



Website Development

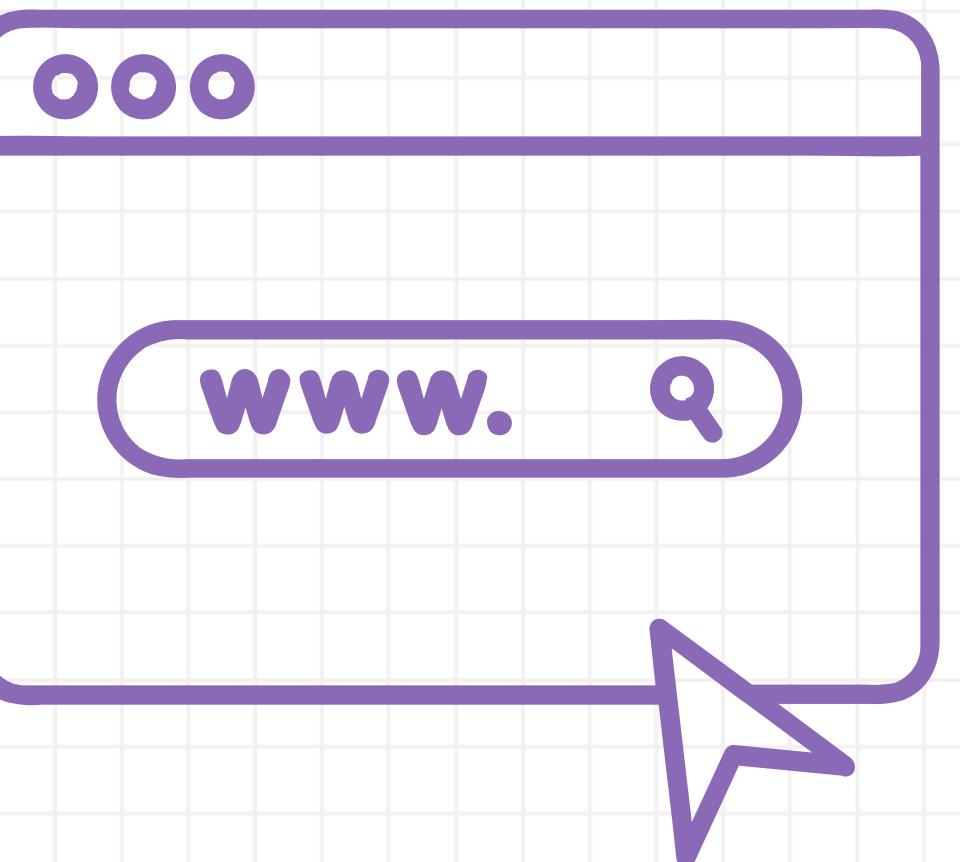
ClickCamp 15

Mahidol University, Computer Engineering

Introduction

Website

- เว็บไซต์คือแหล่งข้อมูลออนไลน์ที่ผู้คนเข้าถึงผ่านอินเทอร์เน็ต เพื่อดูข้อมูล เนื้อหา สินค้า หรือ บริการต่าง ๆ ผ่านหน้าเว็บที่เชื่อมต่อกัน



Frontend vs. Backend



Frontend Development

- เป็นส่วนที่เรามองเห็นบนหน้าเว็บไซต์ และ ผู้ใช้สามารถ interact ได้

Languages

- HTML
- CSS
- Javascript



Backend Development

- เป็นส่วนหลังบ้านที่ผู้ใช้มองไม่เห็น โดยจะจัดการข้อมูล การส่งข้อมูล รับข้อมูล รวมถึงการจัดฐานข้อมูล

Languages

- Python
- C#
- Javascript

Frontend Development

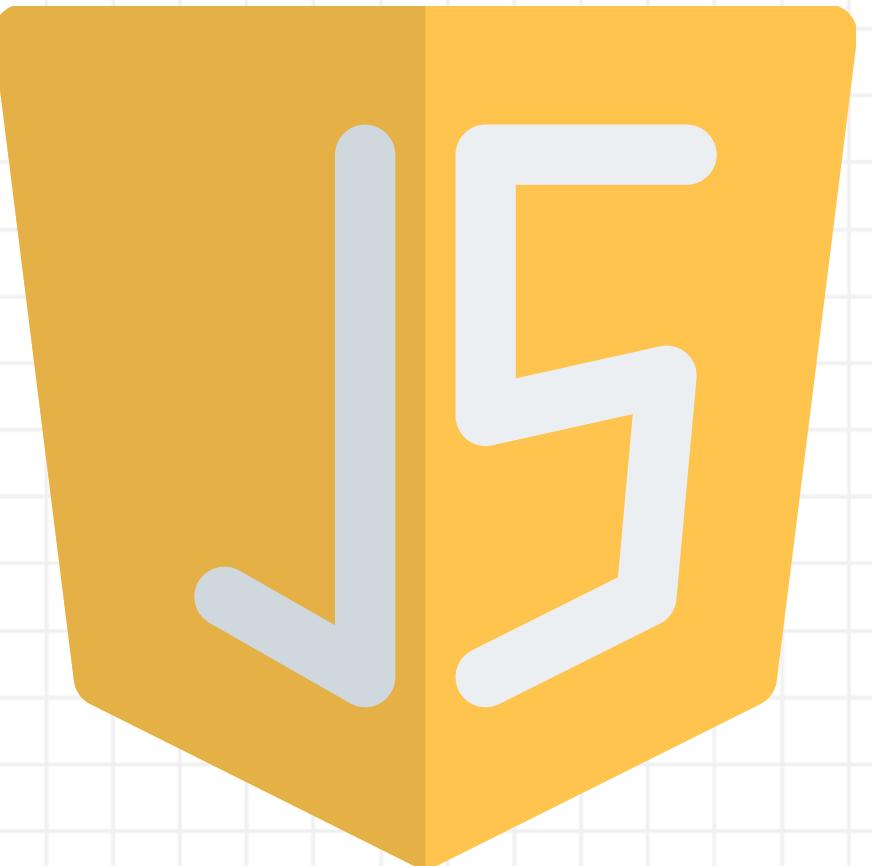
Common Technologies



HTML5



CSS3



Javascript

Course curriculum

Introduction

HTML

- What's HTML
- Tag structure
- Website structure
- Basic to Advance tags

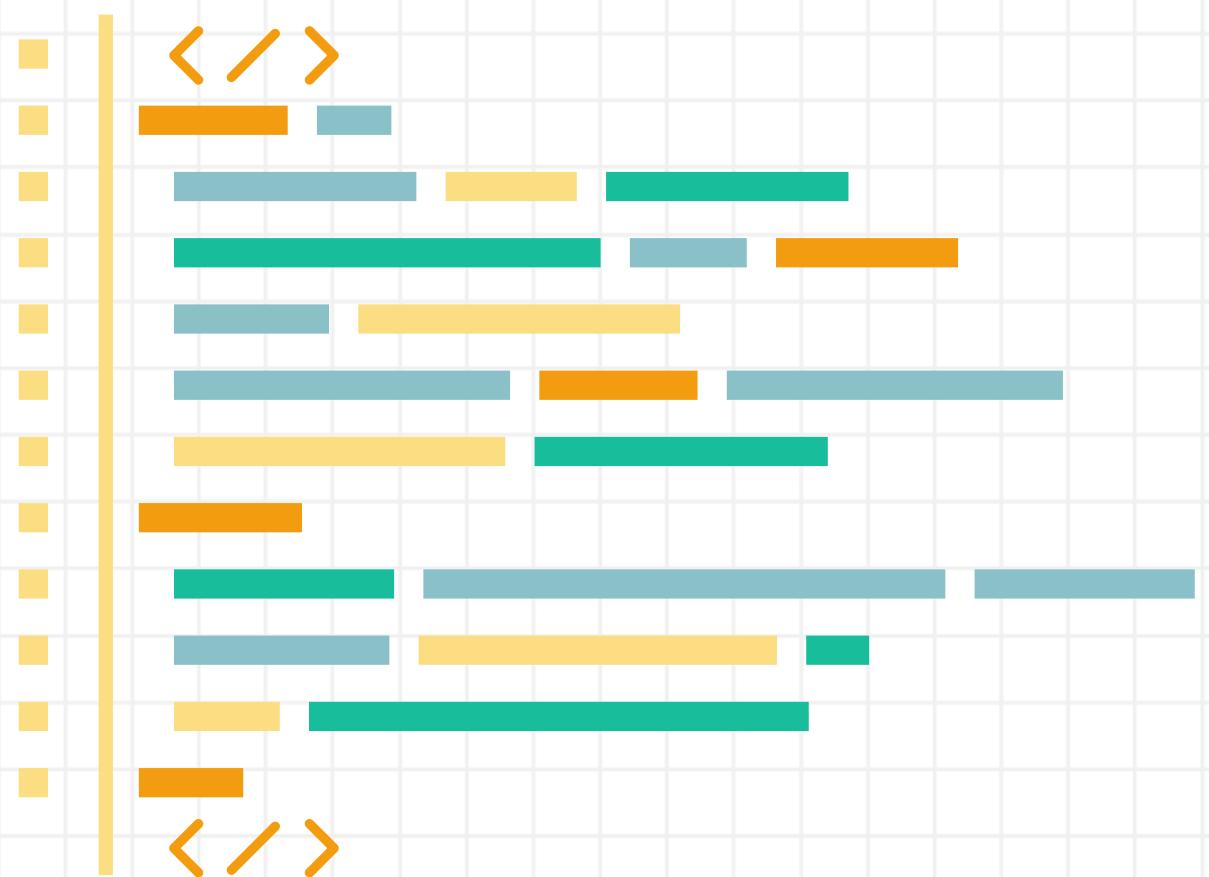
CSS

- What's CSS
- How to add CSS
- Basic properties
- CSS selector
- Box Model
- Display property (Flexbox)
- **Project#1**

