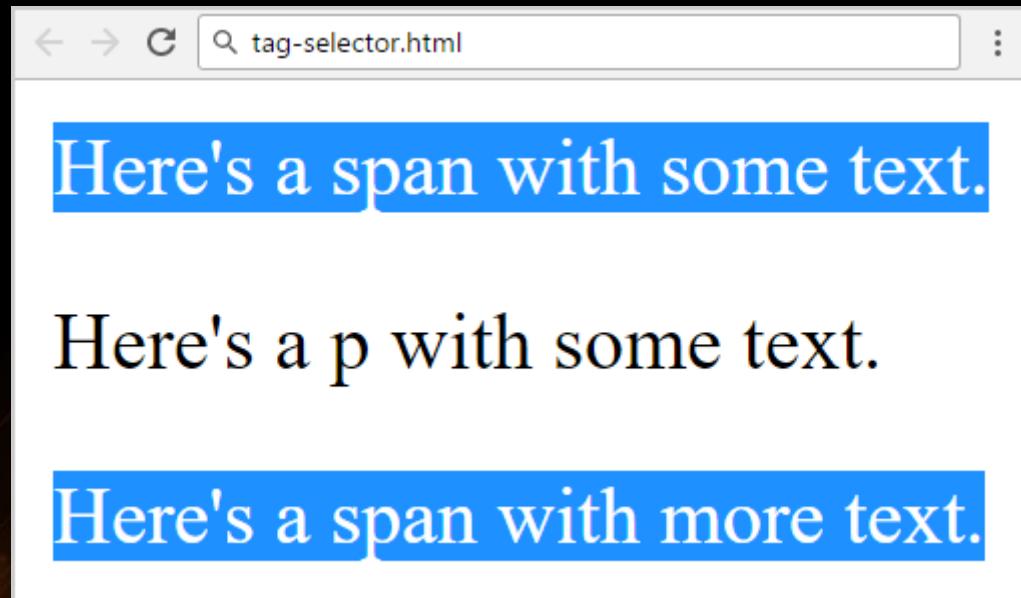
```
#element_id { color: #ff0000; }  
#wrapper { margin: 0 auto; }
```

- By class name (only for HTML):

```
.myClass { border: 1px solid red; }  
.special { font-weight: bold; color: yellow; }
```

Primary Selectors: Select by Tag

```
<span>Here's a span with some text.</span>
<p>Here's a p with some text.</p>
<span>Here's a span with more text.</span>
```

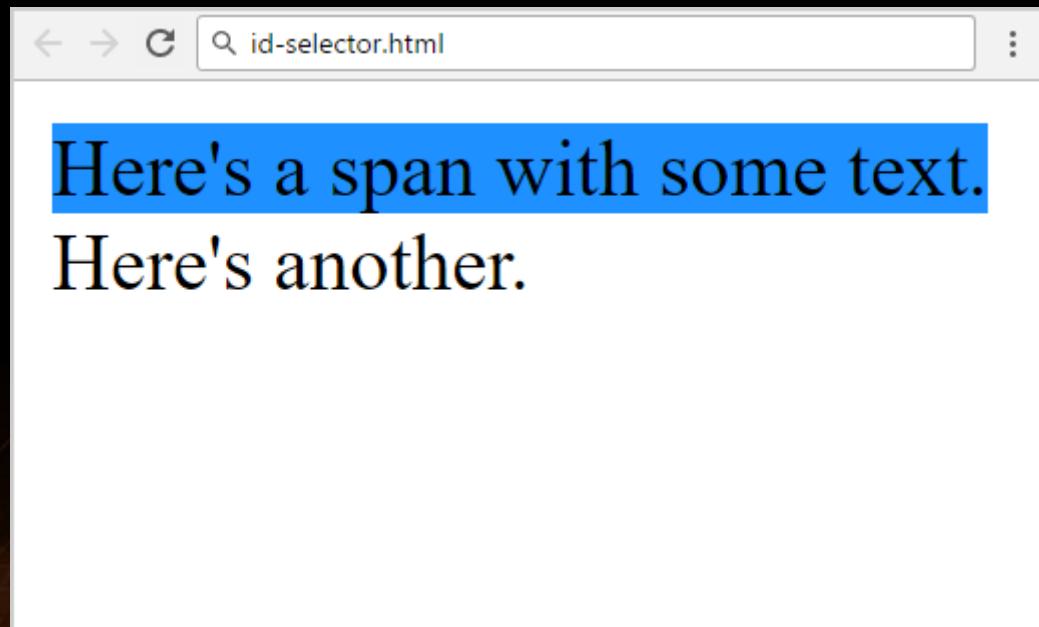


Select all
 elements

```
span {
  background: DodgerBlue;
  color: #ffffff;
}
```

Primary Selectors: Select by ID

```
<span id="top">Here's a span with some text.</span>
<span>Here's another.</span>
```

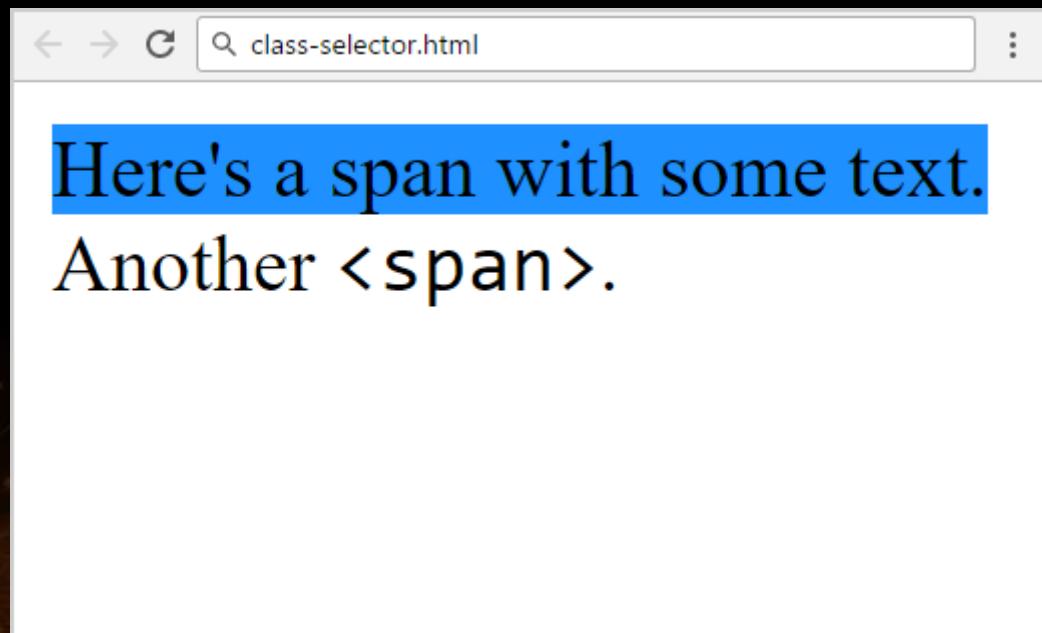


Select ``
with `id="top"`

```
span#top {
    background: DodgerBlue;
}
```

Primary Selectors: Select by Class

```
<span class="sky">Here's a span with some text.</span>
<span>
  Another <span class="code">&lt;span&gt;</span>.
</span>
```



```
span.sky {
  background: DodgerBlue;
}
.code {
  font-family: Consolas;
}
```

span with
class="sky"

