

Introduction to Web Technology

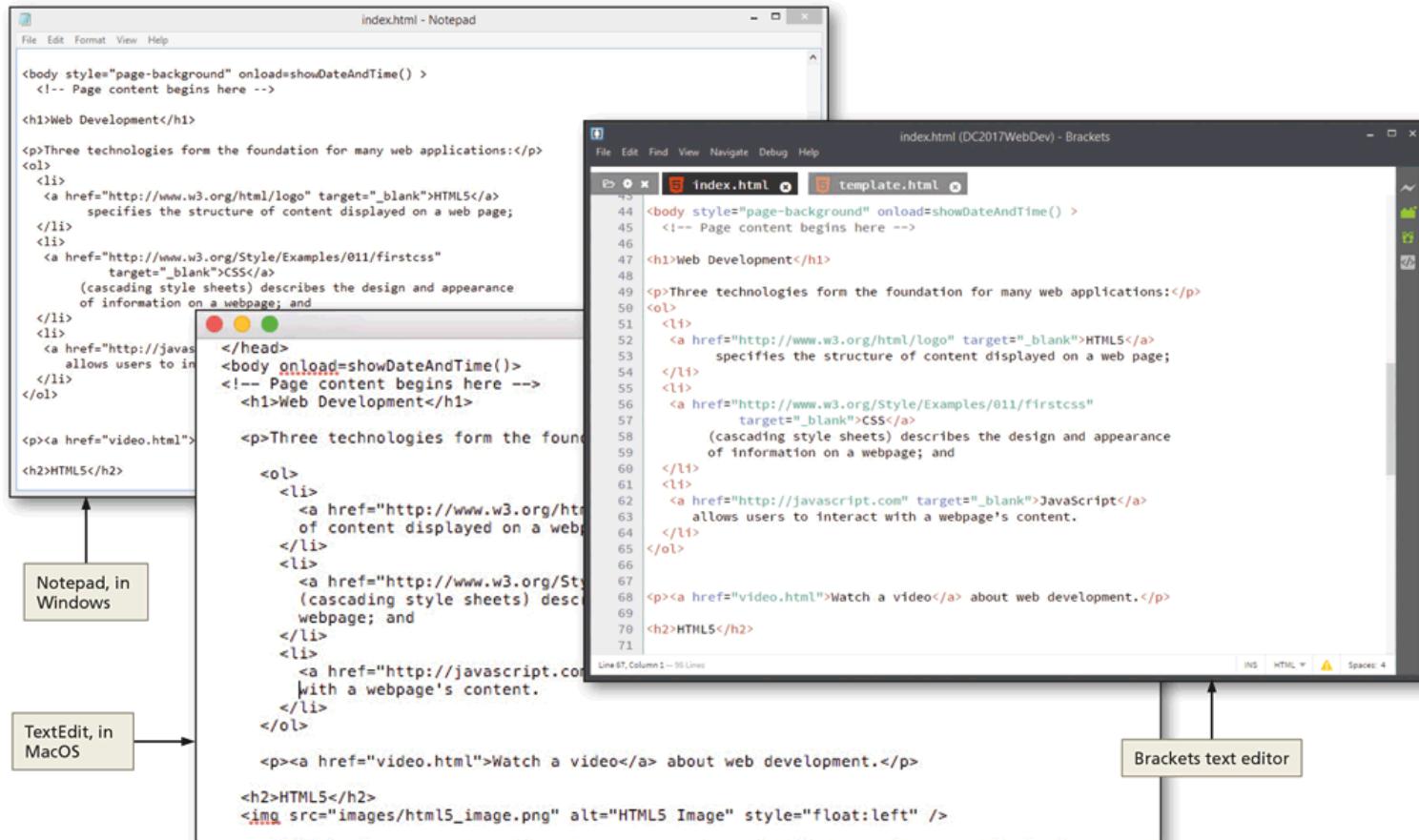


Asst.Prof.Dr. Suvit Poomrittigul

I. Tools for Creating a Website

- ❖ **Text editors**
 - Similar to a word processing program, but lacks most text formatting features
 - Operating systems typically include a text editor
 - A code editor is a type of text editor that contains additional features to help web developers write code

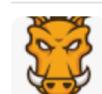
Tools for Creating a Website



Tools for Creating a Website

Web development tools

จากแหล่งที่มาในเว็บ

 Chrome DevTools	 Visual Studio Code Programming language: C, J...	 Sublime Text Programming language: Py...
 Bootstrap Programming language: CS...	 GitHub	 Angular Programming language: Typ...
 Npm Programming language: Jav...	 JavaScript library	 jQuery Programming language: Jav...
 Laravel Programming language: PHP	 Postman	 Sass
 Django Programming language: Py...	 Foundation Programming language: CS...	 React Programming language: Jav...
 Ruby on Rails Programming language: Rub...	 Tailwind CSS Programming language: Typ...	 Atom
 Flutter Programming language: Dart... <i>cross platform Mobile & PC used Dart</i>	 Grunt Programming language: Jav...	 Notepad++ Programming language: C++
 Sketch	 TypeScript	 Vue.js Programming language: Typ... 4

Tools for Creating a Website

● ເນັ້ນທີ່ໂຮງໝານ, ວິໄລ

❖ Content Management Systems (CMS)

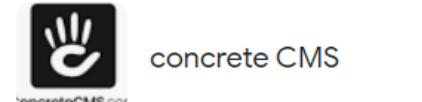
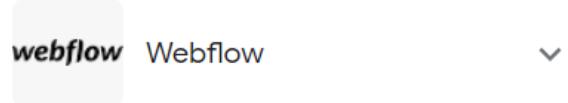
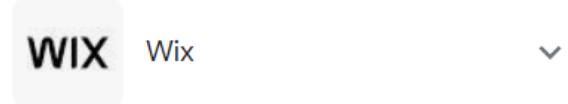
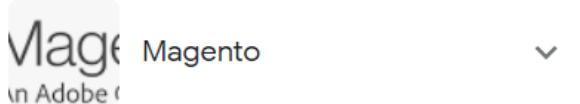
- Enables and manages the publishing, modification, organization, and access of various forms of documents and other files on a network or the web
- Web developer creates a theme, and one or more website content administrators enters the content

Example of CMS

ເວັບສໍາເຮົາຈຸດ



ex. ອີເມຣູນໄຍ່
↳ WordPress
ກໍາ Gallery ຂຶ້ນໂຄງຂອງໃຫ້



Tools for Creating a Website

The screenshot shows the WordPress dashboard. On the left, a dark sidebar lists various tools: Dashboard, Home, Updates (1), Kinsta Cache, Posts, Media, Pages, Lasso, Comments, Rank Math, Elementor, Templates, Appearance, Plugins (1), Users, Tools, and Settings. The main area is titled 'Dashboard' and features a large blue banner with the text 'Welcome to WordPress!' and a link to 'Learn more about the 6.1.1 version.' Below the banner, there are two sections: one for creating content with blocks and patterns, and another for customizing the site's appearance.

Author rich content with blocks and patterns

Block patterns are pre-configured block layouts. Use them to get inspired or create new pages in a flash.

[Add a new page](#)

Start Customizing

Configure your site's logo, header, menus, and more in the Customizer.

[Open the Customizer](#)

Rank Math Overview

Quick Draft

2. Website Technologies

→ fundamental for static web ປຸ່ມຈຳເປົງແລ້ວອະນຸມັດ compiler ຜິດຕະຫຼາດ

- ❖ Hypertext Markup Language (**HTML**) uses a set of codes called **tags** to format documents for display in a browser
- ❖ A complementary technology called **cascading style sheets (CSS)** contains specifications for the fonts, colors, layout, and placement of these HTML elements on a webpage
- ❖ **JavaScript** is a programming language for creating programs that a browser can run to generate content for a website

Programming Language
- ສິນເກມດີເລີດ

2. Web Technologies

- ❖ **World Wide Web (WWW)**: The World Wide Web is based on several different technologies: Web browsers, Hypertext Markup Language (HTML), and Hypertext Transfer Protocol (HTTP).
ມາສັກເອົາຕົວນີ້ ຂ່າງໝັ້ນ HTML5
ທີ່ Browser (ດ້ານ) ຢຳ support
- ❖ **Web Browser**: The web browser is an application software to explore www (World Wide Web). It provides an interface between the server and the client and requests to the server for web documents and services.
→ ເຊັ່ນ HTML, JS ຍຸ່ນເປັນ , Broadcast ໃນຕາມອື່ນຂາດູ້ປົວ
* ໄລກຮະຈາຍຈ້ອນວຸດ
- ❖ **Web Server**: Web server is a program which processes the network requests of the users and serves them with files that create web pages. This exchange takes place using Hypertext Transfer Protocol (HTTP).
- ❖ **Web Pages**: A webpage is a digital document that is linked to the World Wide Web and viewable by anyone connected to the internet has a web browser. ex. Home Page
- ❖ **Web Development**: Web development refers to the building, creating, and maintaining of websites. It includes aspects such as web design, web publishing, web programming, and database management. It is the creation of an application that works over the internet i.e. websites.

Website

The screenshot illustrates the structure of a web page from the Smithsonian National Zoological Park's website. The page is titled "Great Apes & Other Primates". It features a sidebar on the left with links to "Meet the Primates", "Facts", "Exhibit", "News", "Science", and "Enrichment". The main content area contains two sections: "GORILLAS" and "PRIMATES AT THE ZOO". The "GORILLAS" section includes a paragraph about gorillas and a link to "MeetGorillas". The "PRIMATES AT THE ZOO" section includes a paragraph about primates at the zoo and a link to "MeetPrimates". A large image of a gorilla's face is centered between the two sections. The footer features a donation goal of "\$49.50 million" towards a \$60 million goal.

link to external style sheet

h1 header

h2 header

hyperlink

h3 header

image

paragraph

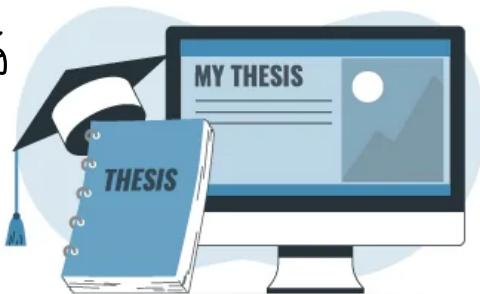
```
<!DOCTYPE html
1 PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
2 "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd"><!-- Inst
3 codeOutsideHTMLIsLocked="false" -->
4 <html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en"><!-- Inst
5 <head>
6
7 <title>Great Apes and Other Primates - National Zoo</title>
8 <link href="https://fonts.googleapis.com/css?family=Oswald:400,300,700" rel="stylesheet">
9 </head>
10 <body>
11
12 <h1>Great Apes & Other Primates</h1>
13
14 <h2>Gorillas</h2>
15 <p>
16 <a href="/Animals/Primates/MeetPrimates/MeetGorillas/">Six western lowland
17 youngest is Kibibi, born in 2009.</a>
18 </p>
19
20 <h3>Primates at the Zoo </h3>
21 <a href="http://www.flickr.com/photos/nationalzoo/3707340844/" title="Goril
22 lla">
23 
24 </a>
25 <p>
26 The Zoo is home to many primates. Orangutans and western lowland gorillas
27 tamarins, Geoffroy's marmosets, and howler monkeys, can be found in the <a href="/anim
28 als/primates/meetprimates/meetlemur">Small Mammal House</a>. Look for gibbons at Gibbo
29 n Ridge and lemurs at Lemur Island. <a href="#">Find out where
30 primates can be seen at the Zoo.</a>
31
32 On mild days, the orangutans can sometimes be seen overhead as they travel
33 along the O Line between the Great Ape House and Think Tank.
34
35 <h3>About Primates</h3>
36 <p>There are 376 species of primates in the world&#8212;from humans and apes
37 to monkeys and prosimians ("premonkeys").</p>
38
39 The smallest primate is the pygmy mouse lemur, which can fit in the palm of your
40 hand. The largest—the gorilla—can weigh more than 400 pounds. Most primates
41 live in warm climates, and most depend on forests for their survival. <a href="#">More
42 Primate Facts</a>
43
44 </body>
45 </html>
```

Static Web VS Dynamic Web

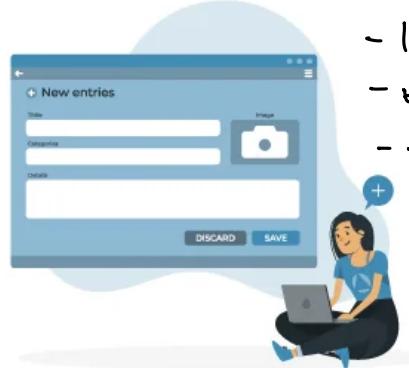
A **static website** is a type of web page that displays fixed and non-changing information. These pages do not have the ability to interact with users or to change content in real time. Instead, static web pages are created and published with fixed content, which is the same for all users who visit the page.



- 1 ქვეთი update 1 გვერდი
- ყაბღისგარე real time



STATIC web



DYNAMIC web

- link ანუ Database
- link ანუ API Shopee
- უკავშირობა user interaction
- HTML, CSS ემულირდება
- მაგ. JS, PHP, Ruby, Python, etc.

A **dynamic web** page is a type of website that generates content in real time based on user interaction. Unlike static web pages, which display the same content to all users, dynamic web pages can be customized and change in real time. This is accomplished by using programming languages, such as PHP, Ruby on Rails, or JavaScript, and connecting to a database.

Web Application

Web Application vs Website

↳ چیزی که Platform نیست

Web Application

Dynamic pages:

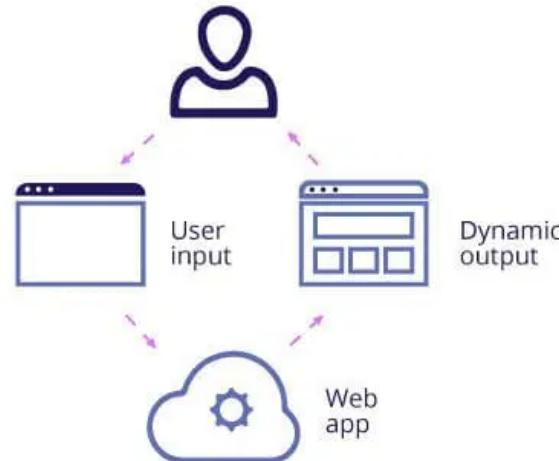
Every user sees the different information each time.



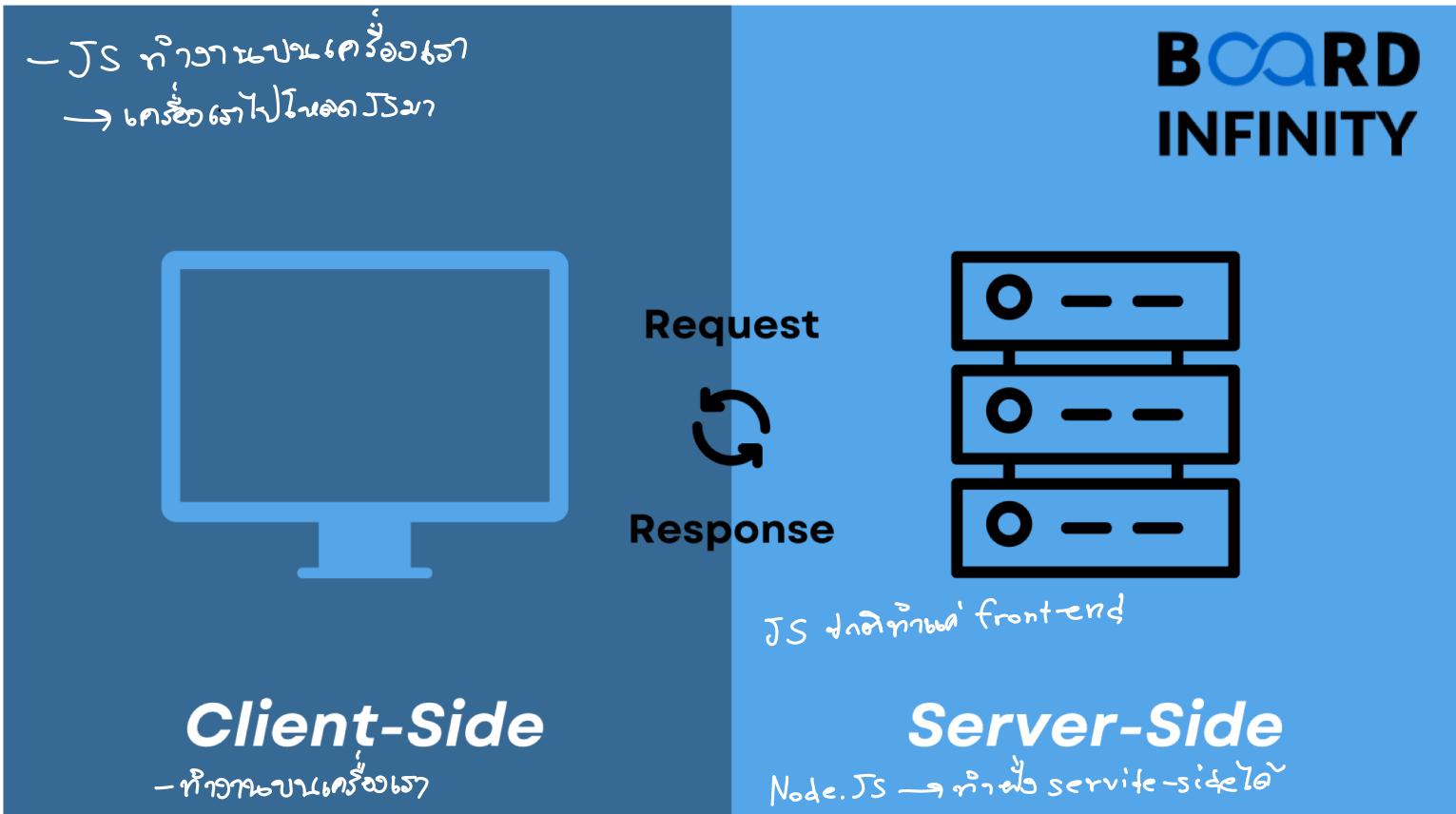
Website

Static pages:

Every user sees the same information each time.



Client Side Scripting and Server Side Scripting



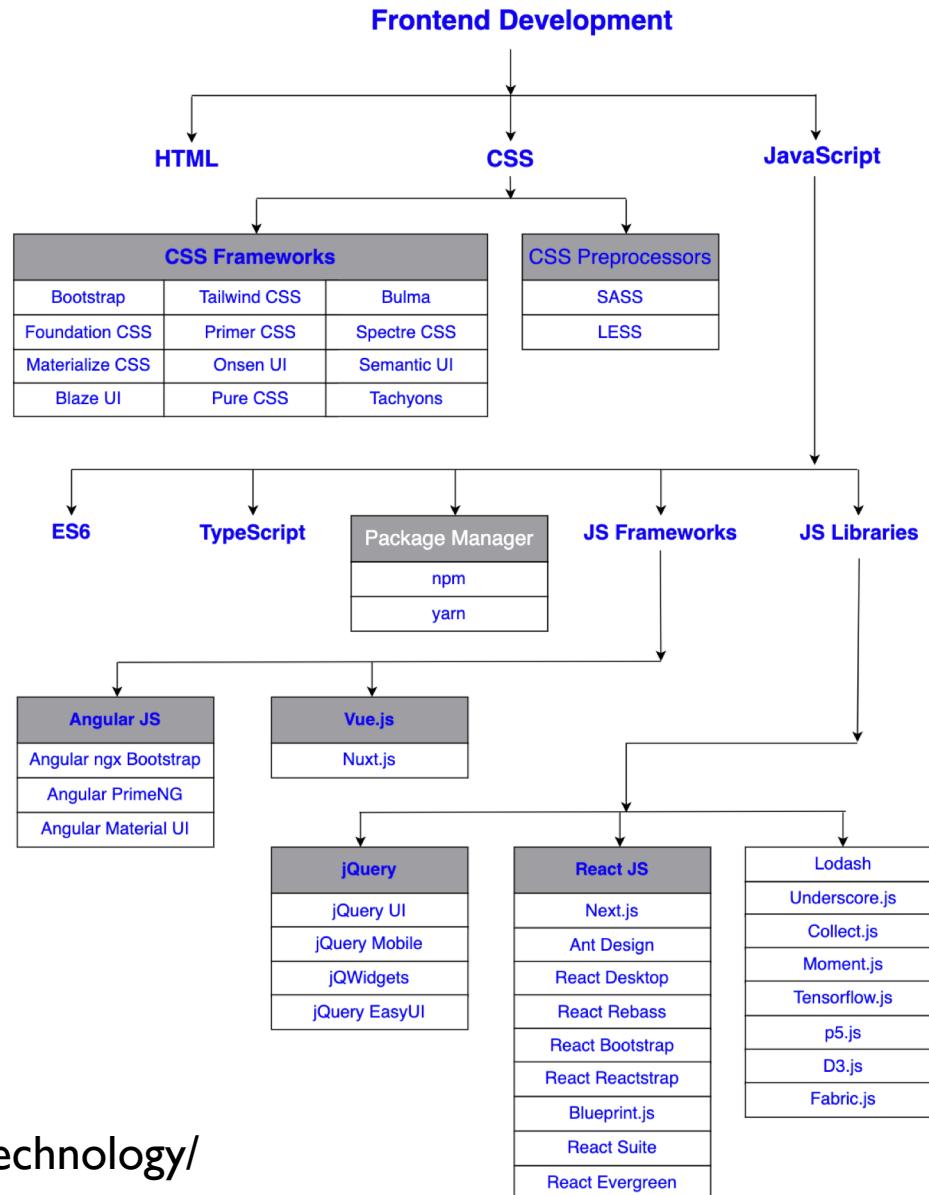
Web Application/Front-End

Web Server/ Website/Back-End

Client Side Scripts

Frontend Development:

The part of a website that the user interacts directly is termed as front end.
It is also referred to as the '**client side**' of the application.

