WEB TECHNOLOGY

1ST SEMESTER 2019

Client-side Technology
HTML and CSS

Asst. Prof. Manop Phankokkruad, Ph.D.
Faculty of Information Technology,
King Mongkut's Institute of Technology Ladkrabang, Thailand

Outline

- Website Overview
- HTML
- CSS

Email Client

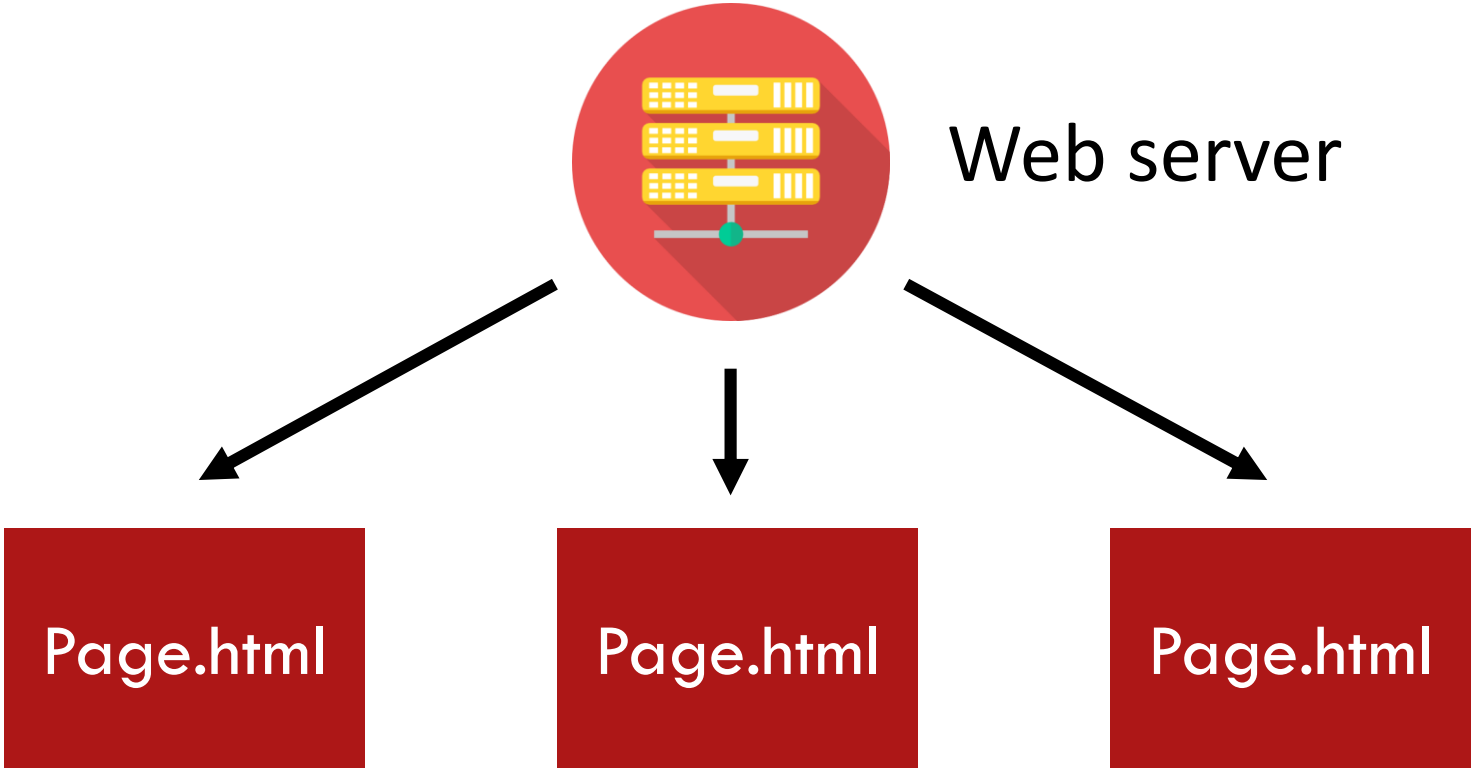
Message composition

The email client is usually set up automatically to connect to the user's mail server, which is typically either an MSA or an MTA, two variations of the SMTP protocol. The email client which uses the SMTP protocol creates an authentication extension, which the mail server uses to authenticate the sender.