Elements with
class="code"

Nested Selectors (1)

- Match relative to element placement
 - Matches direct and indirect child elements:

```
p.item a { text-decoration: underline }
```

- This matches all `<a>` tags that are inside of `<p class="item">`
- * – universal (wildcard) selector (avoid or use with care!)
 - This matches all descendants of the `<p>` element:

```
p * { color: black }
```

- Matching all elements in the page:

```
* { background: #E5E5E5 }
```

Nested Selectors (2)

- **+** selector – used to match “next sibling”:

```
img + .link { float: right }
```

- Matches all siblings with class name **link** after **** element
- **>** selector – matches direct child nodes:

```
p > .error { font-size: 8px }
```

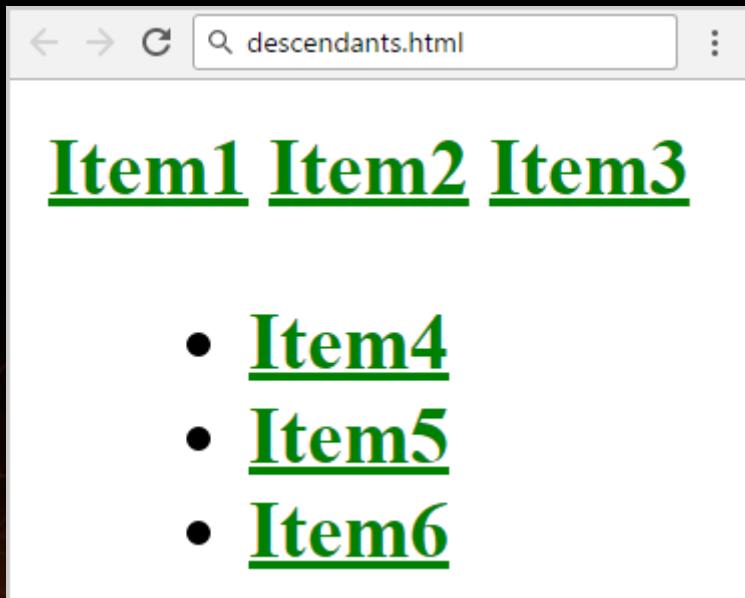
- Matches all elements with class **error**, direct children of **<p>**
- **.class1.class2** (no space!)
- Matches elements with both (all) classes applied at the same time

```
p.post-text.special { font-weight: bold }
```

Nested Selectors: Descendant

```
div.items a {  
    color: green;  
    font-weight: bold;  
}
```

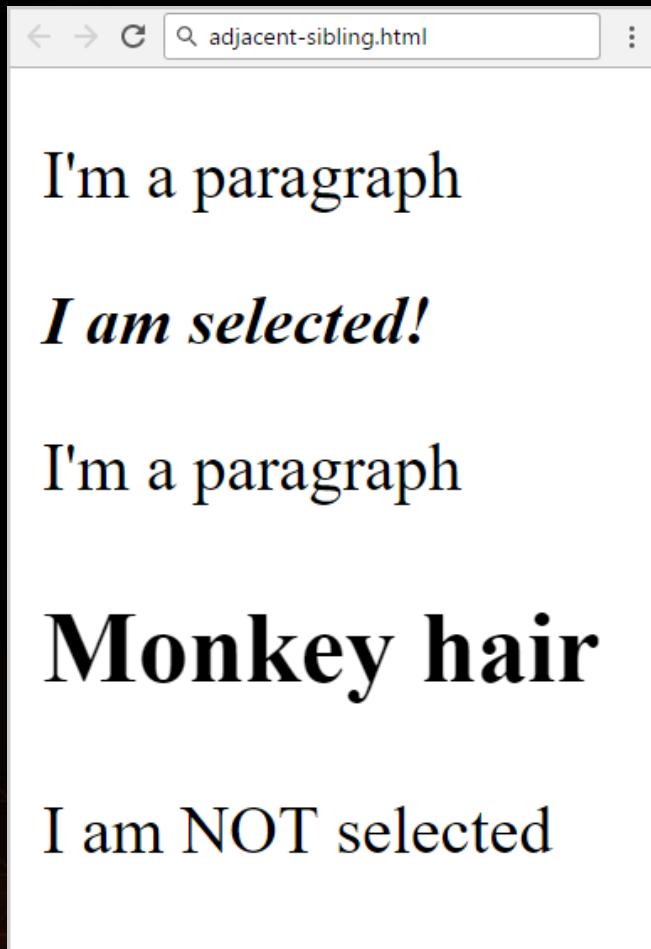
<a> inside <div class="items">



```
<div class="items">  
    <a href="#">Item1</a>  
    <a href="#">Item2</a>  
    <a href="#">Item3</a>  
    <ul>  
        <li><a href="#">Item4</a></li>  
        <li><a href="#">Item5</a></li>  
        <li><a href="#">Item6</a></li>  
    </ul>  
</div>
```

Nested Selectors: Adjacent Sibling

<p> coming after <p>



```
p + p {  
    font-style: italic;  
    font-weight: bold;  
}
```

```
<div>  
    <p>I'm a paragraph</p>  
    <p>I am selected!</p>  
</div>  
<div>  
    <p>I'm a paragraph</p>  
    <h2>Monkey hair</h2>  
    <p>I am NOT selected</p>  
</div>
```