JavaScript

- What's JavaScript
- How to add JavaScript
- Programing basic
- DOM
- **Project#2**
- **Project#3**

HTML5



HTML เป็นภาษา markup

HTML ไม่ใช่ภาษาโปรแกรม

The Story of HTML

World-Wide Web

Tim Berners-Lee, Robert Cailliau
C.E.R.N.
CH - 1211 Genève 23
timbl@info.cern.ch, cailliau@cernext.cern.ch

Abstract

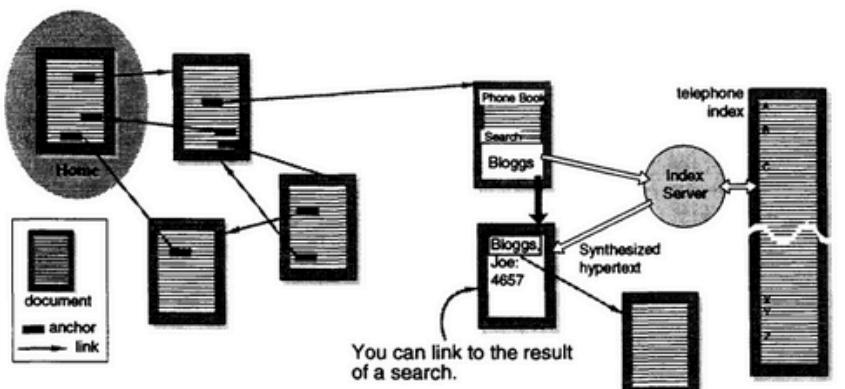
The W3 project merges networked information retrieval and hypertext to make an easy but powerful global information system. It aims to allow information sharing within internationally dispersed groups of users, and the creation and dissemination of information by support groups. W3's ability to provide implementation-independent access to data and documentation is ideal for a large HEP collaboration. W3 now defines the state of the art in networked information retrieval, for user support, resource discovery and collaborative work. W3 originated at CERN and is in use at CERN, FNAL, NIKHEF, SLAC and other laboratories. This paper gives a brief overview and reports the current status of the project.

Introduction

The World-Wide Web (W3) project allows access to the universe of online information using two simple user interface operations. It operates without regard to where information is, how it is stored, or what system is used to manage it. Previous papers give general [1] and technical [2] overviews which will not be repeated here. This paper reviews the basic operation of the system, and reports the status of W3 software and information.

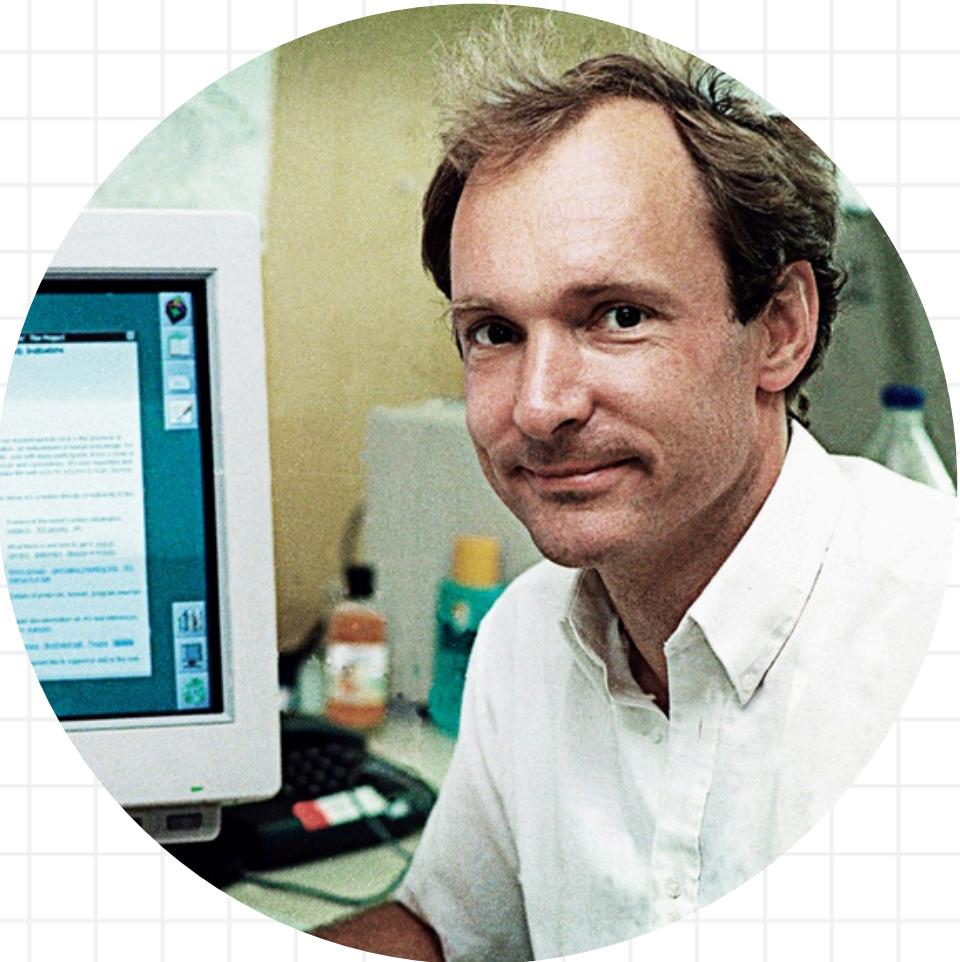
Operation

The W3 world view is of *documents* referring to each other by *links*. For its likeness to a spider's construction, this world is called the *Web*. This simple view is known as the *hypertext* paradigm. The reader sees on the screen a document with sensitive parts of text representing the links. A link is followed by mere pointing and clicking (or typing reference numbers if a mouse is not available).



The diagram illustrates the basic hypertext model. It shows a network of documents represented as rectangles. One document is labeled "Home". Another document contains the text "Phone Book". A third document contains the text "Bloggs". A fourth document contains the text "Bloggs Joe: 4657". An "Index Server" is shown as a circle connected to the "Bloggs" document. A "telephone index" is shown as a vertical rectangle with letters A through Z. Arrows indicate links between documents. A legend at the bottom left shows a document icon with an arrow pointing to the right, labeled "anchor" and "link". A callout bubble says "You can link to the result of a search."

Fig 1. The basic hypertext model is enhanced by searches.



Tim Berners-Lee

<https://cds.cern.ch/record/245440/files/p69.pdf>

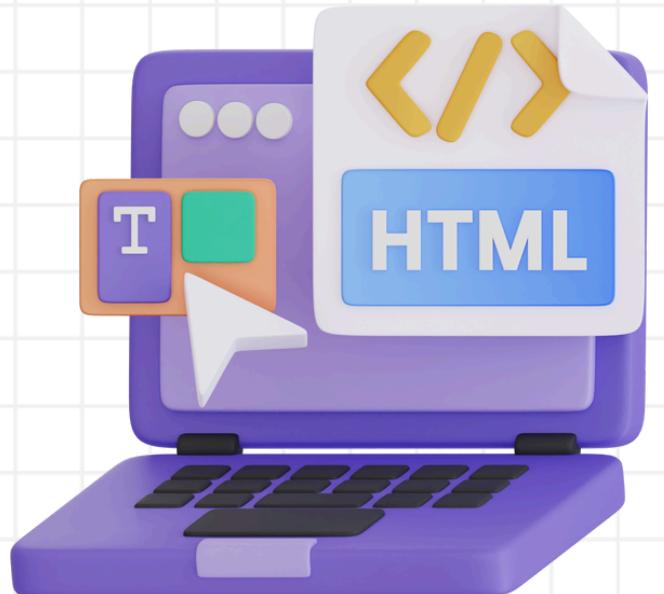
<https://contentsnare.com/html-history/>

HTML คืออะไร

- เป็น HyperText Markup Language สำหรับใช้ในการสร้าง website

HTML มีหน้าที่อะไร

- มีหน้าที่เป็นโครงสร้างของ website ที่จะแสดงข้อมูลต่าง ๆ ของ website สามารถใช้งานร่วมกับภาษาอื่น ๆ ได้



