HTML

Class 2

Calendar

- on the calendar:
  - link to Monday's slides
  - link to next Tuesday's homework

Separation of Duties

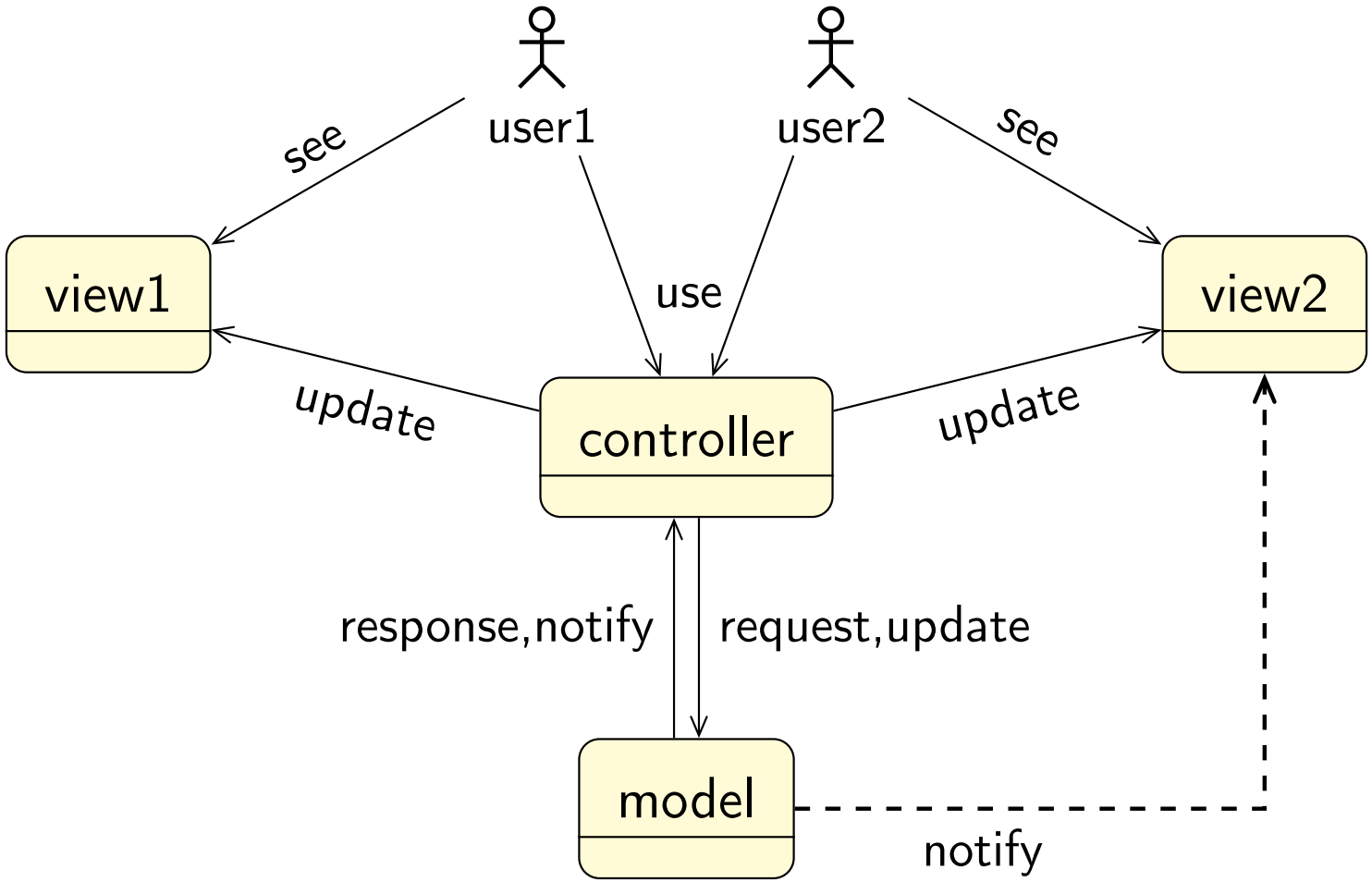
a fundamental concept of modern computer science:
the separation of

