**COMP 307**
Principles of Web Development

# COMP 307

# **Principles of Web Development**

Unit 2 – Frontend Internet Languages

Lecture 3:

Local Development & Introduction to HTML

Contents

What is HTML?
Basic HTML

# Lecture Outline

- ## Local frontend development

- ## Basic HTML4 and 5

    - Focus on building and formatting pages
    - We will talk about FORMS in a later lecture

# Readings

- Textbook reading:

  - Internet and World Wide Web – chapter 4
  - The Full Stack Developer – pages 103-121

- Online HTML educational resources:

  - https://www.w3schools.com/html/
  - https://www.tutorialspoint.com/html/index.htm

# Local Development

Local Development & Introduction to HTML

Contents

# What is local development?

Building websites on your laptop that are not connected to the Internet.

Developers will often start a project this way before transferring the code to a server that is connected to the Internet.

This means we will use a text editor to write the pages and then drag to browser to render.
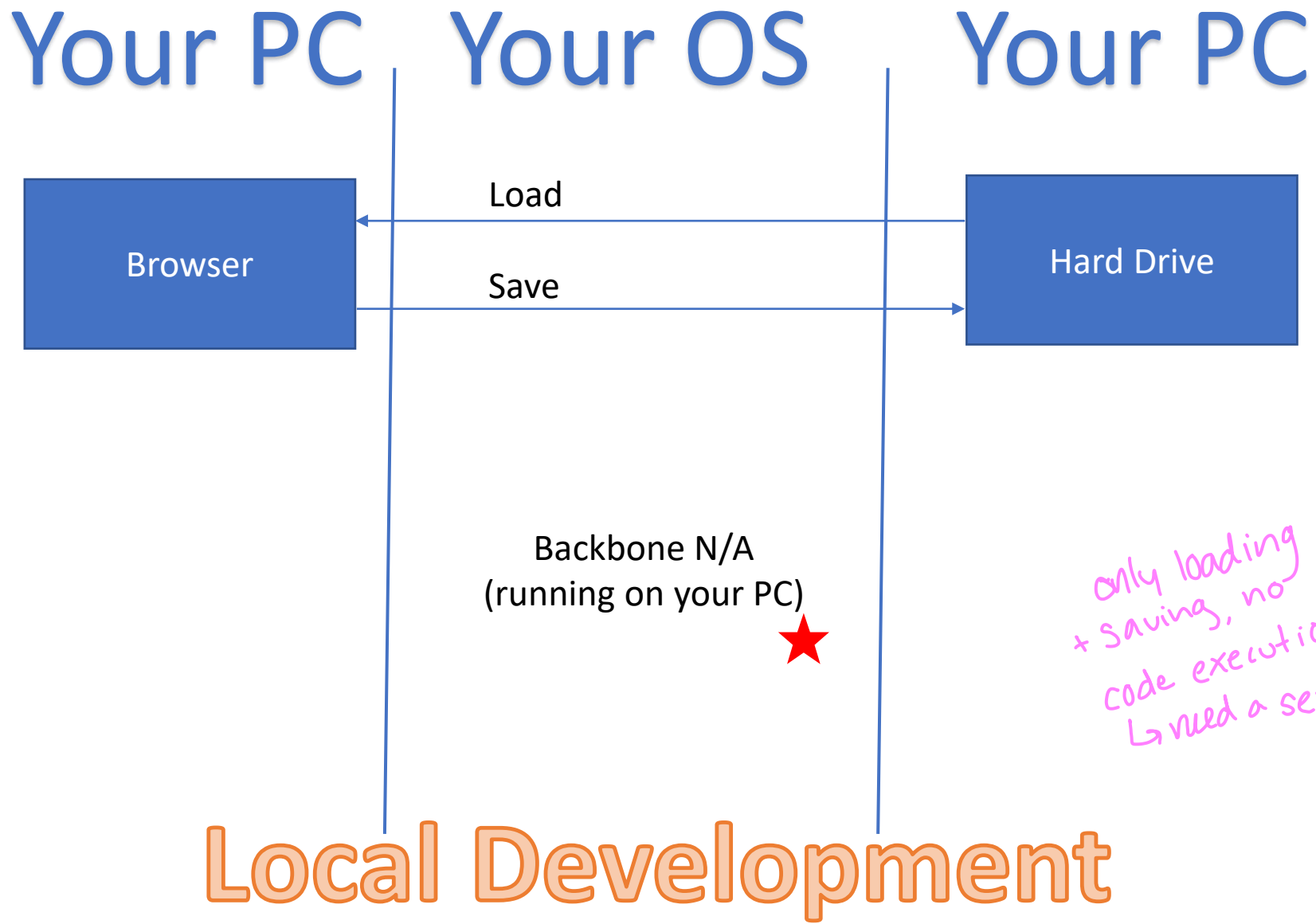
# Local Development

- You create all the programs on your computer and then run them on your browser

- Local development methods:

  - Style 1: Browser only development
  - Style 2: Browser with local server development

- Tools you need:

  - A browser
  - A text editor to write the scripts
  - Optional:
    - Compilers and/or interpreters for advanced programming
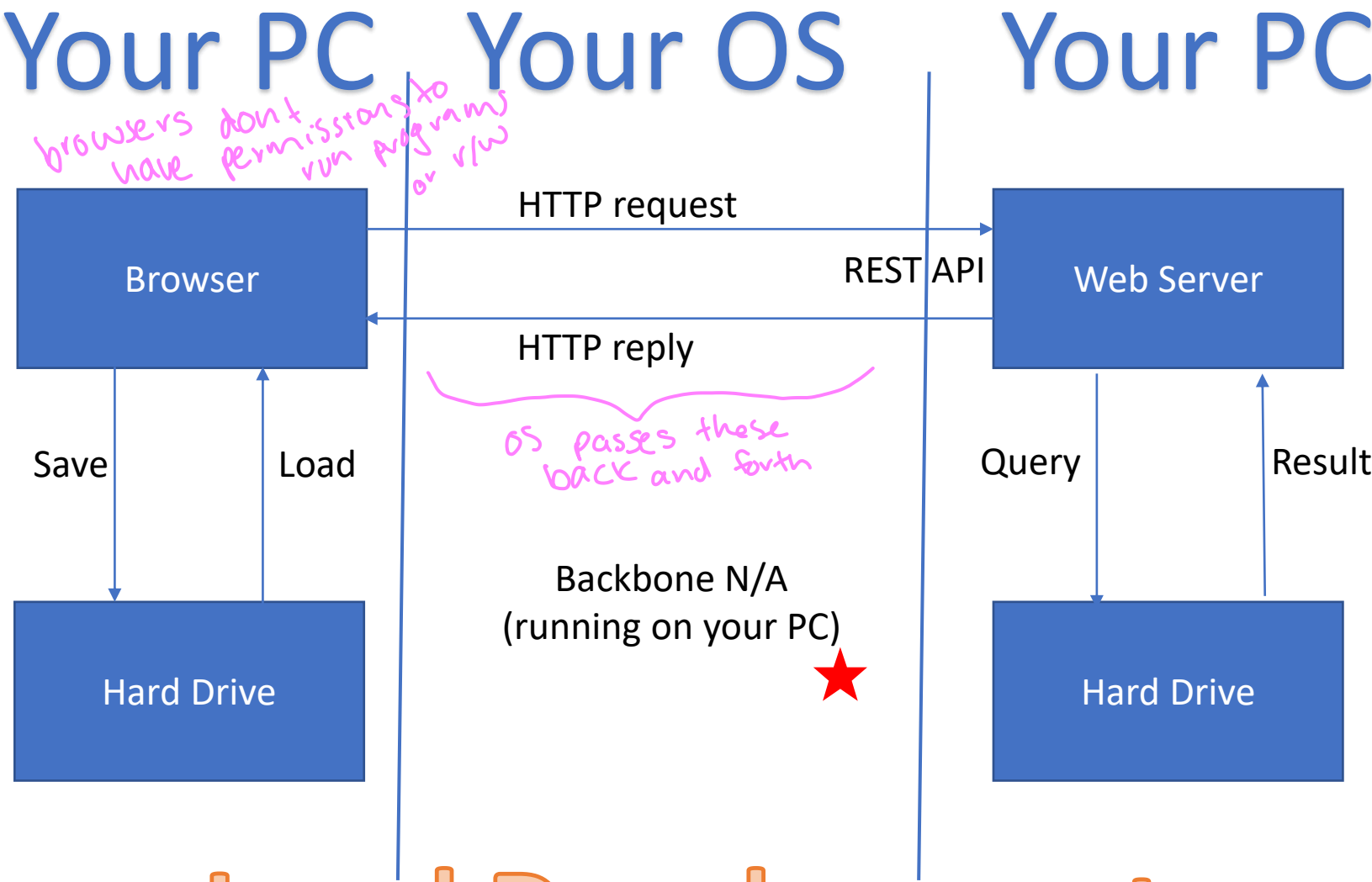    - A local web server (to simulate a true web server)
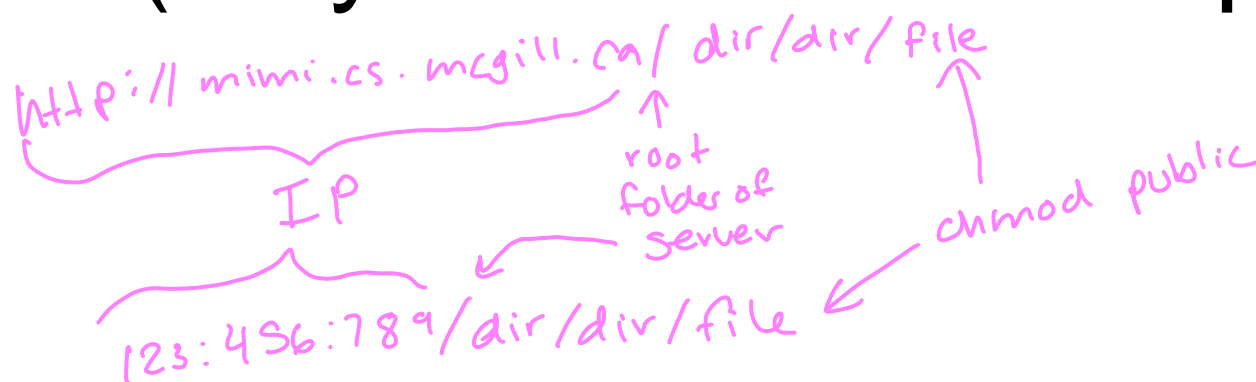
# Browser-to-Local

## Your PC  Your OS  Your PC

| Browser |
|---------|

Load

Save

| Hard Drive |
|------------|

Backbone N/A
(running on your PC)

★

*only loading
+ saving, no
code execution
↳ need a server*

## Local Development

# Browser-to-Local Server

## Your PC   Your OS   Your PC

*browsers dont have permissions to run programs or r/w*

Browser

HTTP request

REST API

Web Server

HTTP reply

*OS passes these back and forth*

Save | Load

Query | Result

Hard Drive

Backbone N/A
(running on your PC)