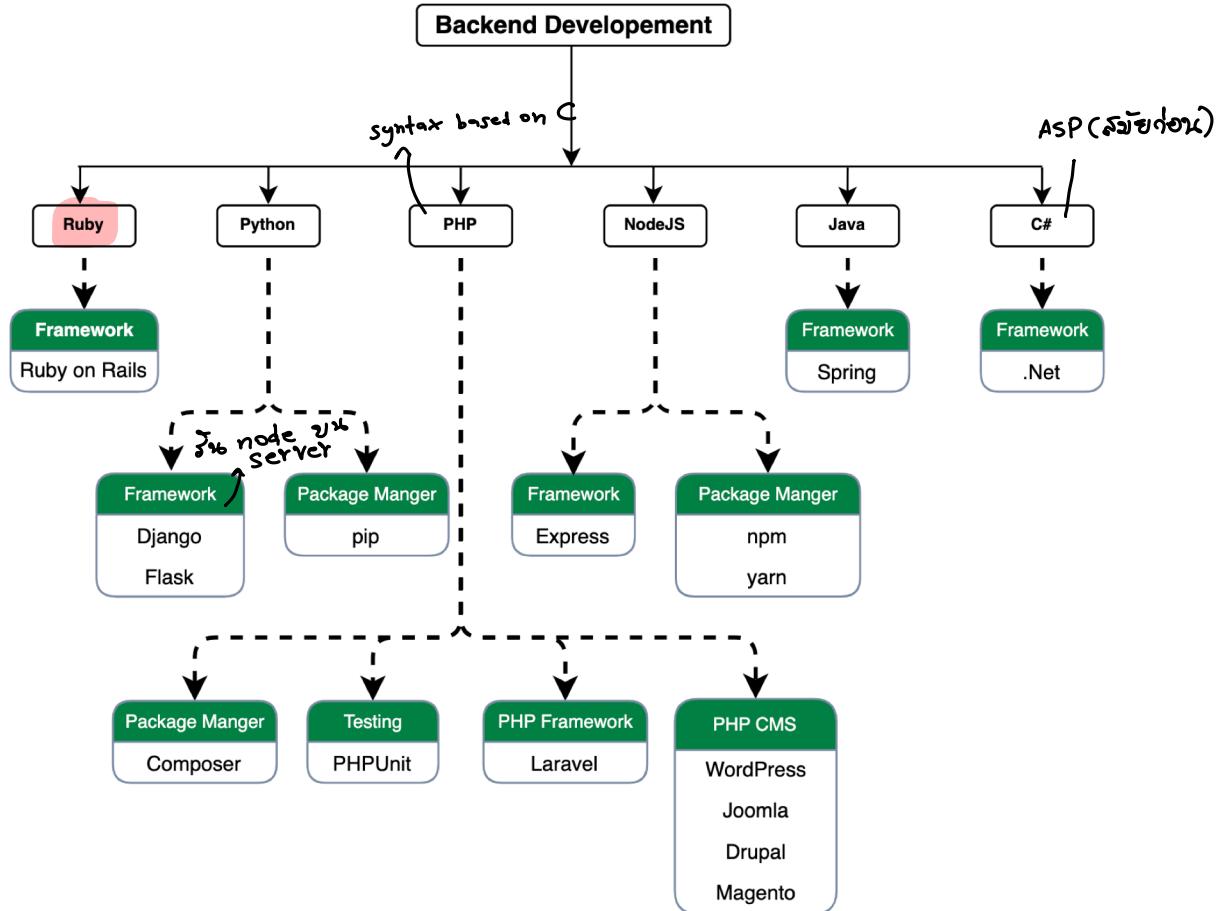


Server Side Scripts

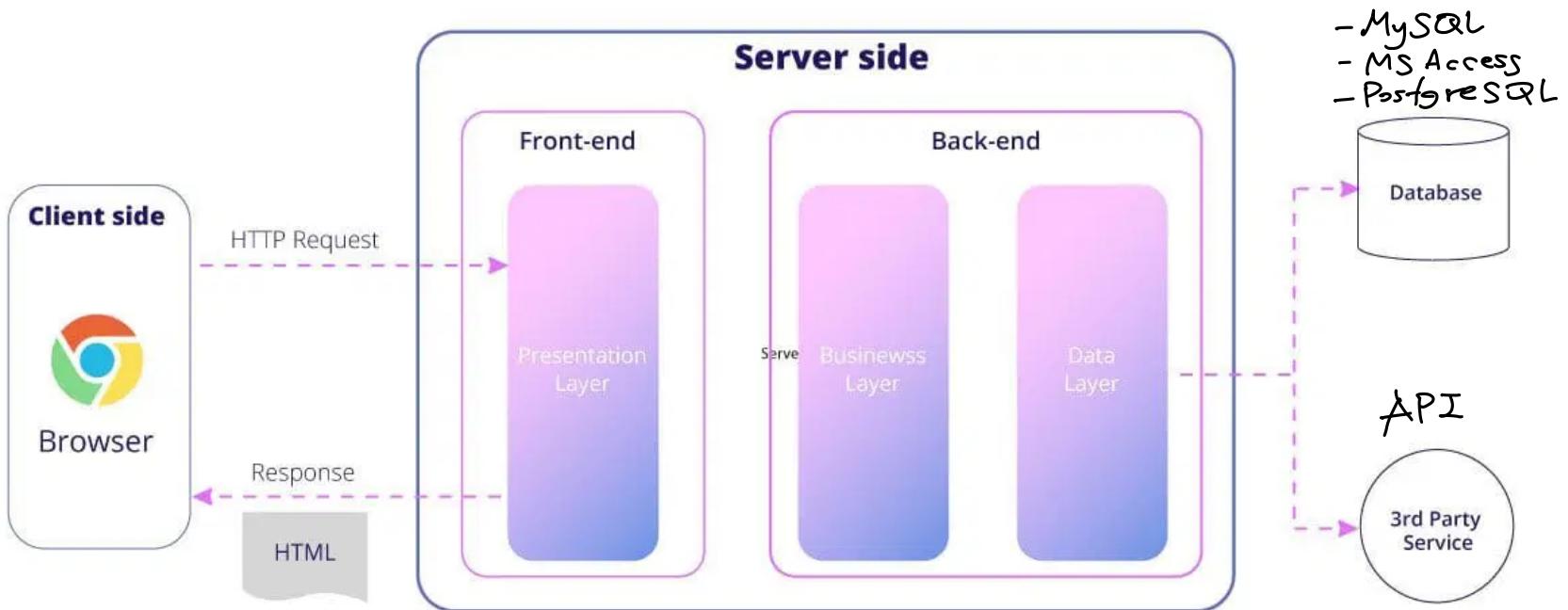
- ຈຳກັງ DB, login ໂດຍ

Backend Development:

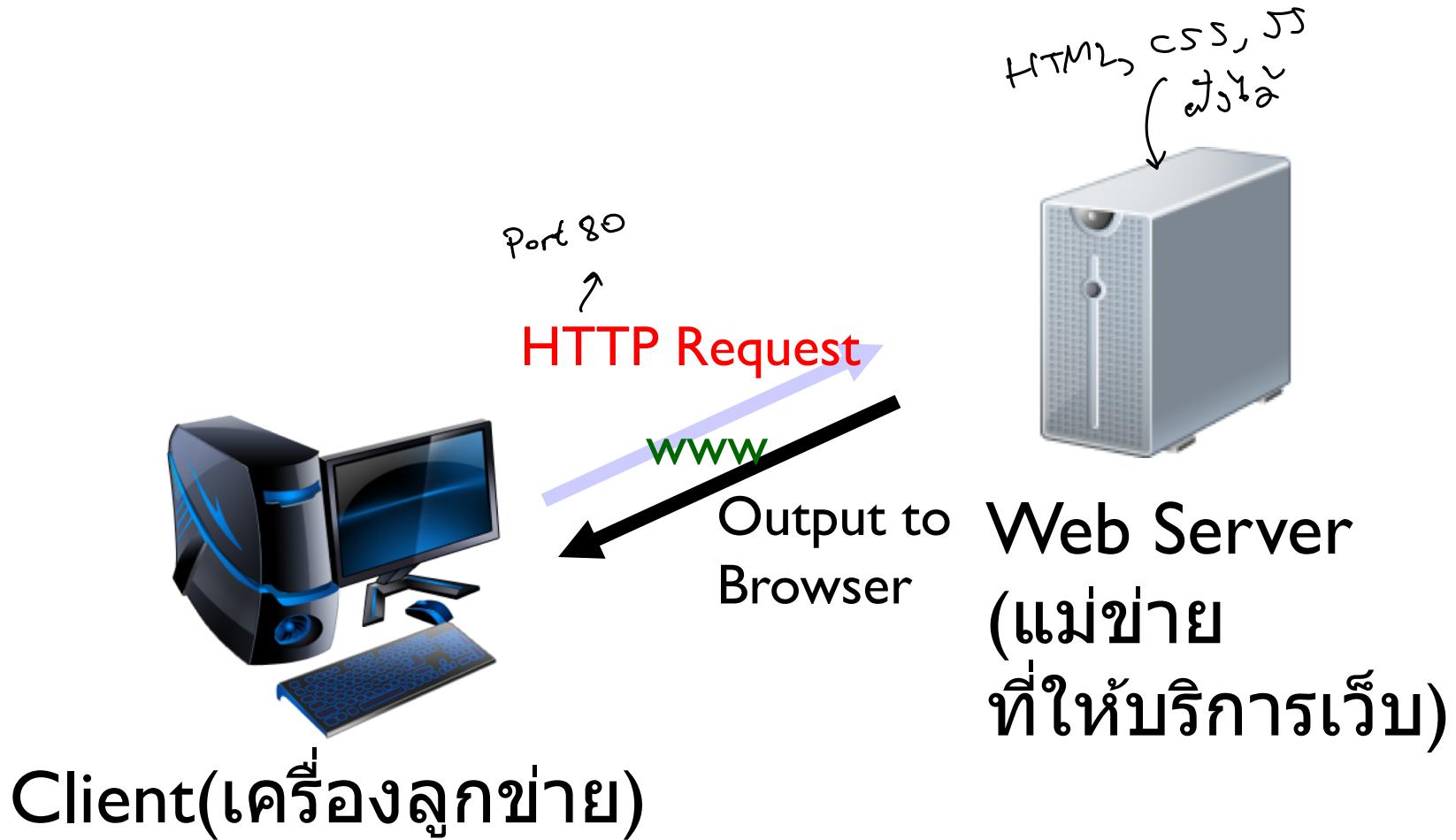
Backend is the ‘server side’ of a website. It is the part of the website that users cannot see and interact. It is the portion of software that does not come in direct contact with the users. It is used to store and arrange data.



SERVER SIDE RENDERING (SSR)



Client-WebServer



❖ Web Server Software

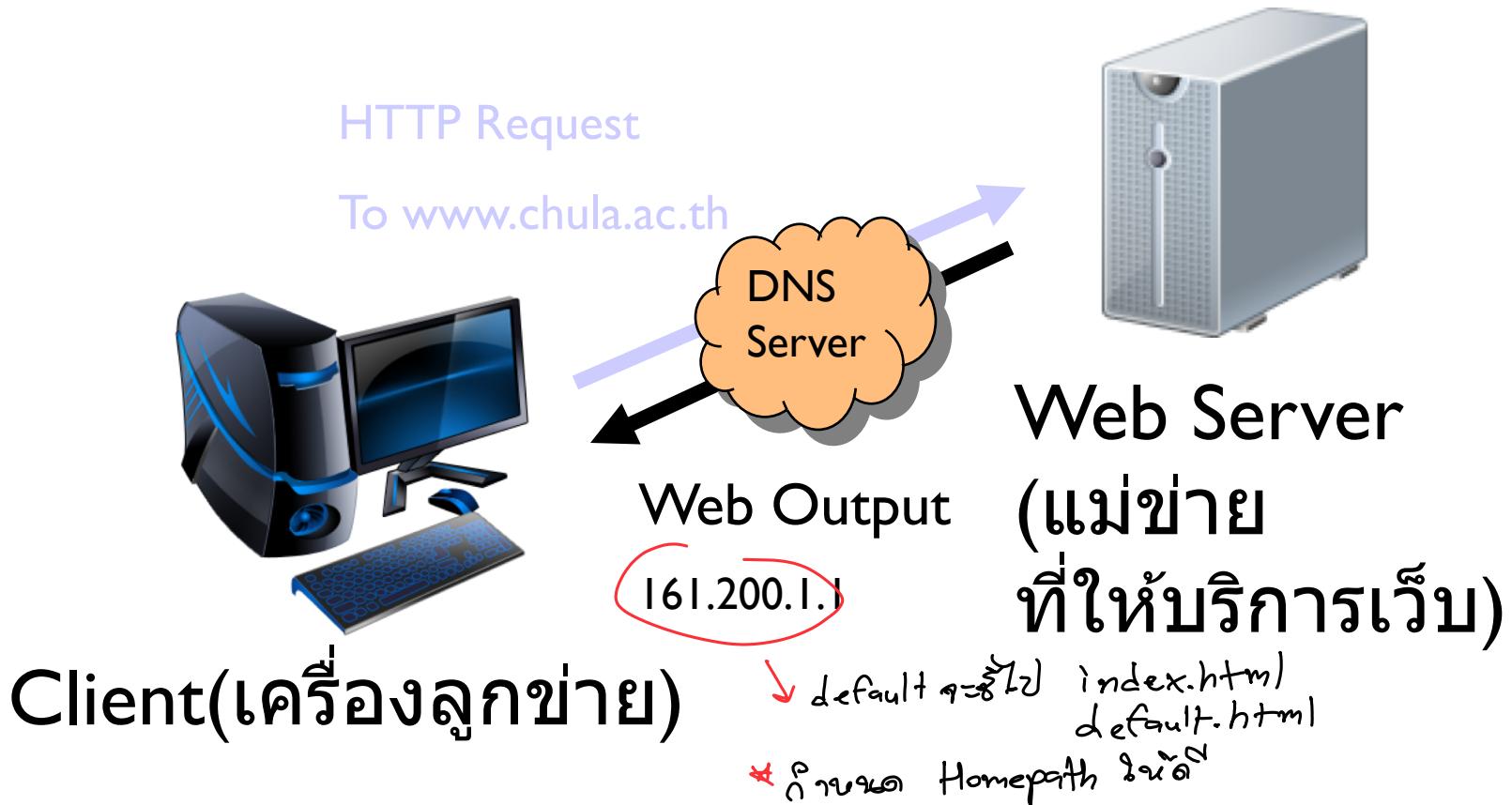
- Microsoft (IIS) ရှိခဲ့သူ Window မြော်စွဲ
- Linux (Apache)
- NginX → Both Windows & Linux

❖ Client Software - Web Browser

- ရုံးဘက်ဆေးစာ

- Internet Explorer(IE)/Microsoft Edge
- Mozilla Firefox → Linux ရုံးစွဲ
- Google Chrome
- Apple Safari

Homepage
→ index.html



HTML

What is HTML?

- HTML stands for **Hyper Text Markup Language**
- HTML is the standard markup language for creating Web pages
- HTML describes the structure of a Web page
- HTML consists of a series of elements
- **HTML elements** tell the browser how to display the content
- **HTML elements** label pieces of content such as "this is a heading", "this is a paragraph", "this is a link", etc.

Ref: https://www.w3schools.com/html/html_intro.asp

Website Technologies

What is an HTML Element?

An HTML element is defined by a start tag, some content, and an end tag:

`<tagname> Content goes here... </tagname>`

The HTML **element** is everything from the start tag to the end tag:

`<h1>My First Heading</h1>`

`<p>My first paragraph.</p>`

Start tag	Element content	End tag
<code><h1></code>	My First Heading	<code></h1></code>
<code><p></code>	My first paragraph.	<code></p></code>
<code>
</code>	<i>none</i>	<i>none</i>

Website Technologies

A Simple HTML Document

Example

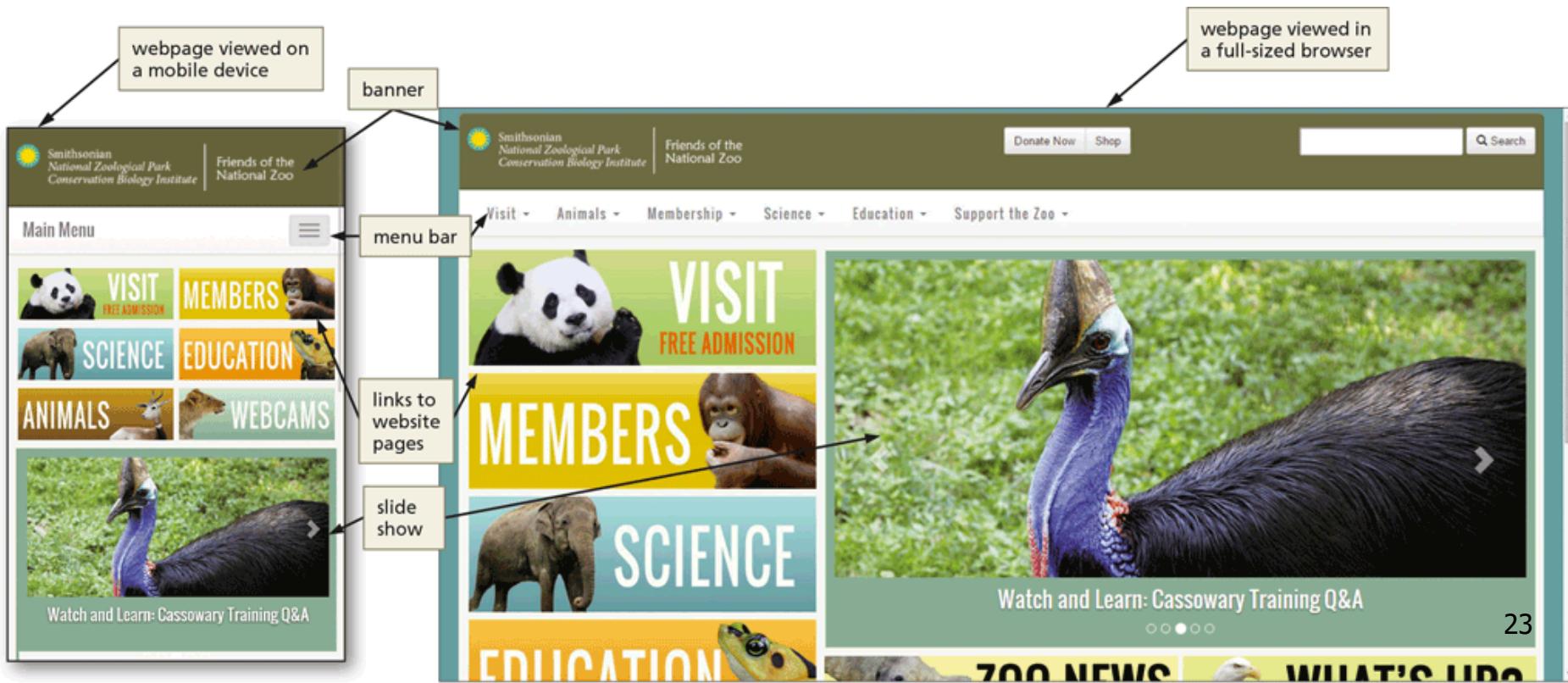
```
<!DOCTYPE html>
<html>
<head>
<title>Page Title</title>
</head>
<body>

<h1>My First Heading</h1>
<p>My first paragraph.</p>

</body>
</html>
```