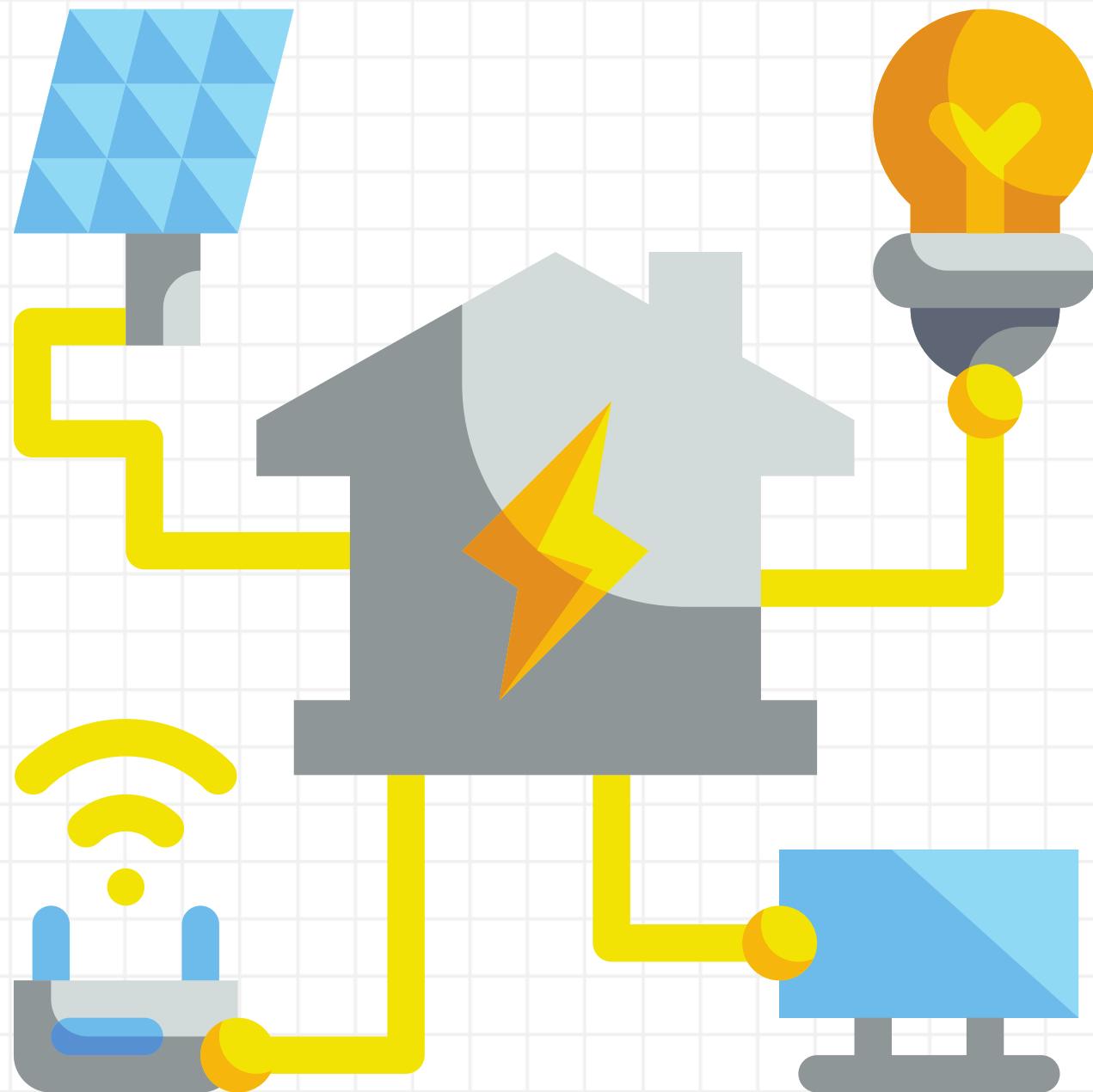
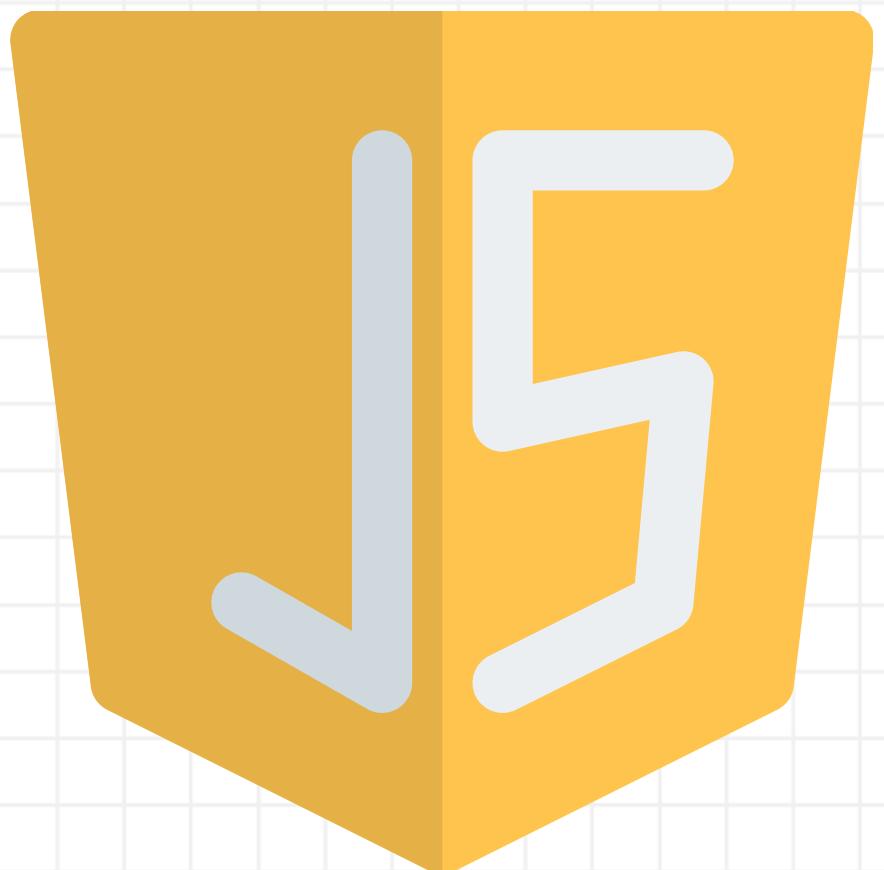
เปรียบเทียบ “โครงสร้าง”



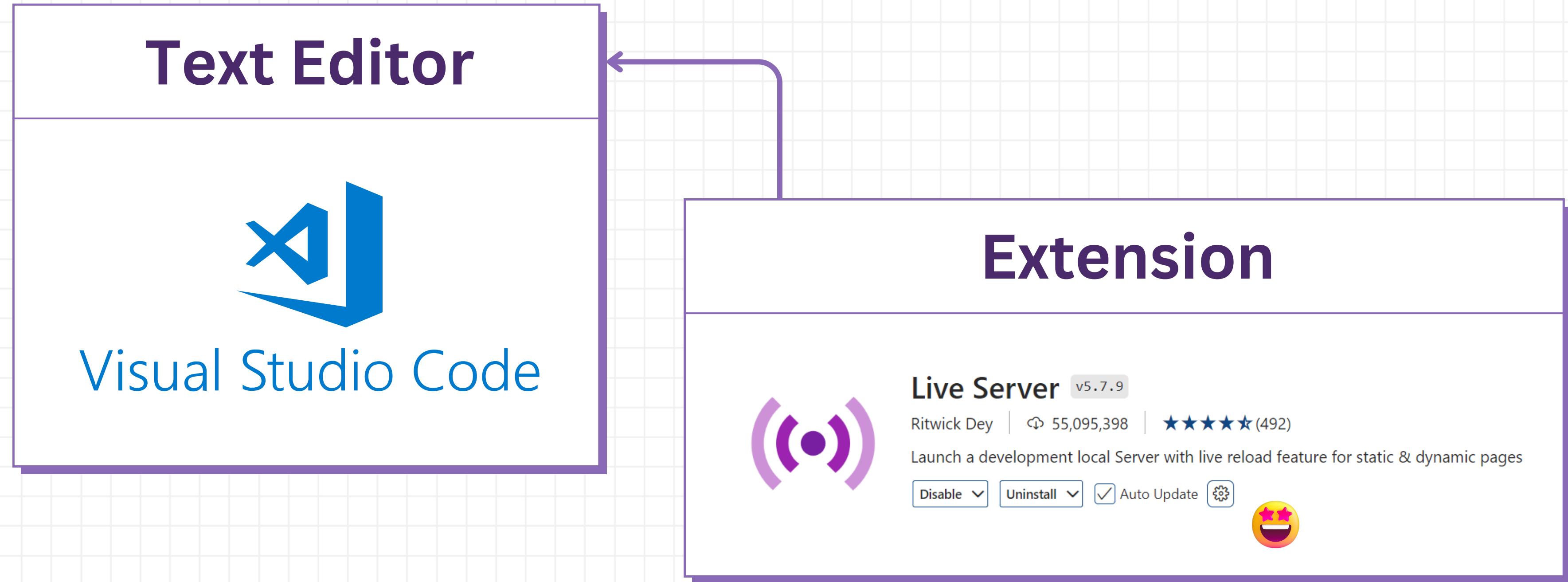
เปรียบเทียบ “การตกแต่งบ้าน”



เปรียบเทียบ “ระบบภายในบ้าน”



