Part 1: Introduction to the Course

Part 2: Introduction to Web Development with HTML5

Web Development and Security (ZEIT3119)

Week 1

Dr. Reza Rafeh

Part 1:
Introduction to the Course

Course Aims

The course aims to introduce students to core concepts and practical skills for developing server-side infrastructure for web applications.

- Analysing the requirements of web applications for real-world problems.
- Applying user-centred design to achieve great user experiences when developing web applications
- Designing and developing web applications using the state of the art web application frameworks
- Developing scalable web systems using well-designed APIs
- Using appropriate testing methods to ensure the quality of web applications
- Identifying and implementing web ethics through appropriate online behaviour, and adopting ethical standards to provide a moral framework
- Using contemporary database systems for managing user information
- Identifying and implementing security threats to a web application
- Analysing the degree of security of a web system

Course Schedule

Week	Date	Topic
1	27/02	Introduction to Web Development HTML5
2	06/03	CSS3 Bootstrap
3	13/03	User-centred Design Prototyping (online recorded lecture)
4	20/03	JavaScript Document Object Model (DOM)
5	27/03	Introduction to Ajax and jQuery
6	03/04	Introduction to PHP
Semester Break		
7	24/04	Relational Databases MySQL Integrating PHP and MySQL (online recorded lecture)
8	01/05	Model–View–Controller (MVC) Laravel API Node.js
9	08/05	Ethics and Session Management
10	15/05	OWASP and Session Fixation
11	22/05	SQL Injection and XSS
12	31/05	Broken Authentication

Course Format

- 2 hours Lecture per week: Monday 1400 - 1600
- 2 hours Lab per week: Monday 1600- 1800
  Wednesday 1500 – 1700
- Please check your timetable to determine which lab to attend.

Course Assessments

- 3 in-class quizzes.
- 2 Group Projects
- Final exam

Assessment	Weight	Group/Individual	Opening Date	Close Date
Quiz 1	5%	Individual	20/03 and 22/03	20/03 and 22/03
Quiz 2	5%	Individual	01/05 and 03/05	01/05 and 03/05
Quiz 3	5%	Individual	01/05 and 03/05	01/05 and 03/05
Project 1	20%	Group and individual	27/02	07/04 23h59
Project 2	25%	Group and individual	24/04	05/06 23h59
Final Exam	40%	Individual	During exam period	

Course Staff

Dr Reza Rafeh - Course Lecturer

Lab Demonstrator for Monday 1600-1800

Dr Faycal Bouhafs - Course Convenor Lab

Lab Demonstrator for Wednesady 1500-1700

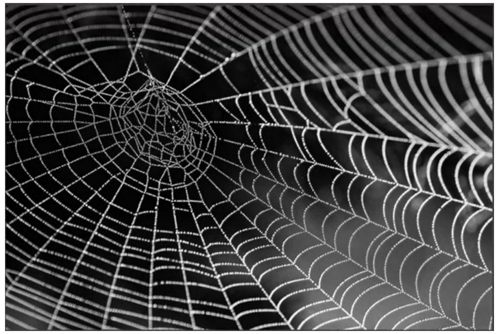
Part 2:

Introduction to Web Development with HTML5

Outline

- World Wide Web
- Website vs Web Application
- Basic Elements
- List
- Table
- Forms
- Hyperlink
- Syntax Validation
- Web Layout
- Web Browsers
- WAMP Installation

Web



- A web is a complex system of interconnected elements.
  - Elements are computers or other technology devices
  - Interconnected: Passing data between them using the Internet
  - Given this infrastructure, many different services can be provided.

Internet

- On the same network:
  - Information delivered: books, brochures, newspapers, music, films
  - Asynchronous conversations can take place. Letters or bills can be delivered using a mail service as email.
  - Synchronous conversations can take place in the form of text messaging, video conferencing, or audio telephony.
- The whole network works because of a set of rules or protocols. Each service is governed by its own rules or protocols which you would need to know in order to be able to use the mail service or a video conferencing service.

Class Activity

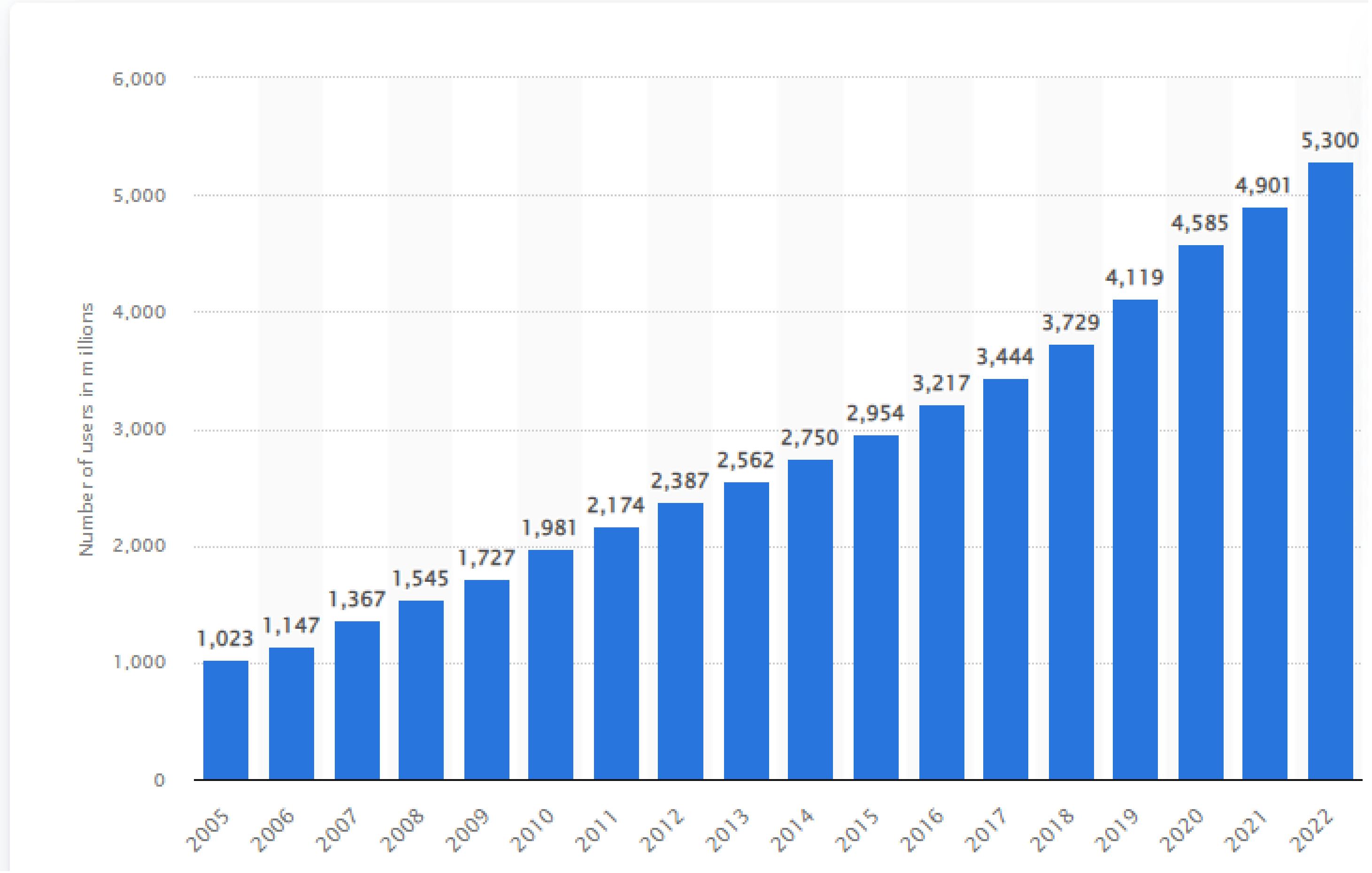
- Give examples of protocols used for these services:
  - Communication protocol for the Internet **TCP/IP**
  - To access web contents (web pages) **HTTP, HTTPS**
  - Email **POP3, SMTP**
  - File transfer **FTP**
  - Video streaming and online gaming **UDP**

What is a Website?

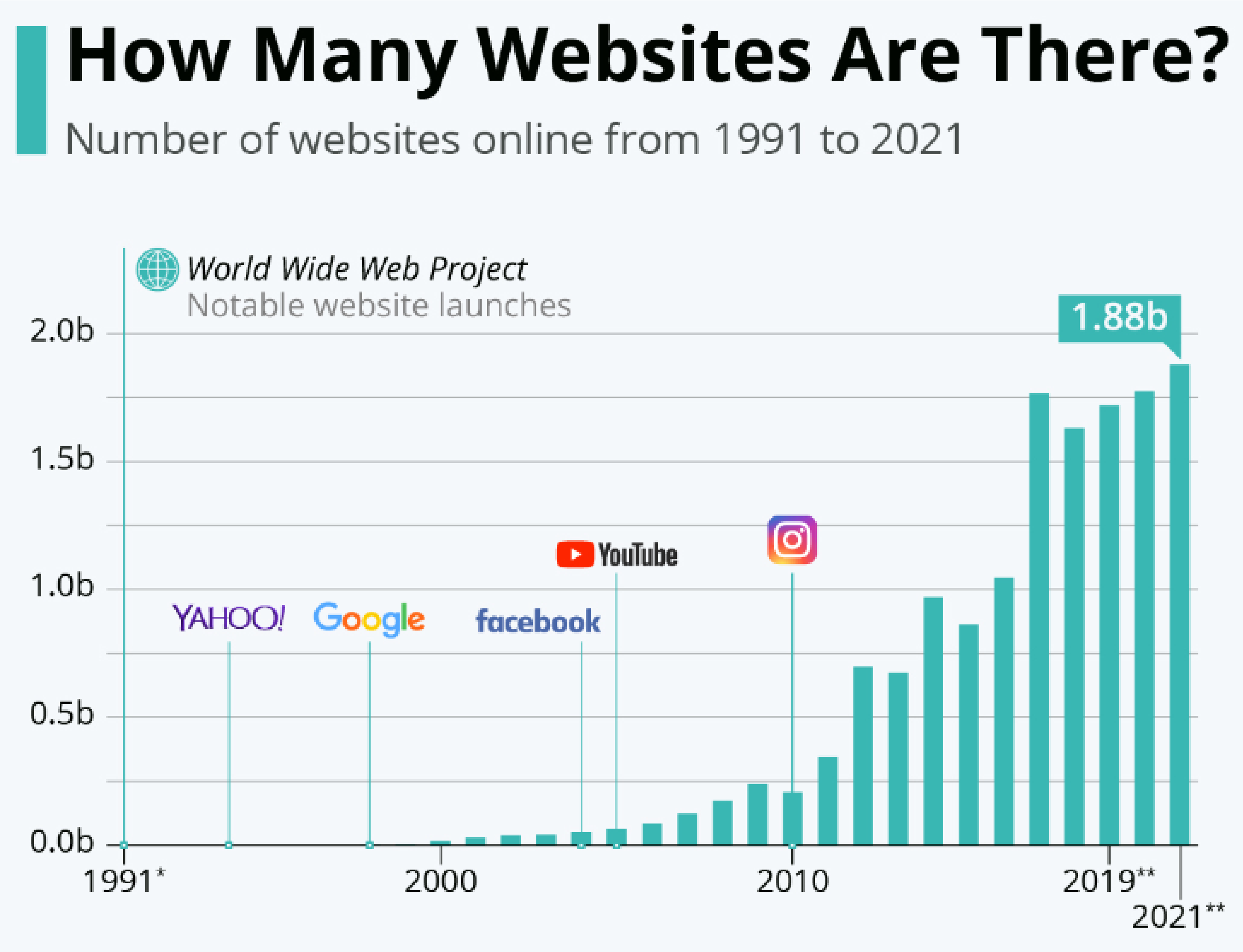
- Pages that can be visited on internet
- Providing information from all around the world
- Need to follow rules and regulations:
  - Client-Sever architecture standard
- Different types of websites
  - Static - information does not change - Wikipedia
  - Dynamic – Update the information based on user e.g login form
  - eCommerce – trading websites

Number of internet users worldwide from 2005 to 2022

(in millions)



Source: www.statista.com



How Many Websites Are There?