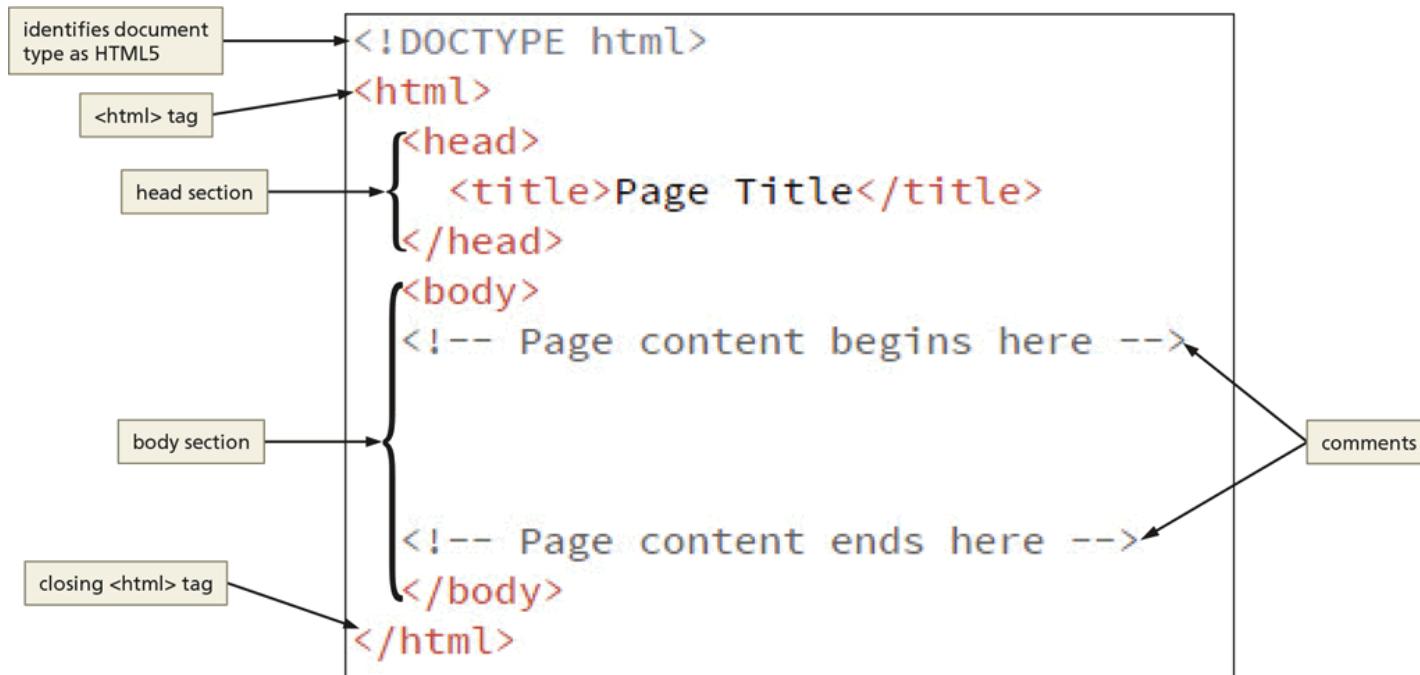
Website Technologies

- ❖ **Responsive webpages** automatically adjust the size of their content to display appropriately relative to the size of the screen of the device on which it is displayed



3. Structure of a Webpage

- ❖ A webpage's source code contains text marked up with **HTML tags** that instruct a browser how to display that content



Structure of a Webpage

- ❖ The World Wide Web Consortium (W3C) oversees the specification of HTML Standards
- ❖ The W3C provides a free, online HTML5 validator application to ensure that a webpage's HTML tags follow the specifications, or rules, for HTML5

4. Creating the index.html File

- ❖ Run the text editor of your choice
- ❖ Navigate to and open the template.html file
- ❖ If necessary, enable the word wrap feature so that you can view all the webpage text without scrolling horizontally
- ❖ Save the file using the file name, index.html. Do not exit the text editor

Creating the index.html File

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>Page Title</title>
5  </head>
6  <body>
7      <!-- Page content begins here -->
8
9
10
11     <!-- Page content ends here -->
12     </body>
13 </html>
```

webpage structure

4.1 Adding the Webpage Title

- ❖ Select the text, **Page Title**, that appears between the **<title>** and **</title>** tags
- ❖ Type Mark's Web Development Page as the title. Replace the name, Mark, with your first name

4.2 Headings

- ❖ Headings **indicate the different sections** of a webpage. HTML supports six levels of headings, which are identified by the following tags: **<h1>, <h2>, <h3>, <h4>, <h5>, and <h6>**

Headings

```
<html>
  <head>
    <title>Headings</title>
  </head>
  <body>
    <h1>Heading level 1</h1>
    <h2>Heading level 2</h2>
    <h3>Heading level 3</h3>
    <h4>Heading level 4</h4>
    <h5>Heading level 5</h5>
    <h6>Heading level 6</h6>
  </body>
</html>
```

The diagram illustrates the hierarchical structure of HTML headings. It shows a block of HTML code on the left, with arrows pointing from specific tags to their corresponding 'Heading level' labels on the right. The code includes an HTML document structure with a title and a body containing six heading tags: h1 through h6. The 'Heading level 1' label points to the first h1 tag. The 'Heading level 2' label points to both the h2 and h3 tags. The 'Heading level 3' label points to the h4 tag. The 'Heading level 4' label points to the h5 tag. The 'Heading level 5' label points to the h6 tag. The 'Heading level 6' label points to the final h6 tag at the end of the body.

Heading level 1

Heading level 2

Heading level 3

Heading level 4

Heading level 5

Heading level 6

4.3 Paragraphs

- ❖ The `<p>` and `</p>` tags are used to identify the beginning and ending of Paragraphs
- ❖ If you have several paragraphs of text on your webpage, these tags will inform the browser to insert additional line spacing above and below the paragraph so that the text is easier to read when displayed in the browser
 - The browser ignores line breaks and line spacing in the HTML file, so it is important to properly define the paragraphs using the `<p>` and `</p>` tags

4.4 Images

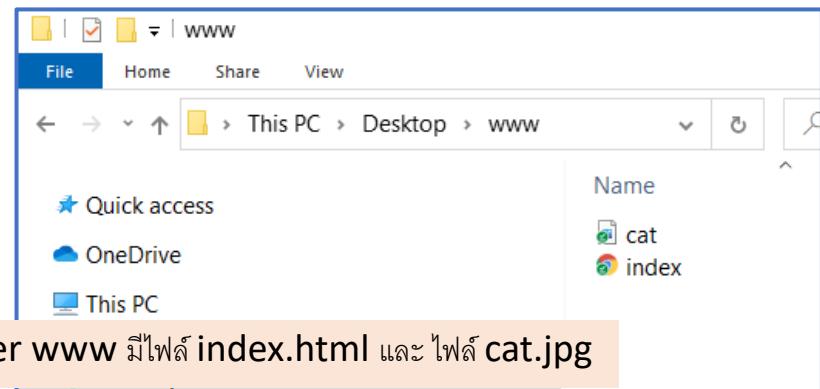
- ❖ Images can be either photos or graphics
- ❖ Images always are stored as separate files, and references to the images appear in the HTML code using the **** tag
- ❖ Common attributes for the **** tag describe the location of an image file, alternate text for the image, and a style that indicates how to position the image
- ❖ ****

การอ้างอิงตำแหน่งไฟล์รูป ใน html

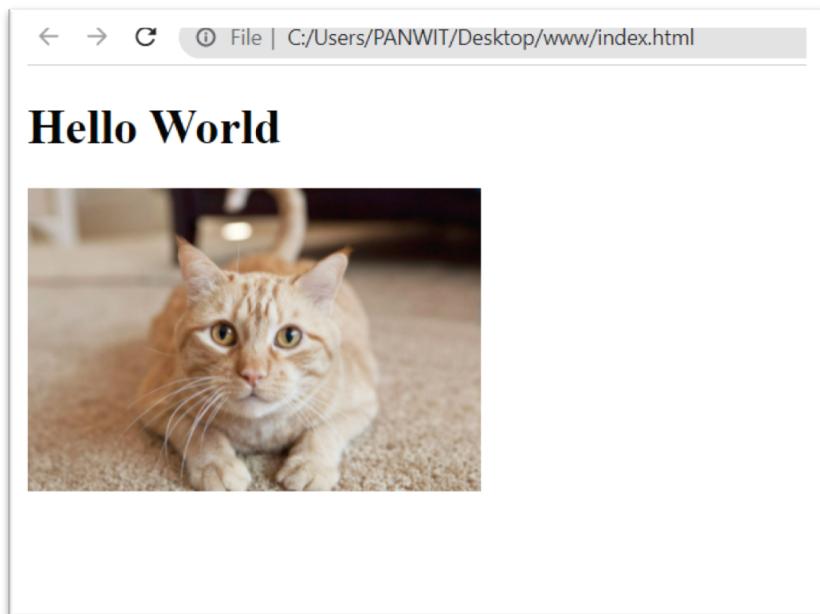
- File นั้นๆ อยู่ใน folder เดียวกัน เช่น ไฟล์รูป และไฟล์ index.html อยู่ใน folder เดียวกัน

ไฟล์ index.html

```
<!DOCTYPE html>
<html>
  <head>
    <title>My Website Title</title>
  </head>
  <body>
    <h1>Hello World</h1>
    
  </body>
</html>
```

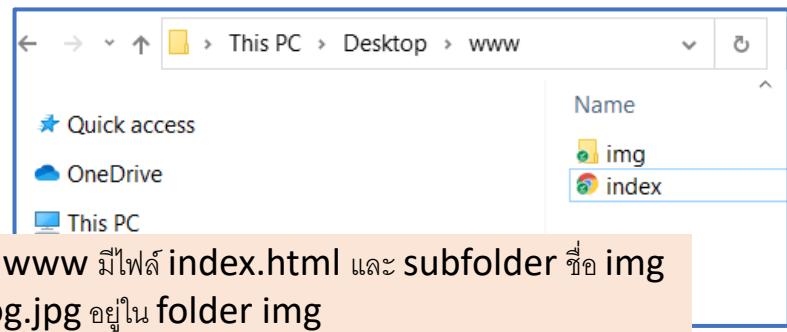


ใน folder www มีไฟล์ index.html และไฟล์ cat.jpg



การอ้างอิงตำแหน่งไฟล์รูปใน html

- File นั้นๆ อยู่คุณละ folder เช่น ไฟล์รูป และไฟล์ index.html อยู่คุณละ folder



ไฟล์ index.html

```
<!DOCTYPE html>
<html>
  <head>
    <title>My Website Title</title>
  </head>
  <body>
    <h1>Hello World</h1>
    
  </body>
</html>
```

Hello World



4.5 Links

- ❖ A link, or **hyperlink**, can be text or an image in a webpage that a user clicks to navigate to another webpage, download a file, or perform another action, such as running an email app and addressing an email message

HYPERLINK WITH ABSOLUTE REFERENCE

```
<a href="http://www.w3.org" target="_blank">HTML5</a>
```

HYPERLINK WITH RELATIVE REFERENCE

```
<a href="video.html"> Watch a video</a>
```

4.6 Unordered and Ordered Lists

Unordered List

```
<ul>
    <li>HTML5</li>
    <li>CSS</li>
    <li>JavaScript</li>
</ul>
```

- HTML5
- CSS
- JavaScript

Ordered List

```
<ol>
    <li>HTML5</li>
    <li>CSS</li>
    <li>JavaScript</li>
</ol>
```

1. HTML5
2. CSS
3. JavaScript

Basic HTML

ไฟล์ index.html

```
index - Notepad
File Edit Format View Help
<!DOCTYPE html>
<html>
<head>
<title>My First HTML</title>
</head>
<body onload="showDateAndTime()">
<h1>My First Heading</h1>
<p>My First Paragraph</p>
<p>HTML links are defined with the a tag:</p>
<a href="https://www.w3schools.com">This is a link</a>
<h2>An Unordered HTML List</h2>
<ul>
<li>Coffee</li>
<li>Tea</li>
<li>Milk</li>
</ul>
<h2>An Ordered HTML List</h2>
<ol>
<li>Coffee</li>
<li>Tea</li>
<li>Milk</li>
</ol>
</body>
</html>
```

แสดงผลที่ Web browser



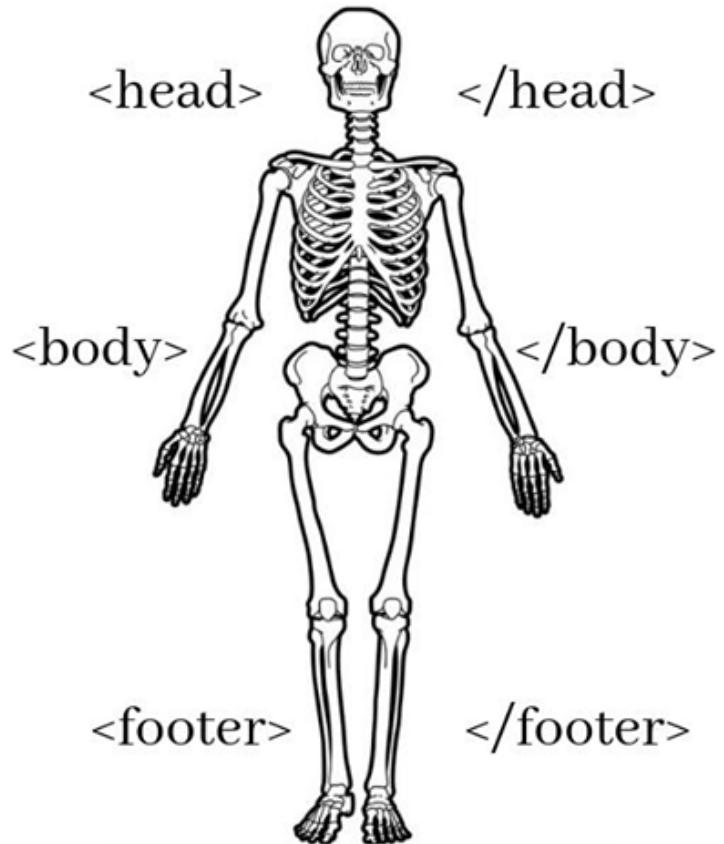
HTML Vs CSS

ກໍານົດລວມຢູ່ຕີ ແຕ່ກົກ້າໃຈໆຂອງ ເຈື່ອ ຈະລັດຕະບຽນ



HTML

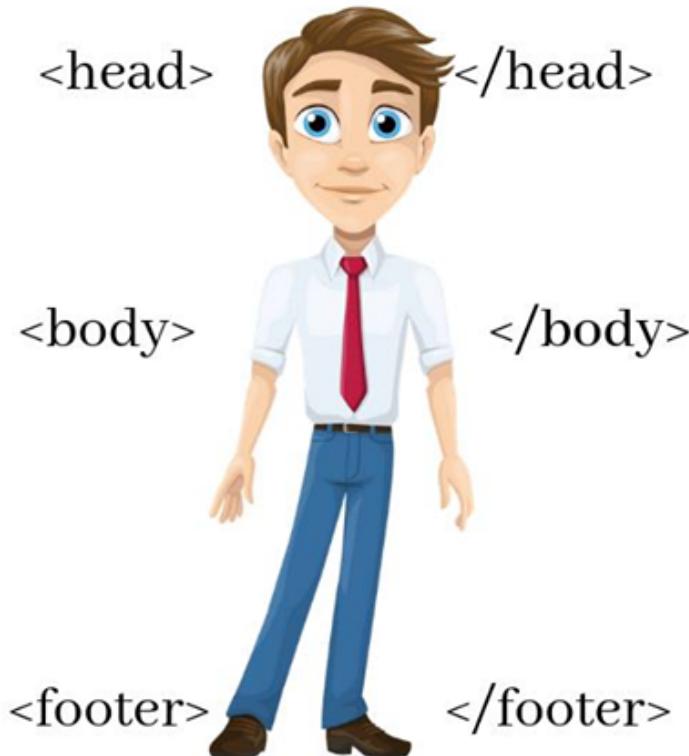
<head>  </head>



Structural Layer

HTML + CSS

<head>  </head>



Presentational Layer

5. HTML with CSS (Cascading Style Sheet)

Internal CSS

ไฟล์ index.html

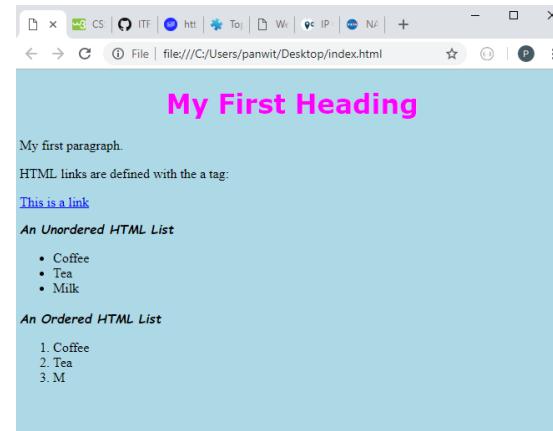
```
<head>
<title>My First HTML</title>
<style>
body {
    background-color: lightblue;
}

h1 {
    font-family: verdana;
    color: #ff00ff;
    text-align: center;
}

h2 {
    font-family: cursive;
    font-size: 15px;
    font-style: italic
}
</style>
</head>
<body>

<h1>My First Heading</h1>
<p>My first paragraph.</p>
<p>HTML links are defined with the a tag:</p>
<a href="https://www.w3schools.com">This is a link</a>
<h2>An Unordered HTML List</h2>
<ul>
```

แสดงผลที่ Web browser



- Embedded styles, which define styles in the **<head> section** of the index.html document, **apply to the entire webpage** on which they are defined

5.1 Applying Styles with CSS

- Although it is possible to customize these elements of webpages using HTML, CSS makes it easier to specify the appearance of similar elements in the same webpage or same website
- Embedded styles, which define styles in the `<head>` section of the `index.html` document, apply to the entire webpage on which they are defined

```
<style>
  h1 {
    font-family:sans-serif;
    color:navy;
    font-style:italic;
  }
  h2 {
    font-family:cursive;
    background-color: navy;
    color:papayawhip;
  }
  p {
    font-family:sans-serif;
    color:rgb(56,0,0);
  }
  ol {
    font-family:sans-serif;
    color:rgb(56,0,0);
  }
  body {
    background-color:#bbccff;
  }
  .fancy {
    font-weight:bold;
    color:red;
    font-style:italic;
  }
</style>
```

5.2 Adding CSS to the index.html File

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>Mark's Web Development Page</title>
5      <style>
6          h1 {
7              font-family:sans-serif;
8              color:navy;
9              font-style:italic;
10         }
11         h2 {
12             font-family:cursive;
13             background-color:navy;
14             color:papayawhip;
15         }
16         p {
17             font-family:sans-serif;
18             color:rgb(56,0,0);
19         }
20         ol {
21             font-family:sans-serif;
22             color:rgb(56,0,0);
23         }
24         body {
25             background-color:#bbccff;
26         }
27         .fancy {
28             font-weight:bold;
29             color:red;
30             font-style:italic;
31         }
32     </style>
33
34
35     </head>
36     <body>
37     <!-- Page content begins here -->
38     <h1>Web Development</h1>
39
40     <p>Three technologies form the foundation for many web applications:</p>
41
42     <ol>
```

styles entered in head section of index.html

5.3 Styling HTML with CSS

- CSS can be added to HTML elements in 3 ways
 - Inline - by using the style attribute in HTML elements
 - Internal - by using a <style> element in the <head> section
 - External - by using an external CSS file

Inline

```
<h1 style="color:blue;">This is a Blue Heading</h1>
```



Internal

```
<!DOCTYPE html>
<html>
<head>
<style>
body {background-color: powderblue;}
h1 {color: blue;}
p {color: red;}
</style>
</head>
<body>

<h1>This is a heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```



External

```
<!DOCTYPE html>
<html>
<head>
<link rel="stylesheet" href="styles.css">
</head>
<body>

<h1>This is a heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

ไฟล์ index.html

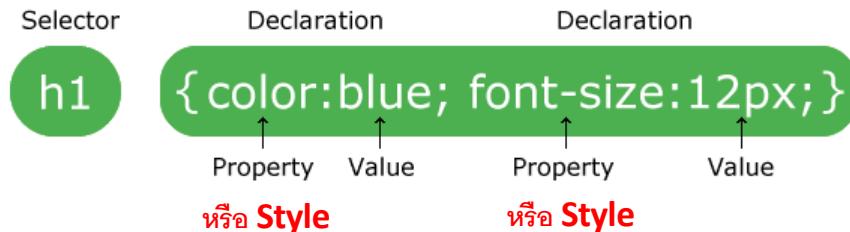
```
body {
background-color: powderblue;
}
h1 {
color: blue;
}
p {
color: red;
}
```

ไฟล์ style.css



5.4 CSS Syntax

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.
- Each declaration includes a CSS **property name** and a **value**, separated by a colon.
- A CSS declaration always ends with a semicolon, and declaration blocks are surrounded by curly braces.

ที่มา https://www.w3schools.com/css/css_syntax.asp

CSS Selector

CSS selectors are used to "find" (or select) the HTML elements you want to style.