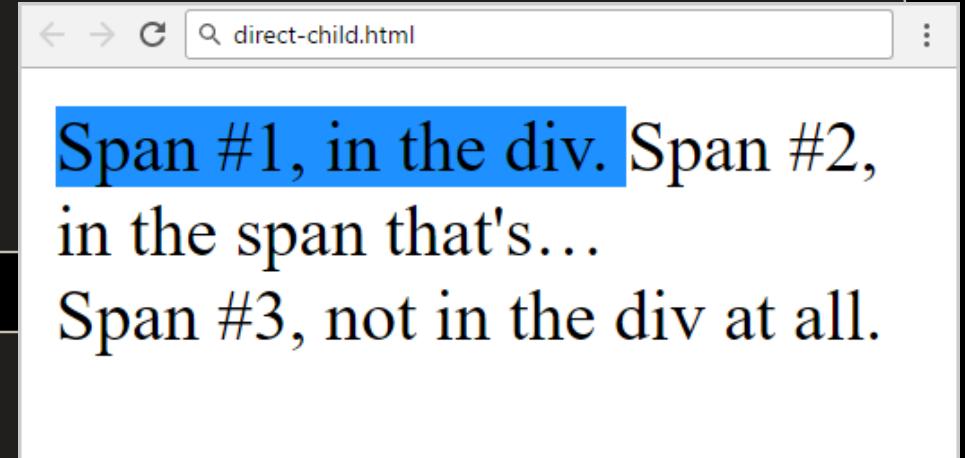
Nested Selectors: Direct Child

```
div > span {  
    background: DodgerBlue;  
}  
  
span { background: #fff; }
```

 directly contained in a <div>

```
<div>  
    <span>Span #1, in the div.  
    <span>Span #2, in the span that's...</span>  
    </span>  
</div>  
<span>Span #3, not in the div at all.</span>
```

Direct child of <div>



Nested Selectors: Multiple Classes



```
.apple { color: red }
.orange { color: orange }
.small { font-size: 16px }
.apple.orange {
  font-style: italic;
}
```

Element with
multiple classes

Elements with
both classes

```
<h2 class="apple orange small">Apple + Orange</h2>
<h2 class="apple">Apple</h2>
<h2 class="orange">Orange</h2>
```

Attribute Selectors

- **[]** selects elements based on attributes
 - Selects **<a>** elements which has **title** attribute:

```
a[title] { color: black }
```

- Selects **<input>** elements with **type='text'**

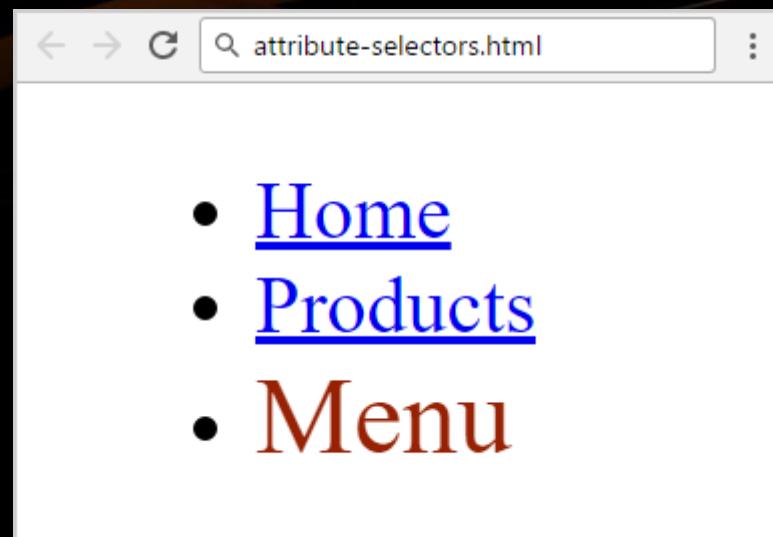
```
input[type='text'] { font-family: Consolas }
```

```
#reg-form input[type='text'] { background: #EEE }
```

- Selects **<a>** elements whose **title** contains the word "**logo**":

```
a[title*=logo] { border: none }
```

Attribute Selectors



Attribute
title="menu"

```
a[title="menu"] {  
    text-decoration: none;  
    color: #962103;  
    font-size: 22px;  
}
```

```
<ul>  
    <li><a href="#">Home</a></li>  
    <li><a href="#">Products</a></li>  
    <li><a href="#" title="menu">Menu</a></li>  
</ul>
```

Combined CSS Selectors

- Selectors can be combined with commas:

```
h1, .link, #top-link { font-weight: bold; }
```

- This matches **<h1>** tags, elements with class "**link**", and the element with id "**top-link**"
- Resetting the browsers default margins and paddings:

```
html, body, div, h1, ul, li, a, img, span, form, legend,  
input, button, textarea, fieldset {  
    margin: 0;  
    padding: 0;  
}
```

Multiple Selectors (Element, Element)



```
<h1>Welcome...</h1>
<h2>My name is...</h2>
<p>I live in Duckburg.</p>
<p>My best friend is...</p>
```

```
h1, h2, p {
    background: yellow;
}
```

Combining Multiple Selectors

- We can combine selectors to achieve more specific rules

```
<h1 id="header" class="intro">HTML and CSS</h1>
```



```
h1#header.intro:hover {  
    text-decoration: underline;  
    color: #c00;  
}
```

- Don't put spaces between combined selectors
- "A space B" means B descendant of A; AB means "A and B"

Problem: Photo Shoot Effect

- You are given a HTML file
- Create a Web page like the screenshot on the right
- Constraints:
 - You are **NOT** allowed to change the **HTML** file
 - Just write the missing **CSS**



Solution: Photo Shoot Effect – CSS (1)

```
@import  
"https://fonts.googleapis.com/css?family=Lato:300,400,900";  
* {  
    margin: 0;  
    padding: 0;  
}  
  
body {  
    font-family: Arial, sans-serif;  
    color: #fcfcfc;  
    background-color: #355664;  
    width: 960px;  
    margin: 0 auto;  
}
```



Solution: Photo Shoot Effect – CSS (2)

```
h1, h2, h3 {  
    font-family:  
        forte, Arial, sans-serif; }  
  
h1 {  
    font-size: 50px;  
    padding: 10px 0 0;  
    text-align: center;  
    text-transform: uppercase}  
  
h2 {  
    font-size: 40px;  
    letter-spacing: 15px;  
    text-align: center;
```

```
text-transform: uppercase;  
border-bottom: 3px solid  
#416371;  
padding-bottom: 20px;}  
  
h3 {  
    text-align: right;  
    font-size: 30px;  
    font-weight: normal;  
    margin: 0 0 10px;}  
  
p {  
    line-height: 20px;  
    padding-bottom: 5px; }
```

Solution: Photo Shoot Effect – CSS (3)

```
figure + p {  
    color: #94c258;  
}  
  
img {
```

```
    border: none;  
    width: 500px;  
    height: 300px;  
}
```

```
div.article-content {  
    margin: 15px 0;  
}
```

```
nav {  
    background: #355664;  
    padding: 0 5px;  
}  
nav ul li {  
    display: inline-block;  
}
```