Number of websites online from 1991 to 2021

Source: www.statista.com

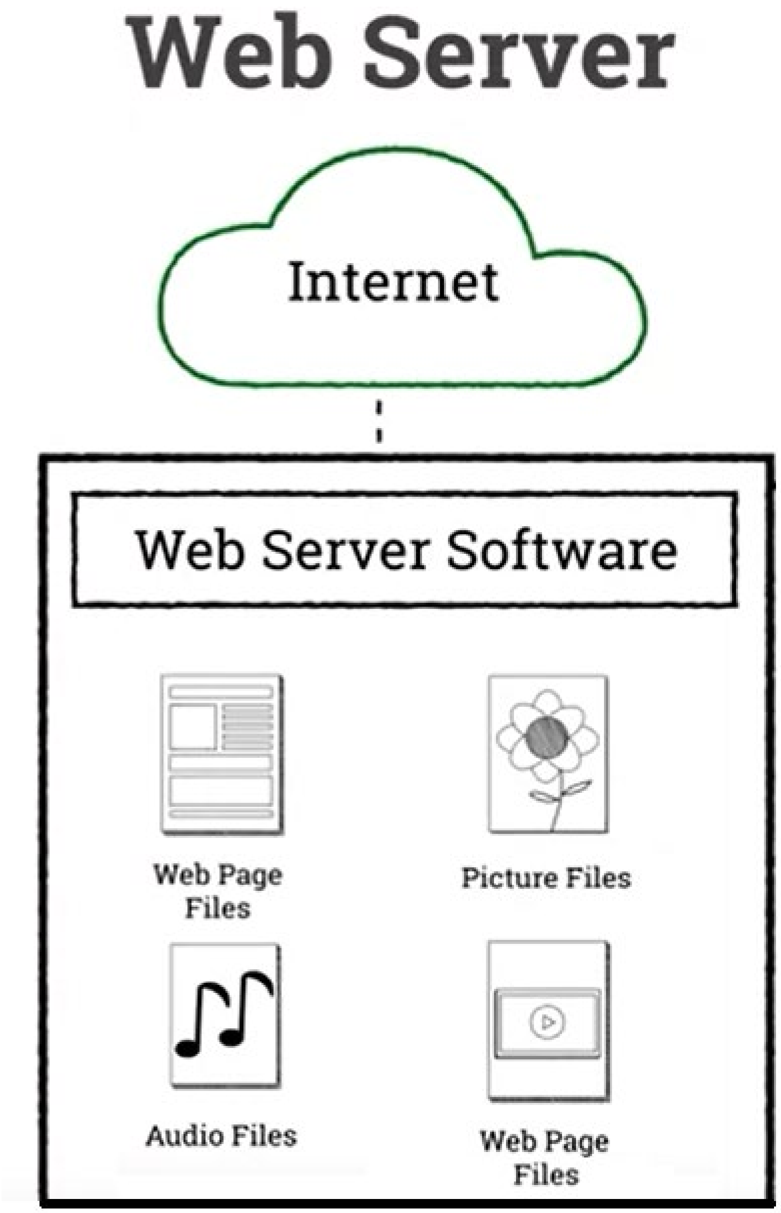
Only 17% of websites are active

Browsing vs Navigation

- Imagine reading a newspaper
  - We could read from cover to cover: browsing
  - We could jump straight to the back to the sports pages: navigation
- Imagine using a library
  - we could browse along shelves looking for any interesting book: browsing
  - We could look for a book by its topic number, going directly to the book based on its topic number: Navigation
- We could use similar techniques to navigate or browse the Web. Navigation can happen using hyperlinks, buttons, menu items, etc.

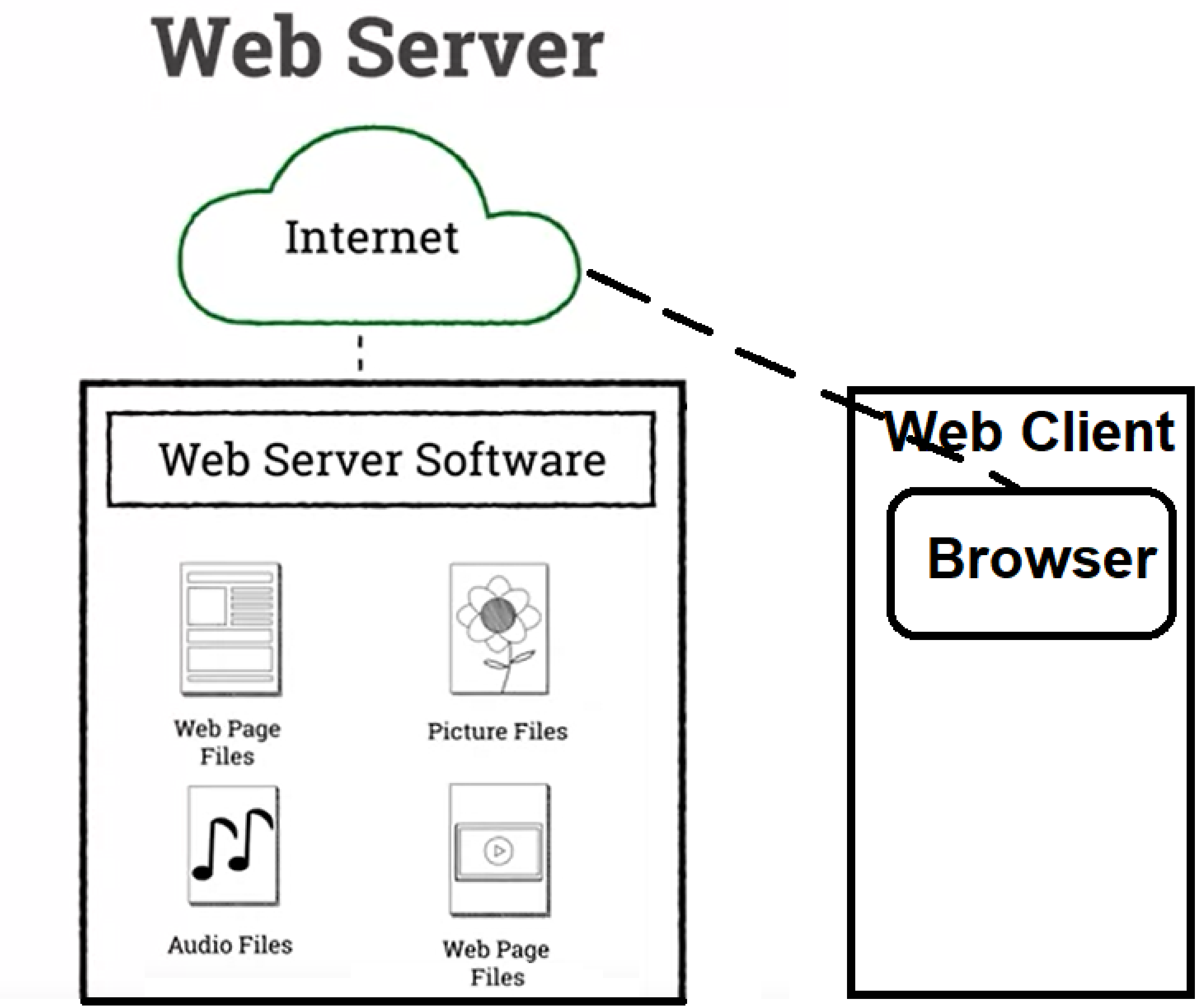
Web Server

- Server has software that allows it to send not only web pages, but also picture files and other media across the Internet
- File store for web pages and any additional material such as pictures, audio and video clips.



Web Client

- Some software running on a device to retrieve the information that is available in other places
- The software is able to present some or all text, pictures, graphics, sound, video, games, or other applications.
- This software is known as browser.



Website - Basic Parts

- Websites consist of three parts
  - GUI
    - web pages that you visit
    - For GUI building HTML is used from long time
  - Coding
    - logic that provides functionality or makes website dynamic
  - Database
    - manages data provided by end user

HTML5, CSS and JavaScript

- Web pages are written using HTML
- HTML stands for HyperText Markup Language.
- Web browsers render the page according to the HTML code.
- CSS allows for styling and customized look and feel with your HTML code
- JavaScript lets us create dynamic interaction and gives us the ability to manipulate html elements using the Document object model (DOM).

HTML5, CSS and JavaScript

- HTML – For structure and content.
- CSS – For layout, look and feel.
- JavaScript – For dynamic elements.

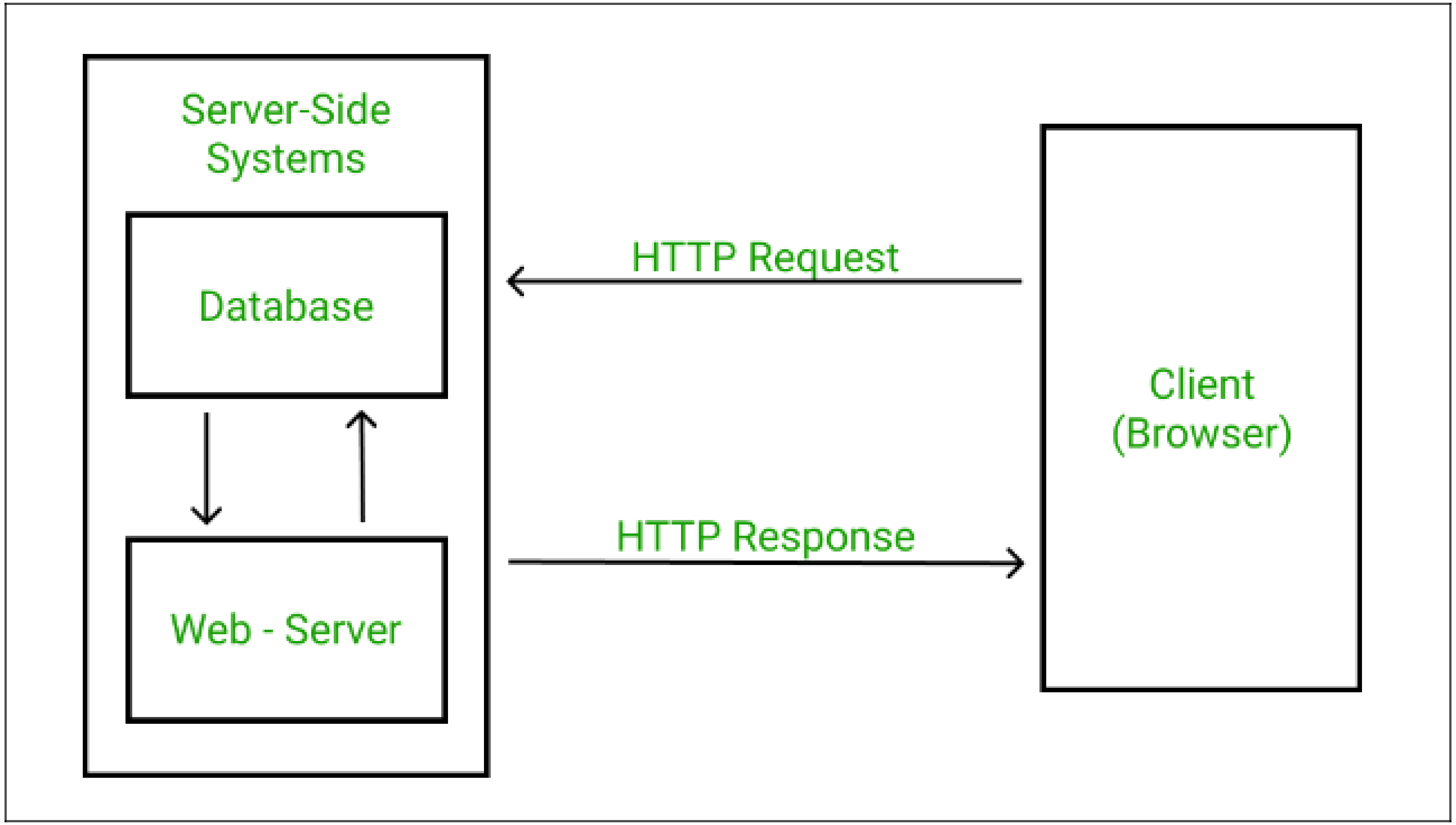
Note: All web browsers use HTML, CSS and JavaScript to display webpages you visit.

HTML5, CSS and JavaScript

- Section 1 (Client Side)
  - HTML5
  - CSS3
  - JavaScript

- Section 2 (Server Side)
  - PHP
  - MySQL

HTTP Request-Response Cycle

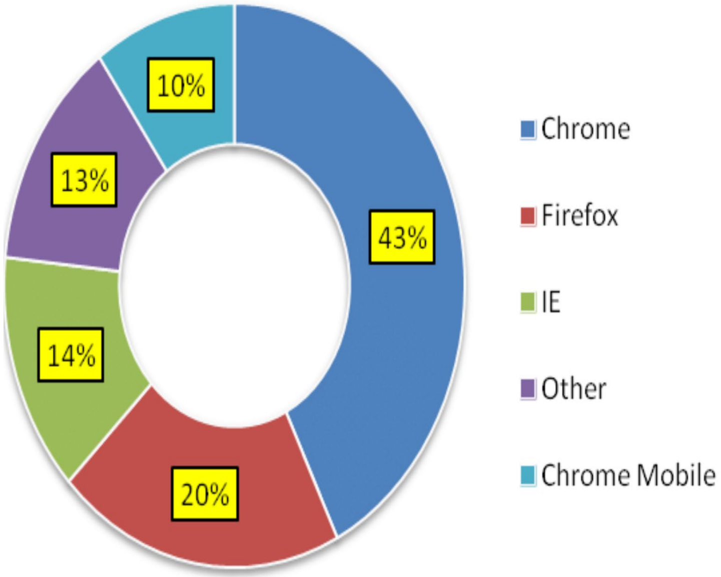


Web Browsers

- Web design should look great in all major browsers.
- First, ensure the web design to be properly rendered in the

  commonly used browsers
- Then, add enhancements with CSS / JavaScript and other

  new technologies in most recent versions.



Web Browsers

- One useful tool in Mozilla Firefox browsers is Page Inspector, which can be invoked through the shortcut Ctrl + Shift + C (luckily, this shortcut is the same on Google Chrome browsers, to invoke a similar tool called “Elements”).

- The tool allows you to modify the codes on the fly, aiding web developers in editing and viewing at the same time.
- It can be a disadvantage as well, the Page Inspector reveals the HTML code which makes it easy to copy web pages. This means that is easy to fake websites.
- The client side (in this case web browsers) is supplied with every relevant code to display a web page. The clients can also reveal the codes supplied to them.

Web Application

- Web Application refers to a package that displays dynamic web pages.
- It is an application program that is stored on a remote server and delivered through internet through browser interface e.g IE, Chrome, firefox
- They don’t need to be downloaded
- To operate a web application it requires a web server, application server and a database
- The display of dynamic pages depends on the data that the users previously supply.

Benefits

- Allowing multiple users accessing the same version of application
- They don’t need to be installed or downloaded
- Can be accessed through multiple browsers
- They can be accessed through various platform like laptops, desktops or mobile

Web Application vs Website

Web Application	Website
Web application is designed for interaction with end users.	Website basically contains static content.
The user of web application can read the content of web application and also manipulate the data.	The user of website only can read the content of website but not manipulate .
The web application site should be precompiled before deployment.	The website does not need to be precompiled .
The function of a web application is quite complex.	The function of website is simple.
Web application is interactive for users.	Website is not interactive for users.
The browser capabilities involved with a web application is high.	The browser capabilities involved with web site is high.
Integration is complex for web application because of its complex functionality.	Integration is simpler for web site.
Web application mostly requires authentication	In website authentication is not necessary.