All CSS Simple Selector

1. The CSS **element** Selector

```
p {  
    text-align: center;  
    color: red;  
}
```

3. The CSS **class** Selector

```
.center {  
    text-align: center;  
    color: red;  
}
```

5. The CSS **group** Selector

```
h1, h2, p {  
    text-align: center;  
    color: red;  
}
```

2. The CSS **id** Selector

```
#para1 {  
    text-align: center;  
    color: red;  
}
```

4. The CSS **universal** Selector

```
* {  
    text-align: center;  
    color: blue;  
}
```

เพิ่ม https://www.w3schools.com/css/css_selectors.asp

Example of Style

ศึกษาเพิ่มเติมได้ที่

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

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

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

Table 3 Selected Styles

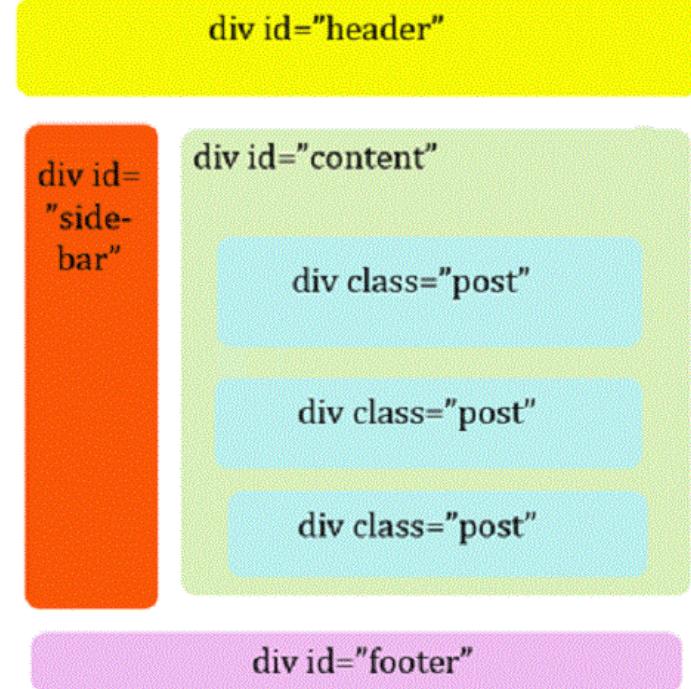
Style	Example	Description
<code>background-color</code>	<code>background-color:yellow;</code>	Specifies the background color of elements, such as <code><p></code> , <code><h1></code> , and <code><body></code>
<code>background-image</code>	<code>background-image:url("images/stripes.jpg")</code>	Sets the background image of a <code><body></code> , <code><p></code> , <code><h1></code> , and other elements to the file whose path is given in the <code>url()</code> function
<code>border</code>	<code>border: 3px red;</code>	Specifies a 4-sided border that is 3 pixels thick
<code>color</code>	<code>color:blue;</code>	Colors can be a web color name, a hexadecimal value, such as <code>#0000FF</code> , or an <code>rgb</code> value, such as <code>rgb(0,0,255)</code> that specifies the red, green, and blue components of the color
<code>float</code>	<code>float:left;</code>	Specifies whether to place an element to the left or right relative to text; often used to position an image to the left or right of text.
<code>font-family</code>	<code>font-family:serif;</code>	Specifies the font for a paragraph, heading, or other text element; use specific font names, such as <code>times</code> , <code>arial</code> , and <code>courier</code> , or family names, such as <code>serif</code> , <code>sans-serif</code> , <code>cursive</code> , or <code>monospace</code>
<code>font-size</code>	<code>font-size:10px;</code>	Specifies font size in pixels
<code>font-style</code>	<code>font-style:italic;</code>	Specifies font style; use <code>normal</code> , <code>italic</code> , or <code>oblique</code> as the value for this style
<code>font-weight</code>	<code>font-weight:bold;</code>	Specifies the font weight for a paragraph, heading, or other text element; use <code>bold</code> for thick characters, or numeric values <code>100</code> through <code>900</code> , in increments of <code>100</code> ; <code>400</code> is the same as <code>normal</code> , <code>700</code> is the same as <code>bold</code>
<code>text-align</code>	<code>text-align:left;</code>	Sets alignment for <code><h1></code> , <code><h2></code> , or <code><h3></code> tags; values can be <code>left</code> , <code>center</code> , or <code>right</code>

5.5 Tag <div>

div หรือ division คือ tag ประเภทหนึ่งของภาษา html

- นำมาใช้แบ่งส่วนต่างๆ ของข้อมูลออกจากกัน
- ใช้สำหรับครอบวัตถุที่ต้องการ เพื่อจัดรูปแบบต่างๆ ให้กับวัตถุในตำแหน่งนั้นๆ
- สามารถใส่ค่าไปตรงๆ ใน div โดยระบุ id หรือ class เพื่อใช้อ้างอิงกับการจัด CSS ได้ตามความเหมาะสม

An HTML page can only have one **unique id** apply to **one specific element**, while a **class** name can be **applied to multiple elements**.



```

<!DOCTYPE html>
<html>
<head>
<style>
/* Style the element with the id "myHeader" */
#myHeader {
    background-color: lightblue;
    color: black;
    padding: 40px;
    text-align: center;
}

/* Style all elements with the class name "city" */
.city {
    background-color: tomato;
    color: white;
    padding: 10px;
}
</style>
</head>
<body>

<h2>Difference Between Class and ID</h2>
<p>An HTML page can only have one unique id applied to one specific element, while a class name can be applied to multiple elements.</p>

<!-- A unique element -->
<h1 id="myHeader">My Cities</h1>

<!-- Multiple similar elements -->
<h2 class="city">London</h2>
<p>London is the capital of England.</p>

<h2 class="city">Paris</h2>
<p>Paris is the capital of France.</p>

<h2 class="city">Tokyo</h2>
<p>Tokyo is the capital of Japan.</p>

</body>
</html>

```

Difference Between Class and ID

An HTML page can only have one unique id applied to one specific element, while a class name can be applied to multiple elements.

My Cities

London

London is the capital of England.

Paris

Paris is the capital of France.

Tokyo

Tokyo is the capital of Japan.

การใช้ id ใน CSS จะถูกกำหนดด้วยเครื่องหมาย #

การใช้ class ใน CSS จะถูกกำหนดด้วยเครื่องหมาย .

CSS frameworks

จากแหล่งที่มาในเว็บ

 Bootstrap	 Tailwind CSS	 Foundation
 Bulma	 Pure	 Semantic UI
 Materialize	 Skeleton	 Milligram

ความคิดเห็น

 tailwindcss Docs Components Blog Showcase

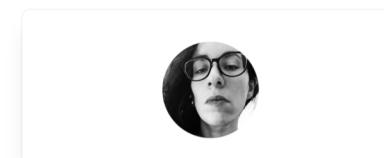
Rapidly build modern websites without ever leaving your HTML.

A utility-first CSS framework packed with classes like `flex`, `pt-4`, `text-center` and `rotate-90` that can be composed to build any design, directly in your markup.

Get started

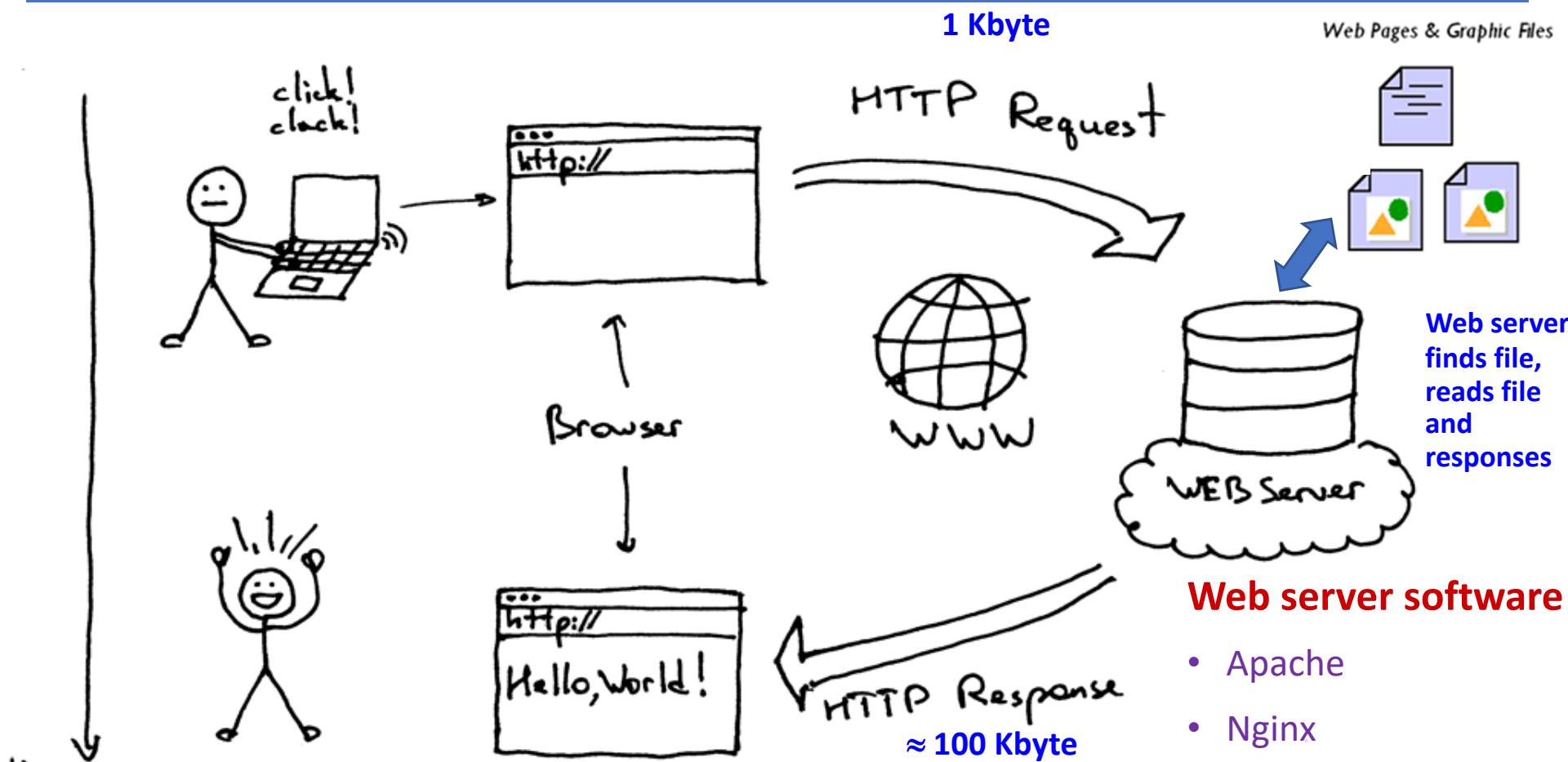
Quick search...

⌘ K



```
1 <figure class="bg-slate-100 rounded-xl p-8 dark:bg-slate-800">
2   
3   <div class="pt-6 text-space-y-4">
4     <blockquote>
5       <p class="text-lg font-medium">
6         Tailwind CSS is the only framework that I've seen sc
7         on large teams. It's easy to customize, adapts to any
```

6. ภาพรวม web client-server



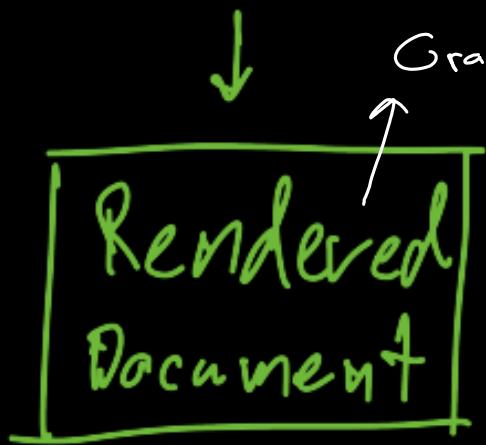
Web Browser Software:

- Chrome, Safari, Firefox, Microsoft Edge

- ## Web server software
- Apache
 - Nginx
 - Caddy
 - Microsoft IIS

client

HTML
Parser



server

index.html

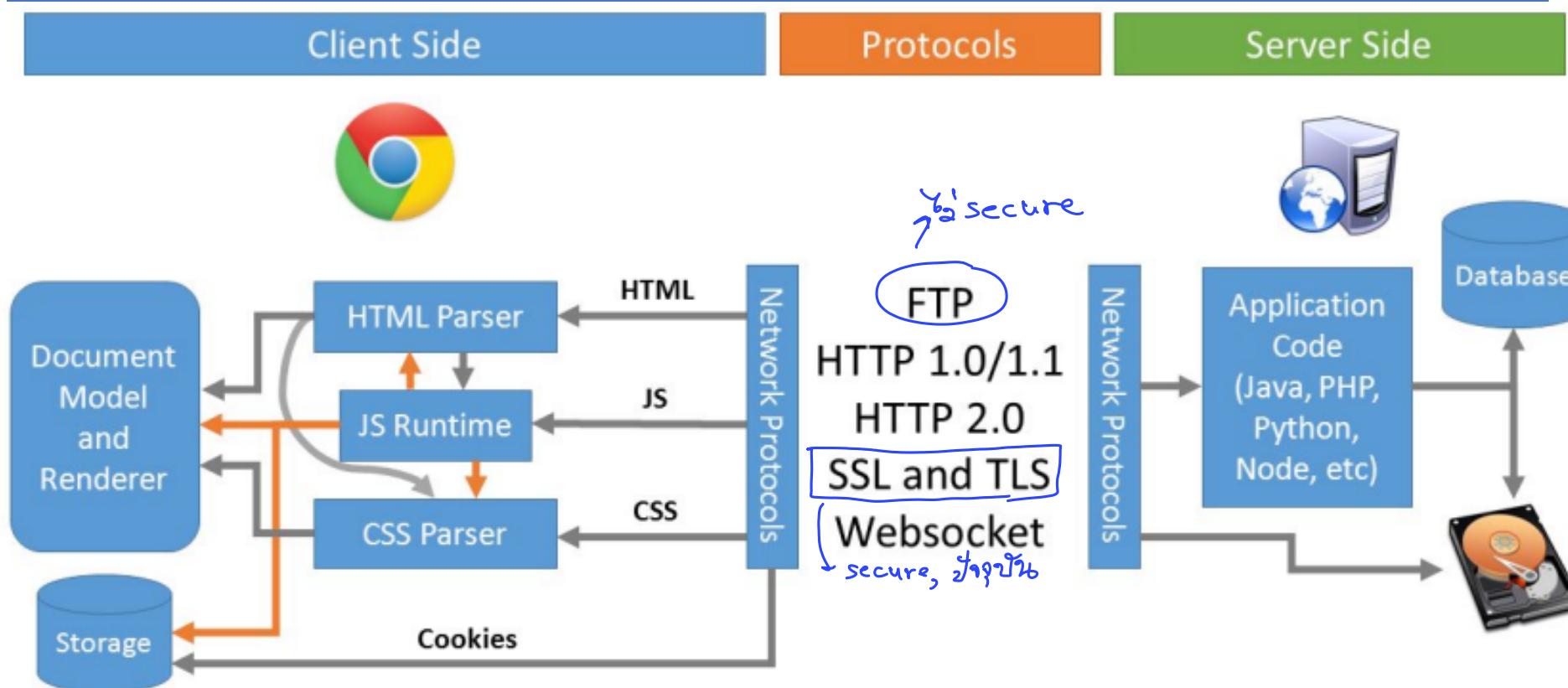


HTML Parser คือตัวแปลง code html ให้กลายเป็น User Interface แสดงผลให้เห็นบน web browser

HTML Parser ทำหน้าที่ render พาก tag ต่างๆ ของ code html ออกมาเป็น User Interface

7. การทำงานของ Web Browser

- HTML, CSS, JS รันไทม์, Image, Database

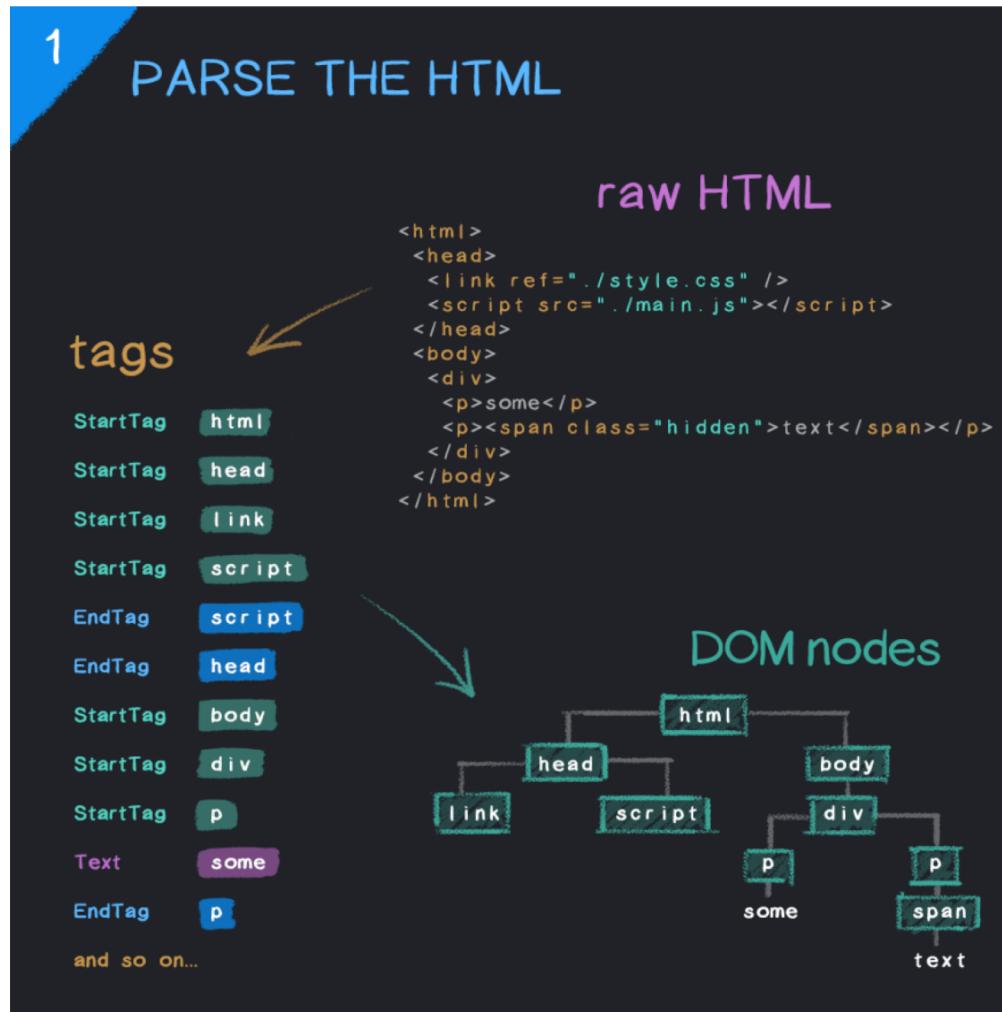


ตัววิเคราะห์คำ แยกคำ ในประโยค

- **HTML Parser** : ทำหน้าที่ render พาก tag ต่าง ๆ ของ code html ออกมาเป็น DOM
- **CSS Parser** : ส่วนไหนใน code html เป็น css ก็จะเข้า CSS Parser ออกมาเป็น CSSOM
- **JS Runtime** : ถ้าใน code html มี JavaScript (js) มันก็จะต้องการ JS runtime environment ในการ run ให้ code js ทำงาน โดยสามารถใช้ js ทำการ อ่าน/แก้ไข html หรือจัดการ css ได้ ซึ่ง browser ทุกตัวบนโลกเดี้ยวนี้มี js runtime อยู่แล้ว

JavaScript Runtime ทำหน้าเพื่อ อ่าน code ภาษา java script และทำงานตามคำสั่ง เพื่อให้ OS ต่างๆ ทำงานได้

7. การทำงานของ Web Browser



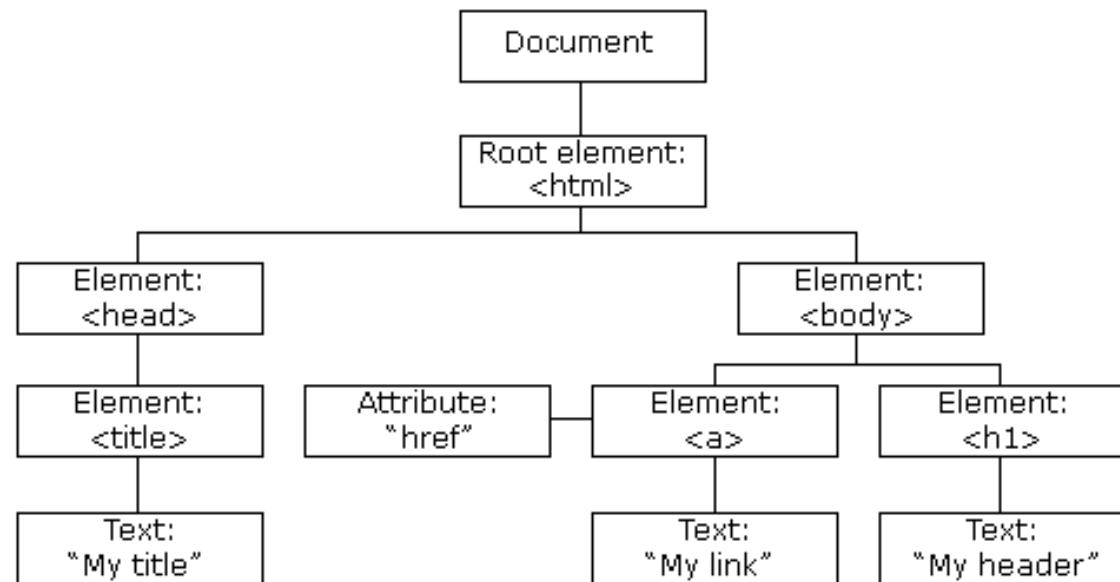
When a web page is loaded, the browser creates a Document Object Model of the page.