Tools



HTML Page Structure

```
<html>

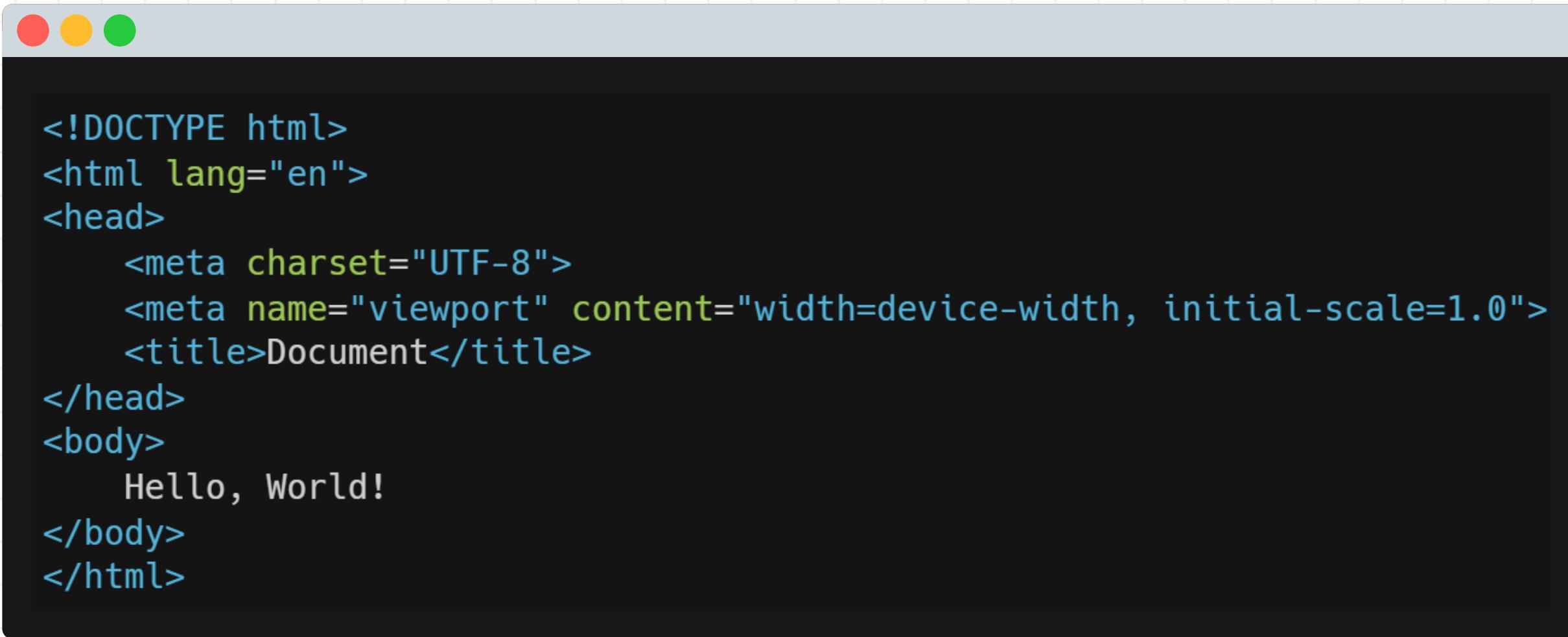
  <head>
    <title>ClickCamp15</title>
  </head>

  <body>
    <h1>This is ClickCamp15</h1>
    <p>it is a camp</p>
    <a>Link to instagram</a>
  </body>

</html>
```

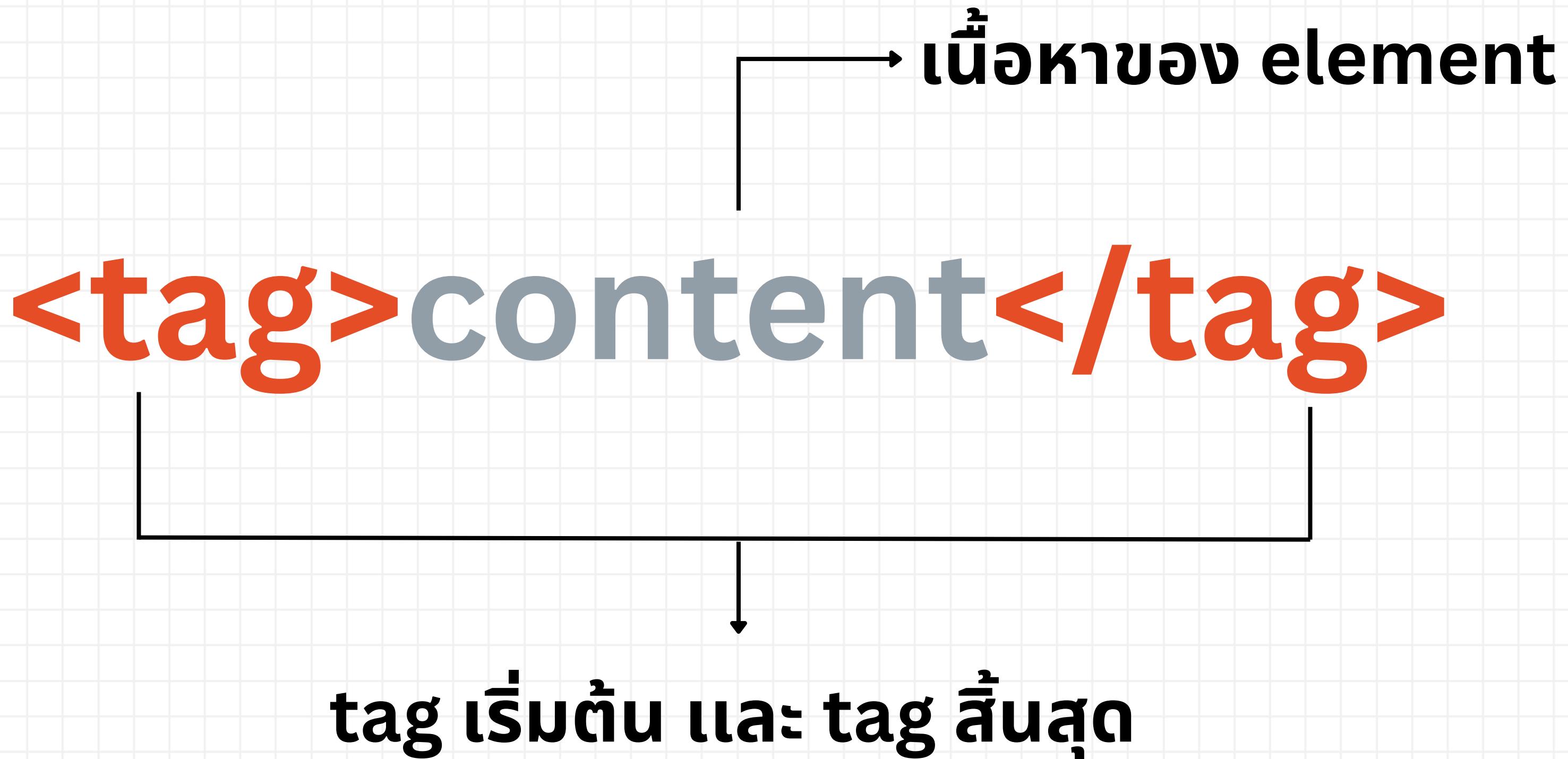
Hello, World!

First Website



```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    Hello, World!
</body>
</html>
```

Element



Element

The diagram illustrates the structure of an HTML element. It features a large rectangular frame with a blue border. Inside, at the top right, is the text "เนื้อหาของ element" with an arrow pointing towards the frame. In the center is the text "<tag>content</tag>". This central text is enclosed in a horizontal rectangle with black borders. A vertical line extends downwards from the bottom center of this rectangle to the text "tag เริ่มต้น และ tag สิ้นสุด" located below it.

เนื้อหาของ element

<tag>content</tag>

tag เริ่มต้น และ tag สิ้นสุด

Basic Tags

```
<h1>content</h1>
<h2>content</h2>
<h3>content</h3>
:
<h6>content</h6>
```



Basic Tags

Hello ClickCamp15

A simple paragraph.

Paragraph that with a line break.

Text after line break

This is the end of p tag.

This is what a tag. can do hyperlink!

Have a great day!

tags

- **<hr>** <!-- เลื่อนตามแนวโน้ม -->
- **
** <!-- เว้นบรรทัด -->
- **<p>** <!-- สร้าง paragraph -->
- **<a>** <!-- สร้าง hyperlink -->

Attribute

<tag attribute="value">content</tag>



กำหนดค่าให้กับ element

Basic Tags

Hello ClickCamp15

A simple paragraph.

Paragraph that with a line break.

Text after line break

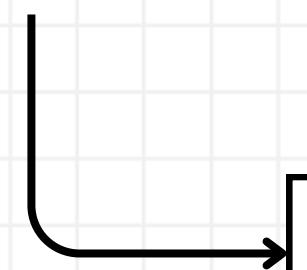
This is the end of p tag.

This is what a tag. can do hyperlink!