Source: https://www.geeksforgeeks.org/difference-between-web-application-and-website/

WAMP/MAMP/LAMP Server

- WAMP - “Windows, Apache, MySQL, and PHP”
- MAMP - “Mac, Apache, MySQL, and PHP,”
- LAMP - “Linux, Apache, MySQL, and PHP.”
- Comes in the form of package that bind the bundled program together
- Always download the latest version

Basic Elements - Definition Of HTML

- HTML, also known as HyperText Markup Language
- It is used to create Web pages
- Using HTML, you can create a Web page with text, graphics, animation, sound and video
- The essence of HTML is tags
- A tag is a keyword enclosed by angle bracket (Example <I>)
- There are opening and closing tags. The text is between the tag

```
<!DOCTYPE html>
<html>
    <head>
        <title>
        </title>
    </head>
    <body>
    </body>
</html>
```

HTML Components

HTML is composed of a bunch of nodes (also tags, elements) organized as a tree.

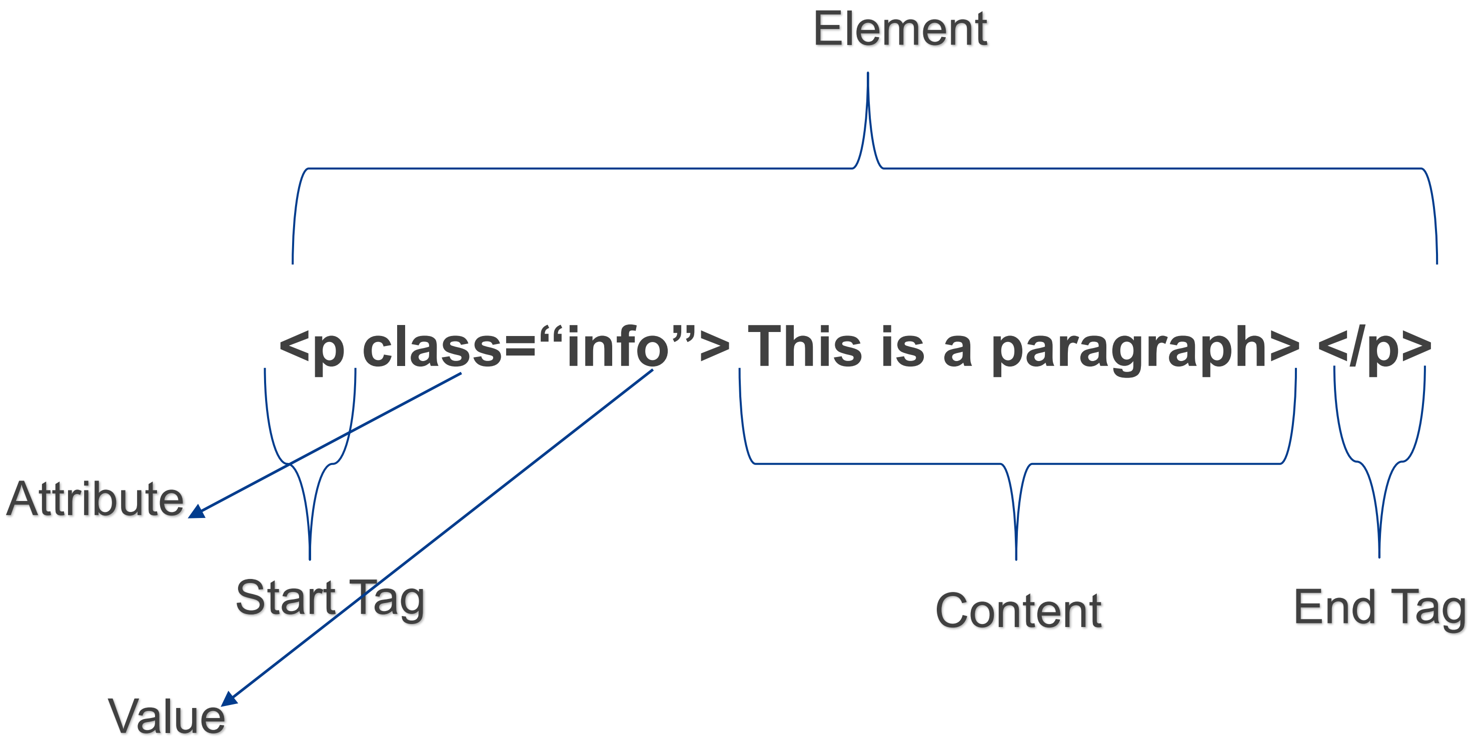
Each HTML node contains

- Content
- Attributes
- Other nodes

HTML nodes represent semantics, or meaning. For example, the <title> node represents the title of the document.

HTML Elements

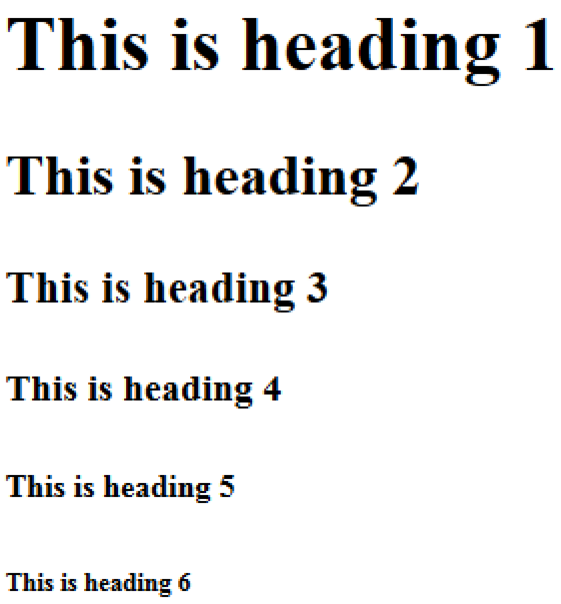
Anatomy of an HTML Element



Basic Elements - Structure of HTML

- <title> Displayed on the browser tab </title>

- <h1> Displayed as bold font weight and in large font size. </h1>



This is heading 1

This is heading 2

This is heading 3

This is heading 4

This is heading 5

This is heading 6

Basic Elements

```
<!DOCTYPE html>

<html>

    <body>

        <h1>My First Heading</h1>

        <p>My first paragraph.</p>

    </body>

</html>
```

Example 2

```
<!DOCTYPE html>
<html>
    <body>
        <h1>Black Goose Bistro</h1>
        <h2>The Restaurant</h2>
        <p>The Black Goose Bistro offers casual lunch and dinner
fare        in a hip atmosphere. The menu changes regularly to highlight
the freshest ingredients. </p>
        <h2>Catering</h2>
        <p>You have fun... <i>we'll handle the cooking </i>. Black Goose
Catering can handle events from snacks for bridge club to elegant
corporate fundraisers.</p>
    </body>
</html>
```

Resources for learning HTML, CSS and JavaScript

- **Brackets**

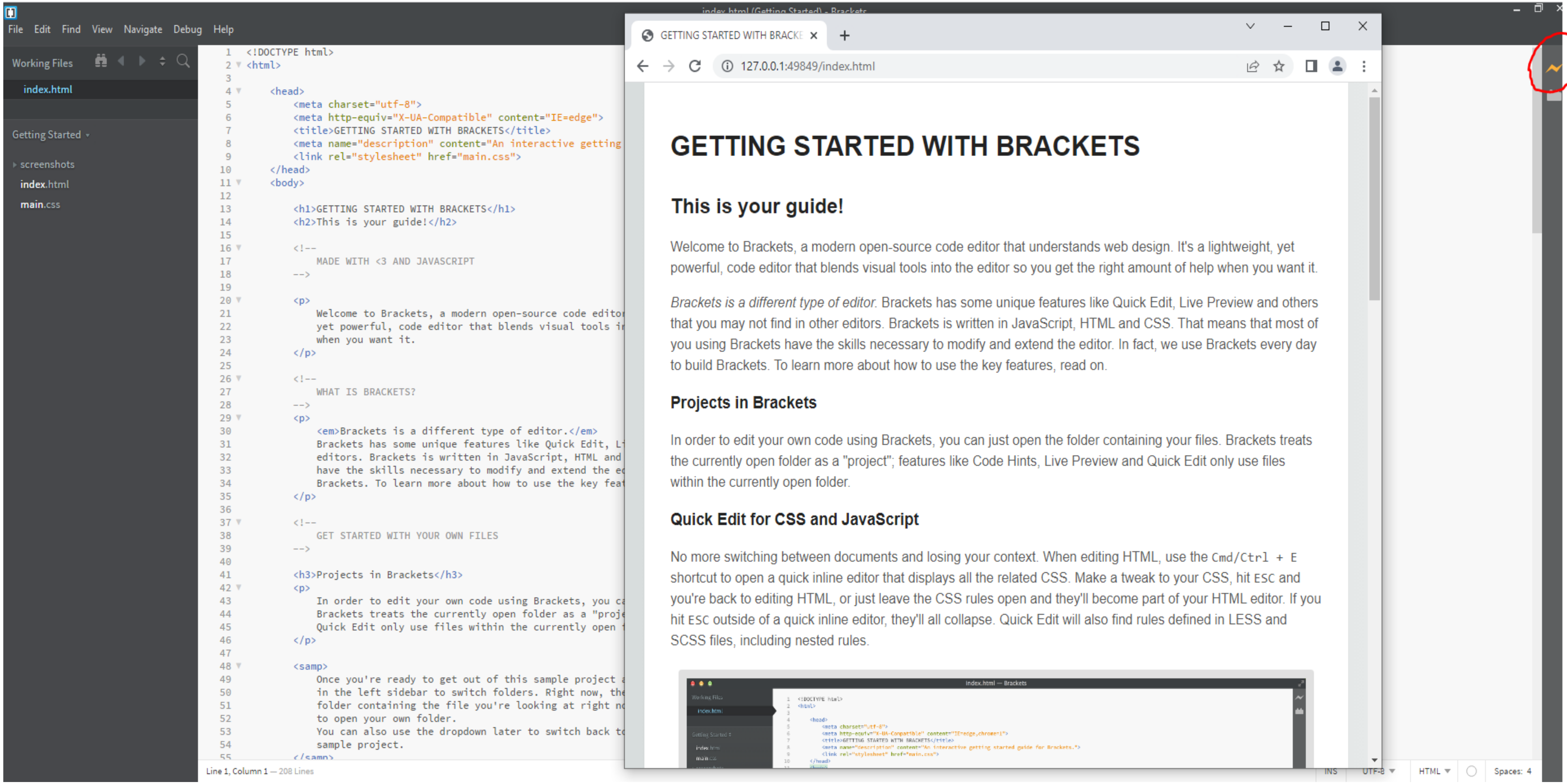
Brackets is a free, modern open-source text editor made especially for Web Development.

Features

- Live preview lets you see changes instantly on screen
- Inline editors that let you work on code side-by-side without any popups

Download Brackets from the following link:

https://sourceforge.net/projects/brackets.mirror/



```
<!DOCTYPE html>
<html>

    <head>
        <meta charset="utf-8">
        <meta http-equiv="X-UA-Compatible" content="IE=edge">
        <title>GETTING STARTED WITH BRACKETS</title>
        <meta name="description" content="An interactive getting
        <link rel="stylesheet" href="main.css">
    </head>
    <body>

        <h1>GETTING STARTED WITH BRACKETS</h1>
        <h2>This is your guide!</h2>

        <!--
            MADE WITH <3 AND JAVASCRIPT
        -->

        <p>
            Welcome to Brackets, a modern open-source code edito
            yet powerful, code editor that blends visual tools i
            when you want it.
        </p>

        <!--
            WHAT IS BRACKETS?
        -->
        <p>
            <em>Brackets is a different type of editor.</em>
            Brackets has some unique features like Quick Edit, L
            editors. Brackets is written in JavaScript, HTML and
            have the skills necessary to modify and extend the e
            Brackets. To learn more about how to use the key fea
        </p>

        <!--
            GET STARTED WITH YOUR OWN FILES
        -->

        <h3>Projects in Brackets</h3>
        <p>
            In order to edit your own code using Brackets, you c
            Brackets treats the currently open folder as a "proj
            Quick Edit only use files within the currently open
        </p>

        <samp>
            Once you're ready to get out of this sample project
            in the left sidebar to switch folders. Right now, th
            folder containing the file you're looking at right n
            to open your own folder.
            You can also use the dropdown later to switch back t
            sample project.
        </samp>
```

Line 1, Column 1 — 208 Lines

GETTING STARTED WITH BRACKETS

This is your guide!

Welcome to Brackets, a modern open-source code editor that understands web design. It's a lightweight, yet powerful, code editor that blends visual tools into the editor so you get the right amount of help when you want it.

Brackets is a different type of editor. Brackets has some unique features like Quick Edit, Live Preview and others that you may not find in other editors. Brackets is written in JavaScript, HTML and CSS. That means that most of you using Brackets have the skills necessary to modify and extend the editor. In fact, we use Brackets every day to build Brackets. To learn more about how to use the key features, read on.

Projects in Brackets

In order to edit your own code using Brackets, you can just open the folder containing your files. Brackets treats the currently open folder as a "project"; features like Code Hints, Live Preview and Quick Edit only use files within the currently open folder.