Solution: Photo Shoot Effect – CSS (4)

```
nav ul li a {  
    color: #ffffff;  
    display: inline-block;  
    font-size: 30px;  
    margin: 5px 2px;  
    padding: 7px 10px 4px;  
    text-transform: uppercase;}  
  
nav ul li:first-of-type > a {  
    color: #7CB3E9;  
    font-size: 30px;  
    text-transform: lowercase;  
}
```

```
nav ul a[title="sweet tabs"] {  
    color: #8EC45B;  
    font-size: 30px;  
    text-transform: none;  
}  
  
nav, article,  
nav ul li a, figure {  
    border-radius: 10px;  
}
```

Solution: Photo Shoot Effect – CSS (5)

```
article {  
background-color: #213E4A;  
margin: 0 0 10px;  
padding: 20px;  
height: 390px;}  
  
figure {  
border: 3px solid #142830;  
float: right;  
height: 300px;  
width: 500px;  
margin-left: 15px;  
}
```

```
footer {  
border-top: 3px solid #142830;  
padding-top: 15px;  
margin-bottom: 30px;  
text-align: center;  
font-size: 20px;  
}
```

Pseudo Selectors

Relative to Element Content or State

```
html, body, div { padding: 0; margin: 0; font-size: 100%; }

/* Start of "Micro clearfix" */

.cf { zoom: 1; }
.cf:before,
.cf:after { content: ""; display: table; }
.cf:after { clear: both; }
```

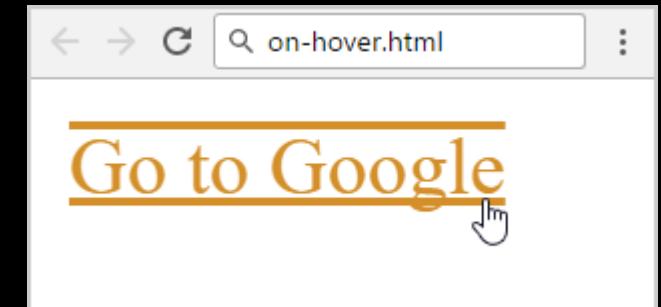
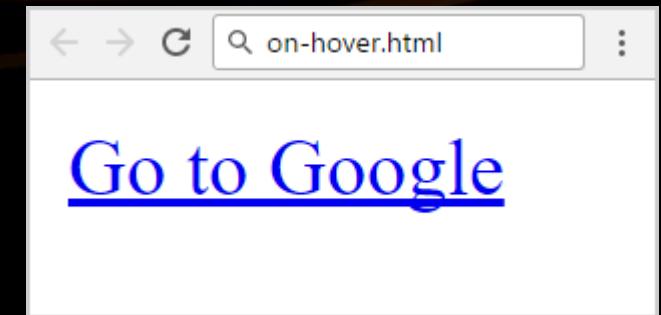
:before and :after

```
<head>
<body>
  <h1>Micro clearfix demo</h1>
  <div class="container cf">
    <div class="section">Float</div>
    <div class="section">Float</div>
  </div>
```

Pseudo Selector (:hover)

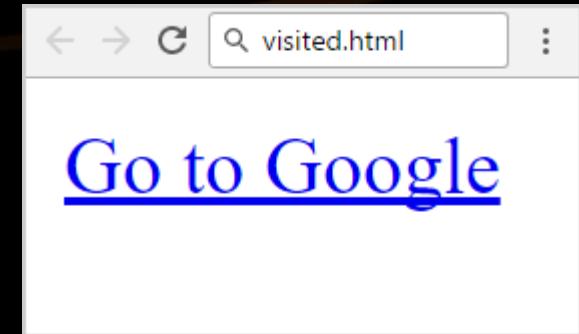
```
<a href="#">Go to Google</a>
```

```
a:hover {  
    color: #D48F29;  
    text-decoration:  
        underline overline;  
}
```

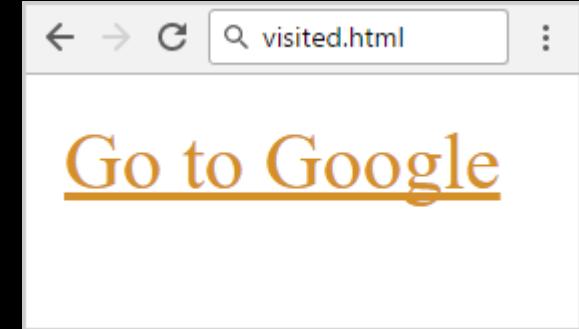


Pseudo Selector (:visited)

```
<a href="#">Go to Google</a>
```



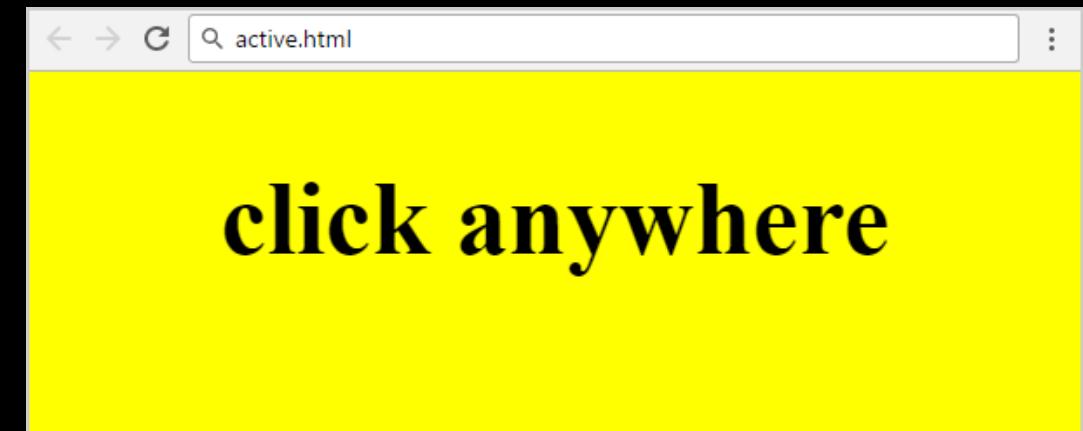
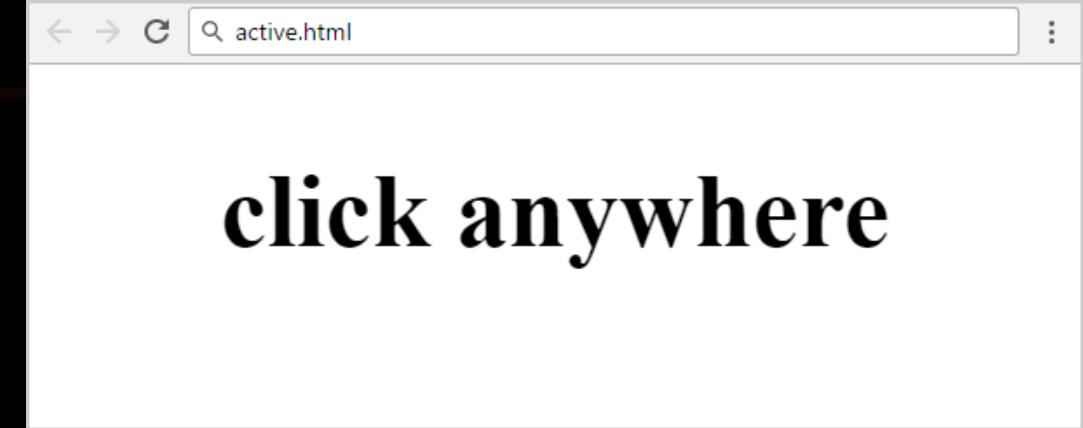
```
a:visited {  
    color:#D48F29;  
}
```