Have a great day!

tags

- **<hr>** <!-- เลื่อนตามแนวโน้ม -->
- **
** <!-- เว้นบรรทัด -->
- **<p>** <!-- สร้าง paragraph -->
- **<a>** <!-- สร้าง hyperlink -->



```
<a href="link">content</a>
```

Attribute

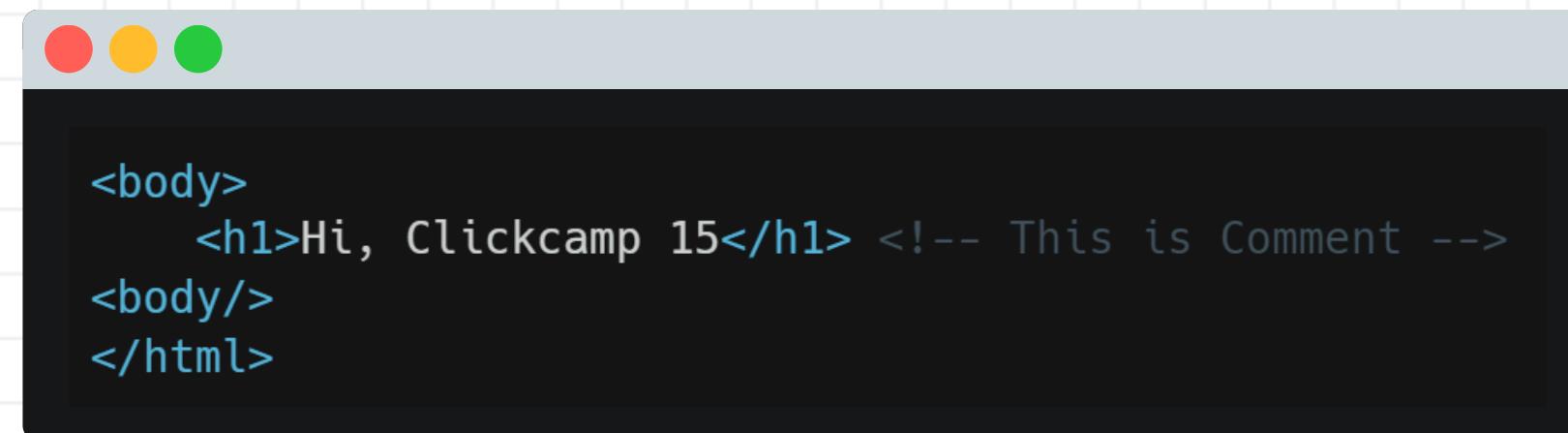
`<p style="color: red;">content</p>`

`<p style="color: red; font-size:30px;">content</p>`

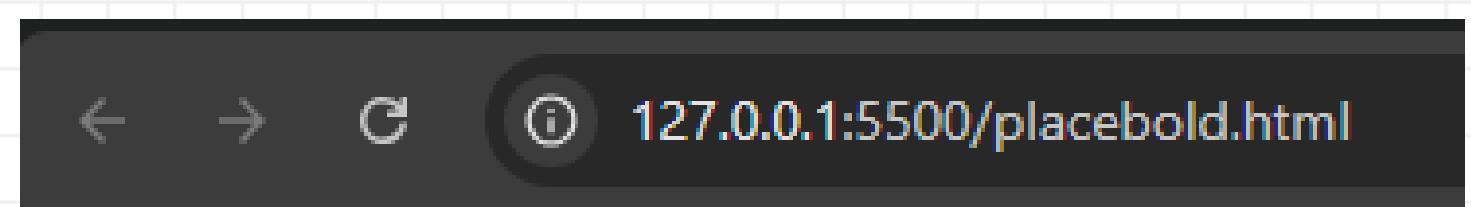
Comments

- Comments คือการที่เราเขียนอธิบายภายใน HTML ของเราเพื่อปั่งบอกระไรบ้างอย่างโดยไม่ให้ส่งที่เขียนแสดงผลผ่านหน้าจอ

<!-- Comment/คอมเม้นท์ -->



```
<body>
  <h1>Hi, Clickcamp 15</h1> <!-- This is Comment -->
</body>
</html>
```



Hi, Clickcamp 15

Generic Container



tags

- **<div> <!-- จับกลุ่ม element -->**

3 Methods adding CSS

Inline CSS

- style = “color : red;”

