Computers, Programming, & JavaScript

A gentle introduction

Day 2

Review of Key Points from last session

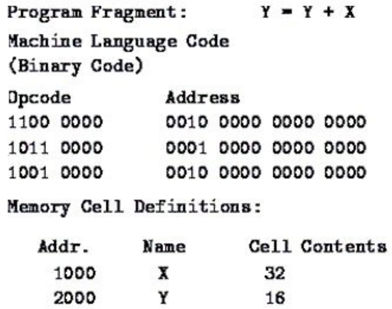
- A computer is a machine for executing programs
- A program is a set of instructions telling a computer what to do
- We write programs in programming languages
- A programming language is a framework that enables us to:
  1. Express simple ideas
  2. Combine simple ideas into a complex idea
  3. Give a complex idea a "label", and then manipulate it as a simple idea

...and, thus, we manage complexity

- A programming language has precise form (syntax) and precise meaning (semantics).

Programming languages: low-level & high-level

The computer understands it own (low-level) machine language



Program Fragment: Y = Y + X

Machine Language Code
(Binary Code)

Opcode	Address
1100 0000	0010 0000 0000 0000
1011 0000	0001 0000 0000 0000
1001 0000	0010 0000 0000 0000

Memory Cell Definitions:

Addr.	Name	Cell Contents
1000	X	32
2000	Y	16

Humans use ***high-level programming languages***

```
def greeting(name):
     print("Hello, " + name + "!")
```

Our focus is ***JavaScript***, one of the many high level programming languages

JavaScript

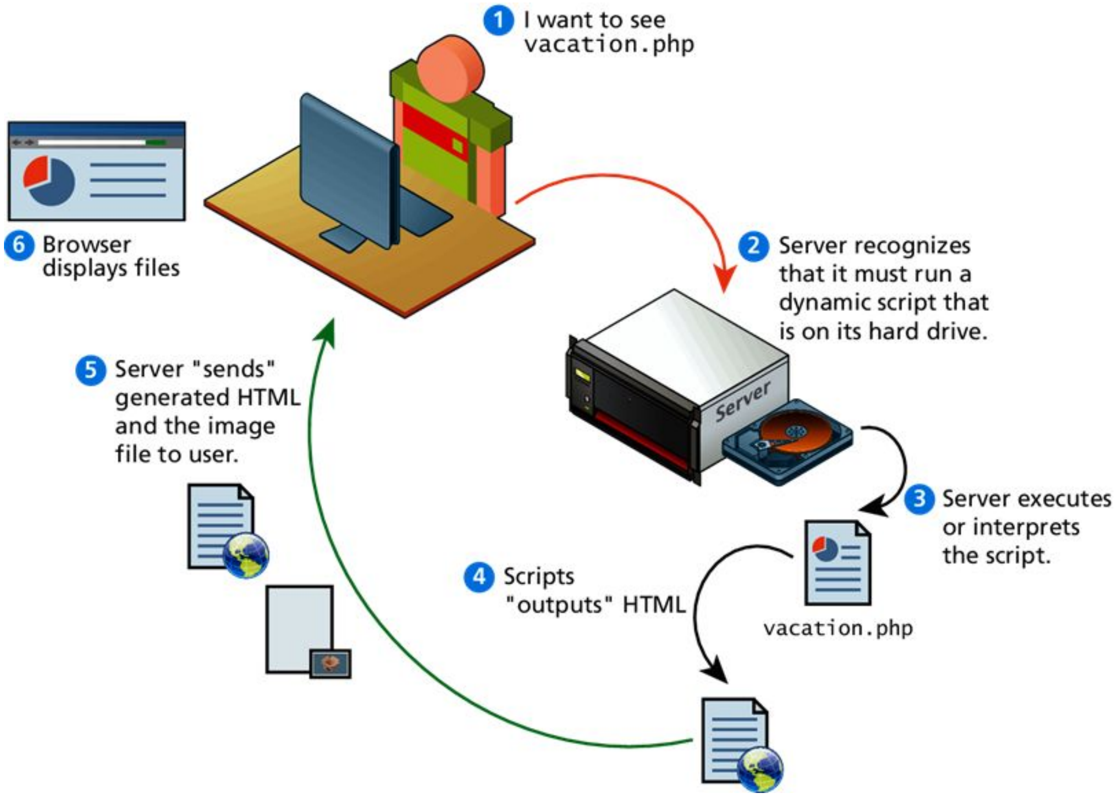
- Is one of the world’s most popular languages
- Is “the world's most misunderstood programming language”
  - (much more complex and powerful than it appears to be)
- Primarily runs right inside the browser:
  - Has no concept of real input or output
  - It is designed to run in a host environment (the browser), and it is up to the host environment to provide mechanisms for communicating with the outside world
- Although it contains the word Java, JavaScript and Java are vastly different programming languages with different uses