DOM หรือ Document Object Model คือโครงสร้างของไฟล์ที่ทาง W3C กำหนดขึ้นเป็นมาตรฐาน

W3C ย่อมาจาก World Wide Web Consortium คือ องค์กรระหว่างประเทศที่ทำหน้าที่จัดระบบมาตรฐานที่ใช้งานบน www

```
1 <html> <!-- Root Element -->
2
3 <head>
4   <title>My title</title>
5 </head>
6
7 <body>
8   <a href="test.html">My link</a>
9   <h1>My header</h1>
10 </body>
11
12 </html>
```

index.html hosted with ❤ by GitHub



7. การทำงานของ Web Browser

2

FETCH EXTERNAL RESOURCES

CSS blocks rendering

JavaScript blocks parser...

unless!

defer waits until parser is finished

async executes as soon as it loads

7. การทำงานของ Web Browser

3 PARSE THE CSS AND BUILD THE CSSOM

```
body { font-size: 16px; }
div { font-size: 14px; }
p { font-weight: bold; }
.hidden { display: none; }
span { display: block; }
```

Any CSS style can potentially be overridden or changed by some other style loaded on the page, which is why the CSSOM cannot be fully constructed until all of the page's stylesheets have been loaded.

```
.hidden { display: none; }
span { display: block; }

Inherited from p
p {
    font-weight: bold;
}

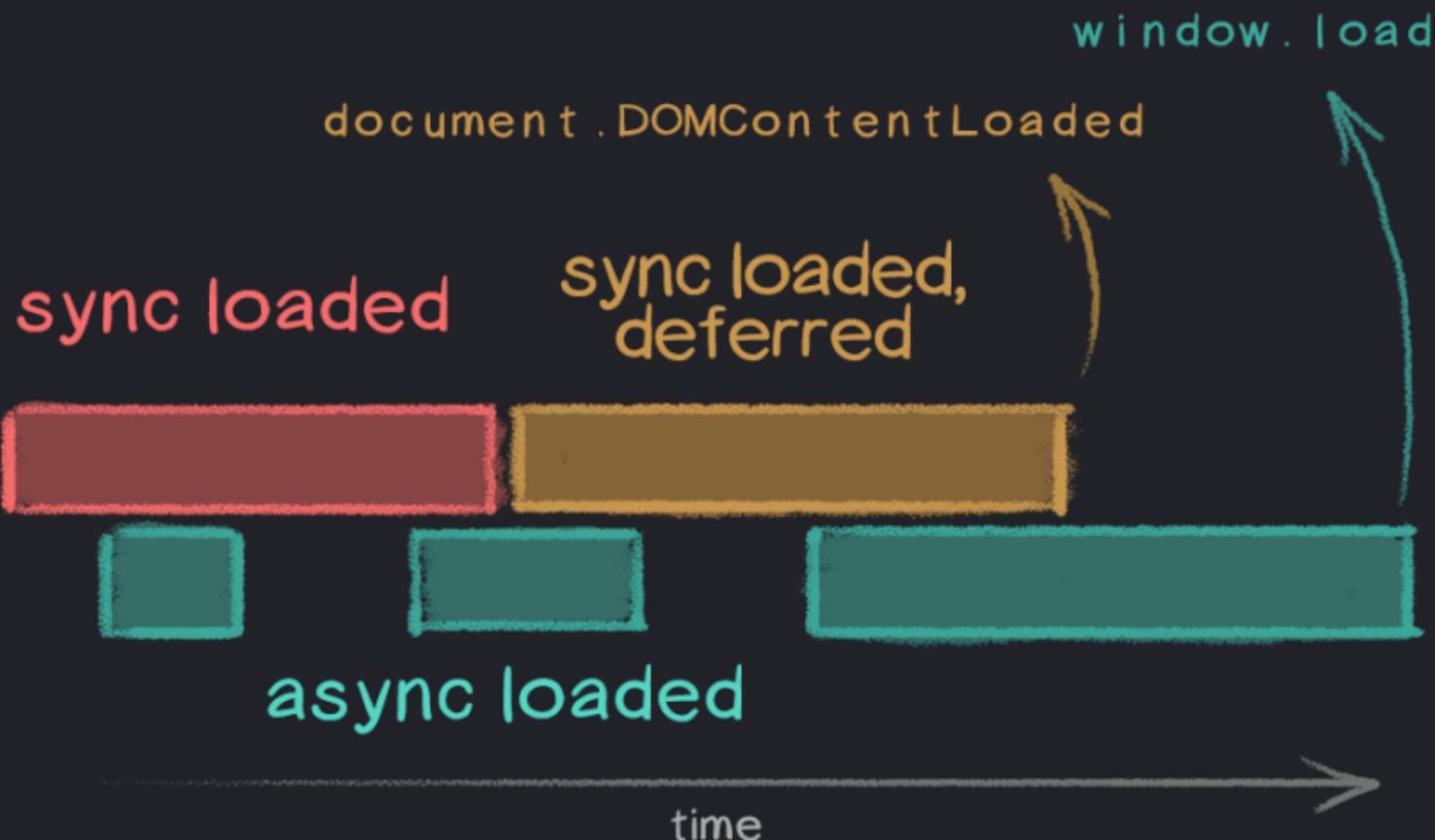
Inherited from div
div {
    font-size: 14px;
}

Inherited from body
body {
    font-size: 16px;
}
```

7. การทำงานของ Web Browser

4

EXECUTE THE JAVASCRIPT

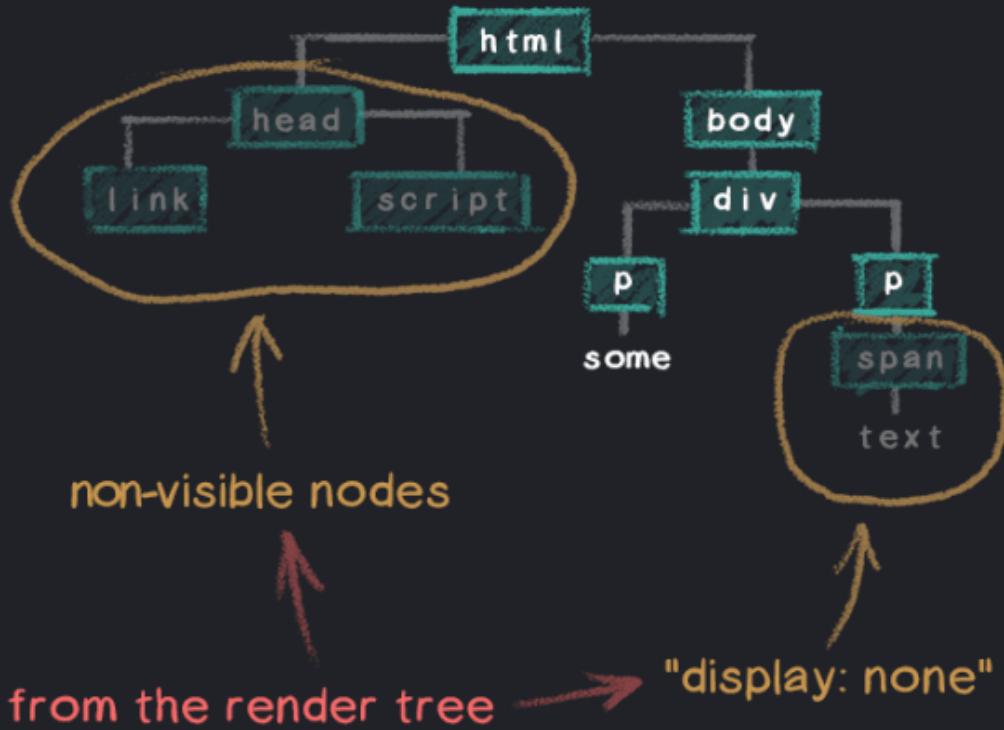


7. การทำงานของ Web Browser

5

MERGE DOM AND CSSOM TO CONSTRUCT THE RENDER TREE

```
body {  
    font-size: 16px;  
}  
  
div {  
    font-size: 14px;  
}  
  
p {  
    font-weight: bold;  
}  
  
span {  
    display: none;  
}
```



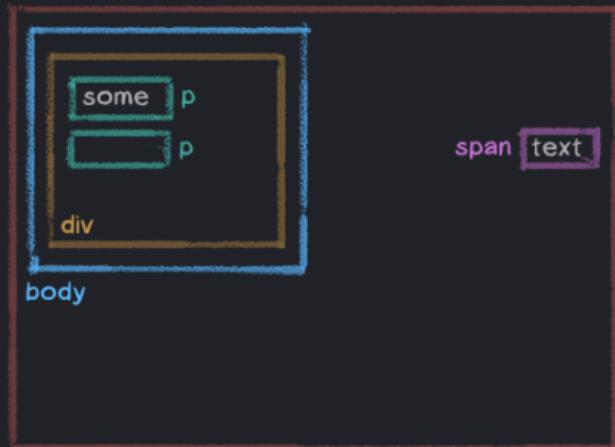
7. การทำงานของ Web Browser

6

CALCULATE LAYOUT AND PAINT

```
<html>
  <head>
    <link href="./style.css" />
  </head>
  <body>
    <div>
      <p>some</p>
      <p><span>text</span></p>
    </div>
  </body>
</html>
```

html



let's add some layout styles...

```
* {
  padding: 25px;
}

html {
  height: 100%;
}

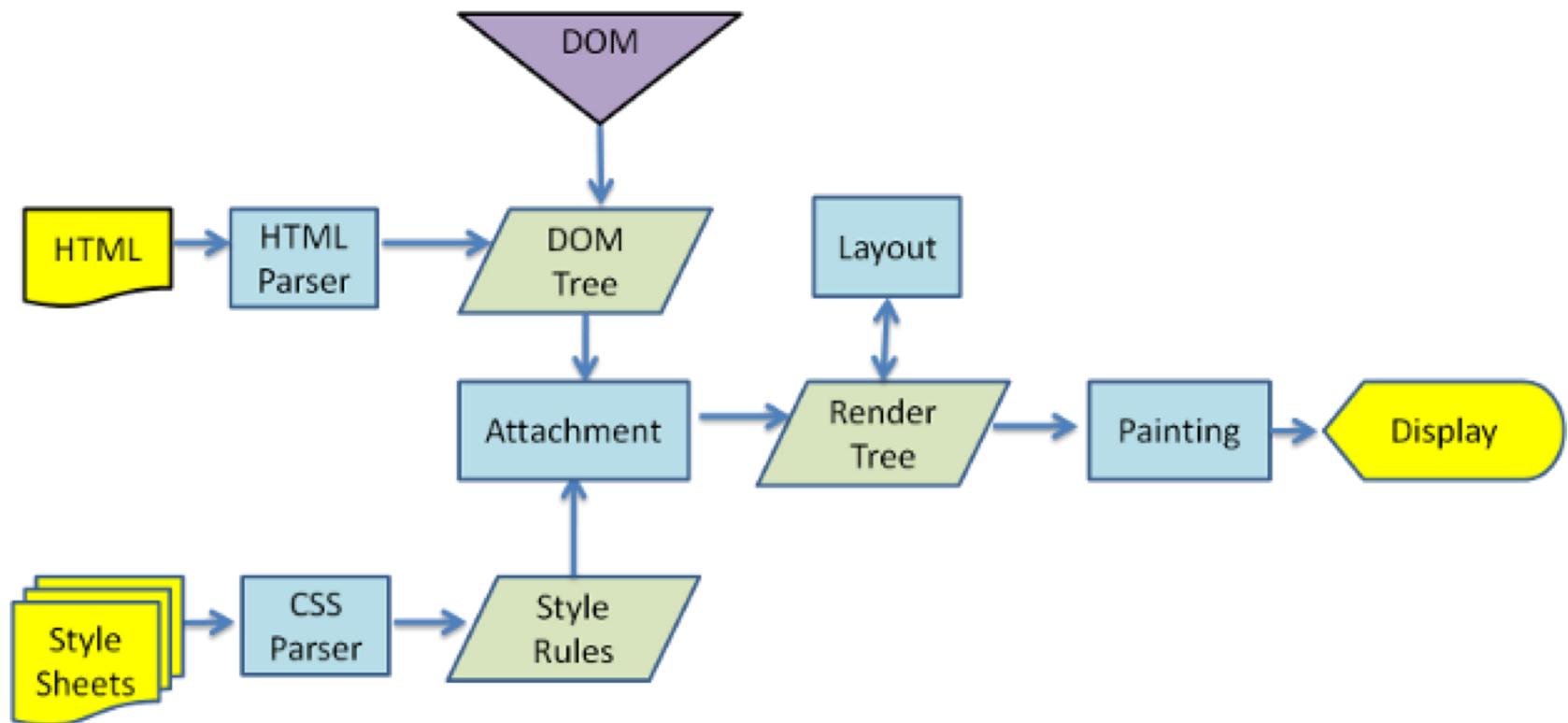
body {
  width: 50%;
}

div {
  height: 200px;
}

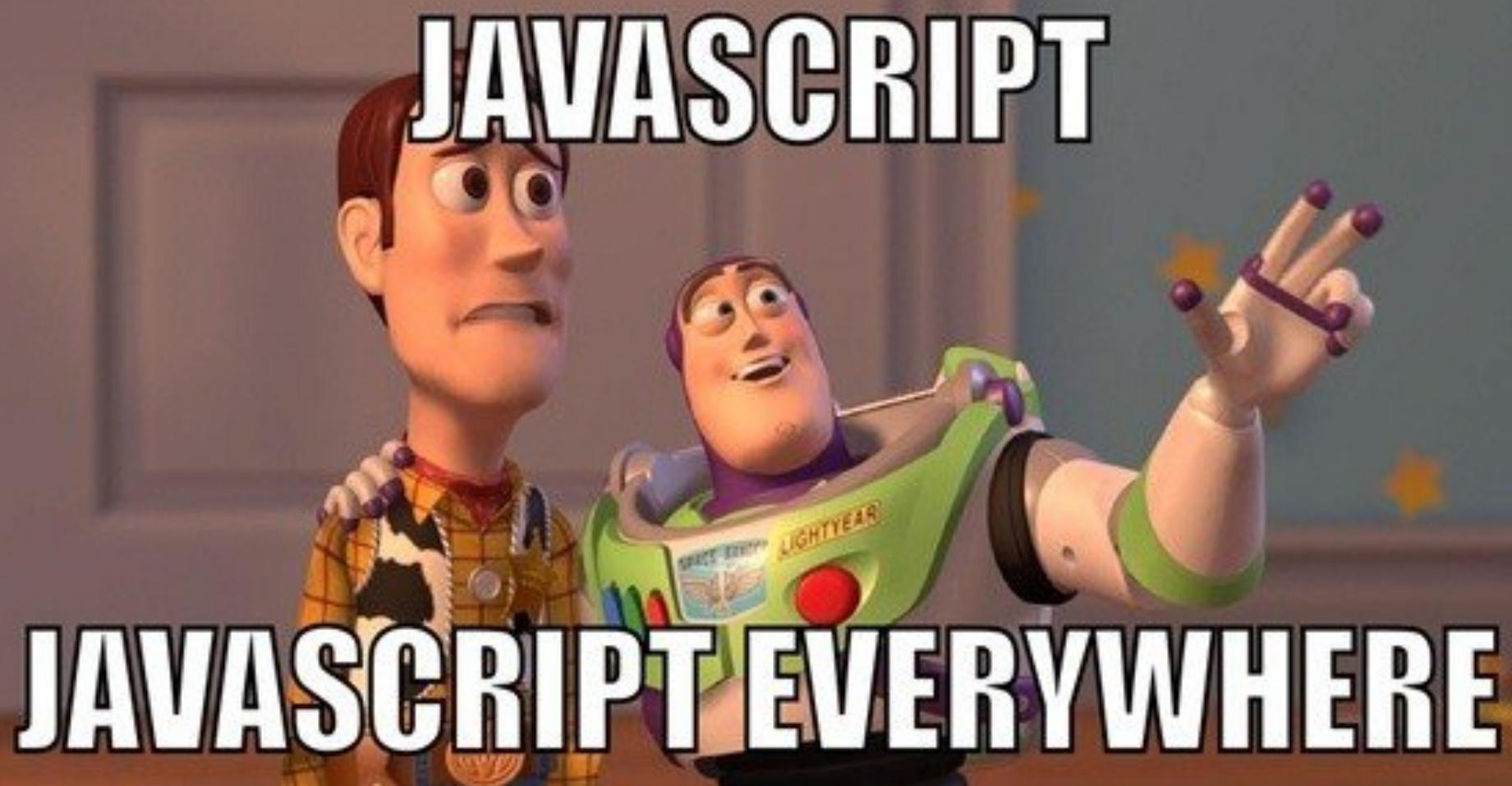
p {
  width: 50%;
}

span {
  position: absolute;
  right: 25px;
}
```

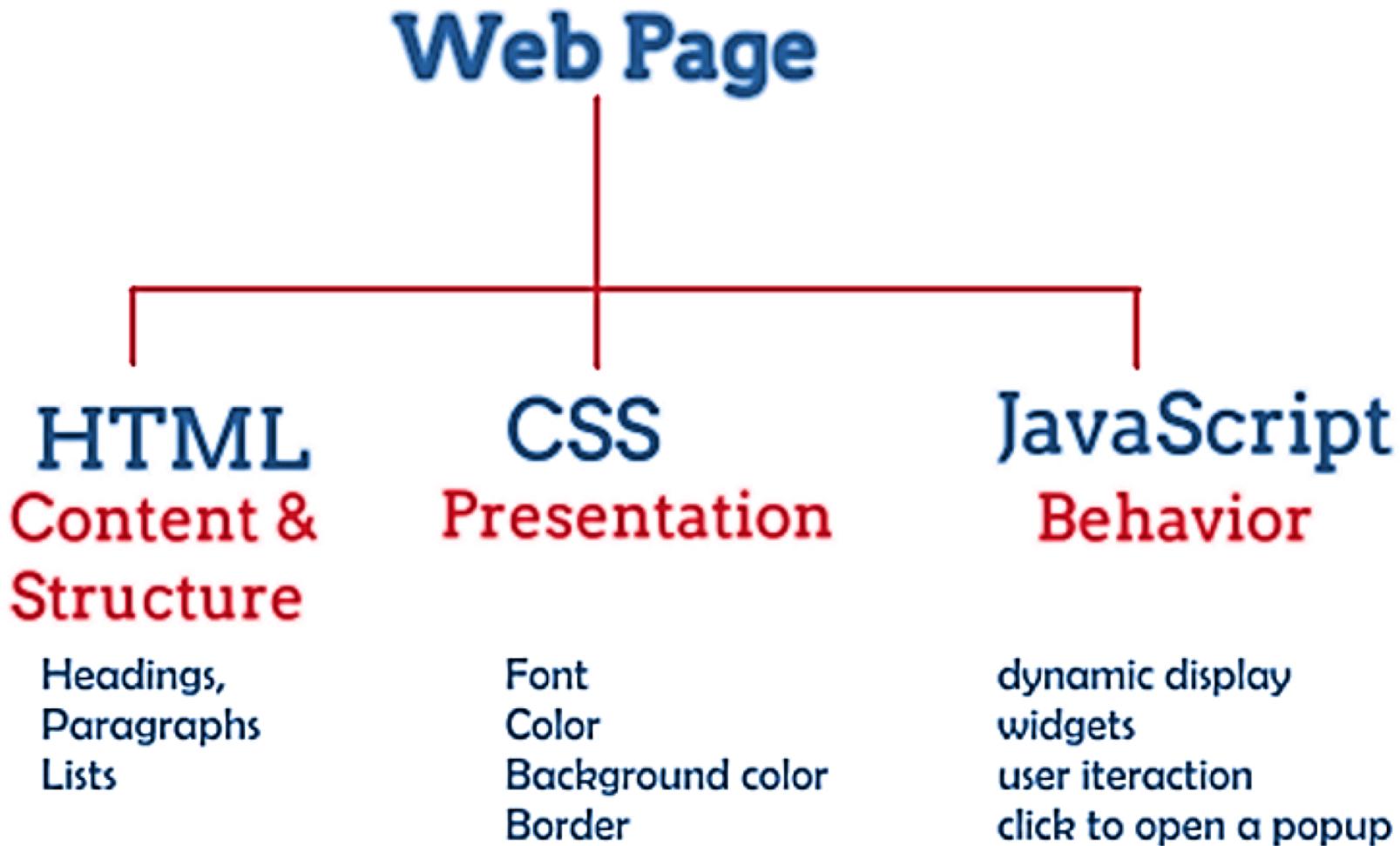
7. การทำงานของ Web Browser



8. JavaScript



Html, CSS, JavaScript



Html, CSS, JavaScript



HTML



HTML + CSS



**HTML + CSS
+ JAVASCRIPT**



house **folder**



index.html



styles.css



scripts.js

8. JavaScript

- เดิมทีภาษา JavaScript ทำงานที่ Web browser เป็นหลัก โดยทำหน้าที่เปลี่ยนแปลงโครงสร้างของ HTML เป็นหลัก