Client settings require the name or IP address of the preferred outgoing mail server, the port number (25 for MTA, 587 for MSA), and the user name and password for the authentication, if any.

Overview

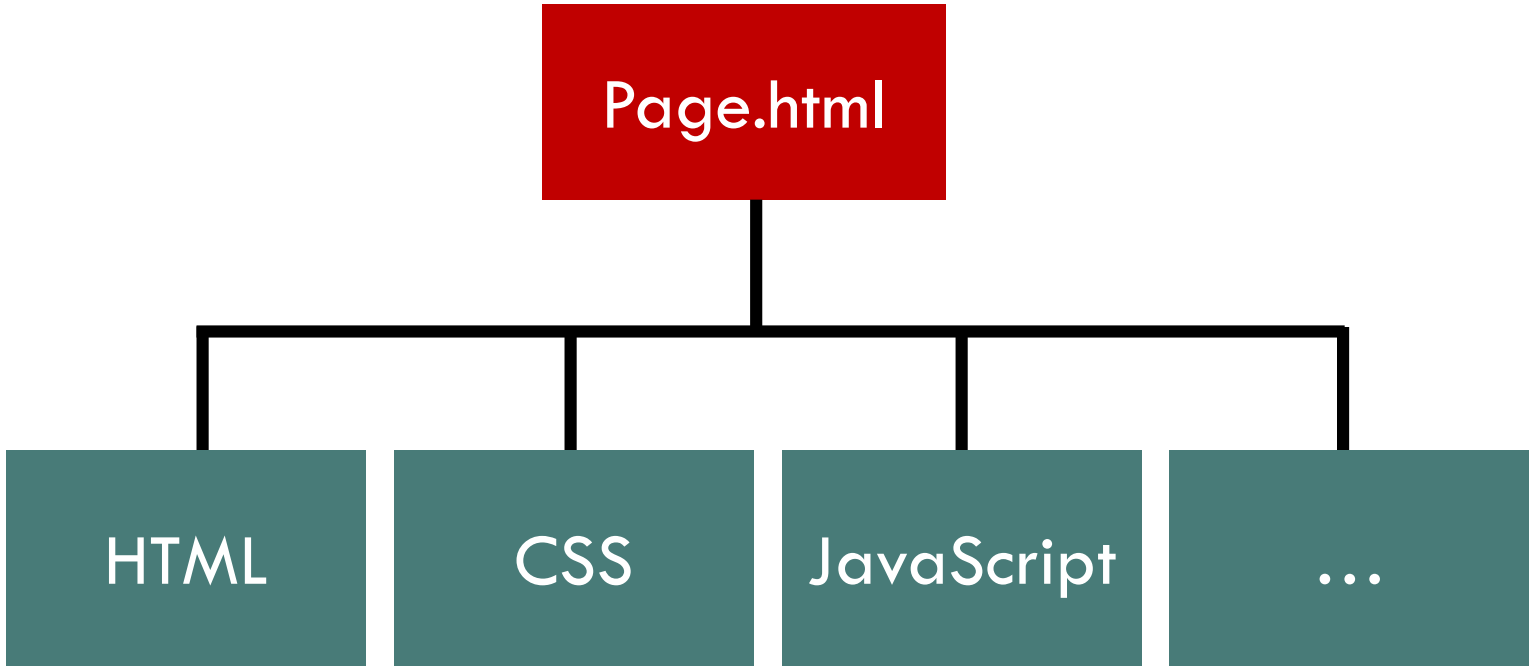
A website is a group of self-contained, individual pages, sent to the browser from the server one-page at a time.