⭐

Hard Drive

## Local Development

# Example Local Development (only browser development)

http://mimi.cs.mcgill.ca/dir/dir/file

IP

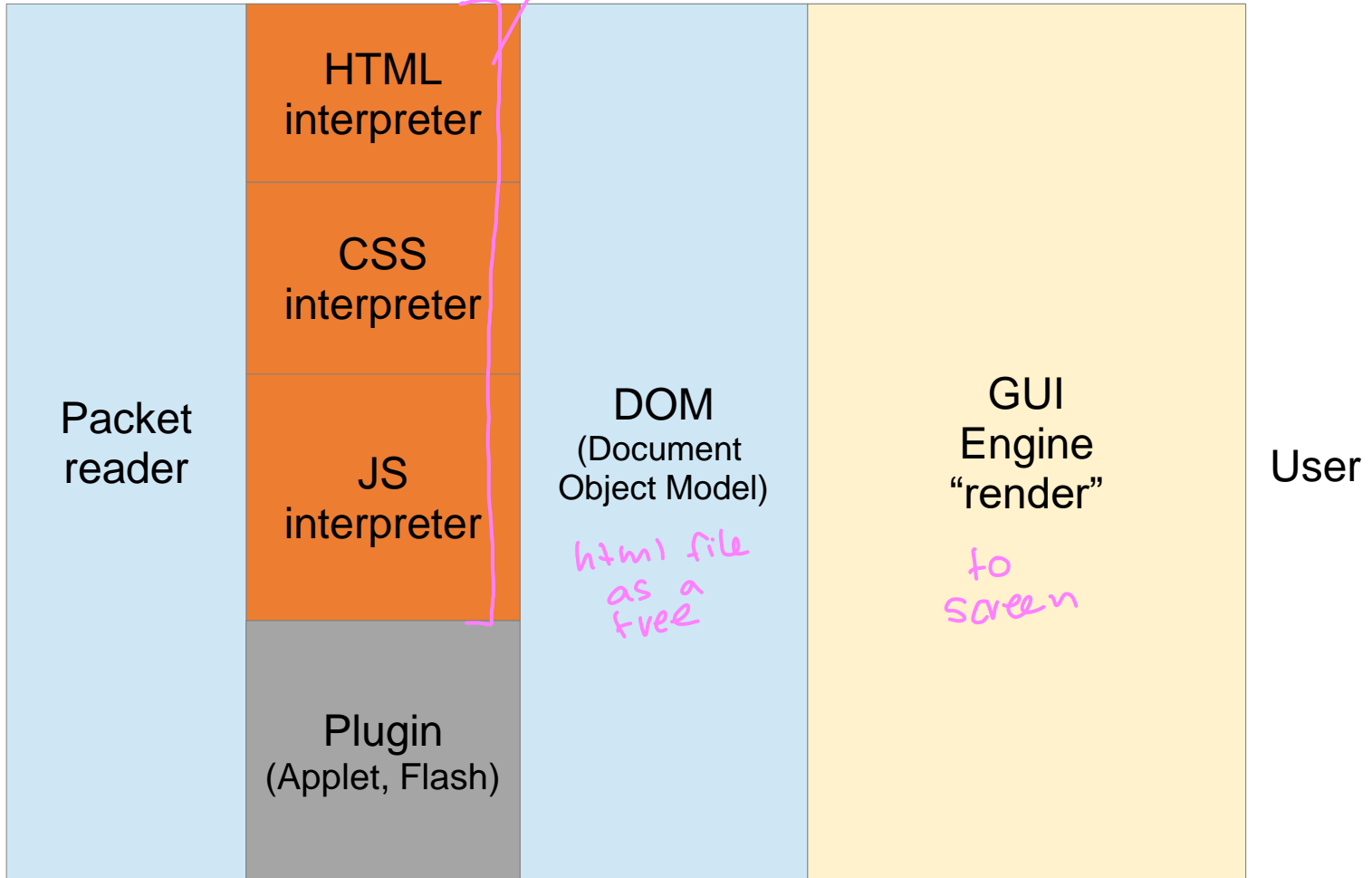↑ root folder of server

123:456:789/dir/dir/file ← chmod public

• Making a HelloWorld.html file locally and running it on a browser.

literally drag the html file and drop in browser
↳ url is the file path

# What is a browser?

Packets
From server

Standard, all browsers have

| Packet reader | HTML interpreter | DOM (Document Object Model) | GUI Engine "render" | User |
|---|---|---|---|---|
| | CSS interpreter | | | |
| | JS interpreter | | | |
| | Plugin (Applet, Flash) | | | |

html file as a tree

to screen

browser

## Contents

McGill

**COMP 307**
Principles of Web Development

# What is HTML?

Local Development & Introduction to HTML

Contents

# What is HTML?

Standardization of information sharing

Timeline of the Internet

Mosaic
1992

On Distributed
Communications
1964

Google
1998

HTML
WWW
1991
Internet
Explorer

Internet
1983

1.7 billion
Internet users
2009

ENIAC
1946

IBM 704
1954

ARPA IPTO
1962

ARPANET
1969

GML
1973

IBM PC
1981

SGML
1986

Archie
1990

Yahoo
1995

1940        1950        1960        1970        1980        1990        2000        2010
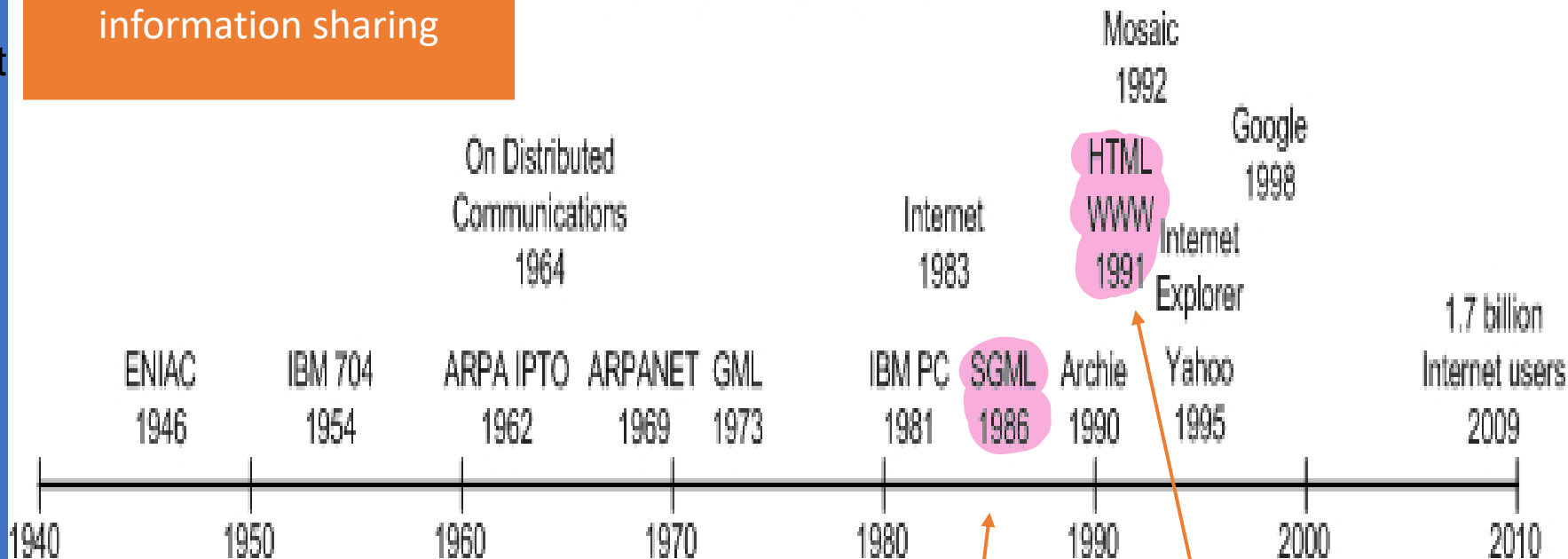
SGML parent of HTML,
XML, and others.
Basic rules for markup