- JavaScript Can Change HTML Content
- JavaScript Can Change HTML Attribute Values
- JavaScript Can Change HTML Styles (CSS)
- JavaScript Can Hide/Show HTML Elements

8. JavaScript

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

<h2>What Can JavaScript Do?</h2>

<p id="demo">JavaScript can change HTML content.</p>

<button type="button" onclick="document.getElementById('demo').innerHTML
= 'Hello JavaScript!'">Click Me!</button>

</body>
</html>
```

- **JavaScript Can Change HTML Content**

What Can JavaScript Do?

JavaScript can change HTML content.

Click Me!

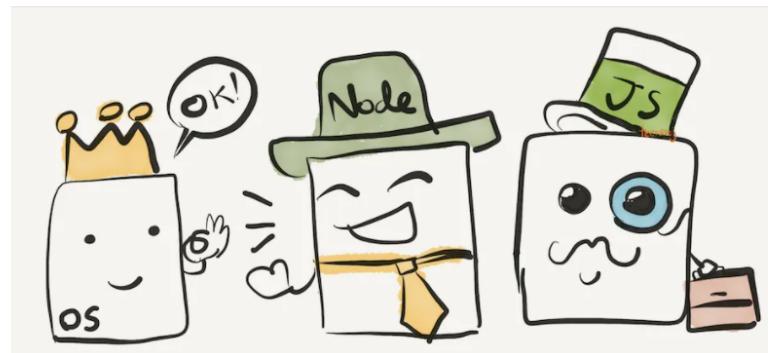
What Can JavaScript Do?

Hello JavaScript!

Click Me!

8. JavaScript

- ต่อมานักคนอยากรู้ว่า JavaScript ทำงานนอก Web Browser ได้เพื่อเพิ่มความสามารถของมัน จึงมีคนคิดค้นสร้างสิ่งที่เรียกว่า Node
- Node หรือ Node.js คือ JavaScript Runtime ที่ถูกสร้างด้วย Chrome's V8 JavaScript Engine
- โดย JavaScript Runtime ทำหน้าเพื่อ อ่าน code ภาษา JavaScript และทำงานตามคำสั่ง เพื่อให้ OS ต่างๆ ทำงานได้



Full Stack JavaScript Tools and Technologies

Front End



Or



React



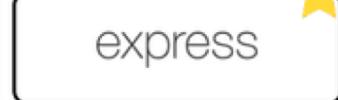
Knockout.



BACKBONE.JS



Back End



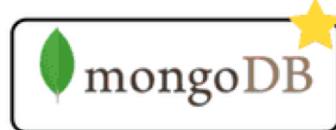
Or



KeystoneJS

For Dynamic Web

Database



Or

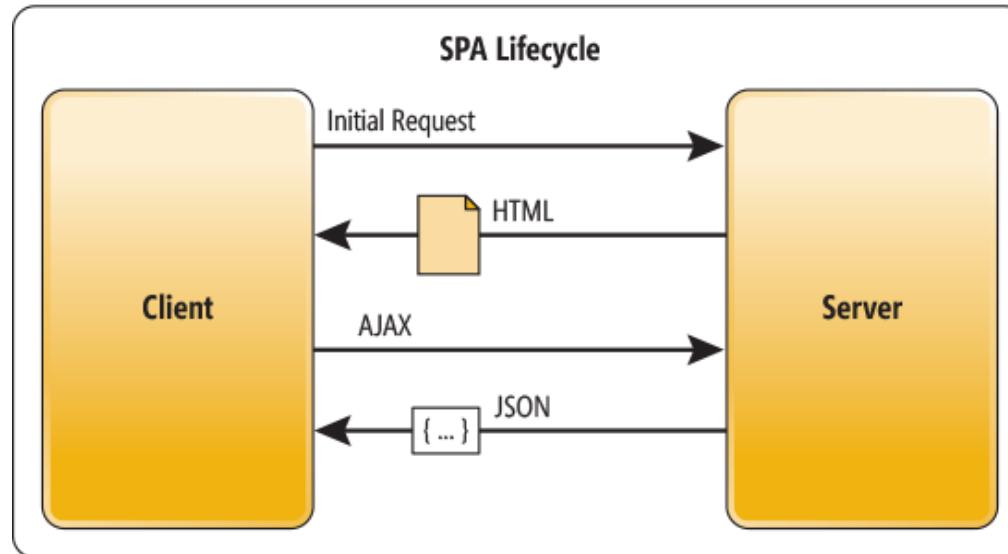
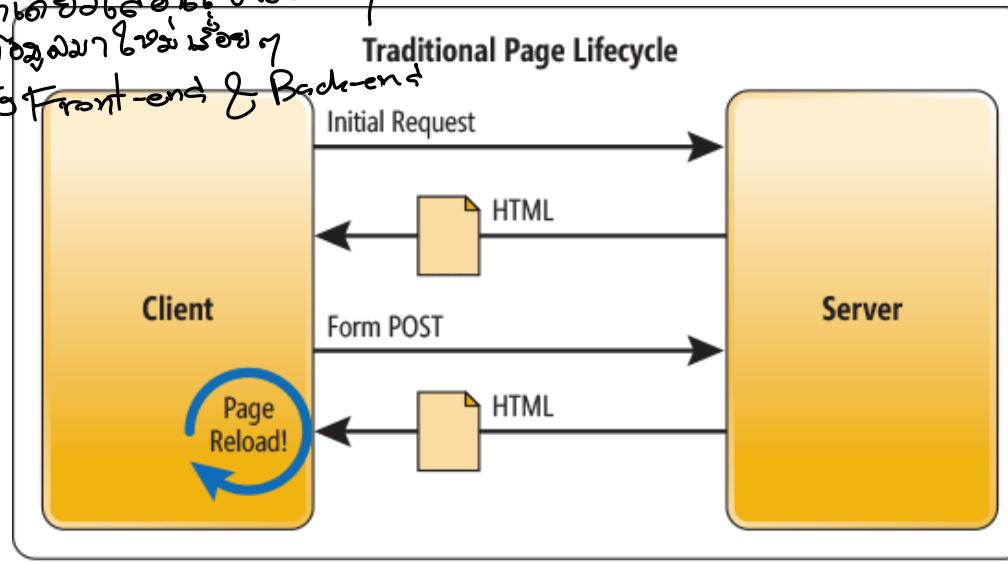


PostgreSQL



9. Single Page Application

- Facebook ខ្សោយការណ៍ទីលាច់លាច់
→ reload រួចរាល់បានចុងក្រោម
→ សំគាល់ Front-end & Back-end



Single Page Application

Server-Side Rendering (SSR)

Step 1

The page loads and requests static assets from the web server.

The data is included in the HTML.



Step 2

The browser has all the information it needs, the page is rendered immediately.



Step 3

The user interacts with the page. This triggers a new request to the server, the page goes blank until it reloads.



Step 4

The new page loads, showing new content to the user.

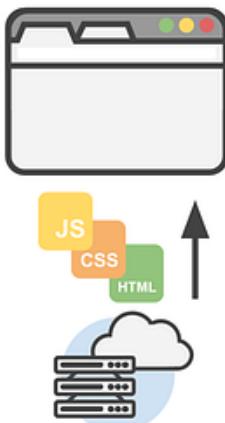


Single Page Application

Single-Page Application (SPA)

Step 1

The page loads and requests static assets from the web server.



Step 2

The frontend framework renders the layout using the HTML and CSS.

The JavaScript is executed and requests data.



Step 3

The JavaScript loads the data in the page.



Step 4

The user interacts with the page, more data is requested and loaded.

The page is updated without having to reload.



Single Page Application

Companies Using Single-page Applications



Twitter



Instagram



Airbnb



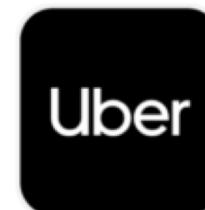
Facebook



Netflix



Gmail

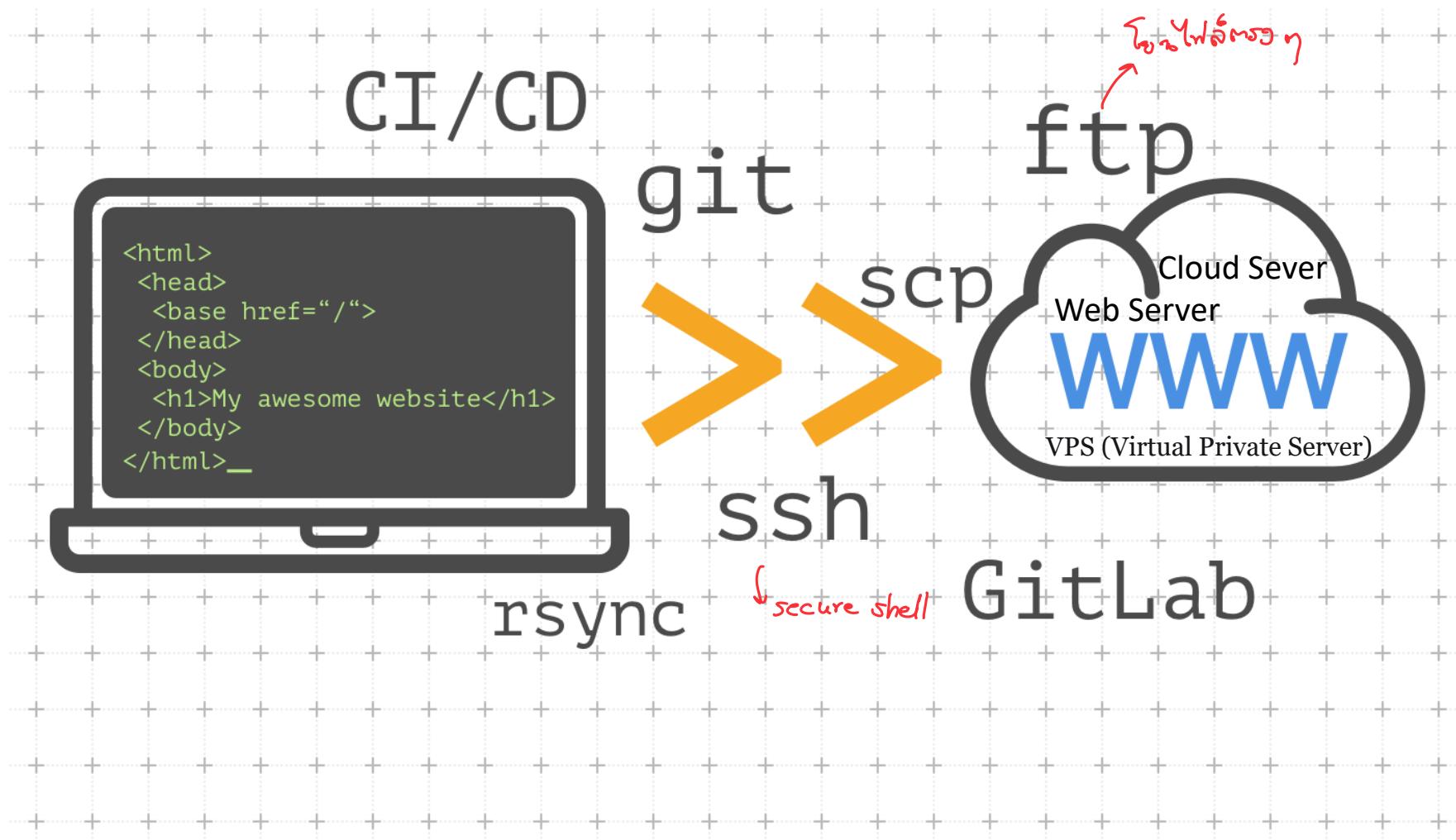


Uber

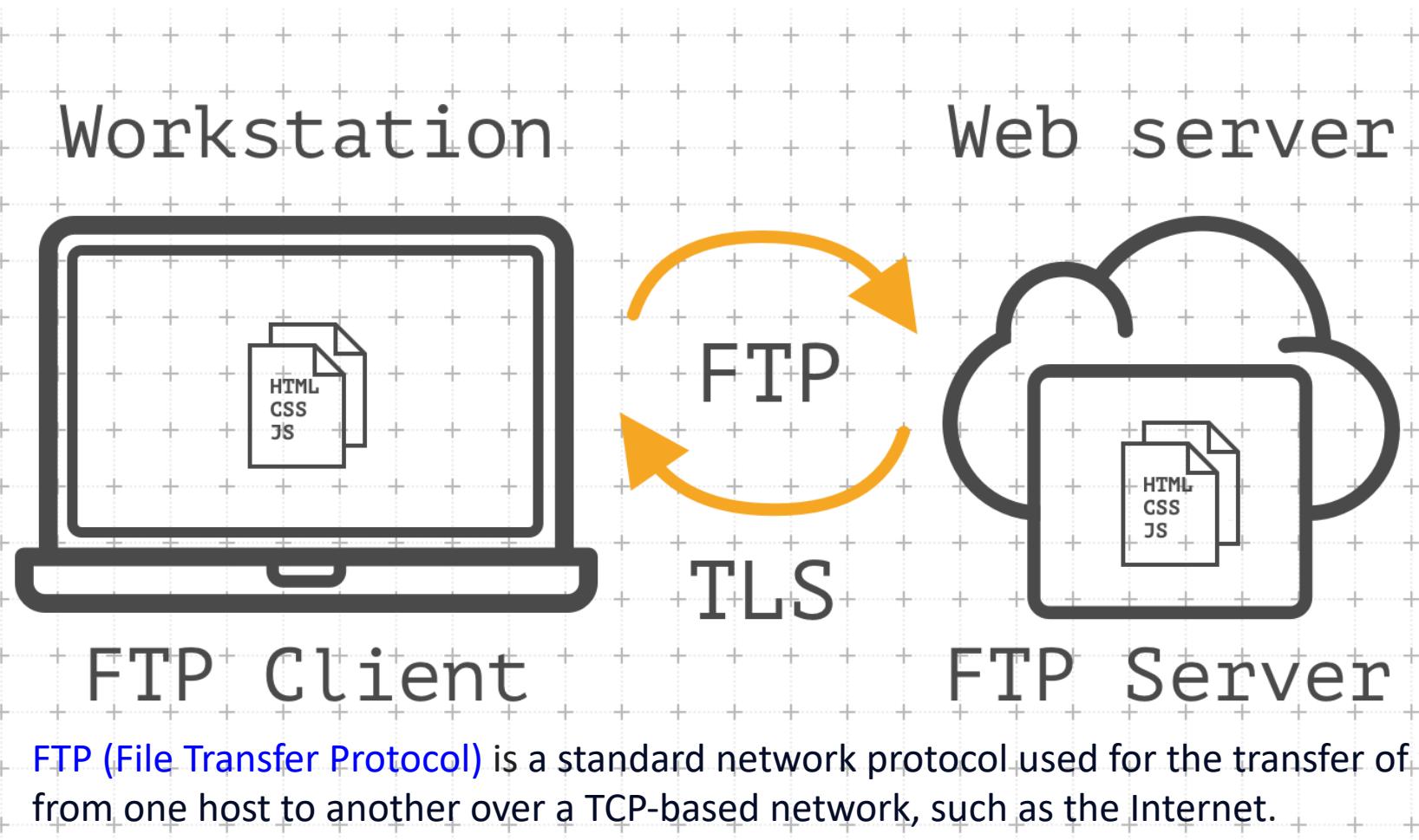
ශුරු web/App මෙහිදැක්වා architecture
නොව single page app. හේතු



Web Deployment Examples



Web Deployment using FTP



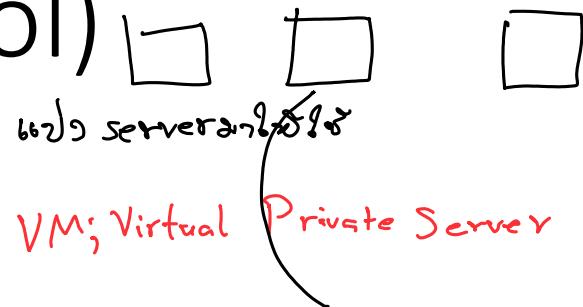
FTP (File Transfer Protocol) is a standard network protocol used for the transfer of files from one host to another over a TCP-based network, such as the Internet.

TLS (Transport Layer Security)

SFTP/SSL (ปัจจุบันในการใช้งาน)

FTP(File Transfer Protocol)

- ชื่อแรกเริ่มไฟล์ใน server



FTP Request

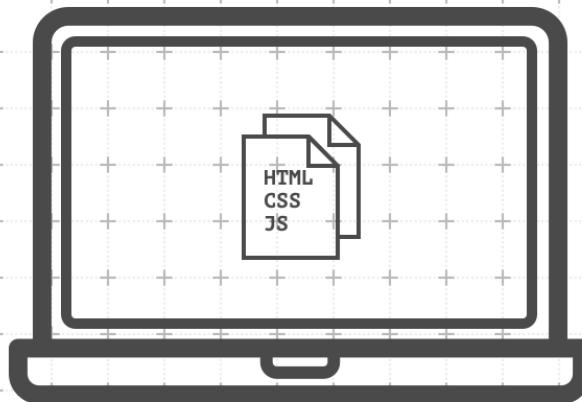


FTP Server
(แม่ข่าย
ที่ให้บริการFile)

Client(เครื่องลูกข่าย)

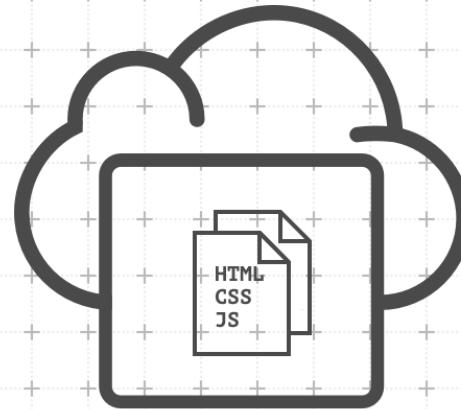
Web Deployment using SCP/SSH

Workstation



scp → ssh

Web server



SCP (stands for 'secure copy') - allow copying files from a local host to a remote host using ssh protocol.

Secure Shell (SSH) a cryptographic network protocol for operating network services securely over an unsecured network. Its most notable applications are remote login and command-line execution.



FileZilla

<https://filezilla-project.org>

:

FileZilla - The free FTP solution

FileZilla - The free **FTP** solution for both **client** and server. Filezilla is open source **software** distributed free of charge.

[Download FileZilla Client](#)

[Download FileZilla Server](#)

[Client Features](#)

[Screenshots](#)



WinSCP

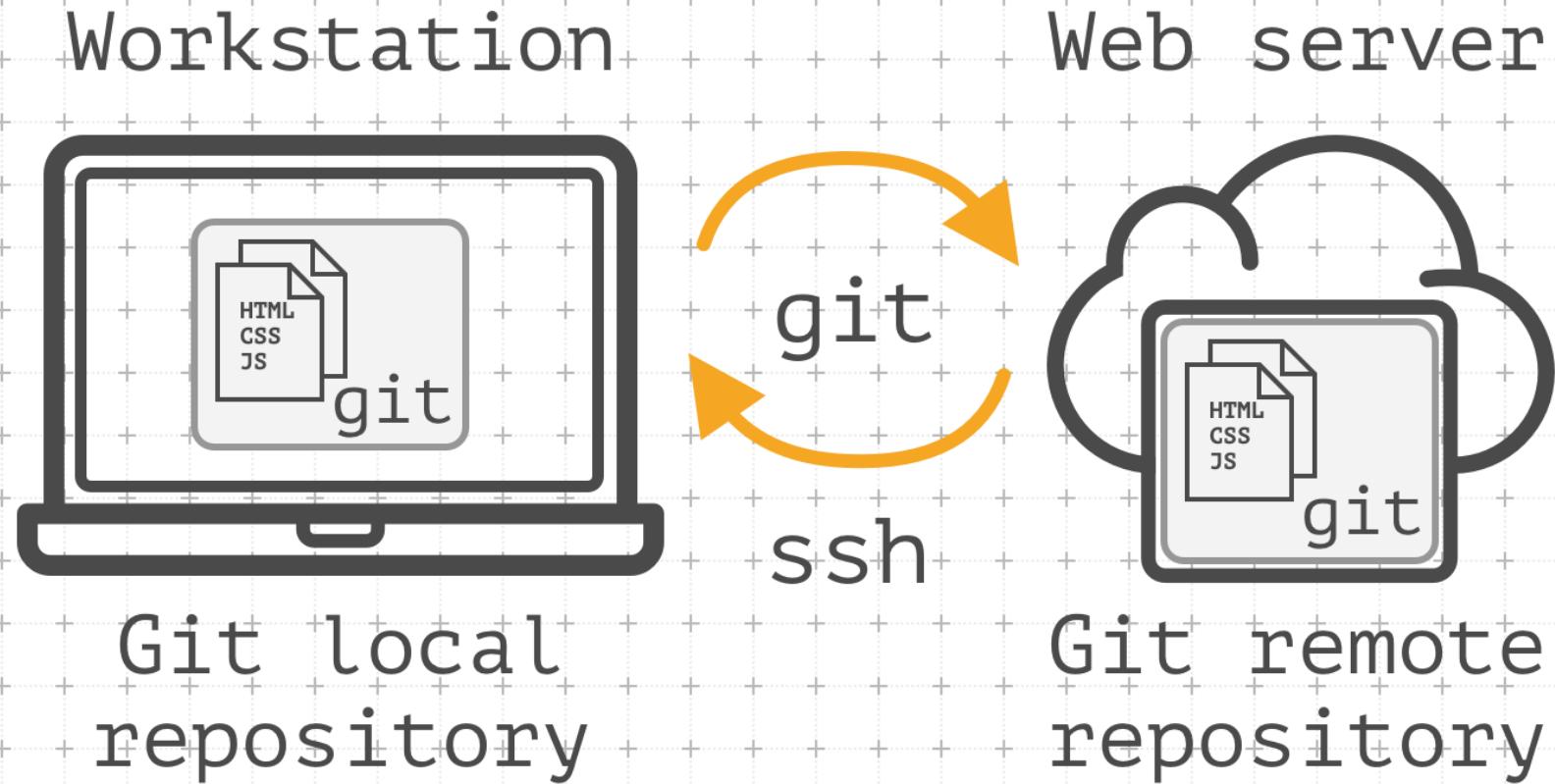
https://winscp.net/eng/docs/free_ftp...

:

Free FTP Client for Windows

10 ก.ย. 2567 — WinSCP is an open source free **FTP client** for Windows. You can get it from WinSCP download page. Latest stable WinSCP version is 6.3.5.

Web Deployment using SSH/GIT