Overview

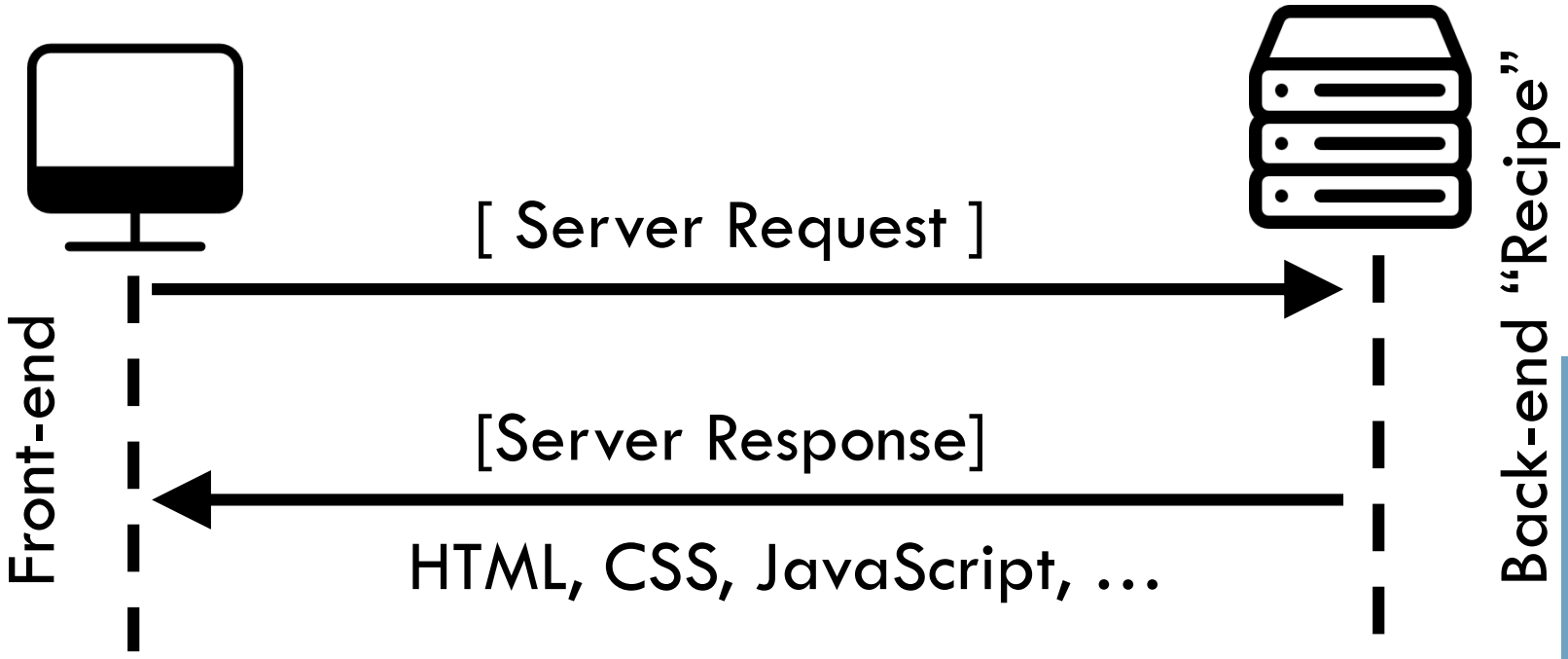
A dynamic web content is built when it is requested, by the user directly, or programmatically while a user is on a page.

Most websites contain both static and dynamic elements.



Overview

Client-side (front-end) coding includes HTML, CSS and JavaScript. This just means that code will be downloaded from the server and then compiled entirely in the browser.