HTML provided a concept
for hyper-links
(clickable links)

Standardized
Generalized
Markup
Language
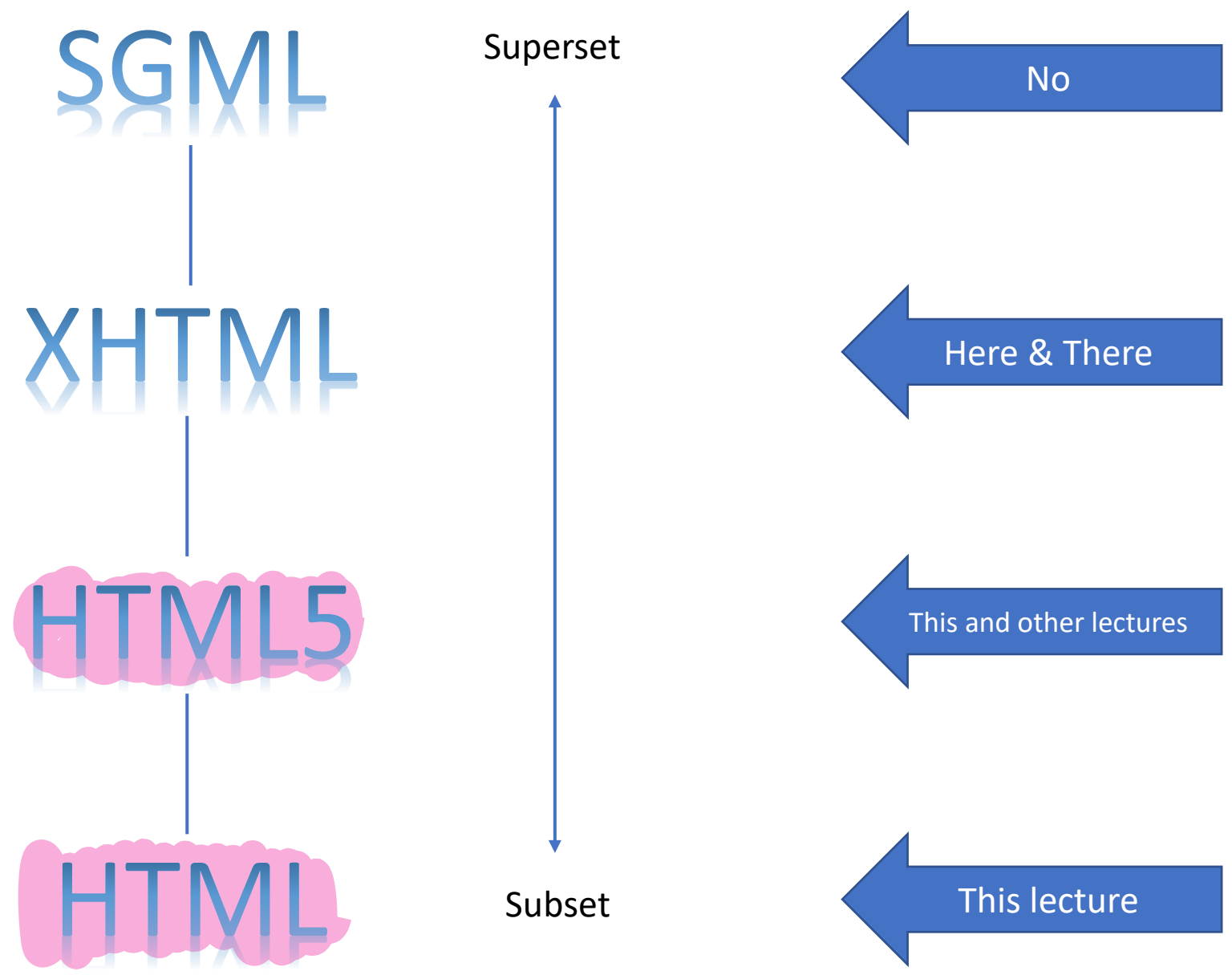
← clickable

Hyper
Text
Markup
Language

McGill                                    Vybihal (c) 2022        Apple used this first in a word processor    12

# HTML in this course

SGML

XHTML

HTML5

HTML

Superset

Subset

No

Here & There

This and other lectures

This lecture

# What is HTML?

It is not a programming language.

It is a text formatting language.