- content
- appearance
- behavior

a widely-used architectural pattern for web systems is
Model-View-Controller (MVC)

- model: the data or contents
- view: the appearance
- controller: the behavior

MVC


user1
user2
see
see
view1
use
view2
update
controller
update
response,notify
request,update
model
notify

MVC on the Web

web systems use various technologies to implement MVC

model:

- Structured Query Language (SQL): relational data
- eXtensible Markup Language (XML): structured data
- JavaScript Object Notation (JSON): structured data
- files: images, video, audio, text, etc.

view:

- Hypertext Markup Language (HTML): page content
- Cascading Style Sheets (CSS): style rules
- browser rendering engines

controller:

- PHP Hypertext Processor (PHP): server logic
- JavaScript: client behavior
- Asynchronous JavaScript and XML (AJAX): client-server interaction

Structure of an HTML Page

```
<!DOCTYPE html>
<html lang="en">
  <head>
    ...
  </head>
  <body>
    ...
  </body>
</html>
```

- this is in a file with extension `.html`
- DOCTYPE is an XML tag that tells browser to interpret the contents as HTML5
- the header stuff gives information about the page
- the body contains the page contents

HTML Elements and Tags

- all HTML content is contained within tags
- tags denote an element
- all tag names are strictly lowercase
- most tags come in open-close pairs with content
  `<p>blah blah blah</p>`
- some tags have void content and are self-closing
  `<hr />` and `<br />`

- every open tag must be closed!
- `<br>` is technically legal in HTML5, but illegal in XML
- for consistency and readability, we will only use `<br />` and will disallow `<br>`

Tags and Attributes

- many tags have attributes with double quotes, no spaces

```
<p class="important">Be careful handling chemicals!</p>
```

- some attributes are optional

```
<p>This is a paragraph without attributes.</p>
```

- some attributes are required

```
<img src="gnus.png" alt="Photograph of a herd of gnus." />
```

Header Elements for Information

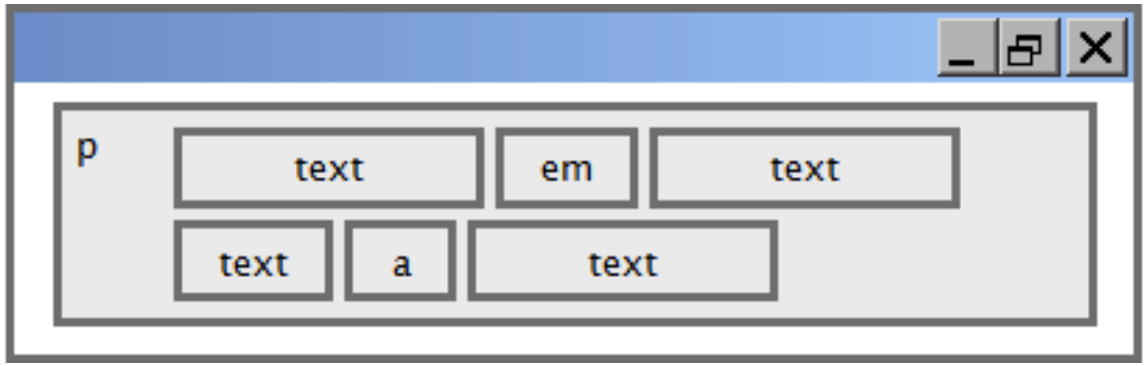
- meta tags give information about the page

```
<meta charset="utf-8" />
<meta name="author" content="Jon Beck" />
```

- the content of the title tag is displayed in the web browser's title bar and when bookmarking the page

```
<title>My Web Page</title>
```