Overview

Three layers of web design:

- Structure - HTML markup, Site planning
- Style - CSS, Imagery
- Behavior - JavaScript

Overview

Common client-side web technologies:

- HTML
- CSS
- JavaScript

HTML and CSS are relatively stable, JavaScript, by means of the application frameworks and utilities developers work with to build web-based applications, is evolving at breakneck speed.

Introduction to HTML

HTML (HyperText Markup Language) is the standard markup language used to create web pages and web applications. Its elements form the building blocks of pages, representing formatted text, images, form inputs, and other structures.

When a browser makes a request to a URL, whether fetching a page or an application, the first thing that is returned is an HTML document. This HTML document may include additional information about its look and layout in the form of CSS, or behavior in the form of JavaScript.

Hypertext + Markup language

Hypertext is text which contains links to other texts.

A **markup language** is a set of markup tags.

The purpose of the tags is to group and describe page content.

DOCTYPE + HTML + HEAD + BODY

<DOCTYPE html>

The doctype is not actually a tag, but a declaration, telling the browser what kind of html you are using. The doctype above declares HTML 5.

<html></html>

The **<html>** element defines the whole HTML document.

DOCTYPE + HTML + HEAD + BODY

<head></head>

The **<head>** element contains special elements that instruct the browser where to find stylesheets, provide meta info, and more.

<body></body>

The **<body>** element contains the document content what is shown inside the browser window.

HTML Basic Structure

Let’s briefly look at a small example file to gain a more concrete understanding of HTML syntax and semantics.

```
<!DOCTYPE html>
<html>
<head>
   <title>Hello World</title>
</head>
<body>
   <p>Hello World!</p>
</body>
</html>
```

HTML : Nesting

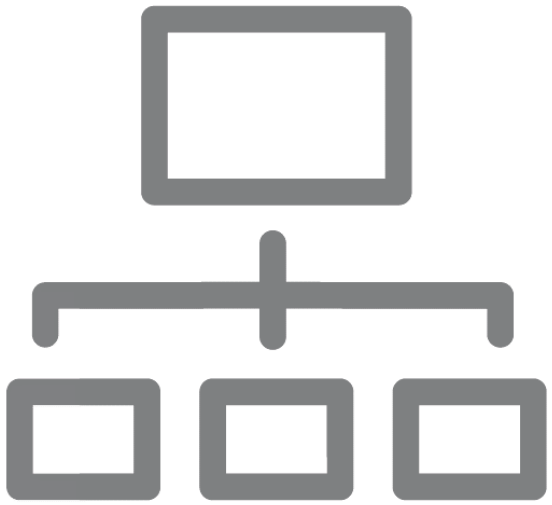
The use of first three tags (html, head and body), introduces and important concept: Nesting, which is when tags “wrap” other tags.

When you create markup, you should indicate nesting by indenting the nested tags with 2 spaces (preferred) or a tab.

```
<html>
  <head></head>
    <body>
      <h1></h1>
      <p></p>
    </body>
</html>
```