```
$ ssh user@remote-host
$ sudo mkdir /opt/webserver/www/example.com
$ sudo chown ubuntu:ubuntu /opt/webserver/www/example.com
$ git init --bare ~/myawesomewebsite.git
Initialized empty Git repository in /home/ubuntu/myawesomewebsite.git

$ cd ~/myawesomewebsite.git
```

Secure Shell (SSH)

Remote Desktop with Command Line

- Windows Cmd
- MacOS Terminal
- Putty



SSH admin@161.200.1.1

Secure Shell
(SSH)



Web Server

Client(เครื่องลูกข่าย)

```
$ ssh user@remote-host
$ sudo mkdir /opt/webserver/www/example.com
$ sudo chown ubuntu:ubuntu /opt/webserver/www/example.com
$ git init --bare ~/myawesomewebsite.git
Initialized empty Git repository in /home/ubuntu/myawesomewebsite.git

$ cd ~/myawesomewebsite.git
```

Web Server Installation

- **OS** (Ubuntu Server/ Windows Server) Secure nén Win 11
- **Web Server Software** (Apache/NginX/IIS)

Optional – Dynamic/BackEnd

- **Database – DBMS** (MySQL/MariaDB/MSaccess)
- **Server Side Script - Interpreter/Complier/Node** (PHP/Node.JS/ASP.NET/Python)

CMS – WordPress - Installation

- **OS** (Ubuntu Server/ Windows Server)
- **Web Server Software** (Apache/NginX)
- **Database – DBMS** (MySQL/MariaDB)
- **Server Side Script – Interpreter** (PHP)

<https://wordpress.org/about/requirements/>

Requirements

To run WordPress, it's recommended your host supports:

- PHP version 7.4 or greater.
- MySQL version 8.0 or greater OR MariaDB version 10.5 or greater.
- HTTPS support

WordPress requirements



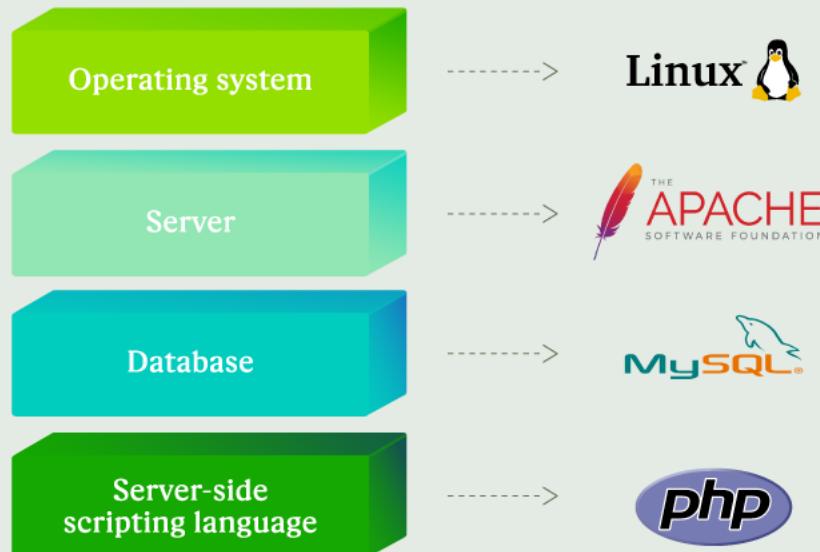
That's really it. [Apache](#) or [Nginx](#) is recommended as the most robust and featureful server for running WordPress, but any server that supports PHP and MySQL will do. That said, for the smoothest experience in setting up—and running—your site, [each host on the hosting page](#) supports the above and more with no problems.

For detailed PHP extension recommendations, see the [Hosting Handbook](#).

Software Stack

Software Stack

→ ໂັດທີ່ເຄື່ອນໄຫວ ຈຸບ



Software Stack Examples

LAMP: Linux/Apache/MySQL/PHP



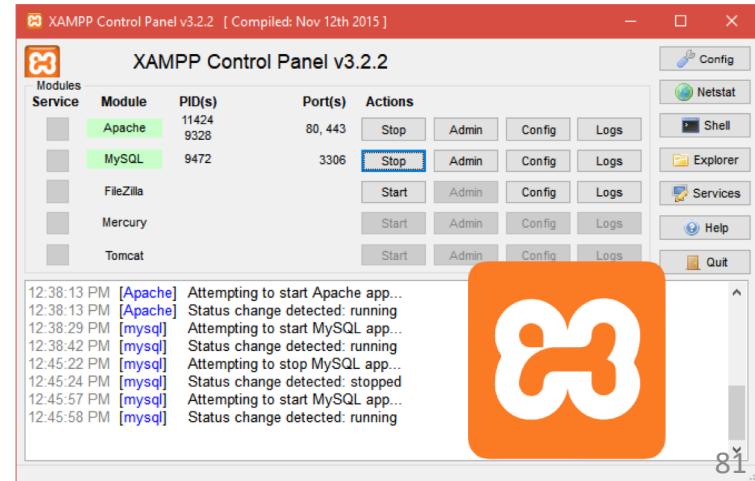
MEAN: MongoDB/Express.js/AngularJS/Node.js



Ruby Stack: Ruby/Ruby on Rails/RVM (Ruby Virtual Machine)/MySQL/Apache/PHP



XAMPP Apache + MariaDB + PHP + Perl



10. Introduction to API

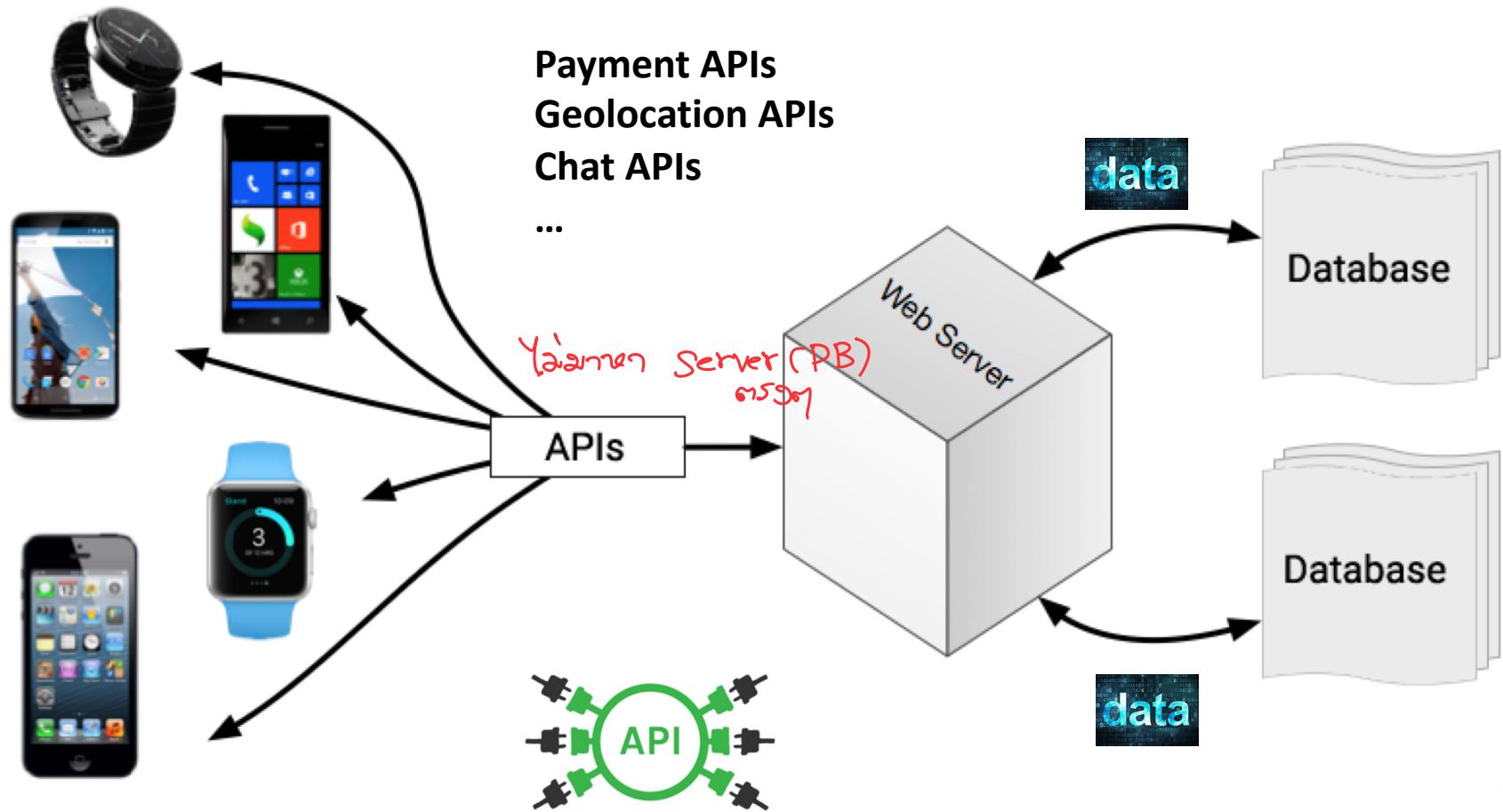


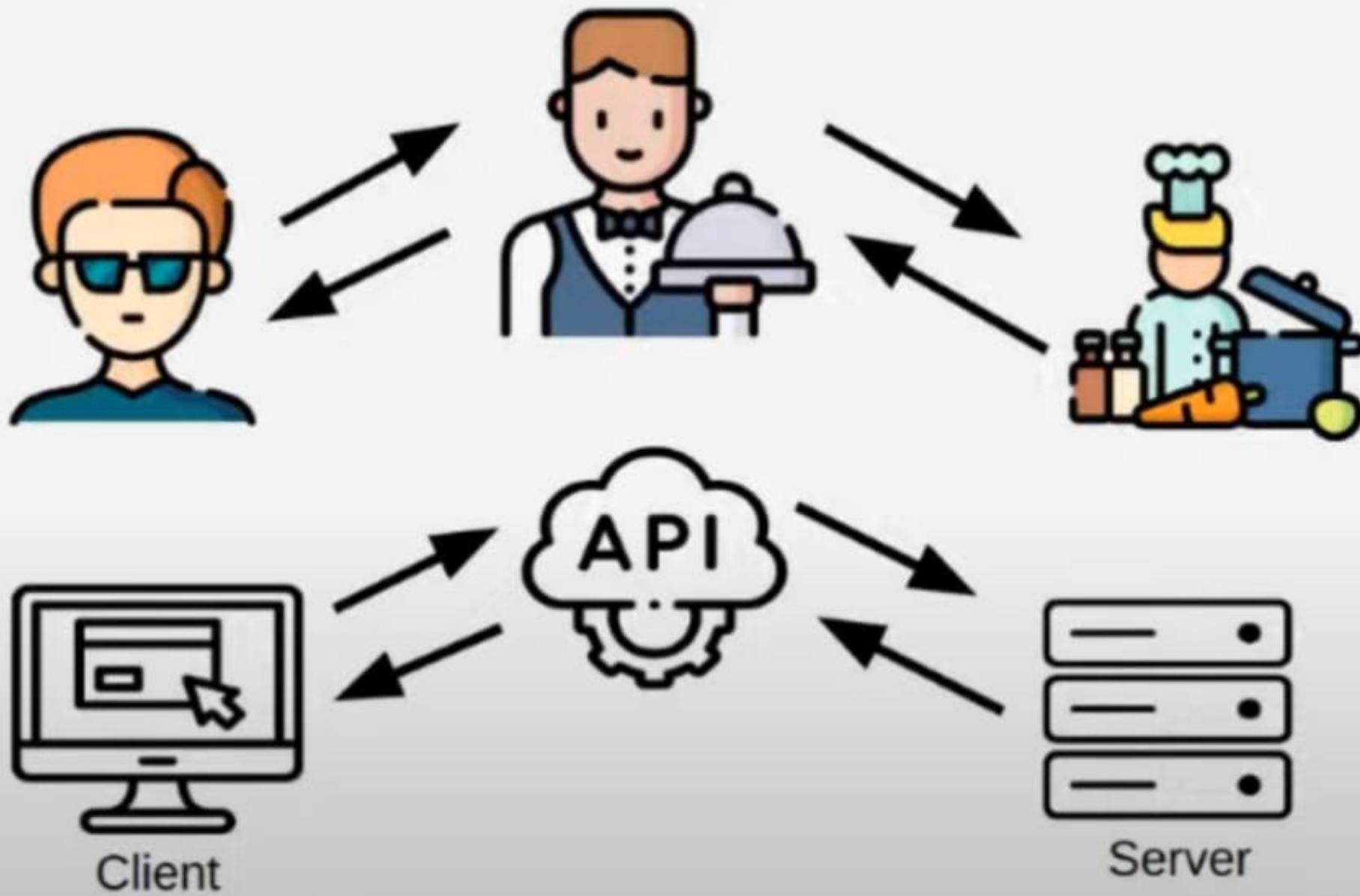
โรงไฟฟ้า ผลิตไฟฟ้า เพื่อให้บริการไฟฟ้า
คำถาม ถ้าไม่มีปลั๊กตัวเมียมามาให้ เราจะใช้ไฟฟ้า อย่างไร ?



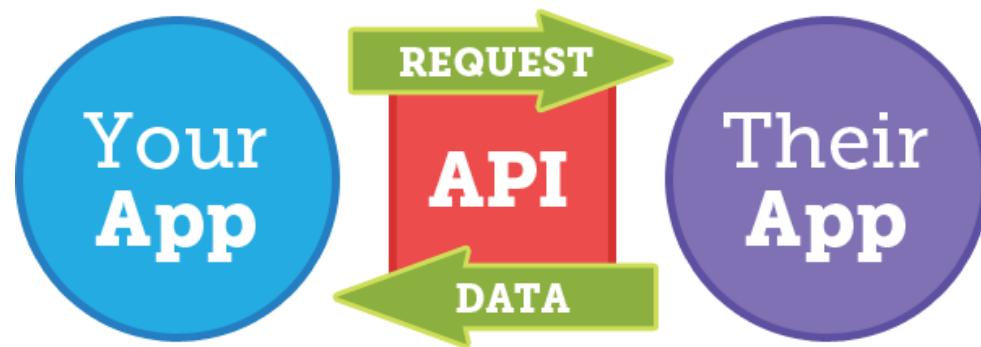
10. Introduction to API

API ຢ່ອມາຈາກ Application Programming Interface





An API allows one piece of software talk to another.



ຕົວຢ່າງ API

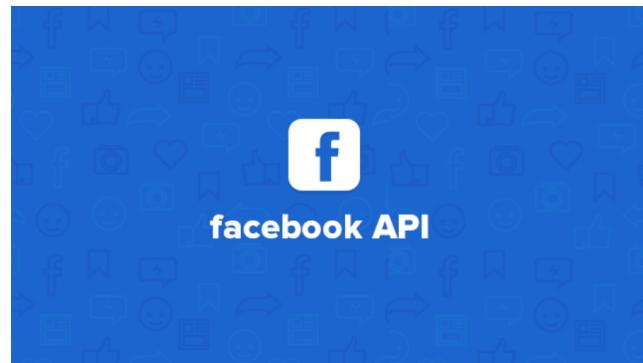
Google Nearby API



Google Maps APIs

A screenshot of the LINE Developers website. At the top, there's a navigation bar with links for LINE Developers, About, News, Products (which is underlined), Documentation, and FAQ. To the right of the navigation is a "Log in to Console" button. Below the navigation, there's a toolbar with various icons. The main content area has a green header with the text "Messaging API". Below the header, it says "Get engaged in interactive conversations with LINE users." and "What is the Messaging API?". It explains that the API lets you develop two-way communication between your service and LINE users. There are two buttons at the bottom: "Start now" and "Documentation". To the right of the text, there's a graphic of two speech bubbles connected by a line, one labeled "Schedule" and the other "LINE", with the text "Today's schedule is..." below them.

YouTube Data API v3



ตัวอย่าง API ให้ข้อมูล อัตราแลกเปลี่ยน

<https://api.exchangeratesapi.io/latest>



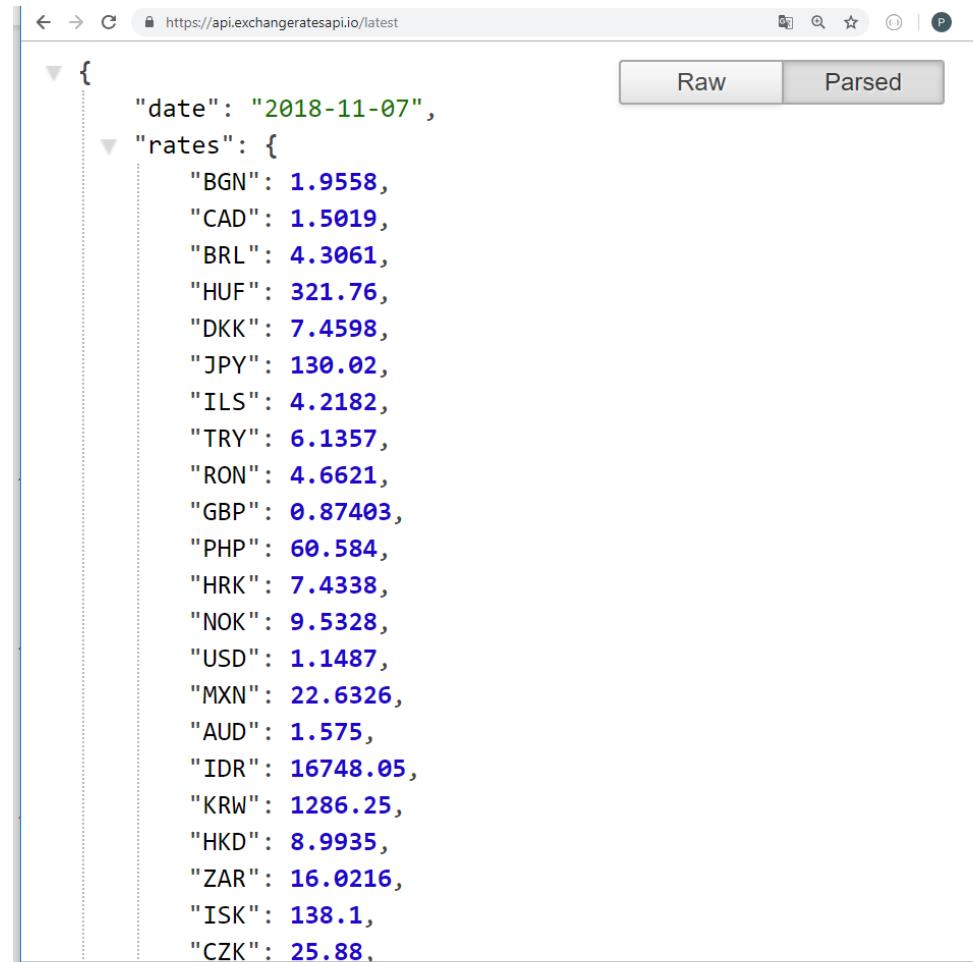
The screenshot shows a browser window with the URL <https://api.exchangeratesapi.io/latest> in the address bar. The page displays a large block of JSON data representing the latest exchange rates. The JSON structure includes a "date" key with the value "2018-11-07", a "rates" object containing rates for various currencies like BGN, CAD, BRL, HUF, DKK, JPY, ILS, TRY, RON, GBP, PHP, HRK, NOK, USD, MXN, AUD, IDR, KRW, HKD, ZAR, ISK, CZK, THB, MYR, NZD, PLN, SEK, RUB, CNY, SGD, CHF, and INR, and a "base" key with the value "EUR".