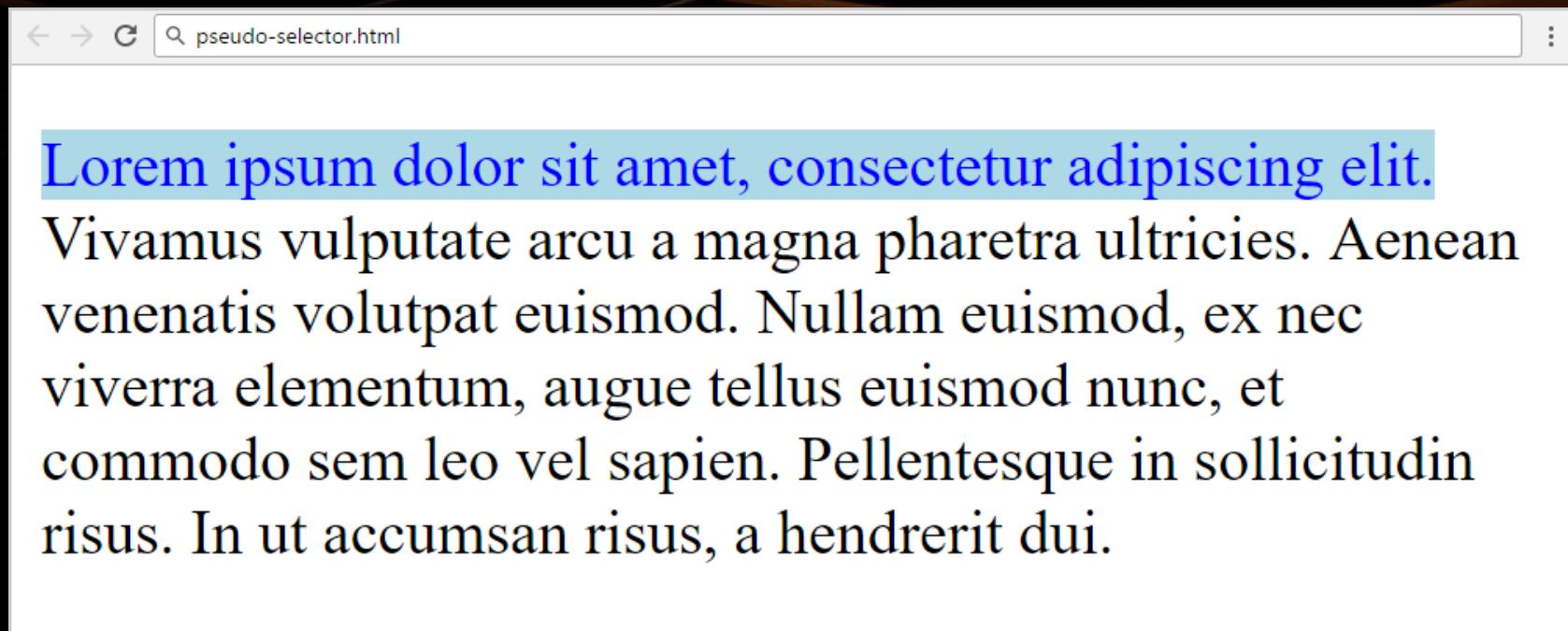
Pseudo Selector (:active)

```
<h2>click anywhere</h2>
```

```
html:active {  
    background: yellow;  
}
```



Pseudo Selector (:first-line)



```
<p>Lorem ipsum dolor  
sit amet, consectetur  
adipiscing elit.  
</p>
```

```
p:first-line {  
    color:blue;  
    background-color: lightblue;  
}
```

Pseudo Selector (::before and ::after)

<h1>CSS Pseudo Selectors</h1>

```
h1::before { /* "« " */  
  content: "\00AB\0020"  
}
```

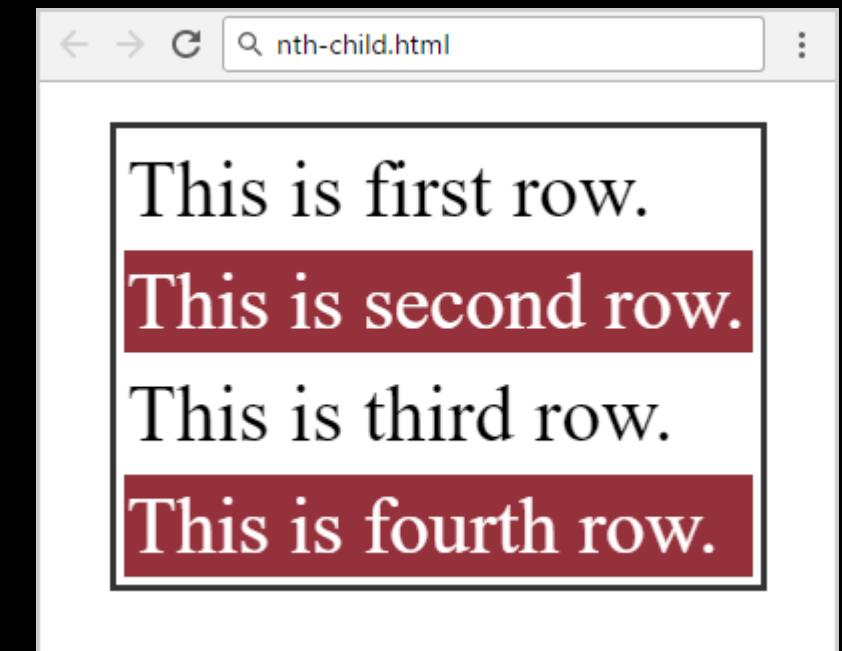
```
h1::after { /* " »" */  
  content: "\0020\00BB"  
}
```