Quick Edit for CSS and JavaScript

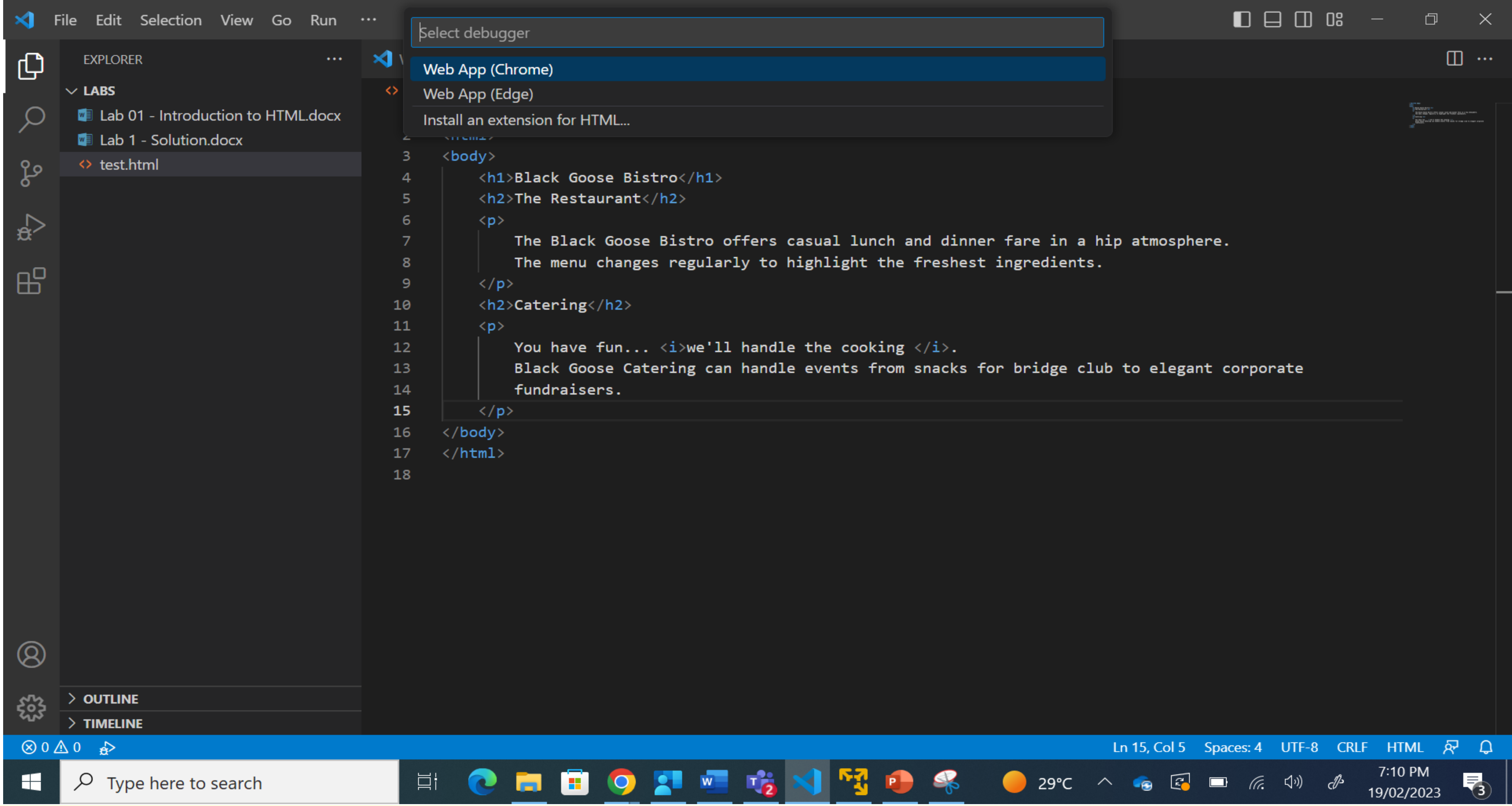
No more switching between documents and losing your context. When editing HTML, use the `Cmd/Ctrl + E` shortcut to open a quick inline editor that displays all the related CSS. Make a tweak to your CSS, hit `ESC` and you're back to editing HTML, or just leave the CSS rules open and they'll become part of your HTML editor. If you hit `ESC` outside of a quick inline editor, they'll all collapse. Quick Edit will also find rules defined in LESS and SCSS files, including nested rules.

Resources for learning HTML, CSS and JavaScript

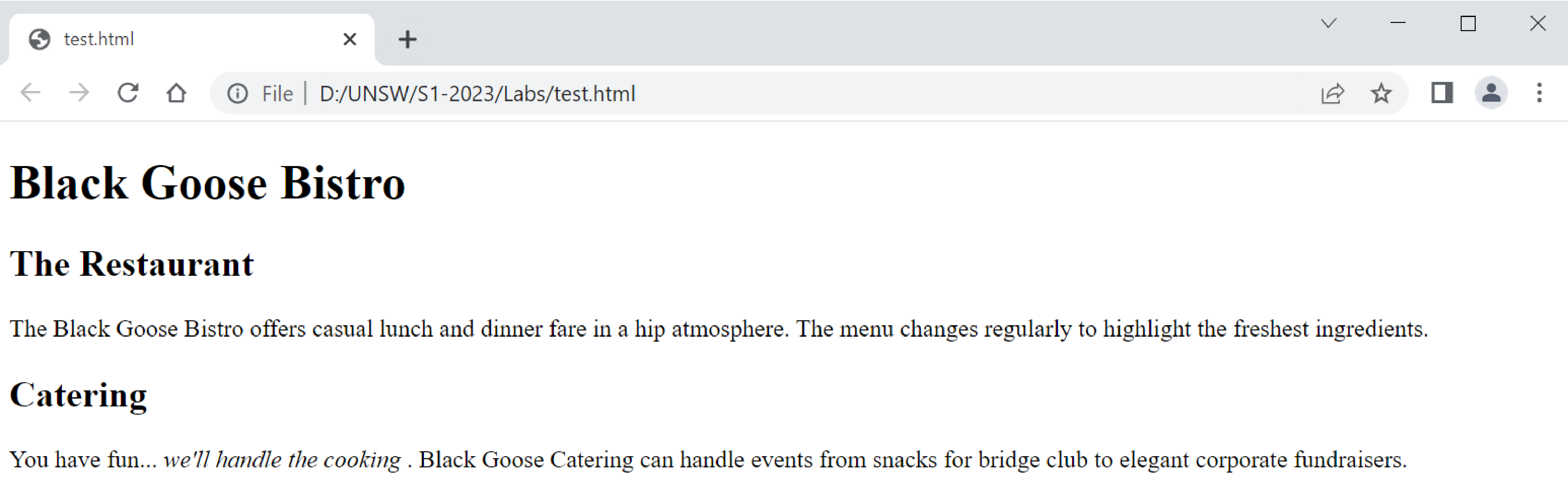
- **Visual Studio Code**

Visual Studio Code provides basic support for HTML programming out of the box. There is syntax highlighting, smart completions with IntelliSense, and customizable formatting. VS Code also includes great Emmet support.

https://code.visualstudio.com/download



```
<html>
<body>
    <h1>Black Goose Bistro</h1>
    <h2>The Restaurant</h2>
    <p>
        The Black Goose Bistro offers casual lunch and dinner fare in a hip atmosphere.
        The menu changes regularly to highlight the freshest ingredients.
    </p>
    <h2>Catering</h2>
    <p>
        You have fun... <i>we'll handle the cooking </i>.
        Black Goose Catering can handle events from snacks for bridge club to elegant corporate
        fundraisers.
    </p>
</body>
</html>
```



Black Goose Bistro

The Restaurant

The Black Goose Bistro offers casual lunch and dinner fare in a hip atmosphere. The menu changes regularly to highlight the freshest ingredients.

Catering

You have fun... *we'll handle the cooking* . Black Goose Catering can handle events from snacks for bridge club to elegant corporate fundraisers.

Resources for learning HTML, CSS and JavaScript

- **Chrome**

Chrome has developer tools which makes it easy to debug your code.

Chrome is the dominant browser these days and it makes sense to develop your web pages using chrome

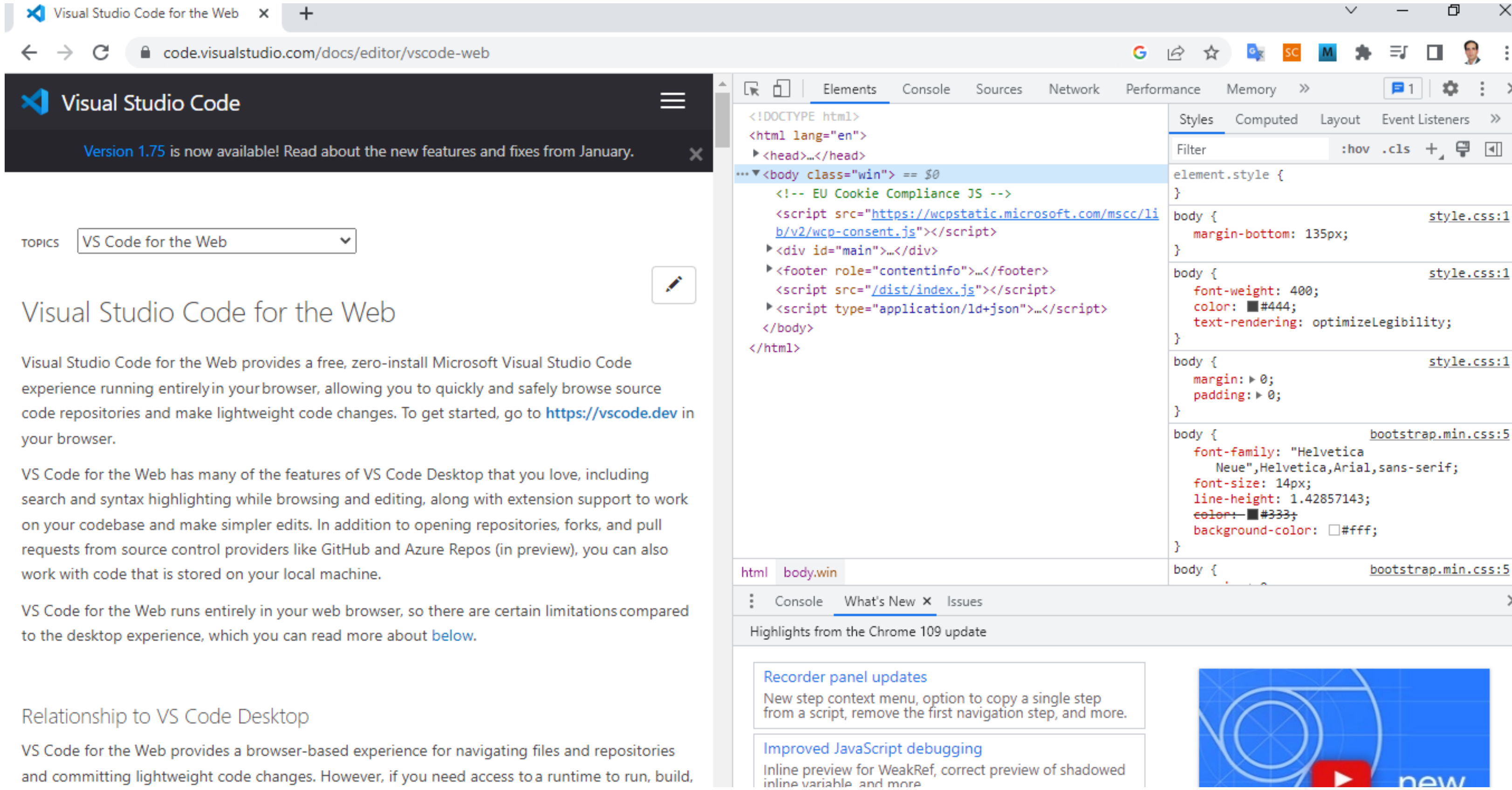
Resources for learning HTML, CSS and JavaScript

- **Chrome**

Developer tools can be accessed as follows:

Click on 3 dots in top right corner → More Tools → Developer Tools.

Or press ctrl+shift+c



Version 1.75 is now available! Read about the new features and fixes from January.

TOPICS VS Code for the Web

Visual Studio Code for the Web

Visual Studio Code for the Web provides a free, zero-install Microsoft Visual Studio Code experience running entirely in your browser, allowing you to quickly and safely browse source code repositories and make lightweight code changes. To get started, go to https://vscode.dev in your browser.

VS Code for the Web has many of the features of VS Code Desktop that you love, including search and syntax highlighting while browsing and editing, along with extension support to work on your codebase and make simpler edits. In addition to opening repositories, forks, and pull requests from source control providers like GitHub and Azure Repos (in preview), you can also work with code that is stored on your local machine.

VS Code for the Web runs entirely in your web browser, so there are certain limitations compared to the desktop experience, which you can read more about below.