```
{"date": "2018-11-07", "rates": {"BGN": 1.9558, "CAD": 1.5019, "BRL": 4.3061, "HUF": 321.76, "DKK": 7.4598, "JPY": 130.02, "ILS": 4.2182, "TRY": 6.1357, "RON": 4.6621, "GBP": 0.87403, "PHP": 60.584, "HRK": 7.4338, "NOK": 9.5328, "USD": 1.1487, "MXN": 22.6326, "AUD": 1.575, "IDR": 16748.05, "KRW": 1286.25, "HKD": 8.9935, "ZAR": 16.0216, "ISK": 138.1, "CZK": 25.88, "THB": 37.66, "MYR": 4.7734, "NZD": 1.6954, "PLN": 4.294, "SEK": 10.3313, "RUB": 75.7709, "CNY": 7.9426, "SGD": 1.5729, "CHF": 1.1444, "INR": 83.28}, "base": "EUR"}
```

ตัวอย่าง API ให้ข้อมูล อัตราแลกเปลี่ยน

<https://api.exchangeratesapi.io/latest>

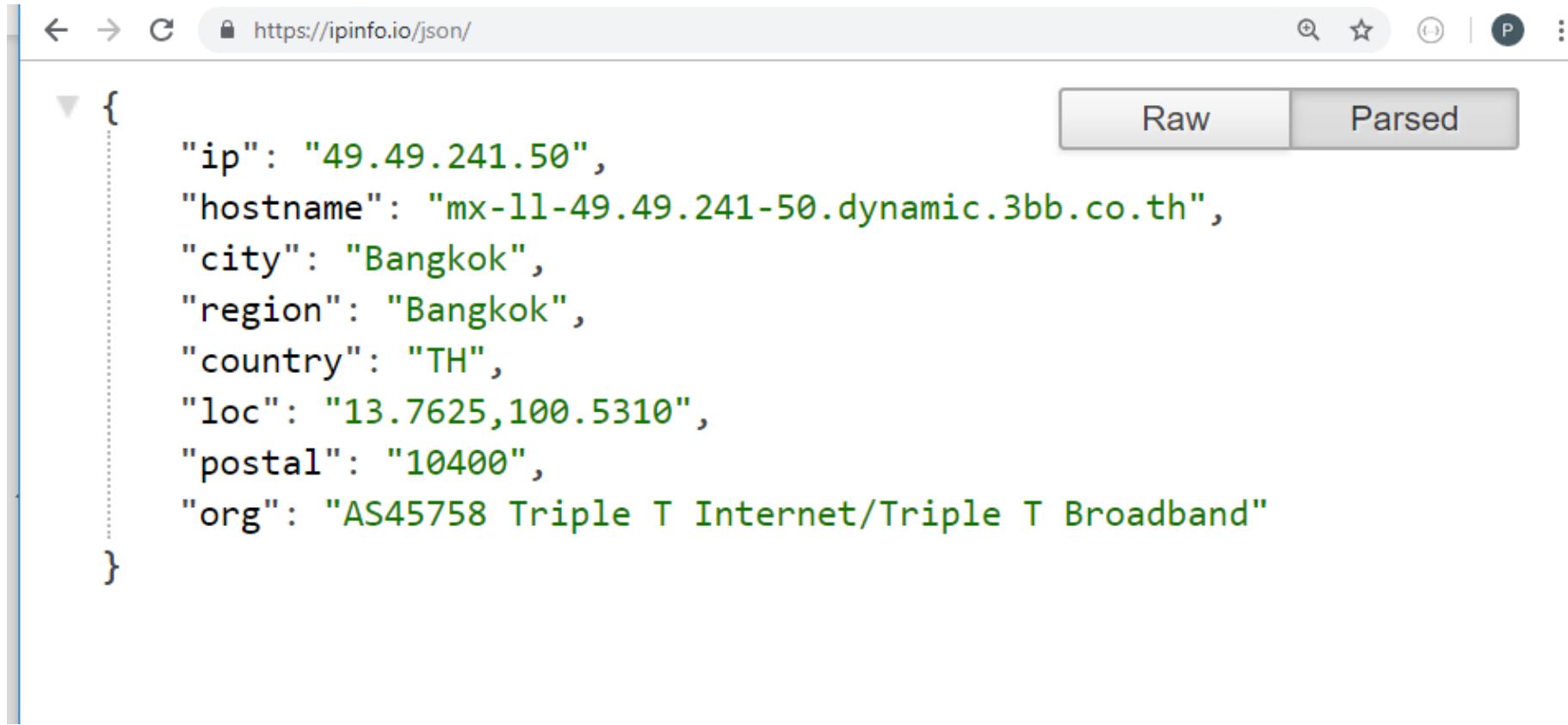
ติดตั้ง plugin **json formatter**



```
{
  "date": "2018-11-07",
  "rates": {
    "BGN": 1.9558,
    "CAD": 1.5019,
    "BRL": 4.3061,
    "HUF": 321.76,
    "DKK": 7.4598,
    "JPY": 130.02,
    "ILS": 4.2182,
    "TRY": 6.1357,
    "RON": 4.6621,
    "GBP": 0.87403,
    "PHP": 60.584,
    "HRK": 7.4338,
    "NOK": 9.5328,
    "USD": 1.1487,
    "MXN": 22.6326,
    "AUD": 1.575,
    "IDR": 16748.05,
    "KRW": 1286.25,
    "HKD": 8.9935,
    "ZAR": 16.0216,
    "ISK": 138.1,
    "CZK": 25.88
  }
}
```

ตัวอย่าง API ให้ข้อมูล IP Address

<https://ipinfo.io/json/>



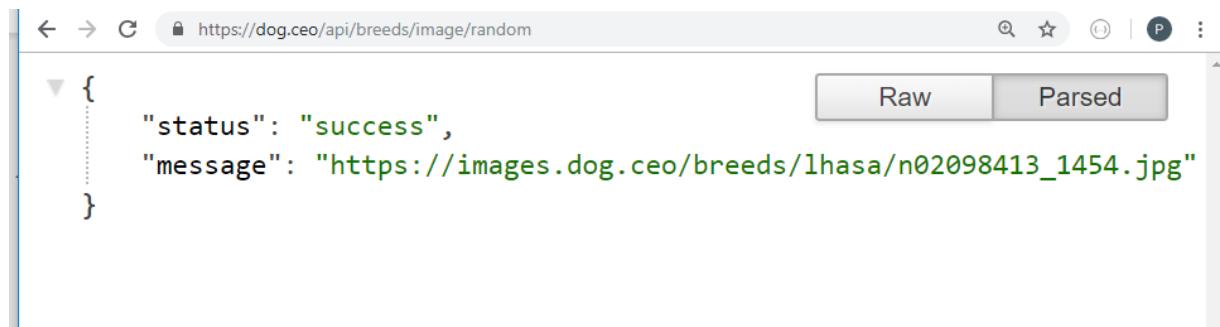
The screenshot shows a browser window with the URL <https://ipinfo.io/json/> in the address bar. The page displays a JSON object representing the location information for the IP address 49.49.241.50. The JSON structure is as follows:

```
{  
  "ip": "49.49.241.50",  
  "hostname": "mx-11-49.49.241-50.dynamic.3bb.co.th",  
  "city": "Bangkok",  
  "region": "Bangkok",  
  "country": "TH",  
  "loc": "13.7625,100.5310",  
  "postal": "10400",  
  "org": "AS45758 Triple T Internet/Triple T Broadband"  
}
```

The "Parsed" tab is selected in the JSON viewer, which highlights the text in green. The browser interface includes standard navigation buttons (back, forward, search) and a toolbar.

ตัวอย่าง API ให้ข้อมูล Random ภาพสุนัข

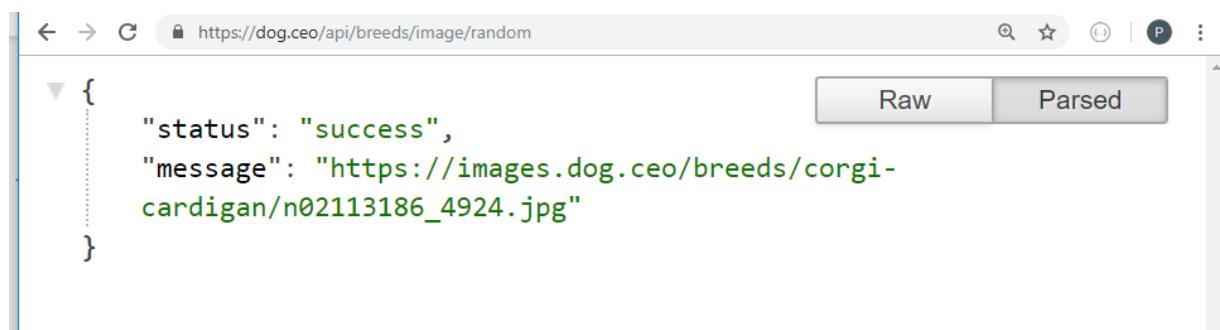
<https://dog.ceo/api/breeds/image/random>



A screenshot of a browser developer tools Network tab. The URL in the address bar is <https://dog.ceo/api/breeds/image/random>. The response is a JSON object with the following structure:

```
{ "status": "success", "message": "https://images.dog.ceo/breeds/lhasa/n02098413_1454.jpg" }
```

The "status" key has a value of "success". The "message" key contains a URL to a dog image: https://images.dog.ceo/breeds/lhasa/n02098413_1454.jpg.



A screenshot of a browser developer tools Network tab. The URL in the address bar is <https://dog.ceo/api/breeds/image/random>. The response is a JSON object with the following structure:

```
{ "status": "success", "message": "https://images.dog.ceo/breeds/corgi-cardigan/n02113186_4924.jpg" }
```

The "status" key has a value of "success". The "message" key contains a URL to a dog image: https://images.dog.ceo/breeds/corgi-cardigan/n02113186_4924.jpg.

ตัวอย่าง API ให้ข้อมูล สภาพอากาศ

<https://www.metaweather.com/api/location/1225448/>

The screenshot shows a browser window with the URL <https://www.metaweather.com/api/location/1225448/>. The page content is a JSON object representing weather data for a specific location. The JSON structure includes an array of consolidated weather data, with one item shown in detail. The detailed item contains various weather parameters such as id, weather state name, wind direction compass, created date, applicable date, minimum temperature, maximum temperature, the current temperature, wind speed, wind direction, air pressure, humidity, visibility, and predictability.

```
{  
  "consolidated_weather": [  
    {  
      "id": 6331049638363136,  
      "weather_state_name": "Heavy Rain",  
      "weather_state_abbr": "hr",  
      "wind_direction_compass": "NE",  
      "created": "2018-11-07T15:36:36.023308Z",  
      "applicable_date": "2018-11-07",  
      "min_temp": 26.06,  
      "max_temp": 31.845,  
      "the_temp": 32.14,  
      "wind_speed": 4.6181500216734275,  
      "wind_direction": 45.53804553582663,  
      "air_pressure": 1016.23,  
      "humidity": 59,  
      "visibility": 11.5789414817466,  
      "predictability": 77  
    }  
  ]  
}
```

ตัวอย่าง API อื่น ๆ สามารถดูได้จาก

<https://github.com/toddmotto/public-apis> หรือค้นจาก google

การเขียน html ดึงค่าข้อมูล ผ่าน API

<https://api.exchangeratesapi.io/latest>

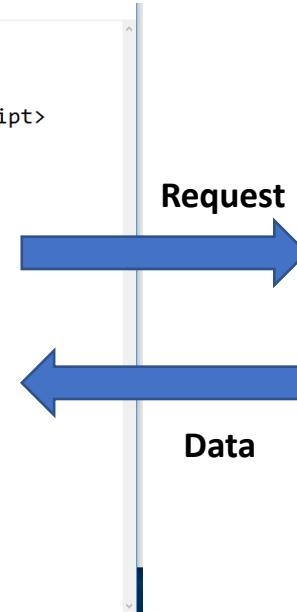
ไฟล์ index.html

```
File Edit Format View Help
<html>
<head>
    <title>Welcome to JSON</title>
    <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js"></script>
</head>
<body>

    <script type="text/javascript">
        $.getJSON('https://api.exchangeratesapi.io/latest', function(response) {
            document.getElementById("data").innerHTML = response.rates.THB
            document.getElementById("data1").innerHTML = response.date
        });
    </script>

    <div style = "color:red" id="data" > </div>
    <div style = "color:blue" id="data1"> </div>

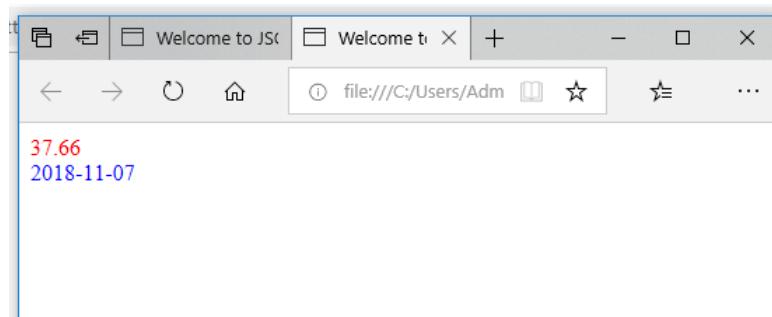
</body>
</html>
```



เปิดไฟล์ ผ่าน Web browser



แสดงผล

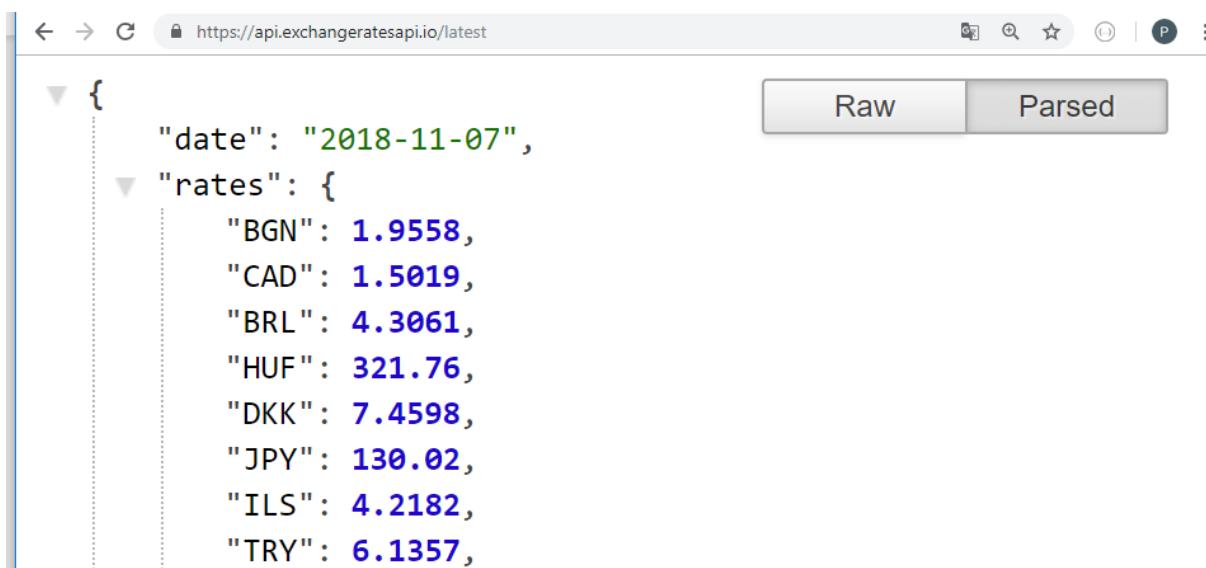


```
"PHP": 60.584,
"HRK": 7.4338,
"NOK": 9.5328,
"USD": 1.1487,
"MXN": 22.6326,
"AUD": 1.575,
"IDR": 16748.05,
"KRW": 1286.25,
"HKD": 8.9935,
"ZAR": 16.0216,
"ISK": 138.1,
"CZK": 25.88,
"THB": 37.66,
"MYR": 4.7734,
"NZD": 1.6954,
"PLN": 4.294,
"SEK": 10.3313,
"RUB": 75.7709,
"CNY": 7.9426,
"SGD": 1.5729,
"CHF": 1.1444,
"INR": 83.28
},
"base": "EUR"
```

9.1 REST API

REST หรือ Representational State Transfer เป็นวิธีในการสร้าง Web Service รูปแบบหนึ่งที่อาศัย HTTP Method (GET, POST, PUT, DELETE) ในการทำงาน และส่งผลกลับมาในรูปแบบของ JSON หรือ XML

เช่น <https://api.exchangeratesapi.io/latest>



REST API

ตัวอย่าง API Document

workshop\$Order Entity CRUD operations

GET `/entities/workshop$Order` Gets a list of entities: workshop\$Order

POST `/entities/workshop$Order` Creates new entity: workshop\$Order

GET `/entities/workshop$Order/{entityId}` Gets a single entity by identifier: workshop\$Order

PUT `/entities/workshop$Order/{entityId}` Updates the entity: workshop\$Order

DELETE `/entities/workshop$Order/{entityId}` Deletes the entity: workshop\$Order

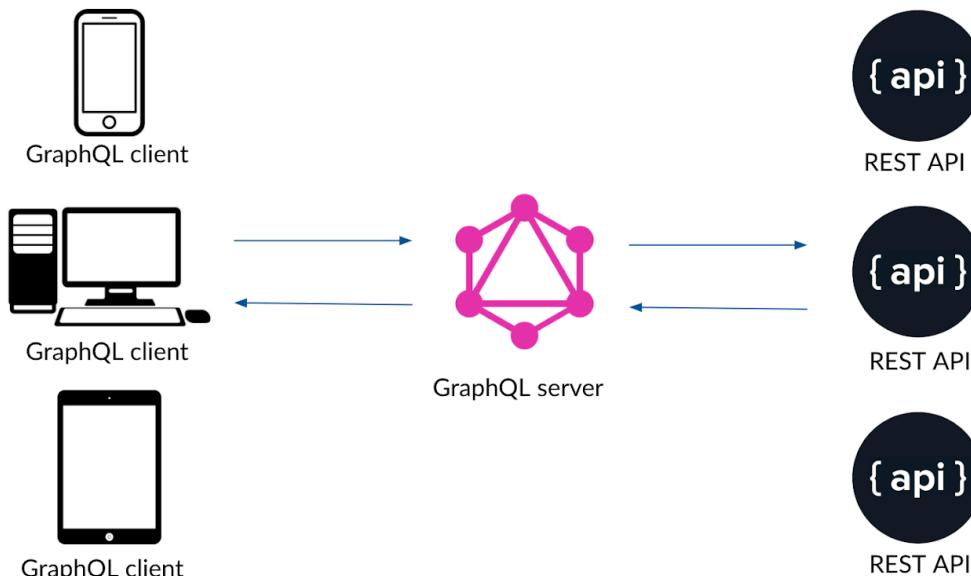
GET `/entities/workshop$Order/search` Find entities by filter conditions: workshop\$Order

POST `/entities/workshop$Order/search` Find entities by filter conditions: workshop\$Order

11. GraphQL

- เนื่องจาก การดึงค่าแบบ Rest API บางครั้งข้อมูลมากเกินความจำเป็น บางครั้งต้องดึงหลายครั้ง เพื่อนำข้อมูลมาประมวลผล หรือเขามาประกอบกัน จึงทำให้เกิดแนวคิด ของ GraphQL

“ *GraphQL is a query language for APIs and a runtime for fulfilling those queries with your existing data stores.* ”



GraphQL