Structural Pseudo-Classes: nth-child(n)

```
<tr><td>This is first row.</td></tr>
<tr><td>This is second row.</td></tr>
<tr><td>This is third row.</td></tr>
<tr><td>This is fourth row.</td></tr>
```

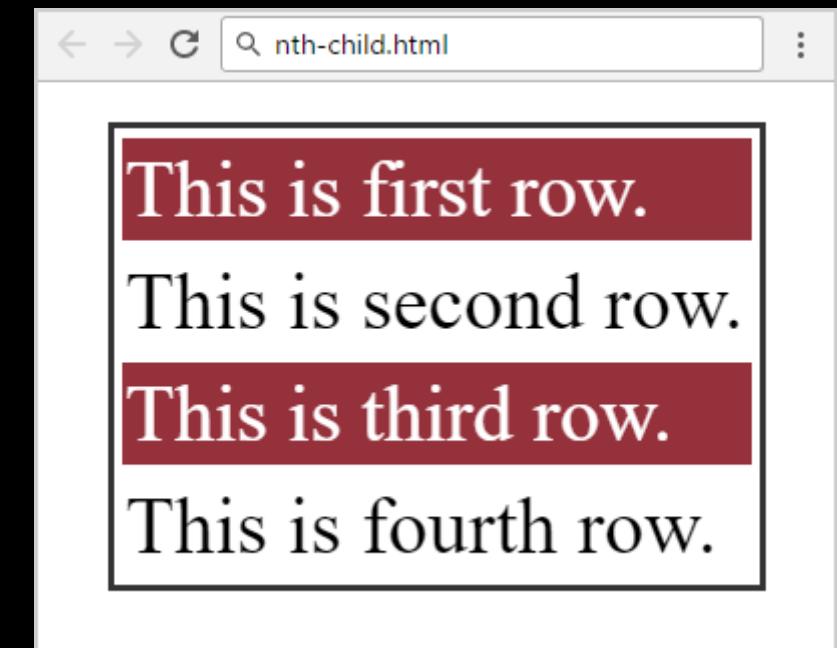
```
tr:nth-child(2n) {
    background: #95313b;
    color: #fff;
}
```



Structural Pseudo-Classes: nth-child(n+1)

```
<tr><td>This is first row.</td></tr>
<tr><td>This is second row.</td></tr>
<tr><td>This is third row.</td></tr>
<tr><td>This is fourth row.</td></tr>
```

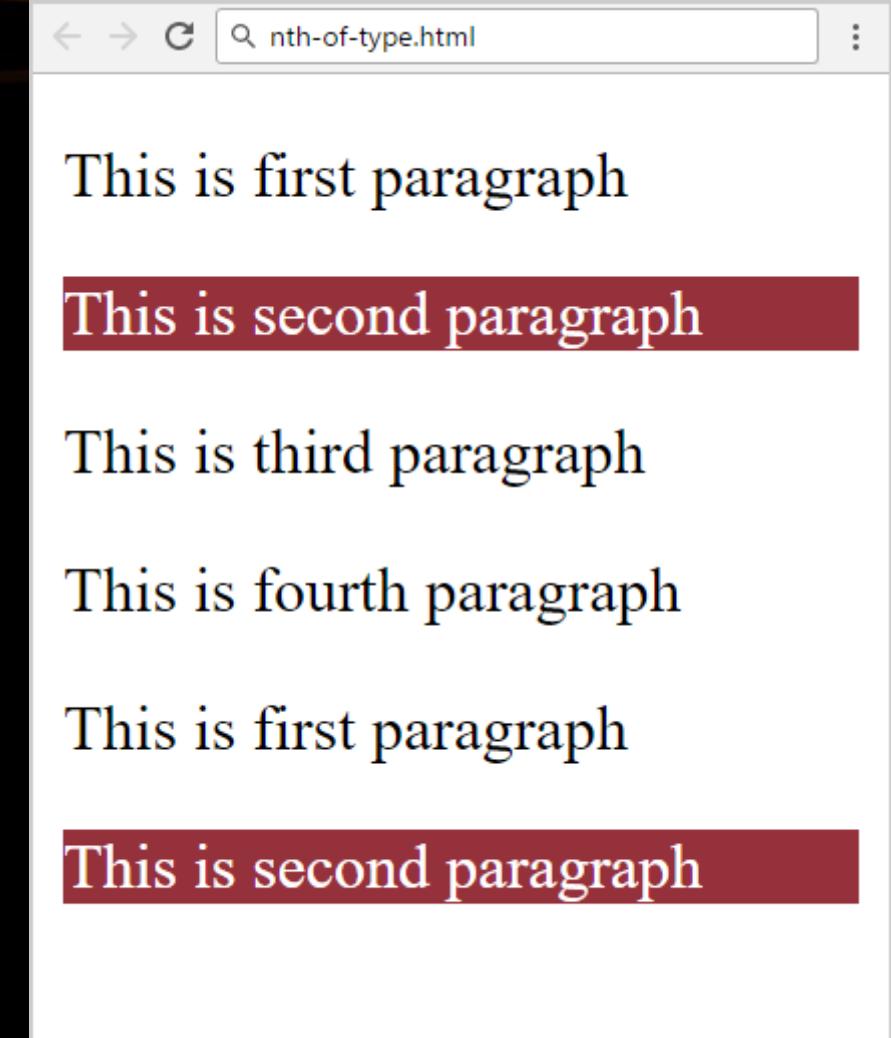
```
tr:nth-child(2n+1) {
    background: #95313b;
    color: #fff;
}
```



Structural Pseudo-Classes: nth-of-type(n)

```
<p>This is first paragraph</p>
<p>This is second paragraph</p>
<p>This is third paragraph</p>
<p>This is fourth paragraph</p>
<div>
  <p>This is first paragraph</p>
  <p>This is second paragraph</p>
</div>
```