HTML: Document Hierarchy

Document Hierarchy: Paren children and siblings

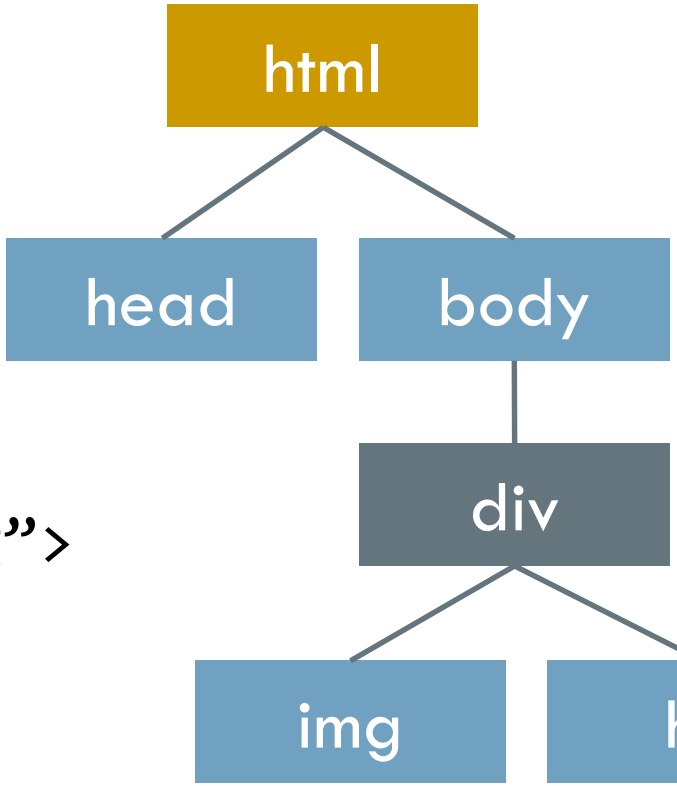


```
<parent x>
  <child y> </child y>
  <child z> </child z>
</parent x>
```

HTML: Document Hierarchy

Document Hierarchy: Parents, children and siblings

```
<html>
<head></head>
<body>
    <div id="logo">
        <img src="dog.jpg">
        <h3></h3>
      </div>
</body>
```



HTML: Elements

Anatomy of an Element

<tag>Content</tag>

An HTML element includes both the HTML tag and content between the tag.

Tags normally come in pairs. The first tag is the start tag, and the second tag is the end tag.

<h1>Main Headline**</h1>**

HTML has a defined set of tag names that the browser understands.

HTML: Elements

The essential element tags

Primary Structure

- html
- head
- body

Head Elements

- title
- meta
- link

Structural Elements

- p
- br
- h1 – h6
- ul
- ol
- a
- img
- div

HTML: Elements

The essential element tags (cont.)

Formatting Elements

- em
- i
- strong
- b
- q
- blockquote
- span

HTML: Attributes

<html lang=”en”></html>

Most elements can have attributes, which provides additional information about the element.

** **

Attributes always follow the same format: name=”value”. You can use either single or double quotes.

HTML: Attributes

The essential attributes

- **href :** This is a link
- **src :** <img src="images/image.jpg" alt="Sam">
- **width** and **height** : <img src="img_student.jpg" width="500" height="600">
- **alt** : <img src="img_student.jpg" alt="Student with a shirt">

Introduction to CSS

CSS (Cascading Style Sheets) is used to control the look and layout of HTML elements.

- CSS styles can be applied directly to an HTML element, defined separately on the same page, or defined in a separate file and referenced by the page.
- Styles cascade based on how they are used to select a given HTML element.

CSS : Cascading + Style Sheet

A **style sheet** is a set of rules defining how an HTML element will be presented in the browser. These rules are targeted to specific elements in the HTML document.

The **cascade** part is a set of rules for resolving conflicts with multiple CSS rules applied to the same elements.

CSS : Cascading + Style Sheet

- **Browser Style Sheet** – the browsers actually put default styling on some elements.
- **External Style Sheet**
  <link href="style.css" rel="stylesheet">

Internal Style Sheet

<style>body { background-color: white; }
</style>

- **Inline (internal) Styles**
  <h1 style="color #F06"> Hello </h1>

CSS : Inheritance

Most elements will inherit many style properties from their parent elements by default.

HTML	relationship
<body> →	parent of site
→	parent of ul and li, child of body
→	
→	parent of li, child of div and body
	child of ul, div, and body
</body>	

CSS : Specificity

Specificity refers to how specific your selector is in naming an element.

