HTML & CSS

CSC309

Mark Kazakevich

Last time we talked about HTTP

- **HTTP**
  - A stateless application layer protocol used for requesting access to resources on the World Wide Web

Today

- We will start talking about what we actually transfer over HTTP
- Specifically: **Web pages**

Web pages

- Web pages are requested by **web browsers**, which interpret and display their contents

HTML

- Web pages are written in **HTML**
  - HyperText Markup Language

- HTML is a **markup language**
  - Describes a web page's **content** and **structure**
  - Not a programming language

HTML basic structure

```
<!DOCTYPE html>
<html>
   <head>
      This is where page metadata and invisible
      content goes
   </head>
   <body>
      This is where visible page content goes
   </body>
</html>
```

HTML Elements

- HTML is written as a collection of **elements** which can contain **content**

```
<p>This is a paragraph element</p>
```

- Elements provide a structure to the document

HTML elements

- Most elements are indicated by an opening <> and closing </> tag
  - `<p>This is a paragraph element</p>`

- Some elements don’t need a closing tag
  - `<br>` line break
  - `<img ...>` image tag

- Elements can contain other elements
  - `<p>This is <strong>imporant.</strong></p>`

HTML boilerplate

- Let’s create an HTML file with all the tags necessary to make a **valid** HTML page

- Can check if HTML is valid using:
  https://validator.w3.org/

HTML simple web page

- Now let’s add some elements to our html file

Semantic tags

- **Semantic Tags/Elements**
  - HTML tags that indicate their expected use
    - `<form>, <table>, <h1>`

- **Design** - meaning of page is always the same regardless of style
- **Accessibility** - screen readers can change voice tone on a tag
- **Search Engine Optimization** - density of keywords is higher when more semantic tags are used

CSS

CSS

- **Cascading Style Sheets**

- A language that describes the “**style**” (layout and appearance) of web pages
  - Separation of content (HTML) and layout (CSS)

- “**Cascading**”: How we style a web page has priority rules
  - We’ll see how this works

Style Rules

- CSS files are simply a set of rules to style different parts of a web page

style.css

```
selector {
    property1: value;
    property2: value;
    ...
}

..more selectors..
```

`selector`: Identifies HTML element (or set of elements)

`property`: Layout property to assign value to

`value`: Value of property

Properties

- What are these ‘properties’?
  - Colour
  - Size
  - Shape
  - Position
  - Font
  - How to align text
  - ...and many more!

Let’s add some CSS to our site

- Need to make a css file
- Need to **link** it in our .html file

Inline and Block elements

- Elements on a web page can be **displayed** in different ways

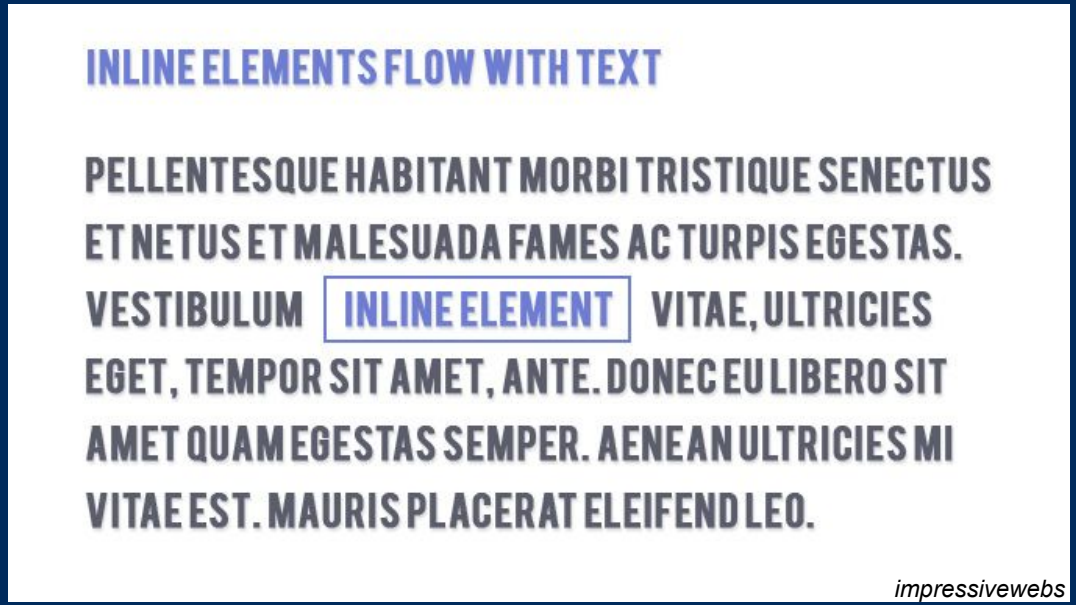
- **Block** elements (such as: `<p>`, `<h1>`, `<h2>`, `<ul>`)
  - Height and width can be specified and changed
    - By default:
      - Width is full width of parent element
      - Height is enough to fit the content
  - Forces creation of newlines