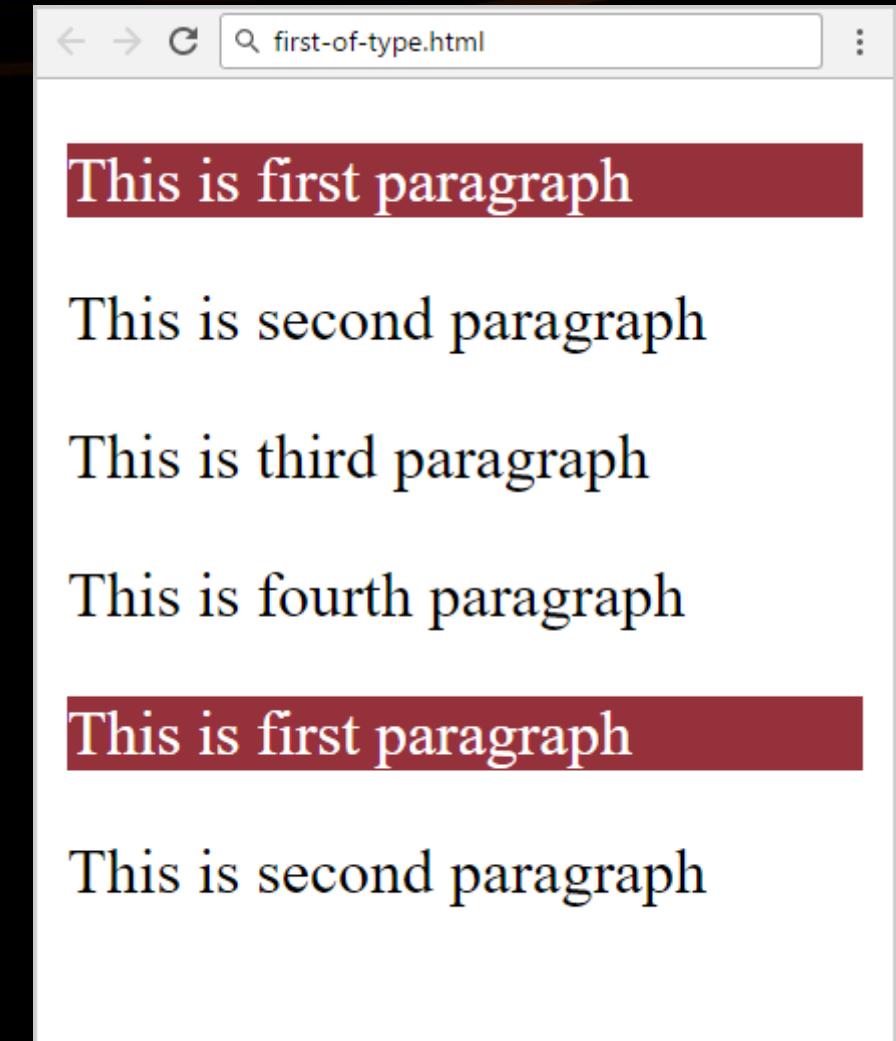
```
p:nth-of-type(2) {
  background: #95313b;
  color: #fff;
}
```



Structural Pseudo-Classes: first-of-type

```
<p>This is first paragraph</p>
<p>This is second paragraph</p>
<p>This is third paragraph</p>
<p>This is fourth paragraph</p>
<div>
    <p>This is first paragraph</p>
    <p>This is second paragraph</p>
</div>
```

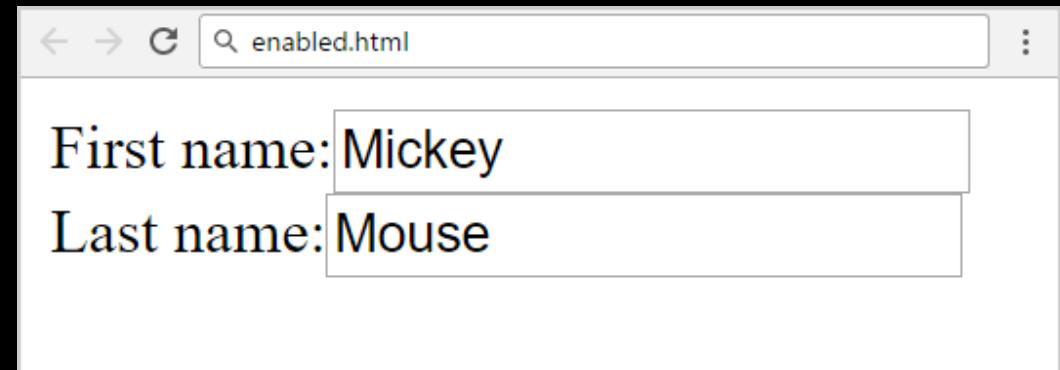
```
p:first-of-type {
    background: #95313b;
    color: #fff;
}
```



The UI Element States

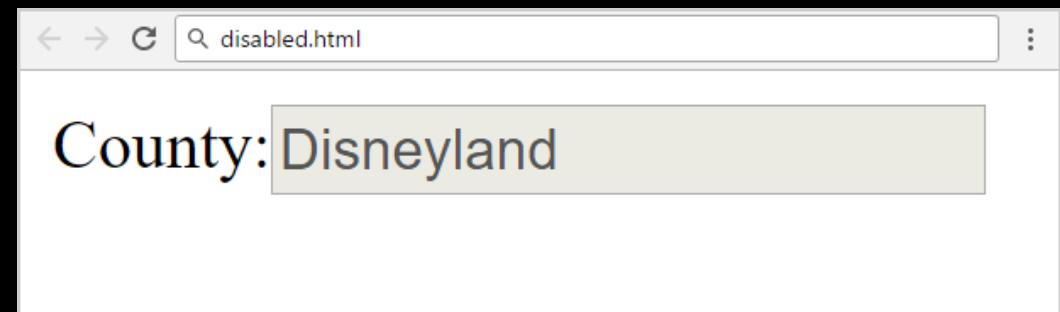
- E:enabled

```
First name:<input type="text">  
<br>  
Last name:<input type="text">
```



- E:disabled

```
County:<input type="text"  
disabled="disabled"><br>
```



Problem: Ancient Architecture

- You are given a HTML file:
- Create a Web page like at the screenshot
 - Constraints:
 - Don't change the HTML file
 - Just write the missing CSS



[←](#) [→](#) [C](#) ancient-architecture.html

Ancient Roman architecture

Posted on 01.01.2017 by Someone



Ancient Roman architecture adopted the external language of classical Greek architecture for the purposes of the ancient Romans, but grew so different from Greek buildings as to become a new architectural style. The two styles are often considered one body of classical architecture. Roman architecture flourished in the Roman Republic and even more so under the Empire, when the great majority of surviving buildings were constructed. It used new materials, particularly concrete, and newer technologies such as the arch and the dome to make buildings that were typically strong and well-engineered. Large numbers remain in some form across the empire, sometimes complete and still in use.