- It helps you describe how a webpage should look like
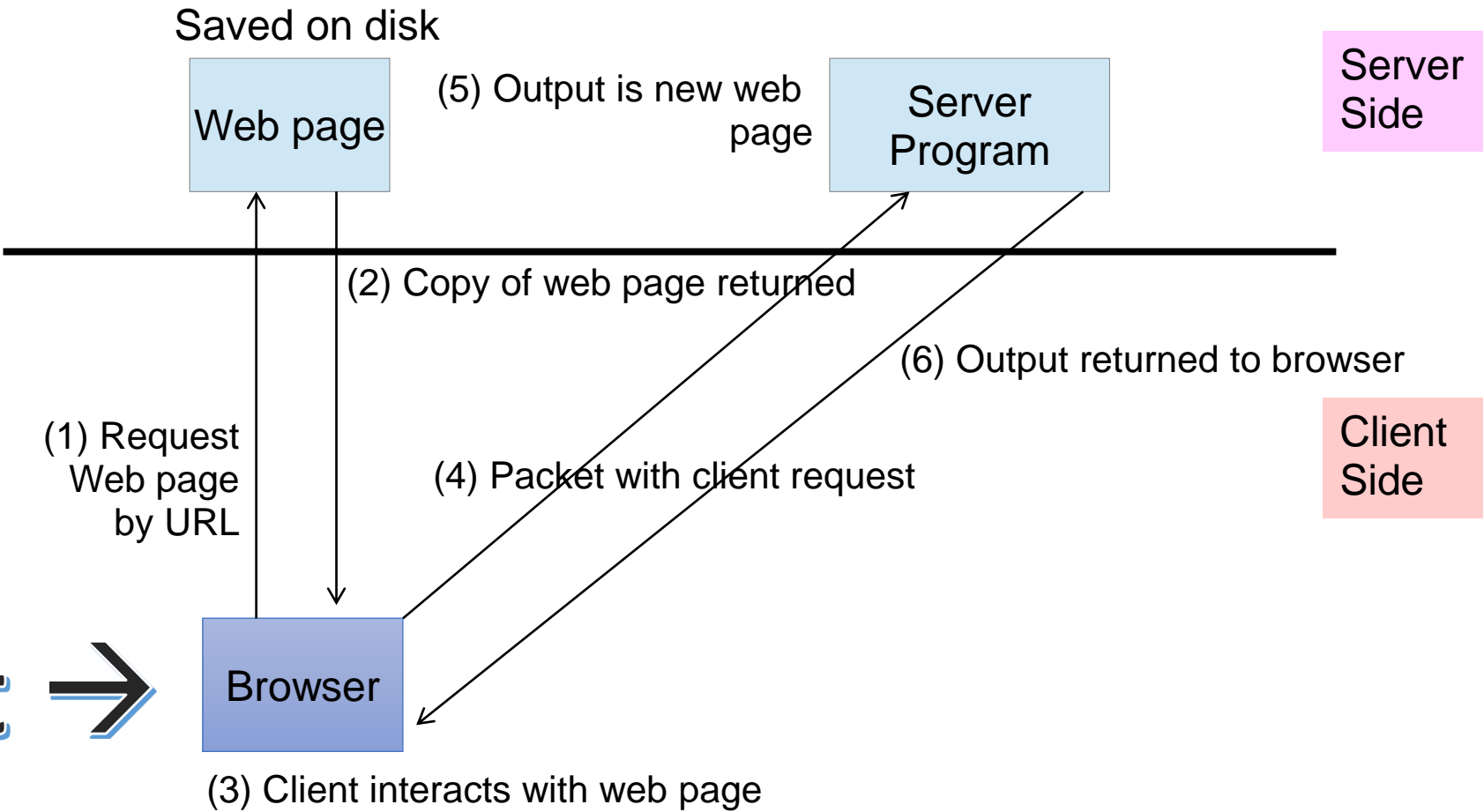
- It allows you to place content on a page

Contents

# What is HTML?

- Document meta data

  - Include files, search engine key word, comments, etc.

- Page formatting

  - Colors, backgrounds, headers, footers

- Layout formatting

  - Location of writing areas and images

- Text formatting

  - Bold, italics, font, colors, etc.

- No programming

# Front End Flowchart

Saved on disk

Web page

(5) Output is new web page

Server Program

(2) Copy of web page returned

(6) Output returned to browser

(1) Request Web page by URL

(4) Packet with client request

**Start →**

Browser

(3) Client interacts with web page

For this lecture, an HTTP reply payload contains HTML.

## Contents

# Basic HTML

Introduction to HTML

Contents

# Does HTML look the same on all browsers?

No !

Let us worry about this later...

Contents

# The fundamental unit in HTML

The **Tag**:

<tag> some text </tag>

<tag attribute="value"> some text </tag>

*modify default tag*

| | |
|---|---|
| <tag> | Represents begin |
| </tag> | Represents end |
| tag | Represents a command applied to text |
| attribute | Modifies the behavior of the command |

Contents

# How do I begin a webpage?

```
<html>
    <head>
    </head>

    <body>
    </body>
</html>
```

HEAD: page meta data

BODY: content & formatting

All HTML code in this file are within these two tags.