Client-side vs server-side

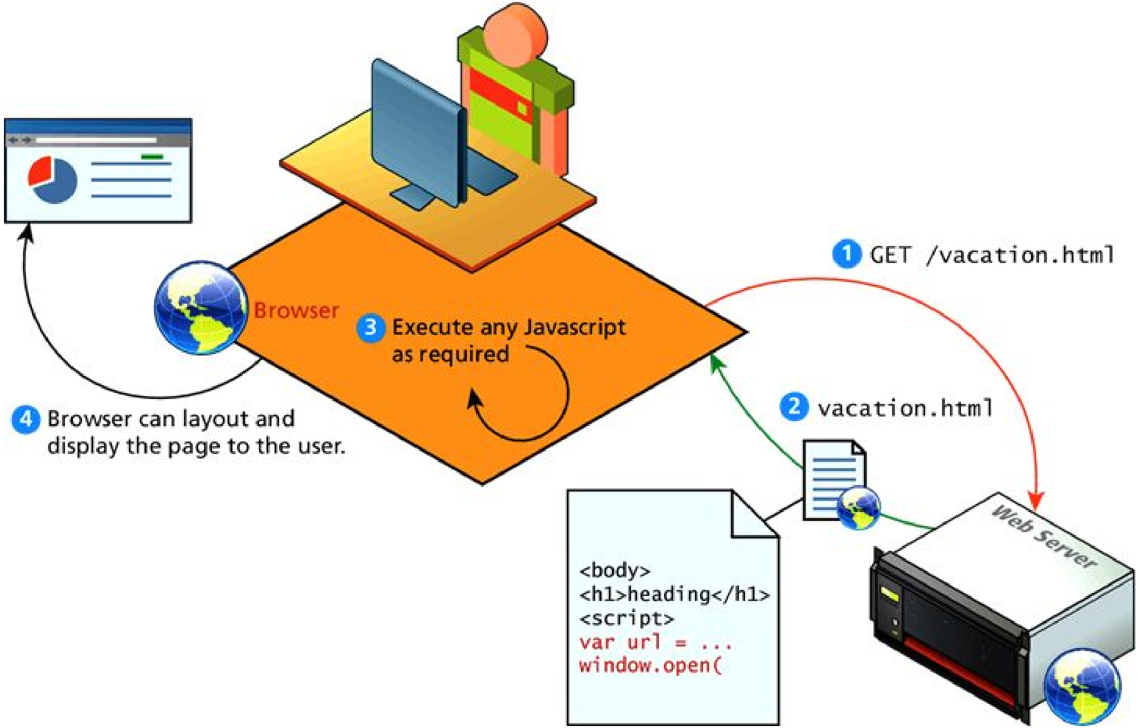
JavaScript is a **client-side** language

PHP, Python, C, C++, and many others can be used on the **server-side** (and so can JavaScript!)

Server-side programming


1 I want to see vacation.php
2 Server recognizes that it must run a dynamic script that is on its hard drive.
Server
3 Server executes or interprets the script.
vacation.php
4 Scripts "outputs" HTML
5 Server "sends" generated HTML and the image file to user.
6 Browser displays files

Client-side programming


1 GET /vacation.html
2 vacation.html
Browser
3 Execute any Javascript as required
4 Browser can layout and display the page to the user.
Web Server

```
<body>
<h1>heading</h1>
<script>
var url = ...
window.open(
```

Comparing server-side and client-side

- On the **server-side** we can:
  - Process user input, get data from external source (databases and more), do any calculations, generate HTML
  - - in any language we like
- On the **client-side** we can:
  - Display basic (silly) visual effects (rollovers!)
    - but that was in the 1990s... Today we can:
  - Process (limited) user input, get data from external source (databases and more), do any calculations, generate HTML
  - - in a few languages, with JavaScript being by far the most portable

Benefits of client-side programs

- Many technological benefits;
- JavaScript can interact with the downloaded HTML in a way that the server cannot, creating a user experience more like desktop software than simple HTML ever could.