Block


BLOCK ELEMENTS EXPAND NATURALLY
AND NATURALLY DROP BELOW OTHER ELEMENTS
impressivewebs

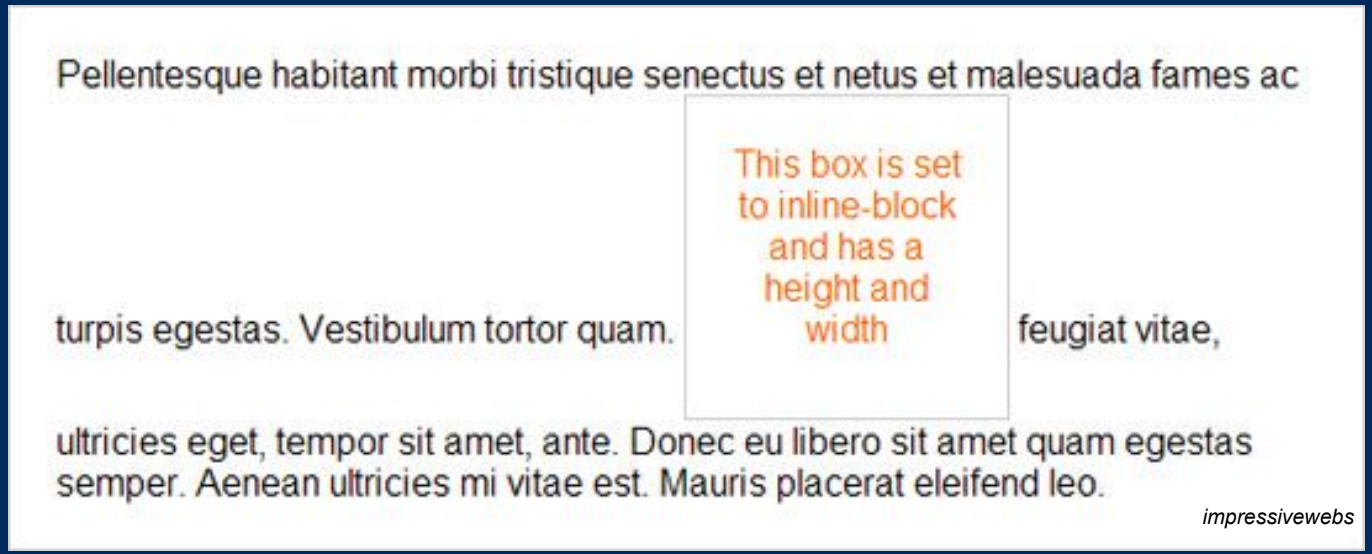
Inline

- **Inline** elements (such as: `<strong>`, `<a>`, `<br>`)
  - Don’t have defined width/height
  - Can’t have block element inside it



Inline-block

- Inline elements that can have a height/width
  - Images <img>



`<div>` and `<span>`

- Two **non-semantic** elements
  - Generic - no specific purpose

- `<span>` is a generic inline element
- `<div>` is a generic block element

- Are used more for creating **natural divisions** throughout your page
  - **Note:** Don't *visually* divide anything themselves
  - You have to indicate how they should appear relative to other elements

Back to CSS

- How do we select elements on a page more specifically?
  - Not just **all** <p>'s or <h1>'s?

- For example, maybe you're writing an article and want to **highlight** some sentences for readers to pay attention to

CSS classes

- Can define your own CSS selectors: **classes**
  - An attribute of an HTML element

In HTML:
This is important

This gives this specific tag a class attribute named “highlight”

CSS classes

- In CSS, selected by putting a **dot** before the class name

```
.highlight {
  background-color: yellow;
}
```

HTML/CSS **id**'s

- **id** attribute in HTML meant to be a unique identifier
  - Only one element should have a particular id

- In CSS, selected by a **hash symbol**

```
#navbar {
   color: white;
}
```

Can also be used as 'anchors' in URL

`http://mysite.ca/index.html#anchor`

Auto-scrolls to position of anchor

Combining selectors

- Different ways to combine selectors
- **Descendant selector**

```
p strong {
        background-color: yellow;
 }
```

applies to all `<strong>` elements that are inside a `<p>`

Combining selectors

- **element.class selector**

```
p.highlight {
        background-color: yellow;
 }
```

applies to all `<p>` elements that have class `highlight`

Combining selectors

- **Multiple element selector**

```
p, strong, h1 {
        background-color: yellow;
 }
```

applies to all `<p>`, `<strong>`, and `<h1>` elements