Relationship to VS Code Desktop

VS Code for the Web provides a browser-based experience for navigating files and repositories and committing lightweight code changes. However, if you need access to a runtime to run, build,

```
<!DOCTYPE html>
<html lang="en">
 ▶<head>…</head>
…▼<body class="win"> == $0
    <!-- EU Cookie Compliance JS -->
    <script src="https://wcpstatic.microsoft.com/mscc/li
    b/v2/wcp-consent.js"></script>
   ▶<div id="main">…</div>
   ▶<footer role="contentinfo">…</footer>
    <script src="/dist/index.js"></script>
   ▶<script type="application/ld+json">…</script>
  </body>
</html>
```

```
element.style {
}
body {                                   style.css:1
   margin-bottom: 135px;
}
body {                                   style.css:1
   font-weight: 400;
   color: ■#444;
   text-rendering: optimizeLegibility;
}
body {                                   style.css:1
   margin: ▶ 0;
   padding: ▶ 0;
}
body {                           bootstrap.min.css:5
   font-family: "Helvetica
      Neue",Helvetica,Arial,sans-serif;
   font-size: 14px;
   line-height: 1.42857143;
   ~~color: ■#333;~~
   background-color: □#fff;
}
body {                           bootstrap.min.css:5
```

Highlights from the Chrome 109 update

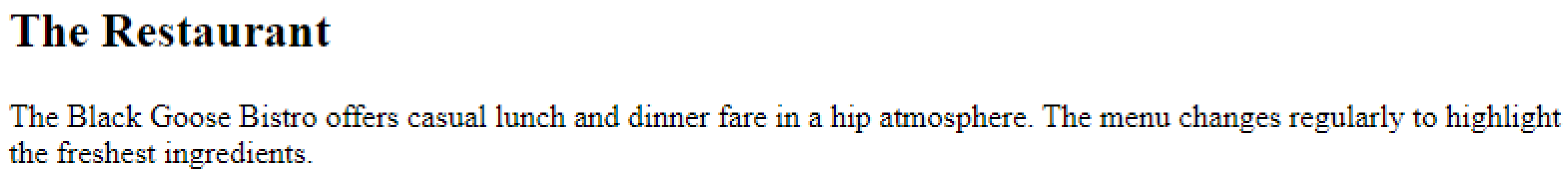
Recorder panel updates
New step context menu, option to copy a single step from a script, remove the first navigation step, and more.

Improved JavaScript debugging
Inline preview for WeakRef, correct preview of shadowed inline variable, and more

Block and Inline Elements

- The heading and paragraph elements start on new lines and do not run together as they did before. That is because by default, headings and paragraphs display as block elements.
- Browsers treat block elements as though they are in little rectangular boxes
- Each block element begins on a new line

```
<h2>The Restaurant</h2>

<p>The Black Goose Bistro offers casual lunch and
dinner fare in a hip atmosphere. The menu changes
regularly to highlight the freshest ingredients.
</p>
```



The Restaurant

The Black Goose Bistro offers casual lunch and dinner fare in a hip atmosphere. The menu changes regularly to highlight the freshest ingredients.

Block and Inline Elements

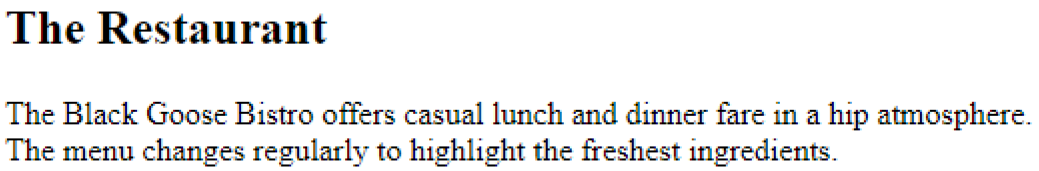
- The text written with <i> or <b> does not start a new line but stay at the flow of the paragraph. These are called as inline element.
- Inline elements do not start new lines; they just go with the flow

```
<p>You have fun... <i>we'll handle the cooking </i>. Black Goose Catering can handle events
from snacks for bridge club to elegant corporate fundraisers.</p>
```

Line Break

- To insert a line break in the text (paragraph or heading), you can use <br> tag:

```
<h2>The Restaurant</h2>

<p>The Black Goose Bistro offers casual lunch and
dinner fare in a hip atmosphere. <br>The menu changes
regularly to highlight the freshest ingredients.
</p>
```



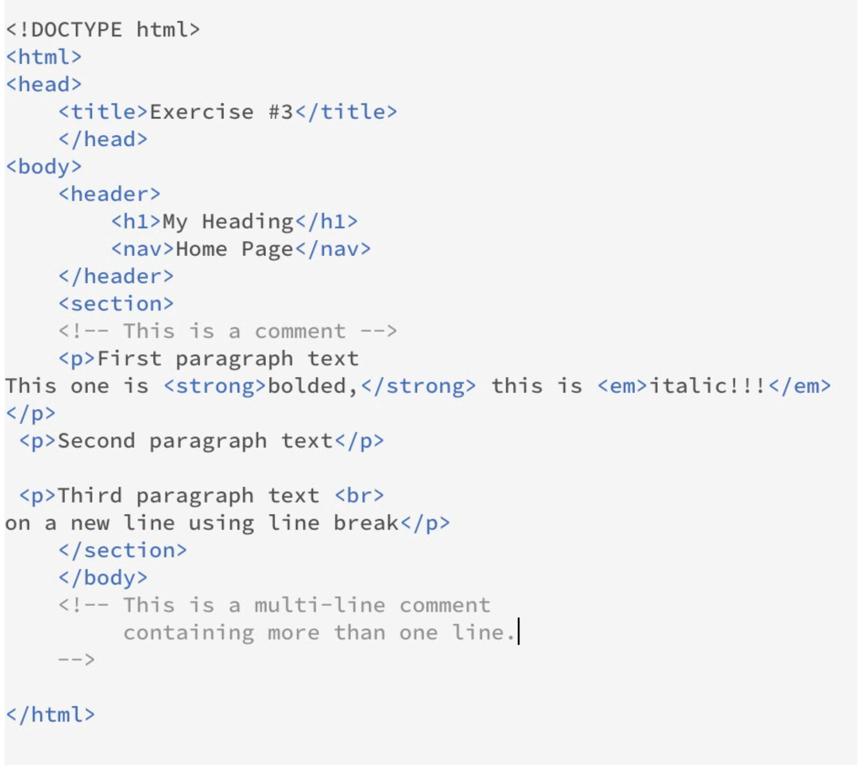
The Restaurant

The Black Goose Bistro offers casual lunch and dinner fare in a hip atmosphere.
The menu changes regularly to highlight the freshest ingredients.

HTML5

HTML Comment

<!-- A Comment → Comments can appear anywhere in a document, as the HTML parser is supposed to ignore them no matter where they appear so long as they are not inside other HTML tag structures. Comments are represented in HTML and XML as content between '<!--' and '-->'



```
<!DOCTYPE html>
<html>
<head>
    <title>Exercise #3</title>
    </head>
<body>
    <header>
        <h1>My Heading</h1>
        <nav>Home Page</nav>
    </header>
    <section>
    <!-- This is a comment -->
    <p>First paragraph text
This one is <strong>bolded,</strong> this is <em>italic!!!</em>
</p>
 <p>Second paragraph text</p>

 <p>Third paragraph text <br>
on a new line using line break</p>
    </section>
    </body>
    <!-- This is a multi-line comment
         containing more than one line.
    -->

</html>
```

HTML5

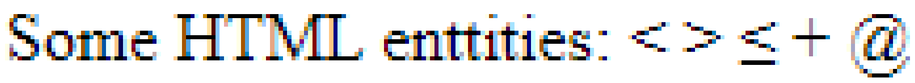
HTML Entities

Common signs are represented using & followed by fixed codes.

All HTML entities are available here:

https://dev.w3.org/html5/html-author/charref

```
<p>
    Some HTML enttities: &lt; &gt;
&le;  &plus; &commat;
</p>
```



Some HTML enttities: < > ≤ + @

HTML5

HTML Text Formatting

<b> - Bold text → <strong> - Important text

<i> - Italic text → <em> - Emphasized text

<mark> - Marked text

<small> - Small text

<del> - Deleted text

<ins> - Inserted text

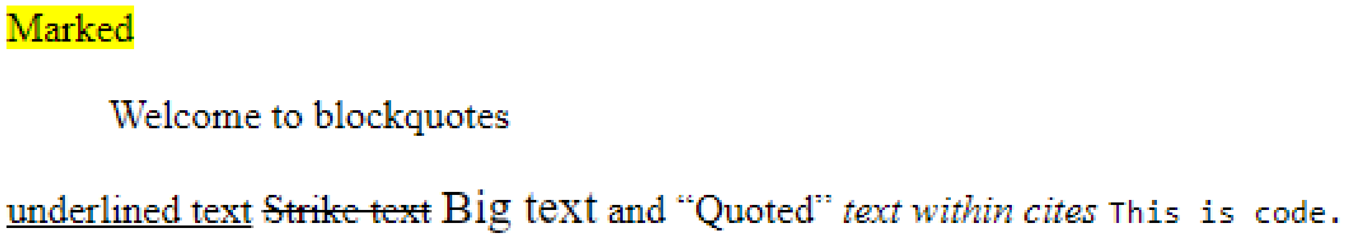
<sub> - Subscript text

<sup> - Superscript text

HTML5

HTML Text Formatting

```
<mark>Marked</mark>
<blockquote>Welcome to blockquotes</blockquote>
<u>underlined text</u>
<strike>Strike text</strike>
<big>Big text</big>
<acronym> and </acronym>
<q>Quoted</q>
<cite> text within cites </cite>
<code>This is code.</code>
```



Marked

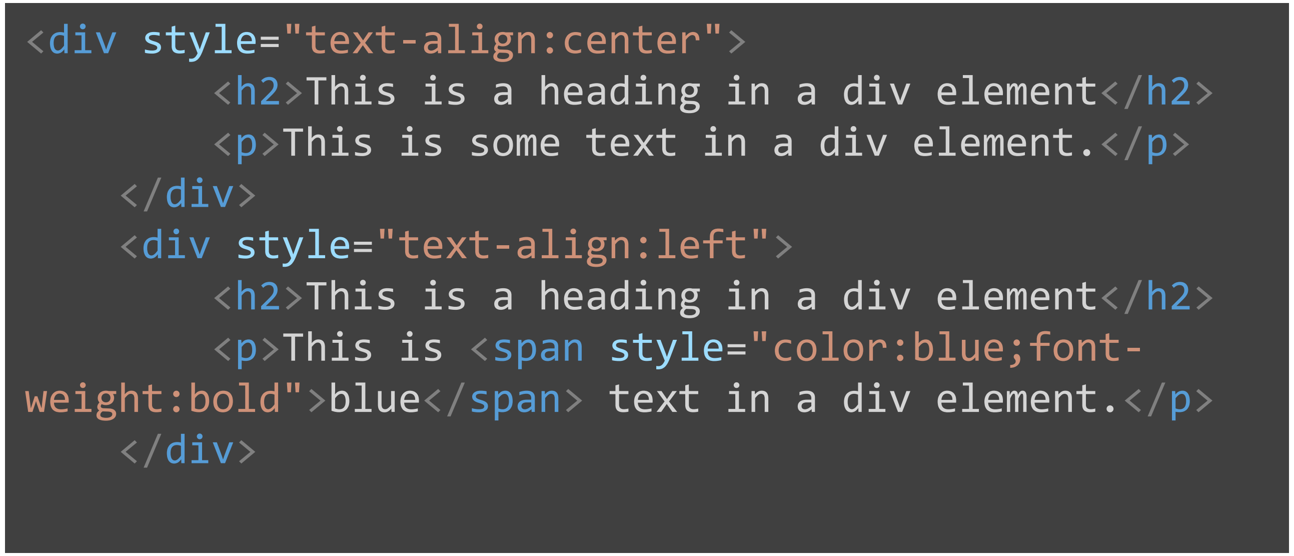
> Welcome to blockquotes

underlined text ~~Strike text~~ Big text and "Quoted" *text within cites* `This is code.`

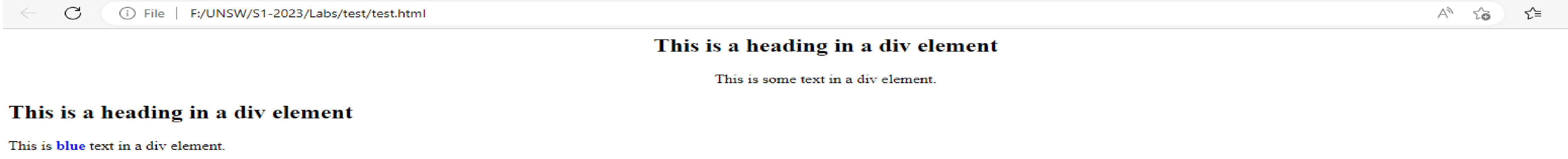
HTML5

HTML Div and Span

- <div> is a block-level element whereas a <span> is an inline element
- The HTML Content Division element (<div>) is the generic container for flow content. It has no effect on the content or layout until styled using CSS.
- The <div> element should be used only when no other semantic element (such as <article> or <nav>) is appropriate.
- The HTML <span> element is a generic inline container for phrasing content which does not inherently represent anything.
- Both can be used to group elements for styling purposes, common attributes.
- CSS styles will be covered in the next lecture.



```
<div style="text-align:center">
        <h2>This is a heading in a div element</h2>
        <p>This is some text in a div element.</p>
    </div>
    <div style="text-align:left">
        <h2>This is a heading in a div element</h2>
        <p>This is <span style="color:blue;font-weight:bold">blue</span> text in a div element.</p>
    </div>
```



This is a heading in a div element

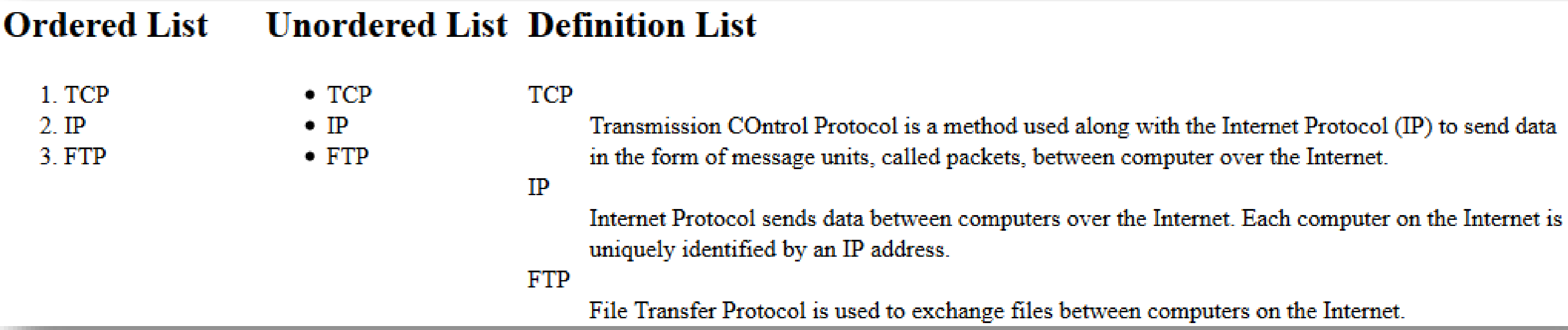
This is some text in a div element.