Block and Inline Elements



- HTML5 defines many categories of content (embedded, phrasing, sectioning, etc.)
- but everyone uses the non-real categories block and inline
- every body element is a block or an inline element
  - block elements contain a region of inline content
  - text flows and wraps within the block
  - a block is set off by a newline break
    - examples: paragraphs, lists, table cells
  - inline elements are small units of content within a block
  - the browser places inline elements within wrapped lines
  - a block element can contain other block or inline elements
  - most inline elements can contain other inline elements
  - in general, an inline element cannot contain a block element

Block Elements

Headings (h1 – h6)

- the outline format of the document
- never skip levels!

Block Elements

Paragraph (p): a paragraph of text

- the most common block element
- words and inline elements are wrapped within the width
- horizontal whitespace is collapsed

Block Elements

Theme change (hr): signal a thematic change in content

- used to be called horizontal rule
- no attributes allowed (old ones are deprecated)
- default rendering is a horizontal line
- a self-closing tag `<hr />`

Block Elements

- other block elements we'll get to soon
  - lists, unordered and ordered, and their items
  - canvas for drawing
  - tables and their column and row contents
  - preformatted text, good for multiline code examples
  - div for a generic section of a page
  - form for sending data to the server
  - grid for industrial strength layout

Inline Elements

Image (img): insert an image

- self-closing tag
- attributes:
  - src: required, can be relative or absolute
  - alt: required, should be informative
  - width and height: optional, speed up rendering, in pixels

```
<img src="../images/stickman.gif" alt="Stickman"
   width="24" height="39" />
<img src="http://www.w3schools.com/images/lamp.gif"
   alt="Lamp" width="15" height="15" />
```

Inline Elements

Code (code): a short piece of source code or a file name

```
<p>
  Java uses <code>System.out.println</code>
  for output.
</p>
```

- inside the p block element are bare words and inline elements
- bare words must be enclosed in some block element

Inline Elements

- other inline elements we'll get to soon
  - span for a generic section of a line
  - button to create a clickable spot
  - input for a form to collect data
  - label to label a form element
  - select for a drop-down menu
  - textarea for a form to collect multi-line text

Text

- text is not an element
- all text must be enclosed within an element
  - block (e.g., p)
  - inline (e.g., span)
- most text just displays itself
- some characters are special, and some aren't on your keyboard
- special characters that don't come out right
  - multiple whitespace: collapsed in by the renderer
  - <: begins an HTML tag
  - >: ends an HTML tag
  - &: begins a character entity

Character Entities

- special characters and characters not on the keyboard
- can be specified by character name or by character number
- examples:

symbol	name	number	description
∀	&forall;	&#8704;	for all
<	&lt;	&#60;	less than
&	&amp;	&#38;	ampersand
‘	&lsquo;	&#2018;	left single quote
’	&rsquo;	&#2019;	right single quote

- google for “html character entities”

Coding Style

Without some discipline, HTML code can become messy. In this class, the following rules apply:

- source code lines must be fewer than 80 characters in width
- all indent levels are 2 spaces, using the space character and no tabs
- if a block element's tags and content can fit on one (properly indented) line, they may do so
- otherwise, a block element opening and closing tag must be on lines by themselves, with the element content indented
- inline elements are simply that: inline, not indented
- blank lines are used for visual clarity

Style Example

```
<body>
  <h1>CS 315 Internet Programming</h1>
  <h2>Spring 2022</h2>

  <p id="lastmodified">6 January 2022</p>

  <h3>Times and Dates</h3>

  <p>
    Please see the <a href="CalendarCS315.html">course
    calendar</a> for information about assignments,
    significant dates, and events.
  </p>
</body>
```

Comments

HTML can contain comments `<!-- this is a comment -->`

- HTML is so simple that comments are rarely needed to explain code itself
- most useful is to label a closing tag whose opening tag is far away

```
<div id="section3">

... lots of content

</div> <!-- section3 -->
```