Client-side programs

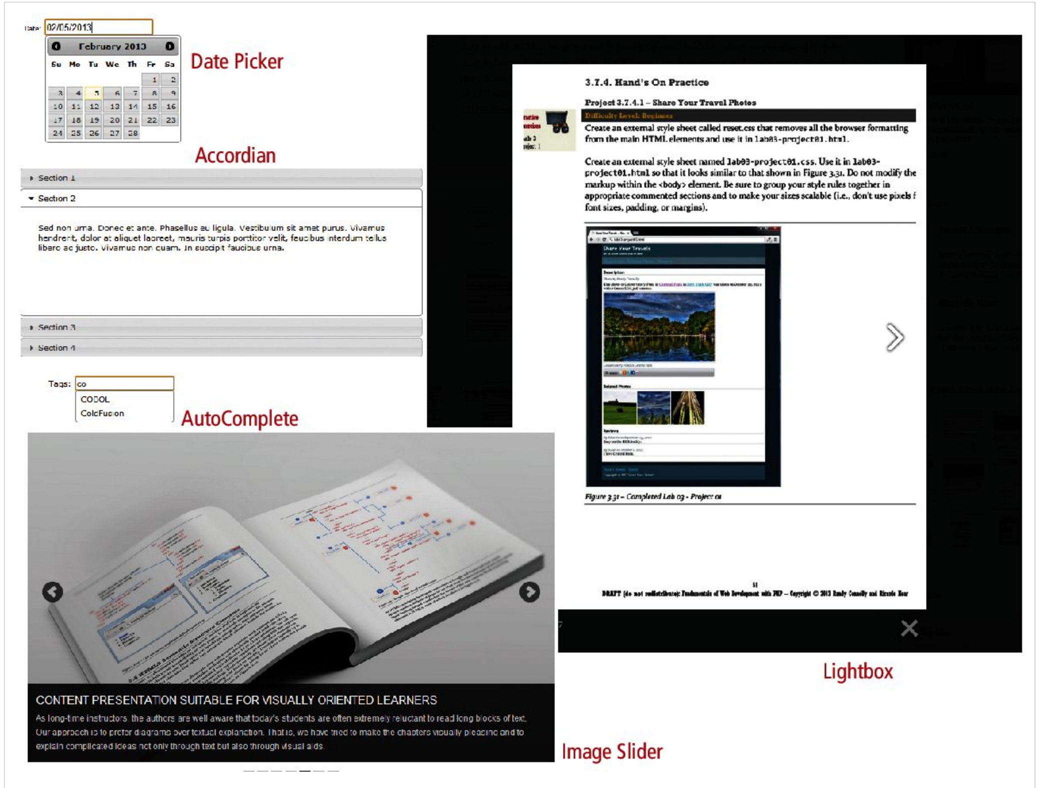
Advantages

- Speed & efficiency:
  - JavaScript can interact with the downloaded HTML in a way that the server cannot, creating a user experience more like desktop software than simple HTML ever could

Disadvantages

- There is no guarantee that the client has JavaScript enabled
- The idiosyncrasies between various browsers and operating systems ***make it difficult to test for all potential client configurations***. What works in one browser, may generate an error in another
- JavaScript-heavy web applications ***can be complicated to debug and maintain***

Which is why we use JavaScript libraries and frameworks!



Date: 02/05/2013

February 2013

Su	Mo	Tu	We	Th	Fr	Sa
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28		

Section 1

Section 2

Sed non urna. Donec et ante. Phasellus eu ligula. Vestibulum sit amet purus. Vivamus hendrerit, dolor at aliquet laoreet, mauris turpis porttitor velit, faucibus interdum tellus libero ac justo. Vivamus non quam. In suscipit faucibus urna.

Section 3

Section 4

Tags: co

CODOL

ColdFusion

3.7.4. Hand's On Practice

Project 3.7.4.1 – Share Your Travel Photos

Difficulty Level: Beginner

Create an external style sheet called reset.css that removes all the browser formatting from the main HTML elements and use it in `lab03-project01.html`.

Create an external style sheet named `lab03-project01.css`. Use it in `lab03-project01.html` so that it looks similar to that shown in Figure 3.31. Do not modify the markup within the <body> element. Be sure to group your style rules together in appropriate commented sections and to make your sizes scalable (i.e., don't use pixels f font sizes, padding, or margins).

Share Your Travels

Figure 3.31 – Completed Lab 03 - Project 01

51

DRAFT (do not redistribute): Fundamentals of Web Development with PHP – Copyright © 2013 Randy Connolly and Ricardo Hoar

CONTENT PRESENTATION SUITABLE FOR VISUALLY ORIENTED LEARNERS

As long-time instructors, the authors are well aware that today's students are often extremely reluctant to read long blocks of text. Our approach is to prefer diagrams over textual explanation. That is, we have tried to make the chapters visually pleasing and to explain complicated ideas not only through text but also through visual aids.

This semester:

1. We learn how to express ideas in JavaScript

2. We learn about JavaScript frameworks

3. We apply our knowledge of JavaScript to work with a JavaScript framework

4. ...to build cool stuff.