Announcements

- Due date changes
  - ∗ Labs 7/8 due before noon this Wed. (May 14)
  - ∗ Project 2B due before noon next Wed. (May 21)

Announcements

FIT100

- Check your email for common problems and solutions for the final pages of Labs 7/8

Announcement

- Lecture Friday is canceled
    - MGH is venue for Undergraduate Research Symposium
    - Noon - 5pm throughout Mary Gates
    - You're excused from lecture to attend!
    - 4th floor labs are still available

Announcement

FIT100

- Reminder
  - ∗ At end of quarter, we will drop your
    - Lowest two Quiz scores
    - Lowest two Lab scores
    - Lowest two Quick Write scores

Announcement

FIT100

- This class has...

  No:
  - Midterms
  - Final

  Yes:
  - Labs
  - Quizzes
  - Projects
  - Quick Writes

Project 2B

- Project 2B and its "quiz" are linked on our online calendar
  - * "Quiz" is
    - Open book
    - Save and resume
    - Retake as often as you want to improve your score
    - Be sure to submit it before the quiz closes!

Quiz and Quick Write

- Next week
  - ∗ Review the questions at the end of these chapters:
    - *Fluency* chapters 18 and 20
    - *QuickStart* chapters 1 and 2
- All JavaScript!

- Topics will include:
  - Variables
  - Values & data types
  - Assignment statements
  - Rules for identifiers
  - Conditionals
  - Loops
  - Arrays
  - Functions
  - Curly brackets

Quick Write

FIT100

- Put everything on the floor except a pen or pencil
- No laptops, pda's, cell phones, etc.
- No notes, textbooks, etc.
- No discussions with your neighbors
- No wandering eyes
- Write for five minutes