This is a heading in a div element

This is **blue** text in a div element.

Lists - Ordered List

- Three types
  - **List tag**: ol, ul, dl
  - **List item tag**: li
  - Other tags: dt, dd



Ordered List

1. TCP
2. IP
3. FTP

Unordered List

- TCP
- IP
- FTP

Definition List

TCP

Transmission COntrol Protocol is a method used along with the Internet Protocol (IP) to send data in the form of message units, called packets, between computer over the Internet.

IP

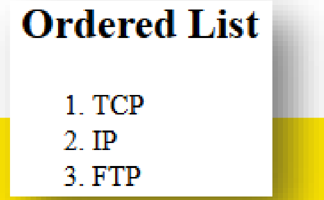
Internet Protocol sends data between computers over the Internet. Each computer on the Internet is uniquely identified by an IP address.

FTP

File Transfer Protocol is used to exchange files between computers on the Internet.

- Ordered List
  - Uses a number system (or lettering system) to sequence the information

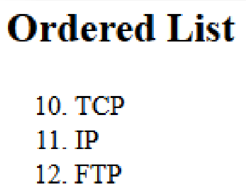
```
<h2>Ordered List</h2>
<ol>
        <li>TCP</li>
        <li>IP</li>
        <li>FTP</li>
</ol>
```



Ordered List

1. TCP
2. IP
3. FTP

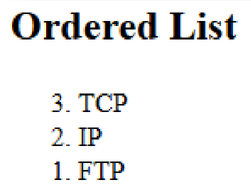
```
<h2>Ordered List</h2>
<ol start="10">
        <li>TCP</li>
        <li>IP</li>
        <li>FTP</li>
</ol>
```



Ordered List

10. TCP
11. IP
12. FTP

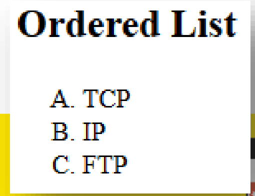
```
<h2>Ordered List</h2>
<ol reversed>
        <li>TCP</li>
        <li>IP</li>
        <li>FTP</li>
</ol>
```



Ordered List

3. TCP
2. IP
1. FTP

```
<h2>Ordered List</h2>
<ol type="A">
        <li>TCP</li>
        <li>IP</li>
        <li>FTP</li>
</ol>
```



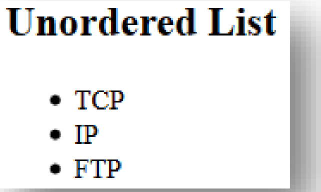
Ordered List

A. TCP
B. IP
C. FTP

Lists -Unordered List

- **Unordered List**
  - Use list markers or bullets (discs, circle, square) to display information.

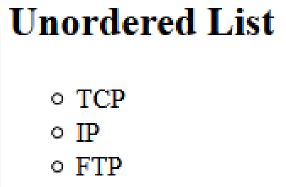
```
<ul>
        <li>TCP</li>
        <li>IP</li>
        <li>FTP</li>
</ul>
```



Unordered List

- TCP
- IP
- FTP

```
<ul type="circle">
        <li>TCP</li>
        <li>IPli>
        <li>FTP</li>
</ul>
```

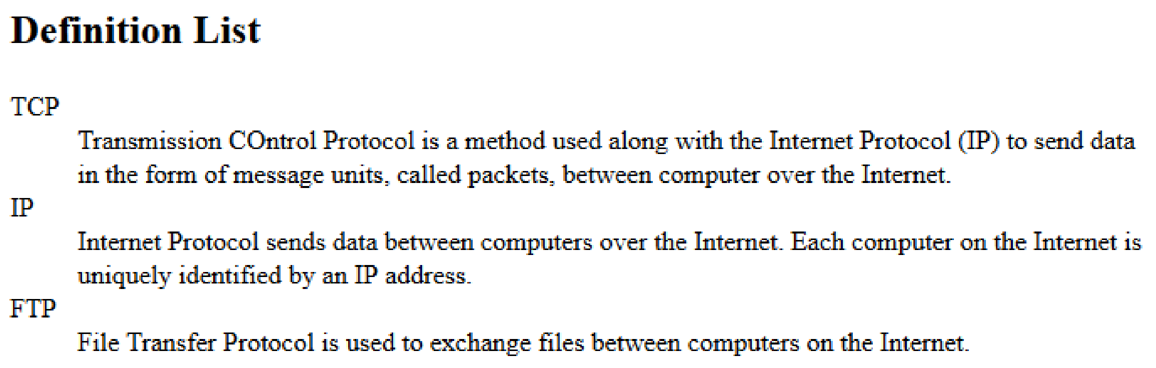


Unordered List

○ TCP
○ IP
○ FTP

- **Definition List (Description List)**
  - Display terms and their definition / description.

```
<dl>
        <dt>TCP</dt>
        <dd>Transmission COntrol Protocol is …<dd>
        <dt>IP</dt>
        <dd>Internet Protocol sends …</dd>
        <dt>FTP</dt>
        <dd>File Transfer Protocol is …dd>
</dl>
```



Definition List

TCP

Transmission COntrol Protocol is a method used along with the Internet Protocol (IP) to send data in the form of message units, called packets, between computer over the Internet.

IP

Internet Protocol sends data between computers over the Internet. Each computer on the Internet is uniquely identified by an IP address.

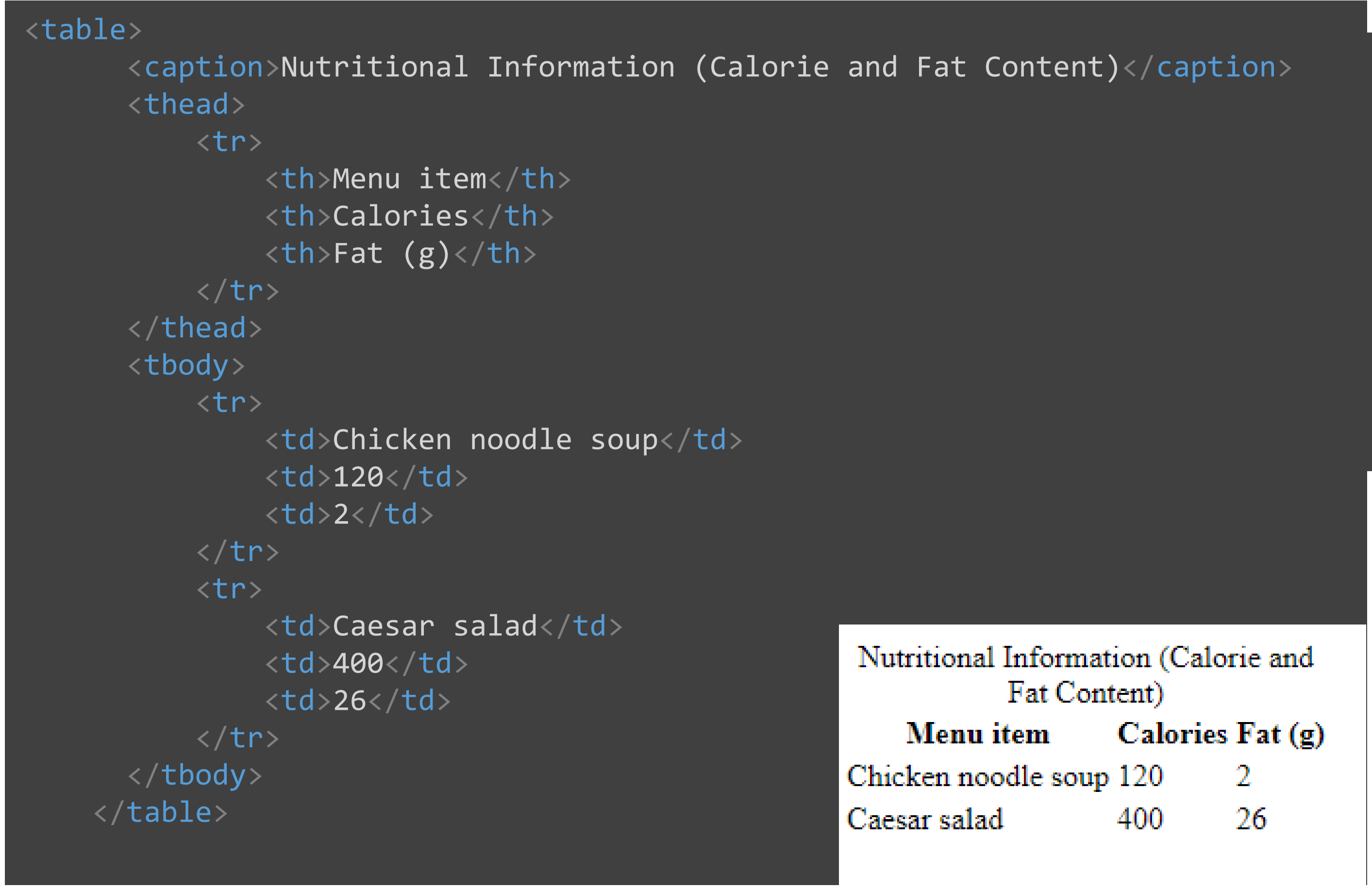
FTP

File Transfer Protocol is used to exchange files between computers on the Internet.

Tables

- Each HTML table is defined with <table>
  - <tr> table row </tr>
  - <th> table header </th>
  - <td> table data </td>
- The text-align property sets the horizontal alignment (like left, right, or center) of the content in <th> or <td>
- By default, the content of <th> elements are center-aligned and the content of <td> elements are left-aligned

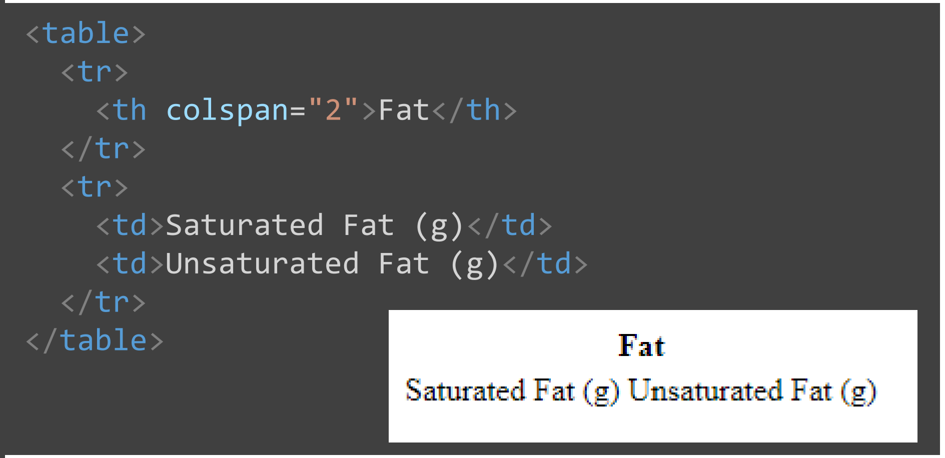
Table - Example



```
<table>
      <caption>Nutritional Information (Calorie and Fat Content)</caption>
      <thead>
          <tr>
              <th>Menu item</th>
              <th>Calories</th>
              <th>Fat (g)</th>
          </tr>
      </thead>
      <tbody>
          <tr>
              <td>Chicken noodle soup</td>
              <td>120</td>
              <td>2</td>
          </tr>
          <tr>
              <td>Caesar salad</td>
              <td>400</td>
              <td>26</td>
          </tr>
      </tbody>
    </table>
```

Nutritional Information (Calorie and Fat Content)

Menu item	Calories	Fat (g)
Chicken noodle soup	120	2
Caesar salad	400	26

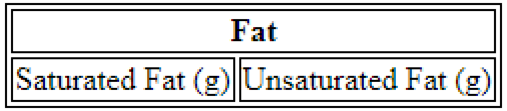


```
<table>
  <tr>
    <th colspan="2">Fat</th>
  </tr>
  <tr>
    <td>Saturated Fat (g)</td>
    <td>Unsaturated Fat (g)</td>
  </tr>
</table>
```

Fat	
Saturated Fat (g)	Unsaturated Fat (g)

Tables With Borders

Adding Style Tag in the header tag to further style the content/table.



Fat	
Saturated Fat (g)	Unsaturated Fat (g)

```
<head>
  <style>
    table, th, td {
      border: 1px solid black;
    text-align: center;
    }
  </style>
</head>
<table>
  <tr>
    <th colspan="2">Fat</th>
  </tr>
  <tr>
    <td>Saturated Fat (g)</td>
    <td>Unsaturated Fat (g)</td>
  </tr>
</table>
```

Forms- How Forms Work?