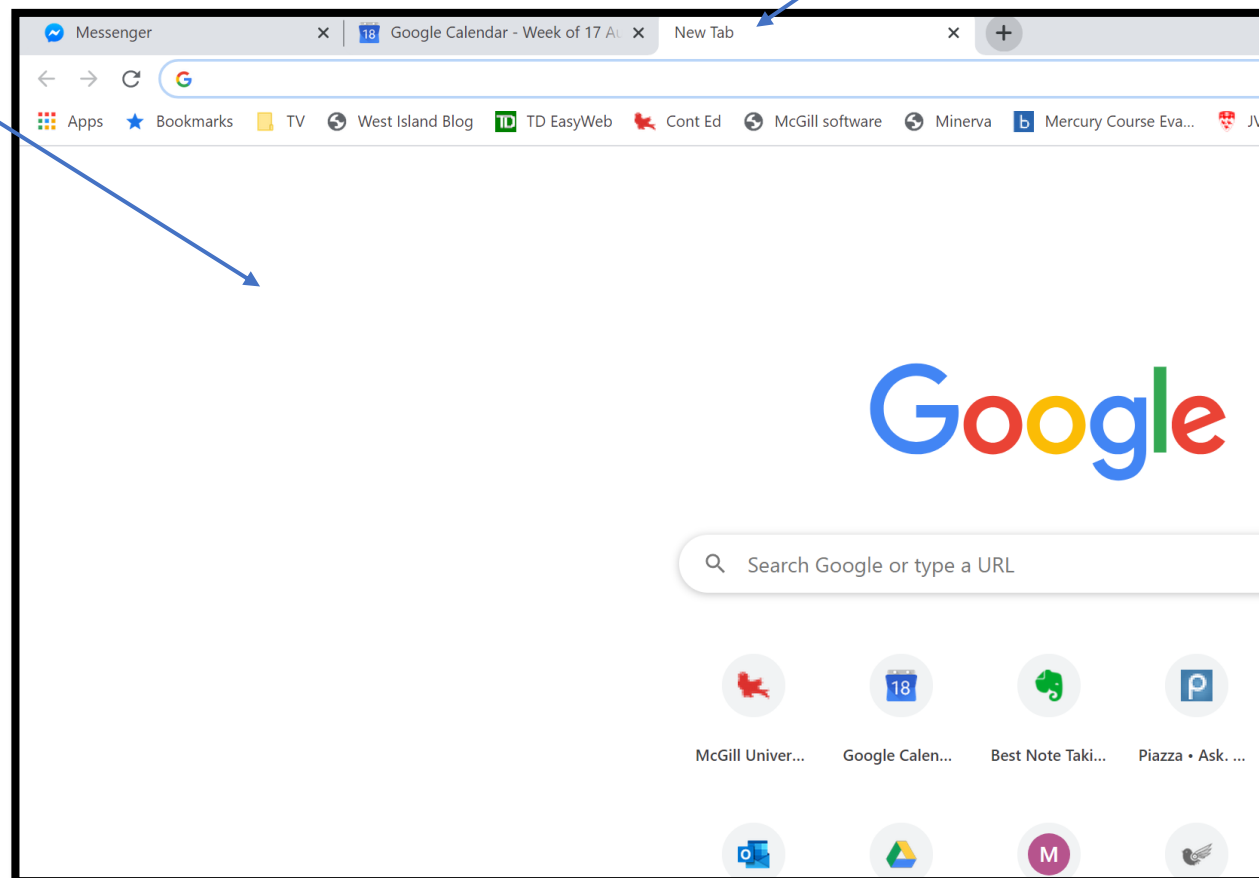# A text file "script"

**Important Note:**
All text formatting is ignored by the browser unless it is an HTML tag. Specifically, spacebar, tab and carriage returns are ignored, except for the first space between words.

Contents

# Simple Content Example

```
<html>
   <head>
       <title>This appears at browser command bar</title>
   </head>

   <body>
       Hello World!
   </body>
</html>
```

# Page Parameters Example

```html
<html>
  <head>
      <meta charset="utf-8">
      <meta name="keywords" content="used by search engines">
      <meta name="description" content="...search engines..">
      <title>This appears at browser command bar</title>
  </head>

  <body>
      Hello World!
  </body>
</html>
```

The "header arguments" do not show on your webpage. They are there to provide information to other things like: browser, server, and search engines.

Charset – tells the browser which ASCII to use
Name – tells search engines and other bots information about your webpage

# Demo

Let us create a simple web page on our local machine and then display it on the browser.

1. Open any text editor (filename.html)
2. Write HTML tags
3. Save file
4. Drag/drop or File/open using your browser

# About the **body** Tag

Your webpage goes here

We need a way to define the extent of your webpage.

<body> -- it start from this point
</body> -- anything after this is not the page

```
<body> … </body>
```

Attributes:

```
<body background="image_file_name"
      bgcolor="color_name_or_code"
      bgproperties="fixed">
```

"fixed" image stays in place as text scrolls

Contents

# Common HTML Tags

Center: &lt;center&gt;&lt;/center&gt;

Underline: &lt;u&gt;&lt;/u&gt;

Bold: &lt;b&gt;&lt;/b&gt;

Italics: &lt;i&gt;&lt;/i&gt;

Strike through: &lt;s&gt;&lt;/s&gt;

Headers: &lt;h1&gt;&lt;h2&gt;&lt;h3&gt;&lt;h4&gt;&lt;h5&gt;&lt;h6&gt;

New line: *most compatible w/ everything* **&lt;br&gt;&lt;/br&gt;** or &lt;/br&gt; or &lt;br&gt;

Paragraph: &lt;p align="position"&gt;&lt;/p&gt;

left, right, center, justify

# Example

```
<html>

  <head>

    <title>Example 2</title>

  </head>

  <body background="bubbles.jpg">

      <center><h1>Welcome to my page</h1></center>

      <p>This is a paragraph</p>

      <p>Another paragraph. It can be <b>VERY</b> long.</p>

  </body>

</html>
```

Assume .jpg file in same directory as .html file

*Power point Notes are not the same as ascii!*

Let us try this on the browser.

Contents

# Common HTML Tags

Super script:     <sup>text_here</sup>

Sub script:      <sub>text_here</sub>

Indent quote:

    <blockquote>text_here</blockquote>

*temperimental on some browses*

# Font and Color

## Fonts:

<font face="verdana" size="2" color="red"> … </font>

Colors are defined in three ways:

- Default colors

- Hexadecimal colors

- RGB colors

# Hexadecimal Colors

Are in hexadecimal: RED  GREEN  BLUE

<font color="#008000">Green text</font>
R   G   B

00 to FF are all the shades of that color

# Colours

• Colours are defined using a hexadecimal notation for the combination of Red, Green, and Blue colour values (RGB).