Browser Object Model and Document Object Model

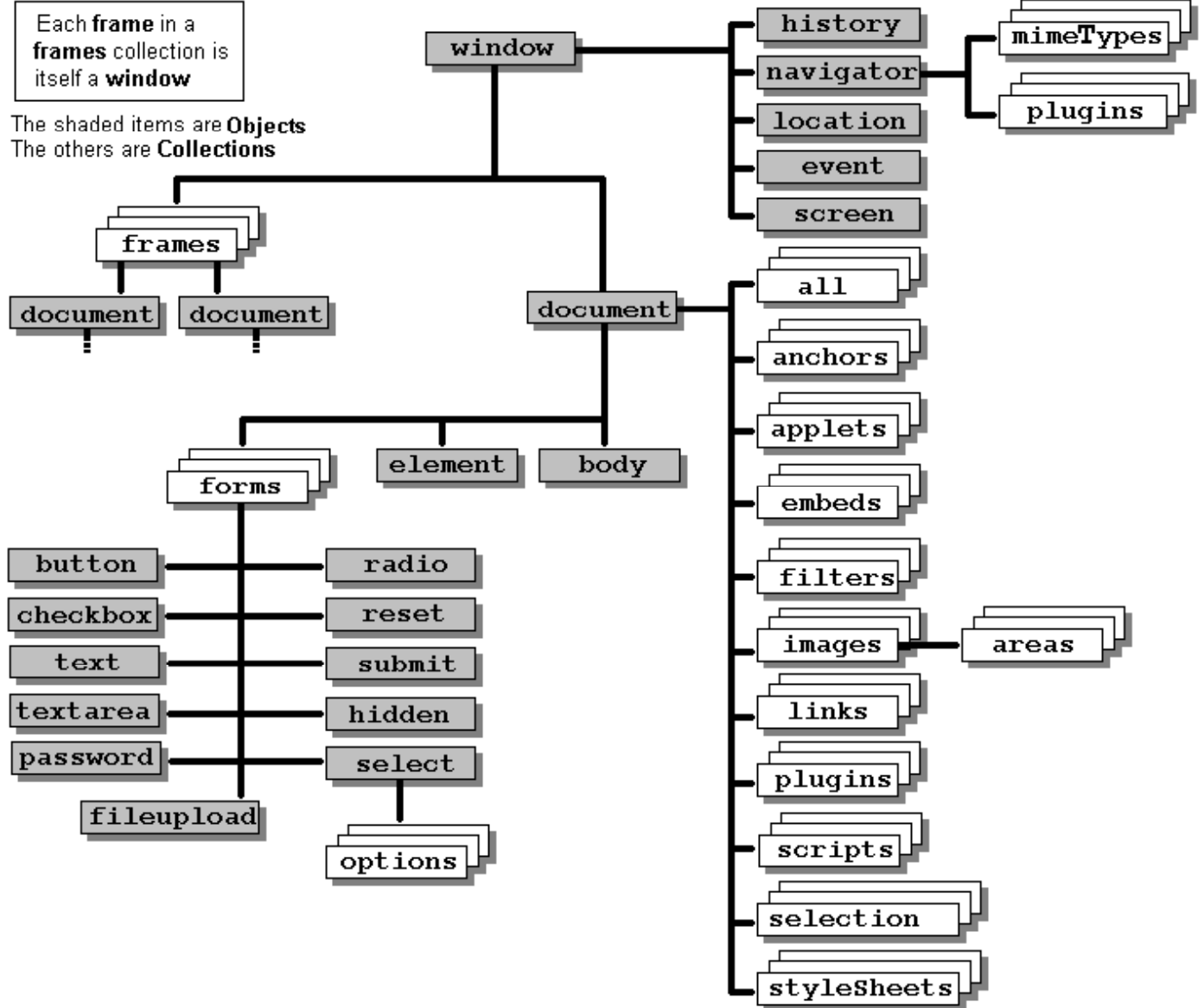## Understanding the document tree

Browser Object Model

- All the images on the page are part of an array
  - **document.images[]**
- Recall an image tag:

- Attributes can be reached—and changed—with dot notation:
  - **document.images[1].src**
  - Another way to do an image rollover

FIT100

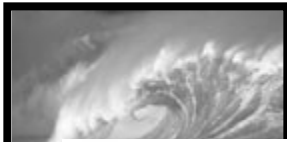Each **frame** in a **frames** collection is itself a **window**

The shaded items are **Objects**
The others are **Collections**

window

- history
- navigator
  - mimeTypes
  - plugins
- location
- event
- screen

frames

- document
- document

document

- all
- anchors
- applets
- embeds
- filters
- images
  - areas
- links
- plugins
- scripts
- selection
- styleSheets

element   body

forms

- button
- checkbox
- text
- textarea
- password
- fileupload
- radio
- reset
- submit
- hidden
- select
  - options

11-27-2007

Video

FIT100

Essentials of the DOM and JavaScript
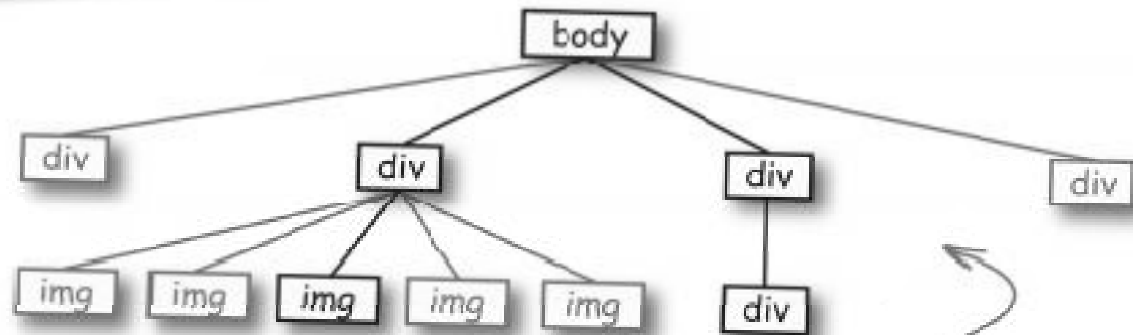
http://www.youtube.com/watch?v=URF2sVQWuxU

Use the Document Object Model.

The web browser uses the Document Object Model to represent your web page. When you change this model with your JavaScript code, the web page will automatically change, too.

The DOM lets you look at this web page...

...in this format

Your JavaScript can work with the DOM much easier than working directly with HTML or CSS.

DOM

- The Document object gives you access to the browser's DOM tree
- You've used it for several labs now:

  document.getElementById("zone1").value

DOM

FIT100

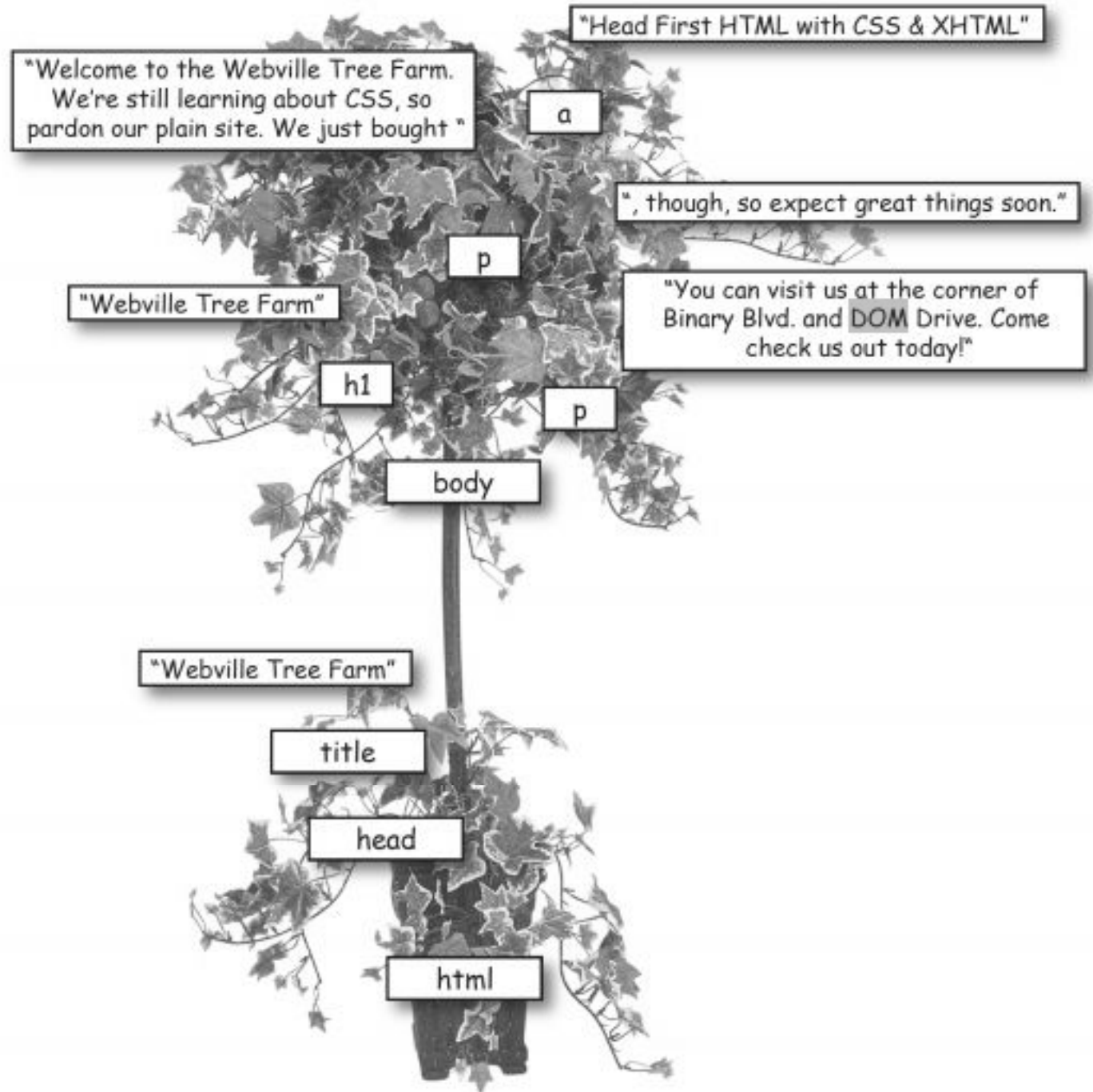- Here's the HTML

```
<html>
 <head>
  <title>Webville Tree Farm</title>
 </head>
 <body>
  <h1>Webville Tree Farm</h1>
  <p>Welcome to the Webville Tree Farm. We're still learning
     about CSS, so pardon our plain site. We just bought
     <a href="http://www.headfirstlabs.com/books/hfhtml/">Head
     First HTML with CSS &amp; XHTML</a>, though, so expect
     great things soon.</p>
  <p>You can visit us at the corner of Binary Blvd. and
     DOM Drive. Come check us out today!</p>
 </body>
</html>
```

- As the browser sees it!

DOM

- getElementByID("id")
  - ∗ attach ID-attributes to HTML tags and access page elements by this notation, instead of having to wade through the hierarchy.

- appendChild()
  - ∗ Use appendChild() and .text or .innerHTML instead of += concatenation

DOM

FIT100

- getElementByTagName("div")
  * Search through all the <id> tags.

D.A. Clements, Instructor

![FIT100]

DOM Exercise

node. Any **single** piece of markup, such as an element or text. The <a> element is an **element** node, while the "Head First HTML with CSS & XHTML" text is a **text** node.

leaf. A piece of markup that has **no children**, such as an element with **no** text content, like <img>, or textual data.
Also known as: leaf node

"Head First HTML with CSS & XHTML"

a

"Webville Tree Farm"

p

child. Any piece of markup that is **contained** by another piece of markup. The text "Head First HTML with CSS & XHTML" is the **child** of the <a> element, and the <p>s in this markup are **children** of the <body> element.
Also known as: **child node, children.**

h1

p

body

html

branch. A branch is a **collection** of elements and content. So the "body" branch is all the elements and text **under** the <body> element in the tree.

parent. Any piece of markup that contains **other markup.** <h1> is the parent of the text "Webville Tree Farm", and <html> is the parent of the **<body>** element.
Also known as: **parent element, parent node.**

root element. The element in a **document** that **contains** all other elements. In HTML, the root element is always **<html>**.