Being Awesome with CSS

A tutorial by Zoe F

CSS: The quick facts

- CSS was created 'cause HTML just isn't made for complex formatting
- The cobbled together CSS to solve the problem
- Everyone hated it
- Eventually they learned to live with it.

Eh? Why do we want it?

- It allows us to get rid of attributes
- A much more flexible design
- Different styles can be used in different situations

Let's see it in action!

http://www.csszengarden.com

Known problems and bugs. ☹

- There's no good WYSIWYG editor for CSS
- Getting things to stack vertically is really tough
- There's still some variation between browsers

CSS: The final frontier

http://www.w3schools.com/Css/css_reference_atoz.asp

The old way:

    <B><h1>Hello</h1>

Tags/Attributes

The new hotness:

H1 {   **Selector**

        font-size:36pt;   **Properties: Values**

        color:blue;

        font-weight:bold

        }

Declaration

    <h1>Hello</hello>

CSS three ways!

- Inline Styles

- Internal Stylesheets

- External Stylesheets

More html

Less html

The 1 minute rule!

Are you stuck on a bug? Wait one minute, then ask the person next toyou.

It's the XP way!

Our materials!

Lab:

http://www.binaryspark.com/classes/CSS_rocks/CSS_lab_instructions.html

Materials:

http://www.binaryspark.com/classes/CSS_rocks/materials.zip

Inline Styles

Old way

<p>
This is a paragaph
</p>

New way

<p style="color:blue; ">
This is a paragraph.
</p>

Internal stylesheets

In the header:

```
<style type="text/css">
        ul {background-color:yellow;
                color:white; }

        p {color:blue}
</style>
```

In the body:
```
<ul>
        <li>list item 1
        <li>list item 2
</ul>
```

External stylesheets

Separating out the CSS and the content completely. Finally.

Creating a Class!

- Classes that can apply to any type of tag:

- Classes that only apply to one or two types of tag:

Calling a Class!

- ON a tag

- With no existing tag

Class-y vocab

- Inheritance

  P.title, P.subtitle, P.isblue, {color:blue;}

  P.Title {text-align:center;}

- Overwriting

  P.title, P.subtitle P.isblue, {

       color:blue ; text-align:center;}

  P.title { color: red;}

DIV's and why they rock

```
<div ID="imadiv">
  <h3>This is a header</h3>
  <p>This is a paragraph.</p>
</div>
```

```
#imadiv {
      color:green;
}
```