Internal CSS

- <style></style>

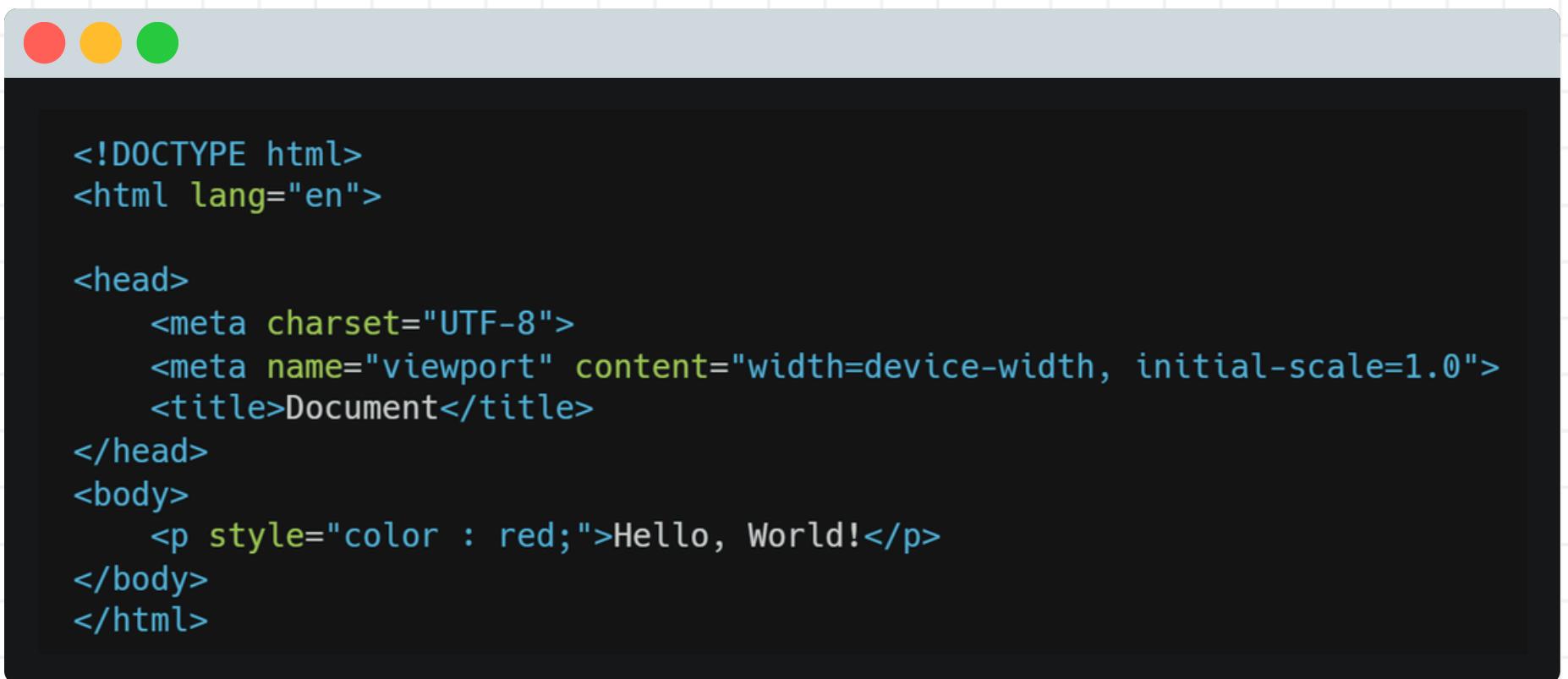
External CSS

- ใช้ไฟล์ .css



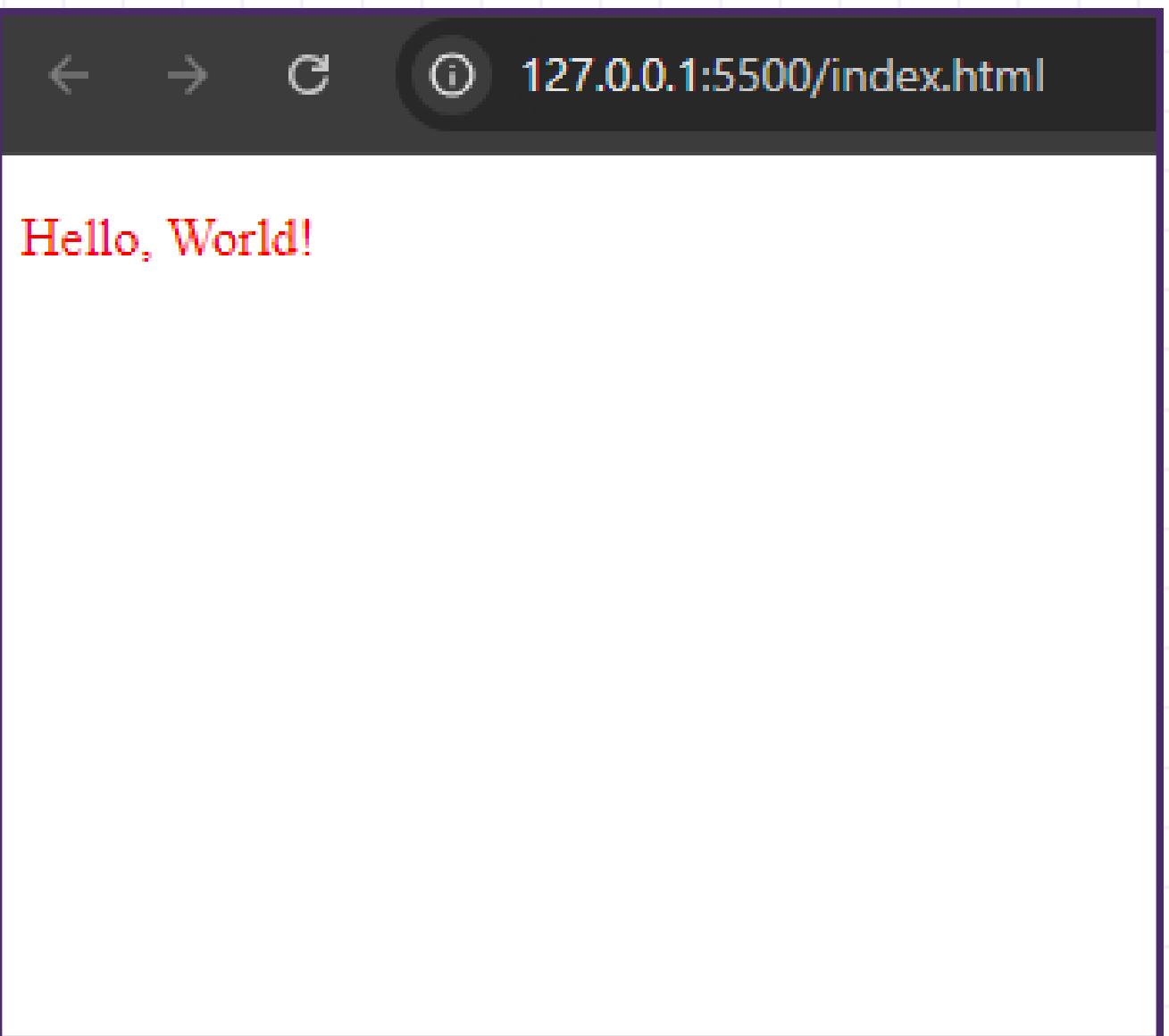
3 Methods adding CSS

Inline CSS



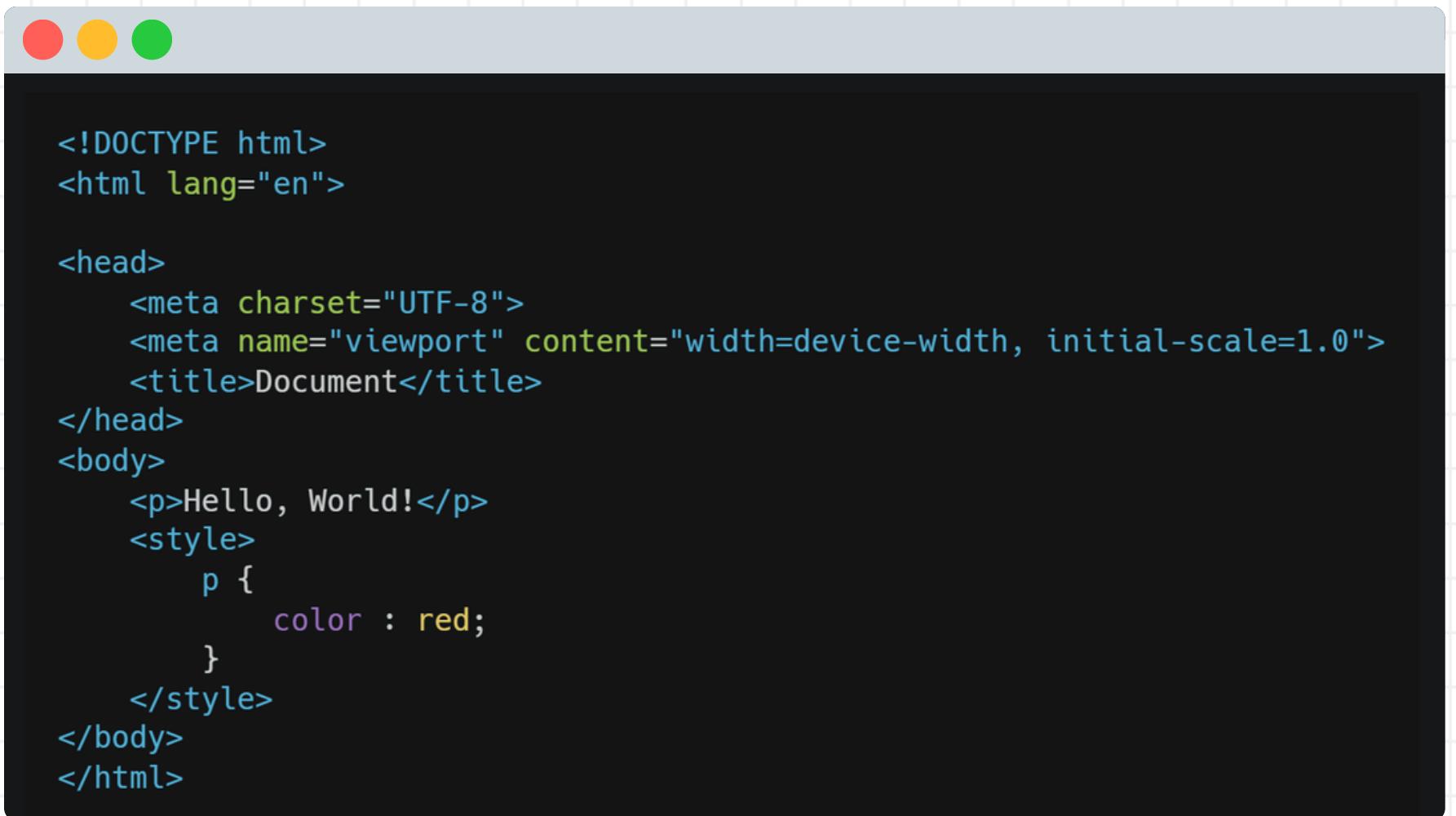
```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    <p style="color : red;">Hello, World!</p>
</body>
</html>
```



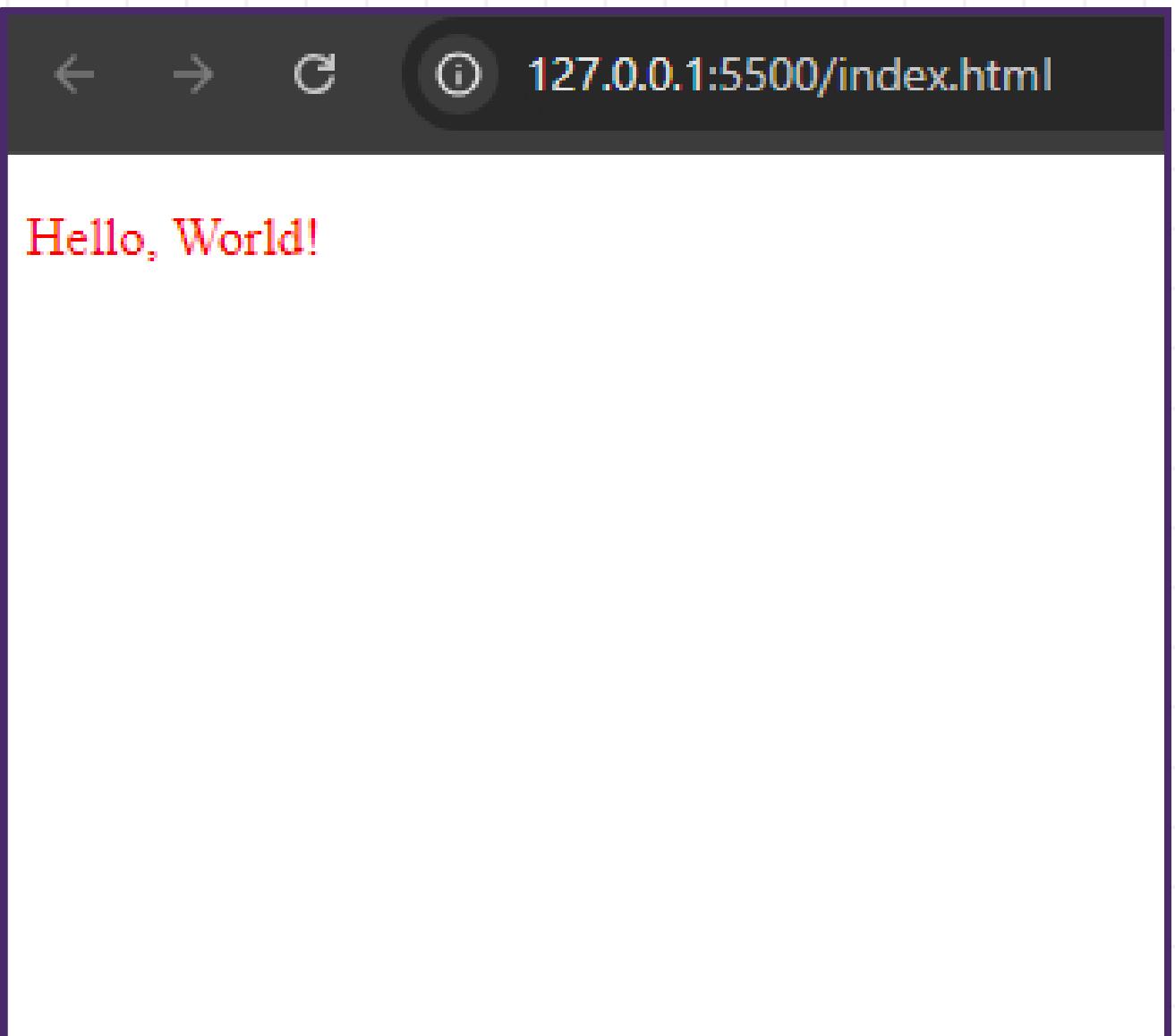
3 Methods adding CSS

Internal CSS



```
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Document</title>
</head>
<body>
    <p>Hello, World!</p>
    <style>
        p {
            color : red;
        }
    </style>
</body>
</html>
```