0 COMMENTS [VIEW POST](#)

Ancient Greek architecture

Posted on 01.01.2017 by Someone



The architecture of ancient Greece is the architecture produced by the Greek-speaking people (Hellenic people) whose culture flourished on the Greek mainland, the Peloponnese, the Aegean Islands, and in colonies in Anatolia and Italy for a period from about 900 BC until the 1st century AD, with the earliest remaining architectural works dating from around 600 BC.^[1] Ancient Greek architecture is best known from its temples, many of which are found throughout the region, mostly as ruins but many substantially intact. The second important type of building that survives all over the Hellenic world is the open-air theatre, with the earliest dating from around 525-480 BC.

0 COMMENTS [VIEW POST](#)

Ancient Indian Architecture

Posted on 01.01.2017 by Someone



Indian architecture is as old as the history of the civilization. The earliest remains of recognizable building activity in India dates back to the Indus Valley cities. Among India's ancient architectural remains, the most characteristic are the temples, Chaityas, Viharas, Stupas and other religious structures. In ancient India, temple architecture of high standard developed in almost all regions. The distinct architectural style of temple construction in different parts was a result of geographical, climatic, ethnic, racial, historical and linguistic diversities. The Rock-cut structures present the most spectacular piece of ancient Indian art specimen. Most of the rock-cut structures were related to various religious communities.

0 COMMENTS [VIEW POST](#)

Solution: Ancient Architecture – CSS

```
@import  
url('https://fonts.googleapis.com/css?family=Indie+Flower');  
/* Also import 'Roboto' font */  
body {  
  font-family: Roboto, sans-serif;  
  background: #e2d7aa;  
  color: #000000;  
  font-size: 16px;  
  width: 1180px;  
  line-height: 24px;
```

```
margin: 0 auto; }  
  
body>div.img-container>img{  
  width: 1100px;  
  height: 450px;  
}  
  
body>div.img-container {  
  width: 1600px;  
  height: 450px;  
  margin-top: 65px;  
}
```

Solution: Ancient Architecture – CSS (2)

```
h1 {  
    font-size: 60px;  
    font-family: 'Indie Flower', cursive;  
    letter-spacing: 5px;  
    text-align: center;  
    margin-top: 65px;  
}  
  
a {  
    text-decoration: none;  
}
```

```
nav {  
    margin: 30px;  
    text-align: center;  
}  
  
nav a {  
    text-decoration: none;  
    font-size: 28px;  
    color: #000000;  
}
```

Solution: Ancient Architecture – CSS (3)

```
nav ul li {  
    list-style: none;  
    display: inline;  
    padding: 0 12px 0 12px; }  
  
nav ul li:first-of-type a,  
nav ul li:last-of-type a {  
    padding-left: 0;  
    color: #0a2d4a;  
    font-weight: bold; }  
  
article {  
    width: 370px;  
    display: inline-block; }
```

```
main {  
    margin-top: 30px;  
}  
main > article img {  
    width: 350px;  
    height: 200px;  
    margin-bottom: 20px;  
}  
p {  
    width: 345px;  
    text-align: justify;  
    margin-bottom: 20px }
```

Solution: Ancient Architecture – CSS (4)

```
article img + p:first-letter {  
    font-family:  
        EucrosiaUPC sans-serif;  
    text-transform: uppercase;  
    font-size: 40px;}  
  
.readmore a, .comments a {  
    background:  
url(images/bg2.jpg);  
    color: #ffffff;  
    padding: 10px;  
    margin-top: 10px;  
    text-transform: uppercase;}
```

```
footer {  
    margin: 0;  
    padding: 10px 0 10px 0;  
    text-align: center;}  
  
article p:last-of-type  
span:first-of-type a::before {  
    content:  
url("images/comment.png");  
    position: relative;  
    padding: 10px;  
    right: 5px;  
    top: 7px;}
```

Other CSS 3 Selectors

- **E:target**
 - An E element being the target of the referring URI
- **E:not(s)**
 - An E element that does not match simple selector
- **E ~ F**
 - An F element preceded by an E element



Default Browser Styles and Precedence

Default Styles and Style Precedence

Default Browser Styles

- Browsers have predefined CSS styles
 - Used when there is no CSS information or any other style information in the document
- Caution: default styles differ in browsers
 - Margins, paddings and fonts differ most often
 - Usually developers reset them

```
* { margin: 0; padding: 0; }
```

```
body, h1, p, ul, li { margin: 0; padding: 0; }
```

CSS Cascade (Precedence)

- There are browser, user and author stylesheets with "normal" and "important" declarations
 - Browser styles (defined by the user-agent, least priority)
 - Normal user styles (defined in the browser's user settings)
 - Normal author styles (external, in head, inline)
 - Important author styles (defined with **!important**)
 - Important user styles (max priority)

```
a { color: red !important ; }
```

Order of Style Definitions

- Conflicting style definitions are resolved by priority
- Priorities of the style definitions
 1. External `<link rel="stylesheet" href="...">`
 2. Styles in the `<head><style>...</style></head>`
 3. Inline style attributes: `<p style="...">`
 4. Using `!important`

```
h1.big {color:red !important}
```

Selector Priority (Specificity)

- Selector priorities depend on:

0

0

0

0

inline styles

number of
id selectors

number of
class
selectors

number of
element
selectors

Selector Priority (Specificity) – Example

```
p { color: #FFF }
```

0, 0, 0, 1

```
.intro { color: #345678 }
```

0, 0, 1, 0

```
#header { color: #000 }
```

0, 1, 0, 0

```
<p style="color: blue">Text</p>
```

1, 0, 0, 0

```
p.intro#header { color: #FFF }
```

0, 1, 1, 1

```
p.intro.big#header { color: #FFF }
```

0, 1, 2, 1

```
p { color: #000 !important }
```

important

CSS References

- CSS Tricks
 - <http://css-tricks.com>
- The CSS documentation at WebPlatform.org:
 - <http://docs.webplatform.org/wiki/css>
- CSS documentation at Mozilla
 - <https://developer.mozilla.org/en-US/docs/CSS>
- CSS3 tutorial
 - <http://www.w3schools.com/css3/>

Summary

1. Primary Selectors

- Tag, element ID, class name

2. Nested Selectors

- Descendant, Adjacent Sibling, Direct Child, Multiple Classes

3. Attribute Selectors

- a[title], input[type='text'], a[title*=logo]

4. Pseudo Selectors

5. Default Styles and Style Precedence



CSS Selectors and Rules



Questions?