• The lowest value that can be given to one light source is 0 (hex #00). The highest value is 255 (hex #FF).

| Colour | Default Names | Hex | RGB |
|---|---|---|---|
| | White | #FFFFFF | 255, 255, 255 |
| | Black | #000000 | 0, 0, 0 |
| | Red | #FF0000 | 255, 0, 0 |
| | Lime (Green) | #00FF00 | 0, 255, 0 |
| | Blue | #0000FF | 0, 0, 255 |
| | Yellow | #FFFF00 | 255, 255, 0 |
| | Fuchsia | #FF00FF | 255, 0, 255 |
| | Aqua | #00FFFF | 0, 255, 255 |
| | Gray | #808080 | 128,128,128 |

# Images

## Images:

```
<img src="filename" align="left" width="10" height="20" alt="text for picture">
</img>
```

left, right, default is center

```html
<html>
<body>

<h1>The img element</h1>

<img src="img_girl.jpg" alt="Girl in a jacket"
width="500" height="600">

</body>
</html>
```

**The img element**

# Commenting

## Comments:        <! .... >

Since a comment has a begin and an end angle bracket it can appear anywhere.

```html
<html>
<body>

<!-- This is a comment -->
<p>This is a paragraph.</p>
<!-- Comments are not displayed in the browser -->

</body>
</html>
```

# What displays on the browser?

```
<html>
 <body>

    <h2>Spectacular Mountain</h2>

    <img src="pic_mountain.jpg" alt="Mountain View"
        width="304" height="228">

 </body>
</html>
```

# List Tags

Bullet lists

<ul> <li></li>... </ul>

Numbered lists

<ol> <li></li>... </ol>

Definition lists

<dl> <dt><dd>... </dl>

Contents

# What displays on the browser?

```
<html>
 <body>

    <p>Shopping list:<p>

    <ul>
        <li>Vegetables</li>
        <li>Bread</li>
        <li>Smoothie</li>
    </ul>

 </body>
</html>
```

# What displays on the browser?

```
<html>
  <head>
      <meta charset="utf-8">
      <meta name="keywords" content="used by search engines">
      <meta name="description" content="...search engines..">
      <title>Lists</title>
  </head>

  <body>
      <center><h1>About HTML</h1></center><br/>
      <p>Some short information about HTML:</p>
      <ul>
          <li>The HTML interpreter ignores any formatting
              you put in the text as you type it in. All
              spacing is ignored, carriage returns are
              ignored, tabs are ignored.
          </li>
          <li>
              By default all text is displayed sequentially
              and is only formatted by the HTML tags.
          </li>
      </ul>
  </body>
</html>
```

*don't need, adds extra space in between* (handwritten annotation pointing to `<br/>`)

## Contents

# Hyper Links

## Off page links

    `<a href="http://web address">Link text</a>`

*or path*

## On page links → *to other points in page*

- The jump off point

    `<a href="#photos">My Photos</a>`

- Jump destination *takes you here*

    `<a name="photos"></a>`

# What displays on the browser?

```html
<html>
 <body>

  <a href="https://www.w3schools.com/html/" target="_blank">
       Visit our HTML tutorial!</a>

  <p>If you set the target attribute to "_blank", the link
       will open in a new browser window or tab.</p>

 </body>
</html>
```

*Where should the link take you*

```
<a target="_blank|_self|_parent|_top|framename">

_blank              new tab
_self               overwrite the current tab      (default)
_parent             opens in parent frame
_top                opens in full body of window
framename           opens in specified iframe
```

# iFrames

```
<!DOCTYPE html>
<html>
 <body>

  <h2>HTML Iframes</h2>

  <p>You can use the height and width attributes to specify the size of the
     iframe:</p>

  <iframe src="demo_iframe.htm" height="200" width="300"
   title="Iframe Example"></iframe>

</body>

</html>
```
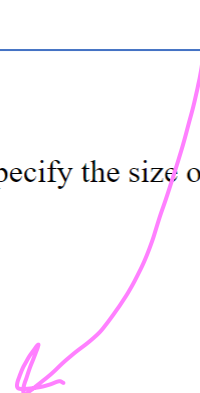
*embed
a webpage
here*

**HTML Iframes**

You can use the height and width attributes to specify the size of the iframe:

**This page is displayed in an iframe**

## Contents

# \<href> and \<iframe>

```
<!DOCTYPE html>

<html>

 <body>

  <h2>Iframe - Target for a Link</h2>

  <iframe src="firstpage.html" name="iframe_a" height="300px" width="100%"
   title="Iframe Example"></iframe>
```

↑ this goes here

```
  <p><a href="helloword.html" target="iframe_a">Click to see hello
   world</a></p>

  <p>When the target attribute of a link matches the name of an iframe, the
   link will open in the iframe.</p>

 </body>

</html>
```

Contents

# Formatting Links

COMP 307
Principles
of Web
Development

Link
Visited link
Active link

```
<html>
  <head>
      <meta charset="utf-8">
      <meta name="keywords" content="used by search engines">
      <meta name="description" content="...search engines..">
      <title>This appears at browser command bar</title>
  </head>

<body text="black" link="blue" vlink="purple" alink="red">

  <a href="https://www.w3schools.com/html/" target="_blank">
      Visit our HTML tutorial!</a>

  <p>If you set the target attribute to "_blank", the link
      will open in a new browser window or tab.</p>

  </body>
</html>
```

Contents

# HTML Tables

```
<table>
    <tr>
        <td>.A..</td>
        <td>.B..</td>
    </tr>

    <tr>
        <td>.C..</td>
        <td>.D..</td>
    </tr>
</table>
```

2X2 Table

- Graphics can be added to cell
- Cells can have a background colour
- Borders can be turned on/off
- Cell padding and spacing

| .A.. | .B.. |
|------|------|
| .C.. | .D.. |

Useful for:
- Displaying tabular information
- Basic page layout

*CSS gives more formatting control*

# What displays on the browser?

```html
<html>
 <head> <title>My Table</title> </head>
 <body>
  <table>
   <tr>
    <th>Firstname</th> <th>Lastname</th> <th>Age</th>
   </tr>
   <tr>
    <td>Jill</td> <td>Smith</td> <td>50</td>
   </tr>
   <tr>
    <td>Eve</td> <td>Jackson</td> <td>94</td>
   </tr>
  </table>
 </body>
</html>
```

*(handwritten annotations: "row" pointing to `<tr>`, "header" pointing to `</th>`, "data" pointing to `</td>`)*

*(handwritten table)*

| Firstname | lastname | age |
|-----------|----------|-----|
| Jill | Smith | 50 |
| eve | Jackson | 94 |