3 Methods adding CSS

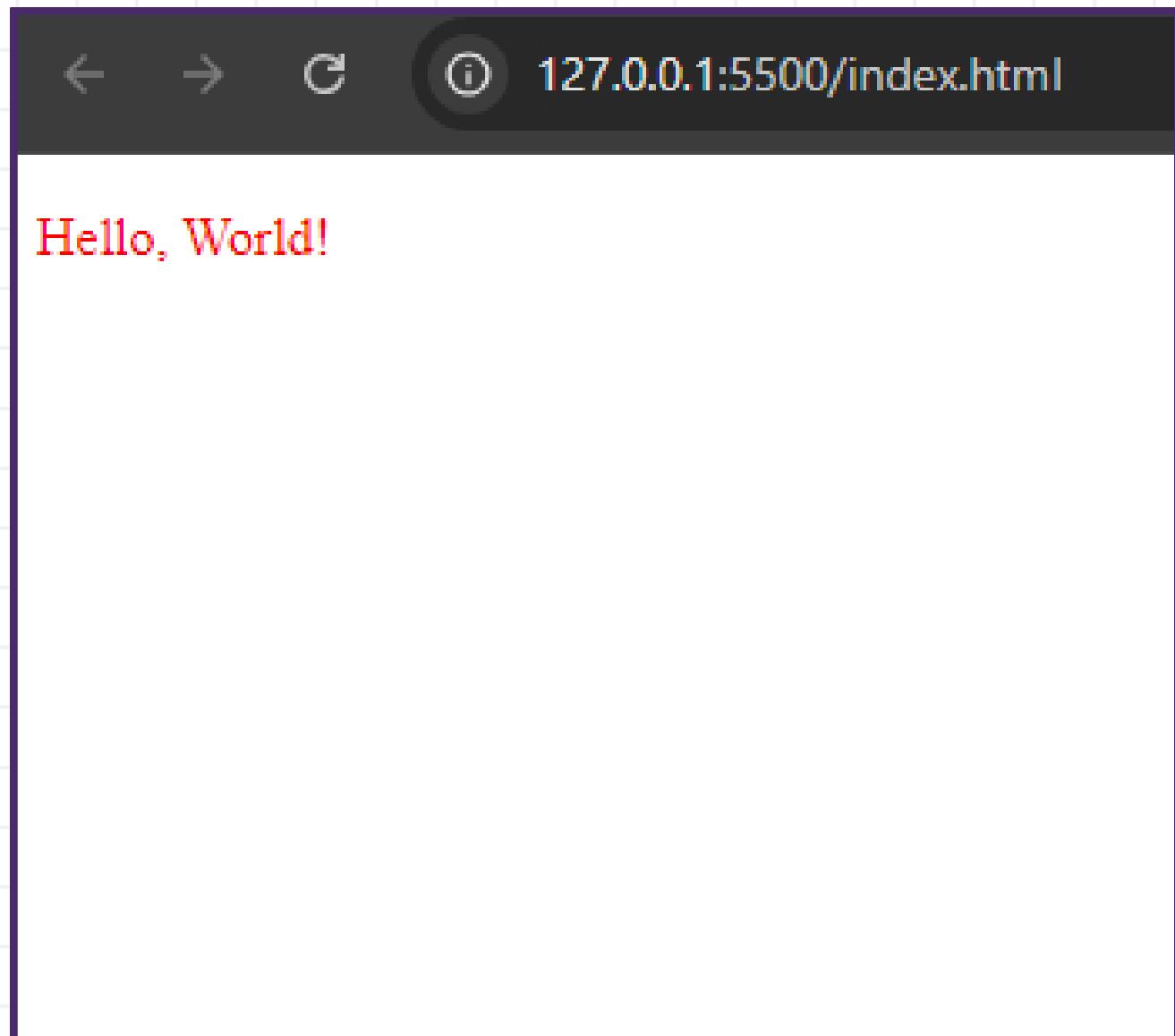
External CSS

index.html

```
<head>
  <link rel="stylesheet" href="style.css">
</head/>
```

style.css

```
p {
  color: red;
}
```



Basic Selector

Element Selector

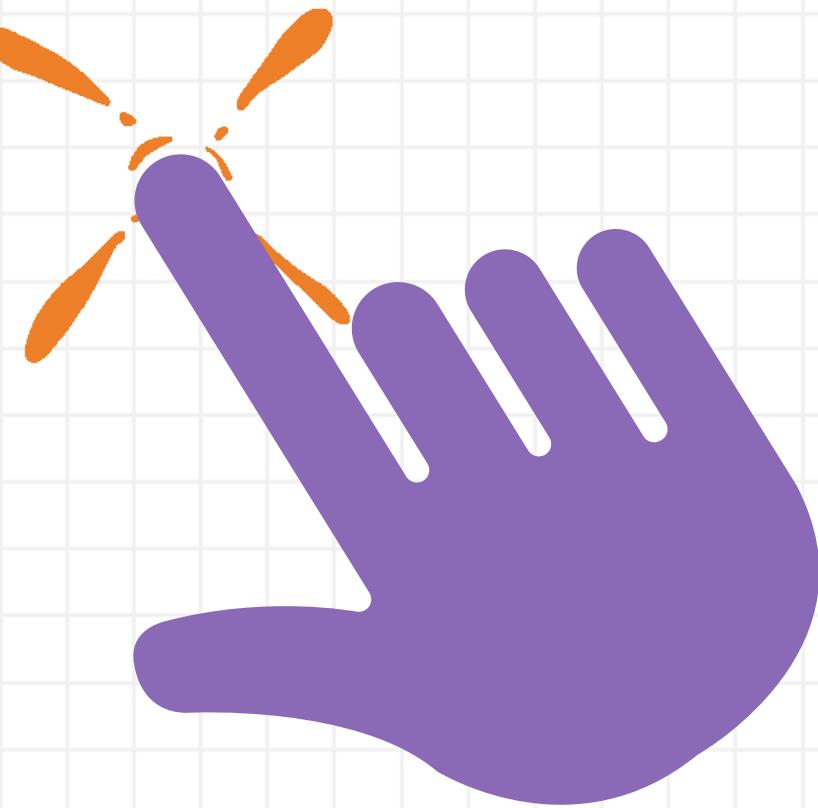
- p, div, h1 - h6

Class Selector

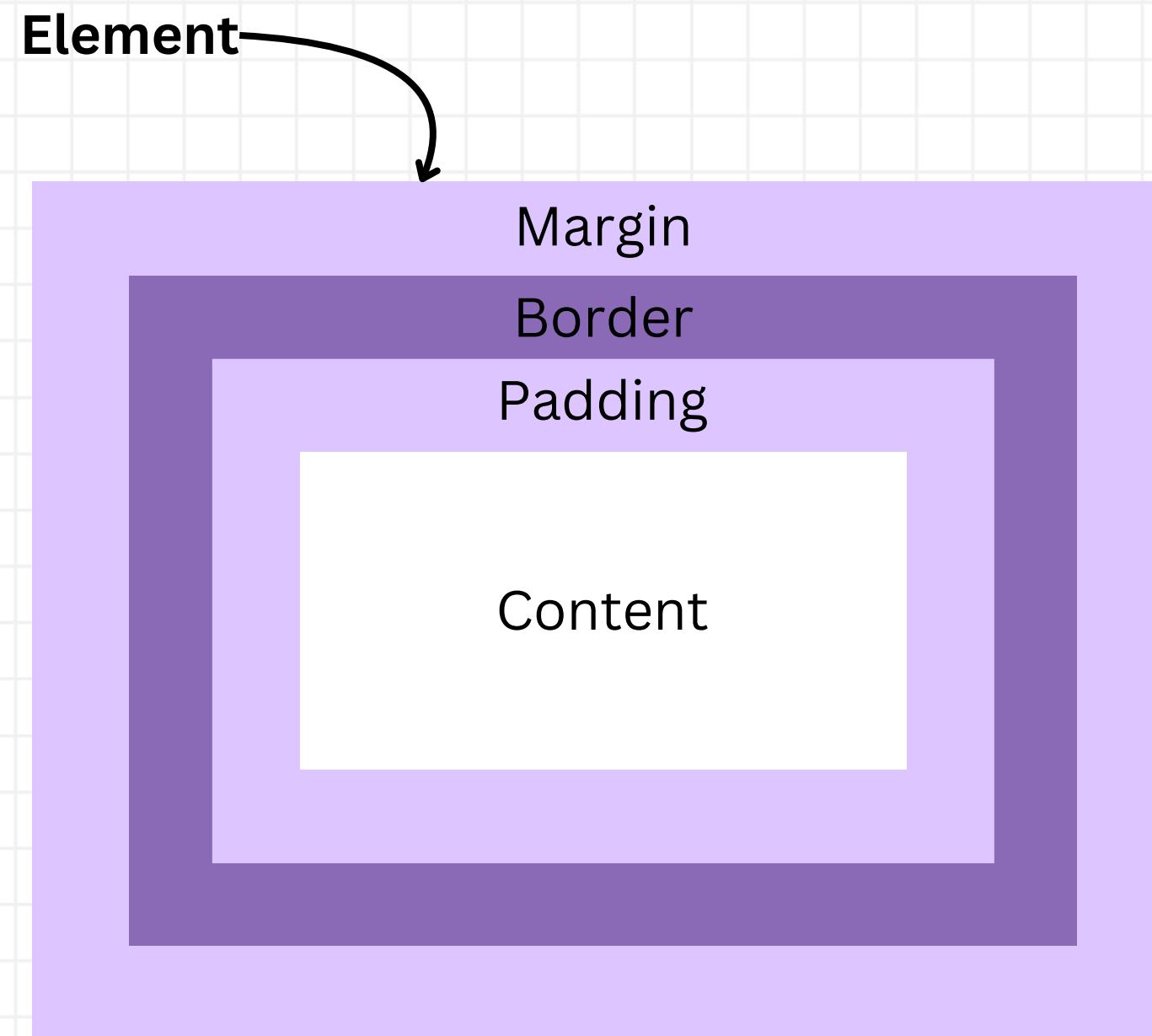
- .class

ID Selector

- #id

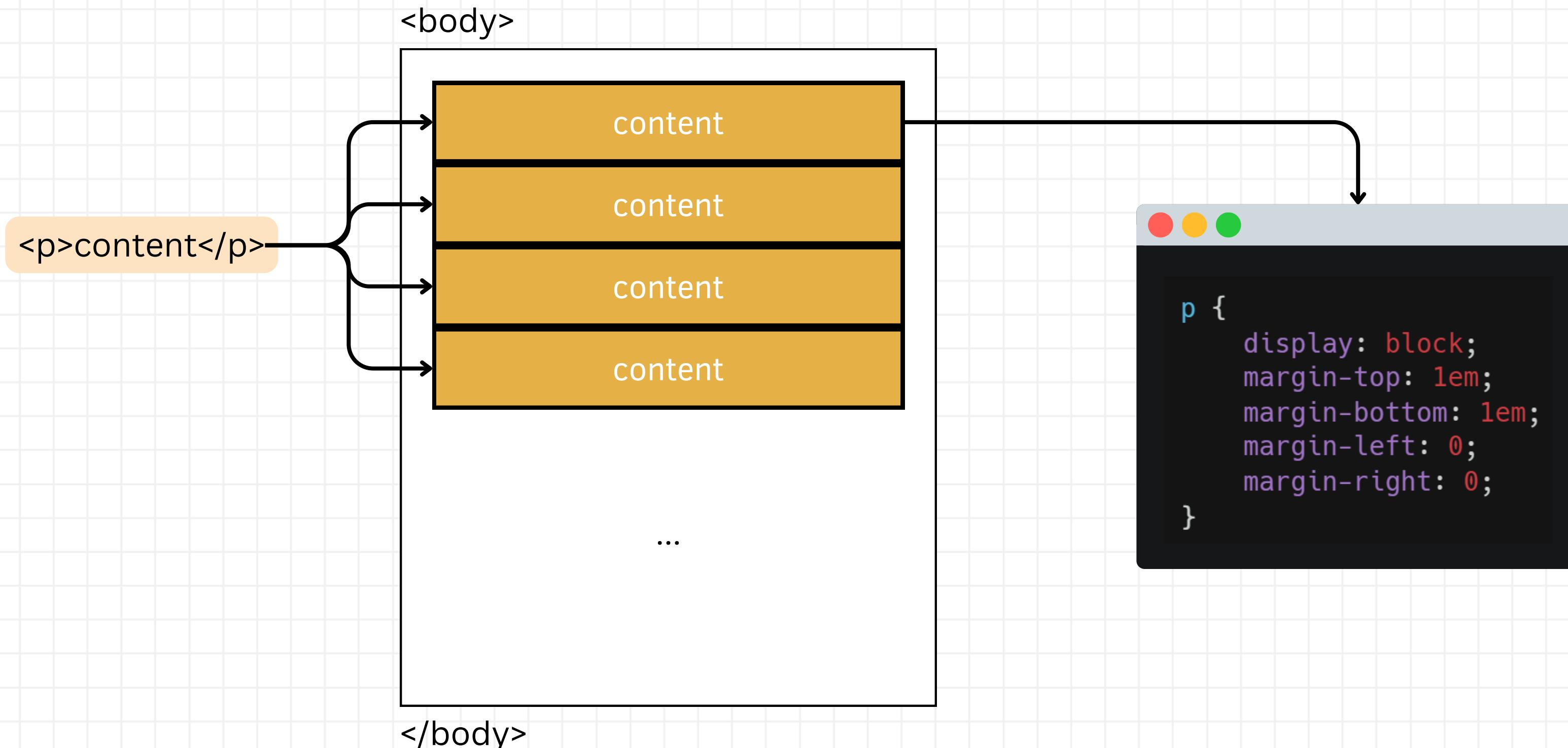


Box Model



```
element {  
    color: xxx;  
    background-color: xxx;  
    text-align: left / right / center / ... ;  
    font-weight: normal / bold /lighter / bolder / ... ;  
  
    width: 200px;  
    height: 100px;  
    padding: 20px;  
    margin: 10px;  
  
    border: width style color;  
}
```

Box Model



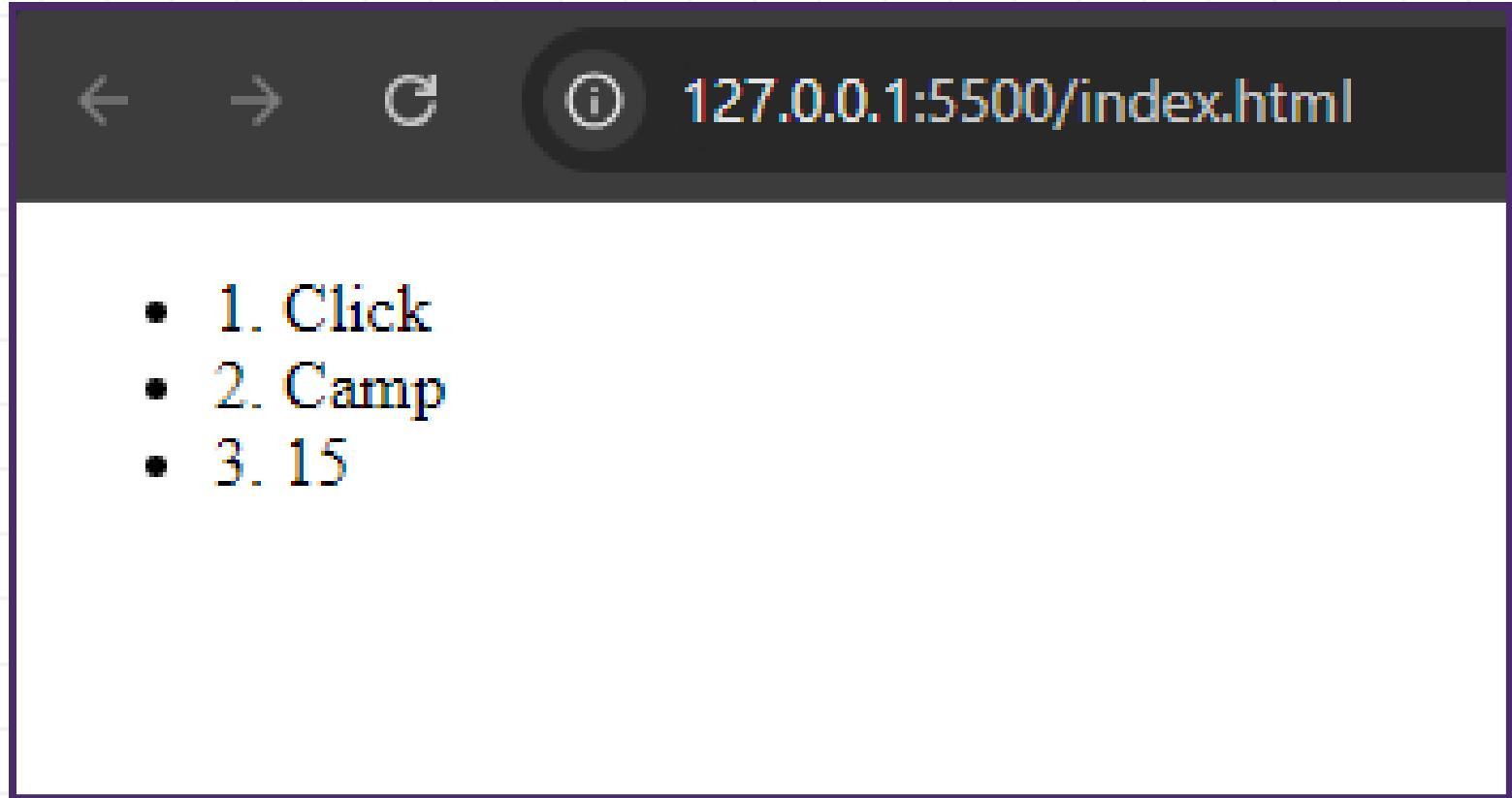