body

make the paragraph 16px, Verdana, red

p

make the paragraph blue

p.pink

make the paragraph pink

16px, Verdana, pink

CSS : Syntax

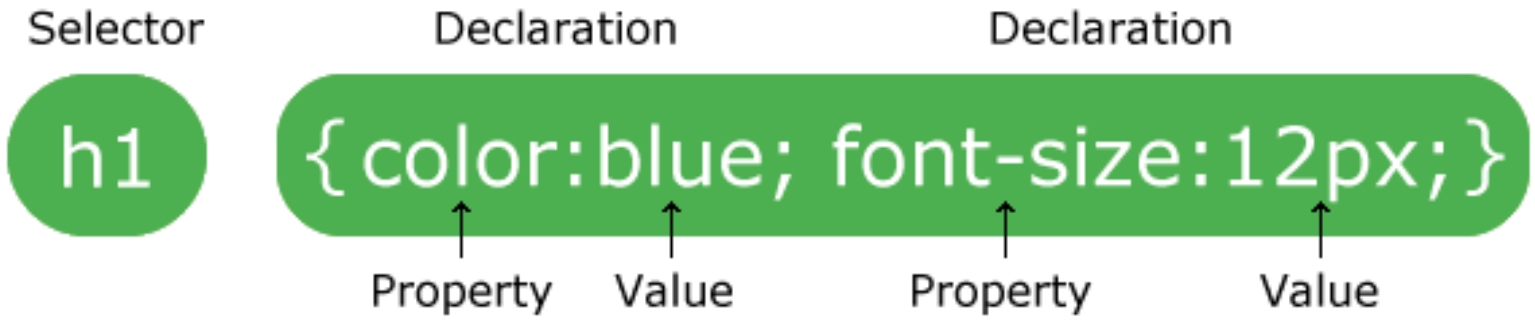
Syntax is the rules for how to write the language.

Three terms for describing the styles:

- CSS rule
- CSS selector
- CSS declaration

CSS : Rule

A CSS rule-set consists of a selector and a declaration block.



The selector points to the HTML element you want to style. The declaration block contains one or more declarations separated by semicolons.

CSS : Selector

The selector associates CSS rules with HTML elements.

```
body {font-family: Arial, Helvetica}
p {color: #666666}
h1 {font-size: 24px}
a {color: blue}
h1, h2 , h3, h4 {
        font-weight: bold
}
```

You can apply styles to multiple selectors in the same rule by separating the selectors with commas.

CSS : Selectors

Four essential selector types :

- ❑ type/element selector
- ❑ id selector
- ❑ class selector
- ❑ descendant selector

type/element → p { **declaration** }

id → #id { **declaration** }

class → .class { **declaration** }

body, p, h1, ul are the simplest type selector, which targets an html element by name.

CSS : Selectors

CSS

#logo {declaration}

.ingredients {declaration}

HTML

<img id = ”logo” src=”” alt=””>

<ul class = “ingredients”>

The most important difference between IDs and classes is that there can be only one ID on a page, but multiple classes.

- An ID is more specific than a class.
- An element can have both an ID and multiple classes.

CSS : Selectors

Descendant Selectors

CSS

```
#sidebar .author {declaration}
```

HTML

```
<div id="sidebar">
    <p class="author"></p>
</div>
```

A space between two selectors indicates a descendant selector. The style is targeted to an element with the class “author” inside the id “sidebar”.

CSS : Multiple classes

CSS

```
.ingredients.time {declaration}
```

HTML

```
<div class="ingredients time">
    <h1></h1>
</div>
```

Elements can have multiple classes, giving you more control. They are written in the CSS in the exact order they appear in the html, with no spaces.

CSS : Declaration

The declaration is always defined in a property/value pair. The two are separated by a colon. How you define the properties will affect how HTML elements are displayed.

element → p { **property: value** }

id → #id { **property: value** }

class → .class { **property: value** }

CSS : Declaration

You can apply multiple declarations to a selector(s) by separating the declarations with semi-colons.

```
p {
      font-family: Arial, sans-serif;
      font-size: 14px;
      color: #666666;
}
/* Comment */
```

More Information

- HTML5 Tutorial
  https://www.w3schools.com/html/
- CSS Tutorial

  https://www.w3schools.com/css/default.asp