- There are two parts for a working form.
- The first part is the form that you see on the page itself that is created using **HTML markup**.
  - Forms are made up of buttons, input fields, and drop-down menus used to collect information from the user.
  - Forms may also contain text and other elements.
- The other component of a web form is an application or script on the server that processes the information collected by the form and returns an appropriate response. It’s what makes the form *work*.
  - Web applications and scripts require programming knowledge (this is what we will study in the next few weeks)

Forms

- Text Input
  - The most important form element is the <input> element.
  - The <input> element can be displayed in several ways, depending on the type attribute

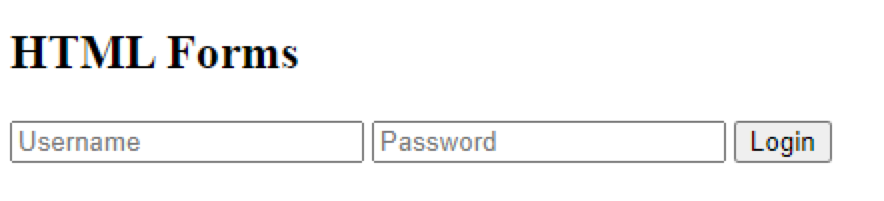
- Radio Button
  - Input<input type="radio"> defines a **radio button**.
  - Radio buttons let a user select ONLY ONE of a limited number of choices

- Submit Button
  - <input type="submit"> defines a button for **submitting** form data to a **form-handler**.
  - The form-handler is typically a server page with a script for processing input data

- Action Button
  - The action attribute specifies where to send the form-data when a form is submitted.

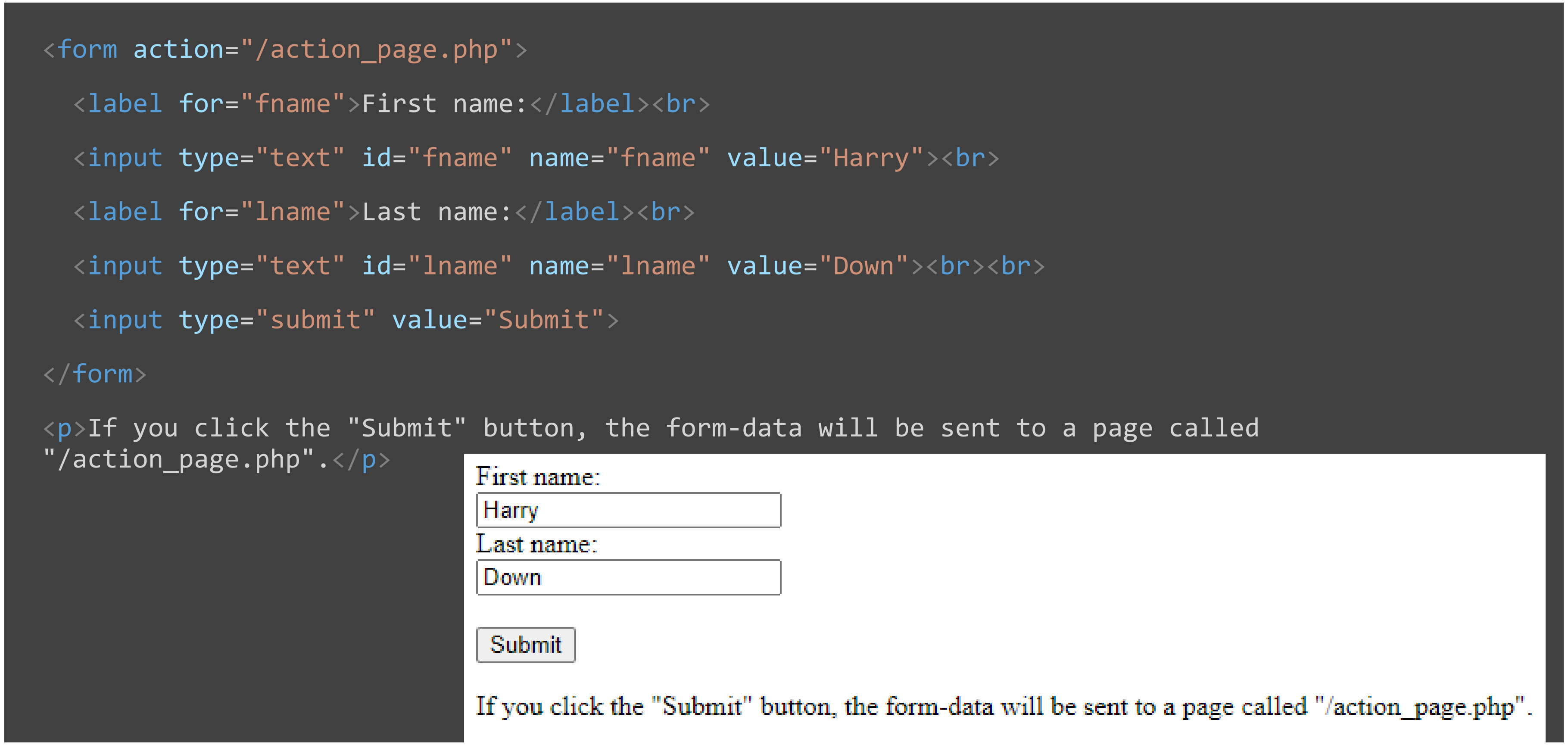
Forms - Example

```
<!DOCTYPE html>

<html>

<body>

<h2>HTML Forms</h2>

<form action="second.html">

<input type="text" placeholder="Username" name="username">

<input type= "password" placeholder="Password"
name="password">

<button type="submit">Login</button>

</form>

 </body>

</html>
```



HTML Forms

Forms - Example



```
<form action="/action_page.php">
  <label for="fname">First name:</label><br>
  <input type="text" id="fname" name="fname" value="Harry"><br>
  <label for="lname">Last name:</label><br>
  <input type="text" id="lname" name="lname" value="Down"><br><br>
  <input type="submit" value="Submit">
</form>
<p>If you click the "Submit" button, the form-data will be sent to a page called
"/action_page.php".</p>
```

First name:

Harry

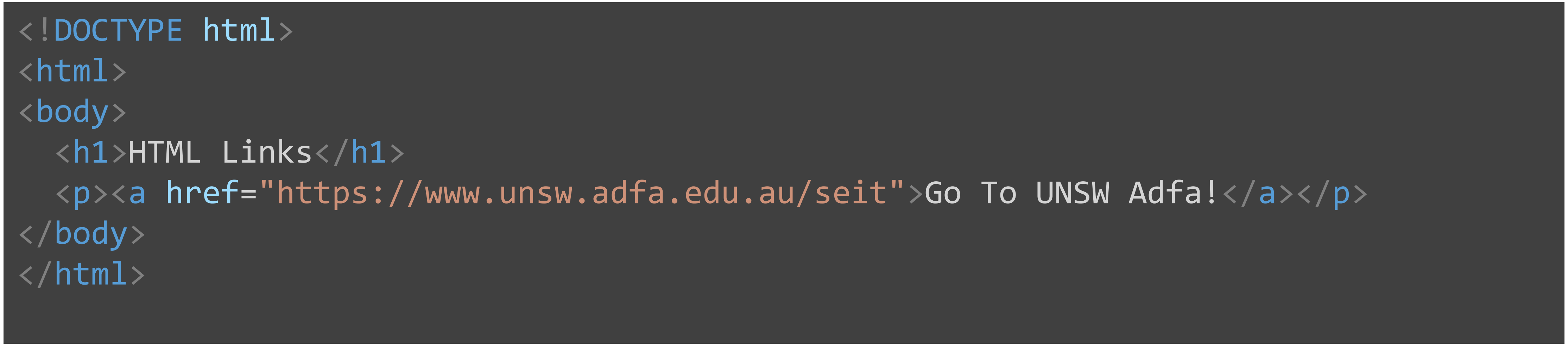
Last name:

Down

If you click the "Submit" button, the form-data will be sent to a page called "/action_page.php".

Hyperlink

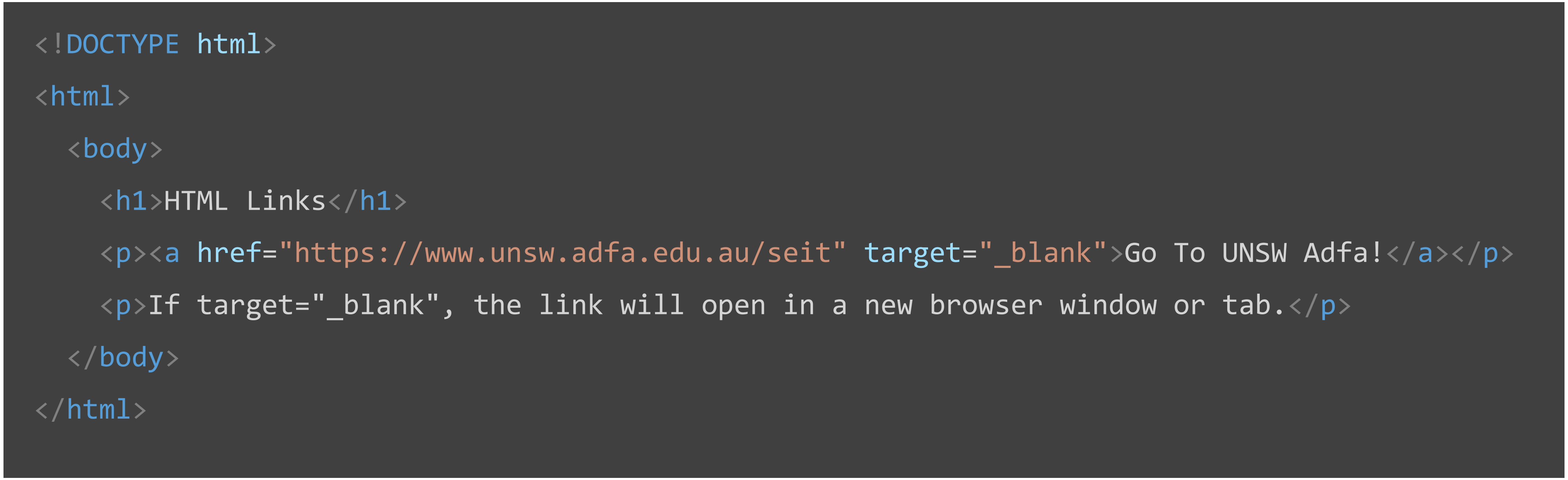
- <a> defines a hyperlink in HTML
- Most important attribute of <a> element is href that indicates the link’s destination
- <a href= “*url*”>link text </a>
- The *link text* part will be visible to the reader. clicking on the link text part will send the user to the specified URL



```
<!DOCTYPE html>
<html>
<body>
  <h1>HTML Links</h1>
  <p><a href="https://www.unsw.adfa.edu.au/seit">Go To UNSW Adfa!</a></p>
</body>
</html>
```

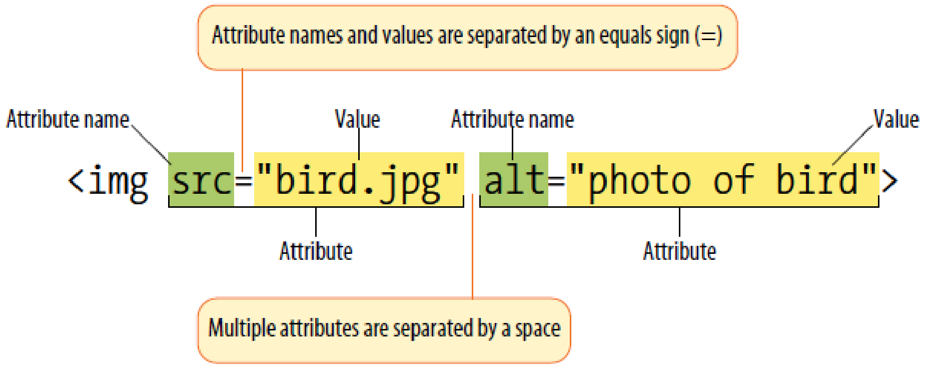
Open a Hyperlink in an New Window

Target attribute will allow to open the website in a new browser window



```
<!DOCTYPE html>
<html>
  <body>
    <h1>HTML Links</h1>
    <p><a href="https://www.unsw.adfa.edu.au/seit" target="_blank">Go To UNSW Adfa!</a></p>
    <p>If target="_blank", the link will open in a new browser window or tab.</p>
  </body>
</html>
```

Adding Image - Hyperlink-Attribute



Attribute names and values are separated by an equals sign (=)
Attribute name
Value
Attribute name
Value
<img src="bird.jpg" alt="photo of bird">
Attribute
Attribute
Multiple attributes are separated by a space

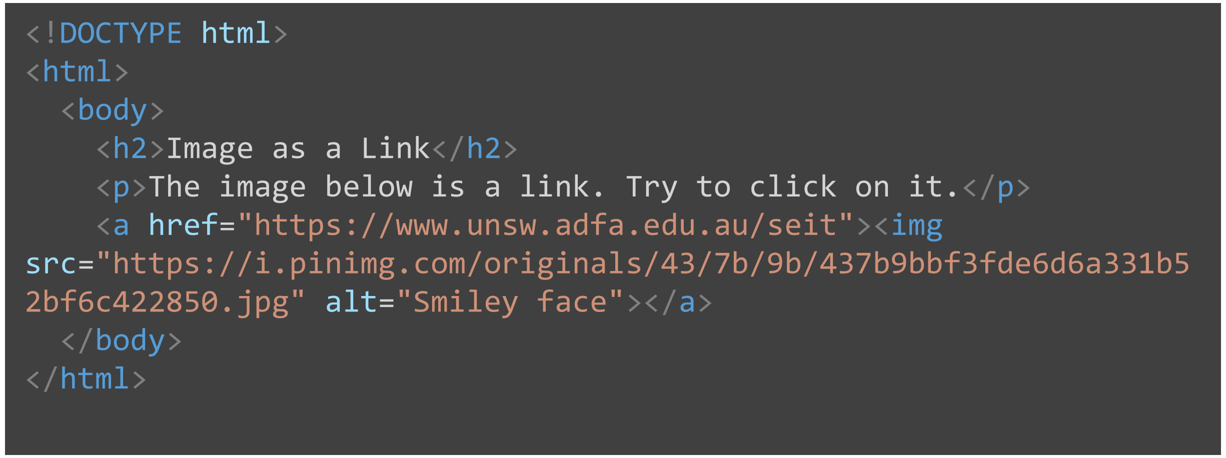
The syntax for an attribute is as follows:

attributename="value“

You can also put more than one attribute in an element in any order. Just keep them separated with spaces.