# Table as a layout

*span across 2 columns*

```
<table width="100%" height="100%" cellpadding="10" cellspacing="0" border="0">
<tr>  <!-- ============ HEADER SECTION ============== -->
  <td colspan="2" height="100" bgcolor="#777d6a"><h1>Website Logo</h1></td></tr>
<tr> <!-- ============ LEFT COLUMN (MENU) ============== -->
  <td width="20%" valign="top" bgcolor="#999f8e">
    <a href="#">Menu link</a><br>
    <a href="#">Menu link</a><br> <br><br><br><br><br><br><br><br><br><br><br><br>
  </td>
  <!-- ============ RIGHT COLUMN (CONTENT) ============== -->
  <td width="80%" valign="top" bgcolor="#d2d8c7">
    <h2>Page heading</h2>
    Blabla bla.</td>
</tr>
</table>
```

# Special Symbols

- Leaving an extra space:  

- Double quote: &#34; &quot;

- É: &#233; &eacute;

  *ascii*

Many more… https://www.w3schools.com/html/html_symbols.asp

# Deprecated

- Useful links:

  - https://www.codehelp.co.uk/html/deprecated.html

  - https://www.w3schools.com/

- Definition of "deprecated"

  - Items marked to be removed from a language

  - These items are often replaced by another command or language (like CSS)

Is it really deprecated?

# HTML5

Introduction to HTML

# What is HTML5

Just the latest version of HTML


- Further standardization

- Internationalization

- New features

# DOCTYPE

`<!DOCTYPE html>`

This is an SGML document definition command (or DTD).  It is optional.

Without it the browser goes into compatibility mode so it can process older styled web sites.

With it, the browser uses all HTML5 interpretation conventions.

# HTML5 Tags

We will see today:

*   "sections"   ← a nice feature that nobody uses

When we talk about Java Script and graphics:

*   <canvas>

When we talk about server communication (using forms):
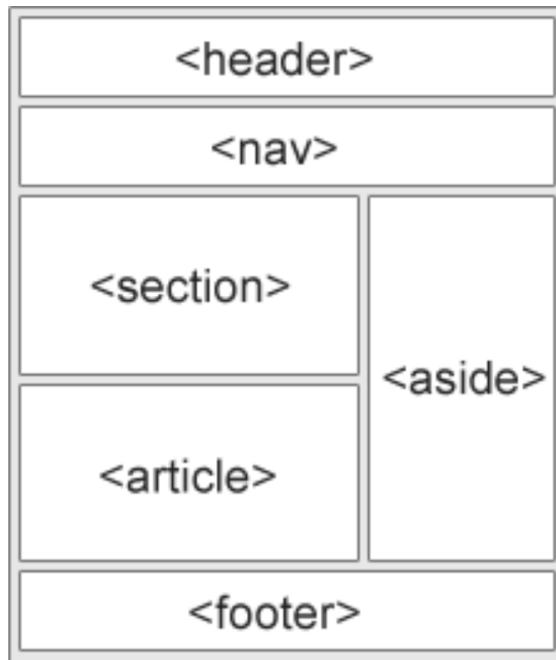
*   <color>

*   <date>

*   <time>

*   <email>

*   Etc.

Full list here: https://www.tutorialspoint.com/html5/html5_new_tags.htm

# Dividing the page into <u>Specific</u> sections

Layout Elements



- `<header>` - Defines a header for a document or a section
- `<nav>` - Defines a set of navigation links
- `<section>` - Defines a section in a document
- `<article>` - Defines an independent, self-contained content
- `<aside>` - Defines content aside from the content (like a sidebar)
- `<footer>` - Defines a footer for a document or a section
- `<details>` - Defines additional details that the user can open and close on demand
- `<summary>` - Defines a heading for the `<details>` element

## Contents

What is HTML?
Basic HTML

https://www.w3schools.com/html/html_layout.asp

*Just words make things more clear!*

They are basic layout division markers and do not style by themselves. To add style use other HTML or CSS commands.

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>CSS Template</title>
    <meta charset="utf-8">
  </head>

  <body>
    <h2>HTML5 Layout</h2>
    <p>Example</p>

    <header>
      <h2>Cities</h2>
    </header>

    <section>
      <nav>
        <ul>
          <li><a href="#">London</a></li>
          <li><a href="#">Paris</a></li>
          <li><a href="#">Tokyo</a></li>
        </ul>
      </nav>

      <article>
        <h1>London</h1>
        <p>London is the capital city of England. It is the most populous city in the
            United Kingdom, with a metropolitan area of over 13 million inhabitants.</p>
      </article>
    </section>

    <footer>
      <p>Footer</p>
    </footer>

  </body>
</html>
```

# Why have sections if they don't do anything?

In programming we use Class and Methods to help organize our code…

In scripting we also want to organize our pages.

Experience has shown that there are common ways to section a script. HTML5 introduces these new section tags.

Contents

# Generic HTML Sections

&lt;div&gt;&lt;/div&gt;   "div"

- Provides the ability to group HTML elements together with an ID

- Can be used with CSS

- Can be used with Java Script

> We will explore this next class, when we explore more advanced techniques

# History of <div>

- HTML 4 tag

- Motivation: a desire to group parts of a webpage into related purpose for styling and organizing.

- Today: basic way developers group elements of a page. More popular than HTML5's section tags… ⬅ do the same things as div

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>CSS Template</title>
    <meta charset="utf-8">
  </head>

  <body>
    <h2>HTML5 Layout</h2>
    <p>Example</p>

    <div>
      <h2>Cities</h2>
    </div>

    <div>
      <div>
        <ul>
          <li><a href="#">London</a></li>
          <li><a href="#">Paris</a></li>
          <li><a href="#">Tokyo</a></li>
        </ul>
      </div>

      <div>
        <h1>London</h1>
        <p>London is the capital city of England. It is the most populous city in the
            United Kingdom, with a metropolitan area of over 13 million inhabitants.</p>
      </div>
    </div>

    <div>
      <p>Footer</p>
    </div>

  </body>
</html>
```

# Prepare for next class

- ## Assignment

  - Mini 1 due next class

- ## Lab

  - Have you finished Lab A?

- ## Try on your own

  - Try out all the HTML from the lecture using your own copypaste.html file and drag-drop into browser.
  - Make your own modifications to see what happens.

Contents

What is HTML?
Basic HTML