<element attribute1="value“ attribute2="value">

Example - Adding an Image



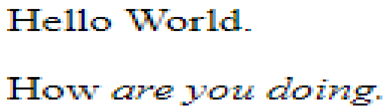
```
<!DOCTYPE html>
<html>
  <body>
    <h2>Image as a Link</h2>
    <p>The image below is a link. Try to click on it.</p>
    <a href="https://www.unsw.adfa.edu.au/seit"><img
src="https://i.pinimg.com/originals/43/7b/9b/437b9bbf3fde6d6a331b5
2bf6c422850.jpg" alt="Smiley face"></a>
  </body>
</html>
```

Syntax Validation

Is the HTML code syntactically valid?

- Browsers are pretty tolerant with syntax error

```
 <p>Hello World.
<p>How <i>are you doing.</p> </i>
```


Hello World.

How *are you doing.*

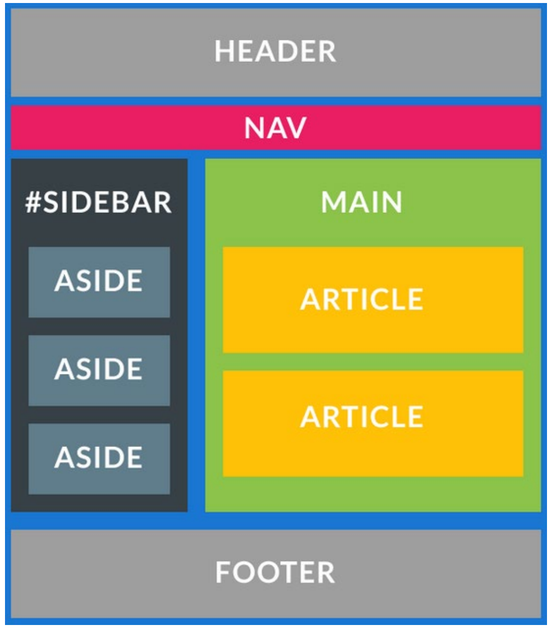
- Invalid code may cause browsers to render the pages slower than otherwise

- Validate HTML code to assure quality and performance
  - Markup Validation Service: https://validator.w3.org
  - HTML5 validator: https://html5.validator.nu/
  - HTML5 lint: http://www.htmllint.net/en/html-lint

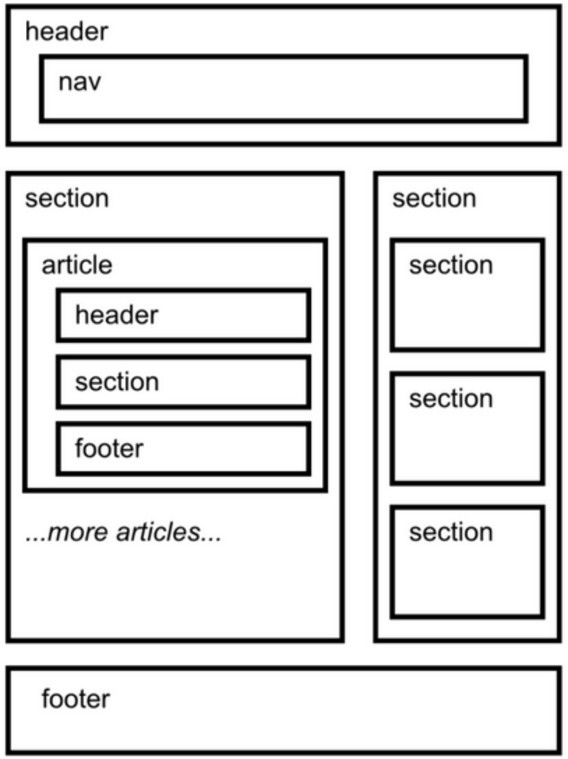
- Test on different browsers (and versions) to ensure the same user experience.

Layout - How to Structure Web Content

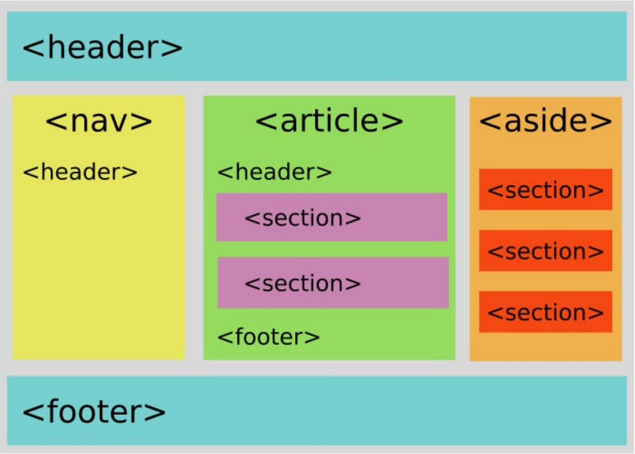
- Web page layout (wireframe)



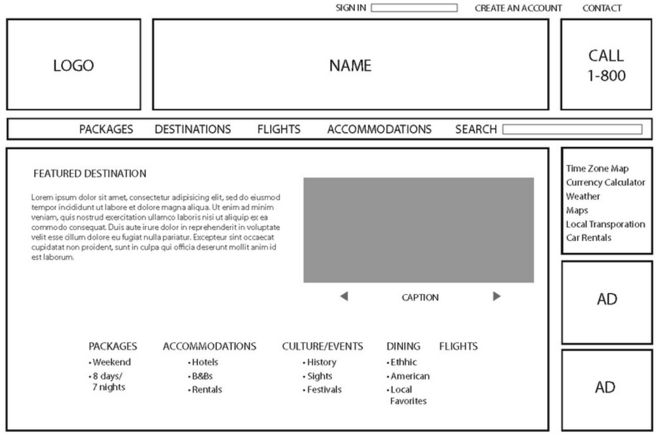
Structural elements
`div(id=sidebar)`



`nav` *inside* `header`



Three column layout



Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

More complex layout

Structural Elements

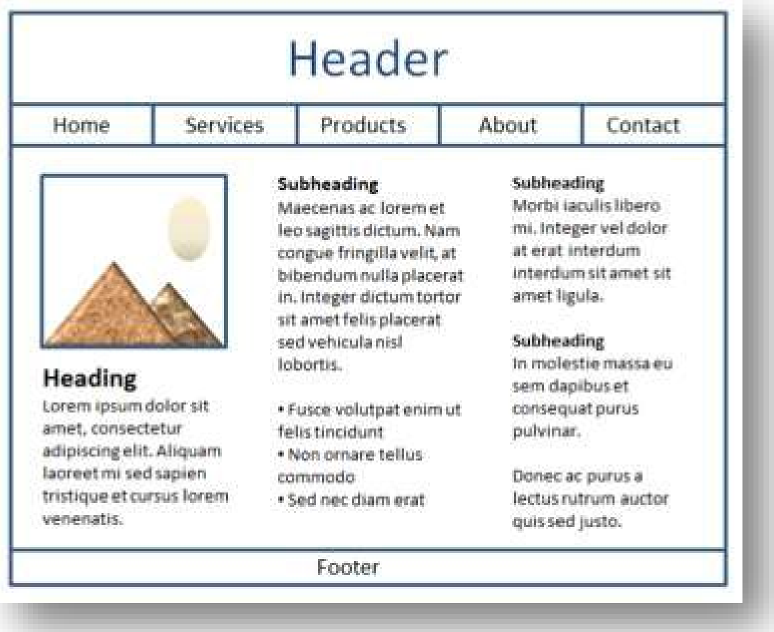
Semantic Elements: `header`, `footer`, `nav`

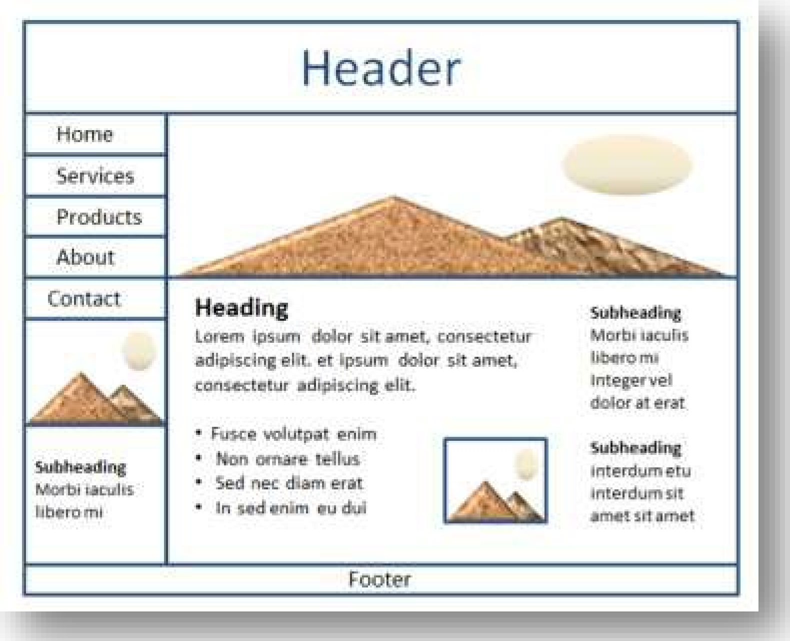
- **Semantic Elements**
  - Provide **semantic / hierarchical** structure to the web content (*meaningful to machines, e.g. screen reader, SE*)
  - **Do not dictate** what presentation methods to be used.
- **<header>**
  - Contain introductory content to a document / article / section.
  - Include headings, introductory text, navigation, meta data (post-date).
- **<footer>**
  - Include the author of the document, copyright information, links to terms of use, contact information, disclaimers.
- **<nav>**
  - Global navigation links, table of contents, previous/next links,

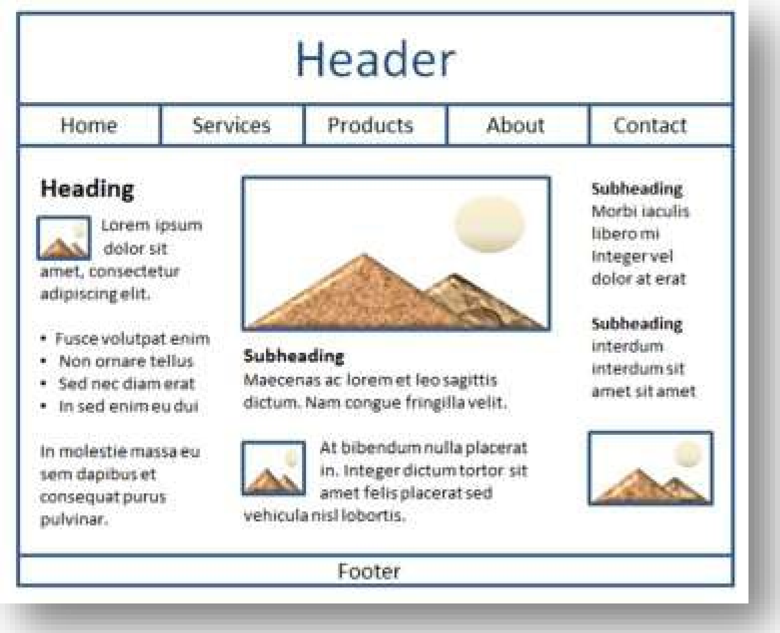
Web Layout

Represent the sketch of a web page blueprint

- Shows the structure of the basic page elements, including logo, navigation, content and footer
- Layout can be selected based on the audience and web page content.


Header
Home
Services
Products
About
Contact
Heading
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aliquam laoreet mi sed sapien tristique et cursus lorem venenatis.
Subheading
Maecenas ac lorem et leo sagittis dictum. Nam congue fringilla velit, at bibendum nulla placerat in. Integer dictum tortor sit amet felis placerat sed vehicula nisl lobortis.
• Fusce volutpat enim ut felis tincidunt
• Non ornare tellus commodo
• Sed nec diam erat
Subheading
Morbi iaculis libero mi. Integer vel dolor at erat interdum interdum sit amet sit amet ligula.
Subheading
In molestie massa eu sem dapibus et consequat purus pulvinar.
Donec ac purus a lectus rutrum auctor quis sed justo.
Footer


Header
Home
Services
Products
About
Contact
Subheading
Morbi iaculis libero mi
Heading
Lorem ipsum dolor sit amet, consectetur adipiscing elit. et ipsum dolor sit amet, consectetur adipiscing elit.
• Fusce volutpat enim
• Non ornare tellus
• Sed nec diam erat
• In sed enim eu dui
Subheading
Morbi iaculis libero mi Integer vel dolor at erat
Subheading
interdum etu interdum sit amet sit amet
Footer


Header
Home
Services
Products
About
Contact
Heading
Lorem ipsum dolor sit amet, consectetur adipiscing elit.
• Fusce volutpat enim
• Non ornare tellus
• Sed nec diam erat
• In sed enim eu dui
In molestie massa eu sem dapibus et consequat purus pulvinar.
Subheading
Maecenas ac lorem et leo sagittis dictum. Nam congue fringilla velit.
At bibendum nulla placerat in. Integer dictum tortor sit amet felis placerat sed vehicula nisl lobortis.
Subheading
Morbi iaculis libero mi Integer vel dolor at erat
Subheading
interdum interdum sit amet sit amet
Footer

Resource

- https://www.w3schools.com/html/default.asp

Final Notes

- Project 1 starts this week.
- Students need to form group of maximum 3 members.
- Each group should choose a topic.
- Topics will be different.
- Write the name of group members and the topic in this Excel file:

Project Groups.xlsx

- Discussion Forum will be open on the

  Moodle
- Please use it to inform your lecturers of your group and topic.
- Deadline 12th March 23h59.