Default Value of each tag

CSS Units

Unit Types	Unit	Description
Absolute Unit	px	Pixels, relative to screen resolution
Relative Unit	em	Relative to parent font size
	rem	Relative to root font size
	vw, vh	Relative to viewport width, height
	%	Relative to parent element

*เป็นแค่ส่วนหนึ่งเท่านั้น

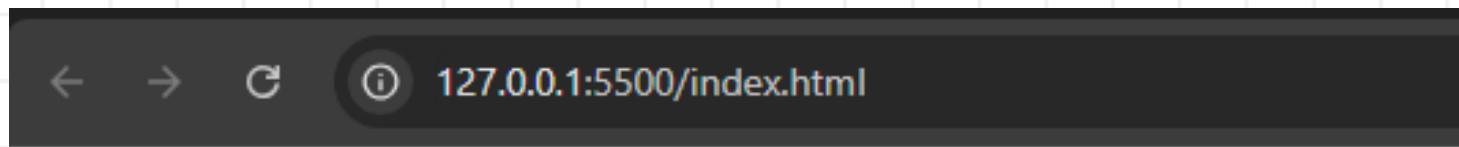
Intermediate Tags



List

- **** <!-- order list -->
- **** <!-- unorder list -->
- **** <!-- list -->

Intermediate Tags



Images

- <!-- សូរាប -->

```

```

How to adjust image size

Attributes

- **width** ปรับขนาดความกว้าง
- **height** ปรับขนาดความสูง



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

Intermediate CSS

Before hover



This is a hover text!

After hover



This is a hover text!

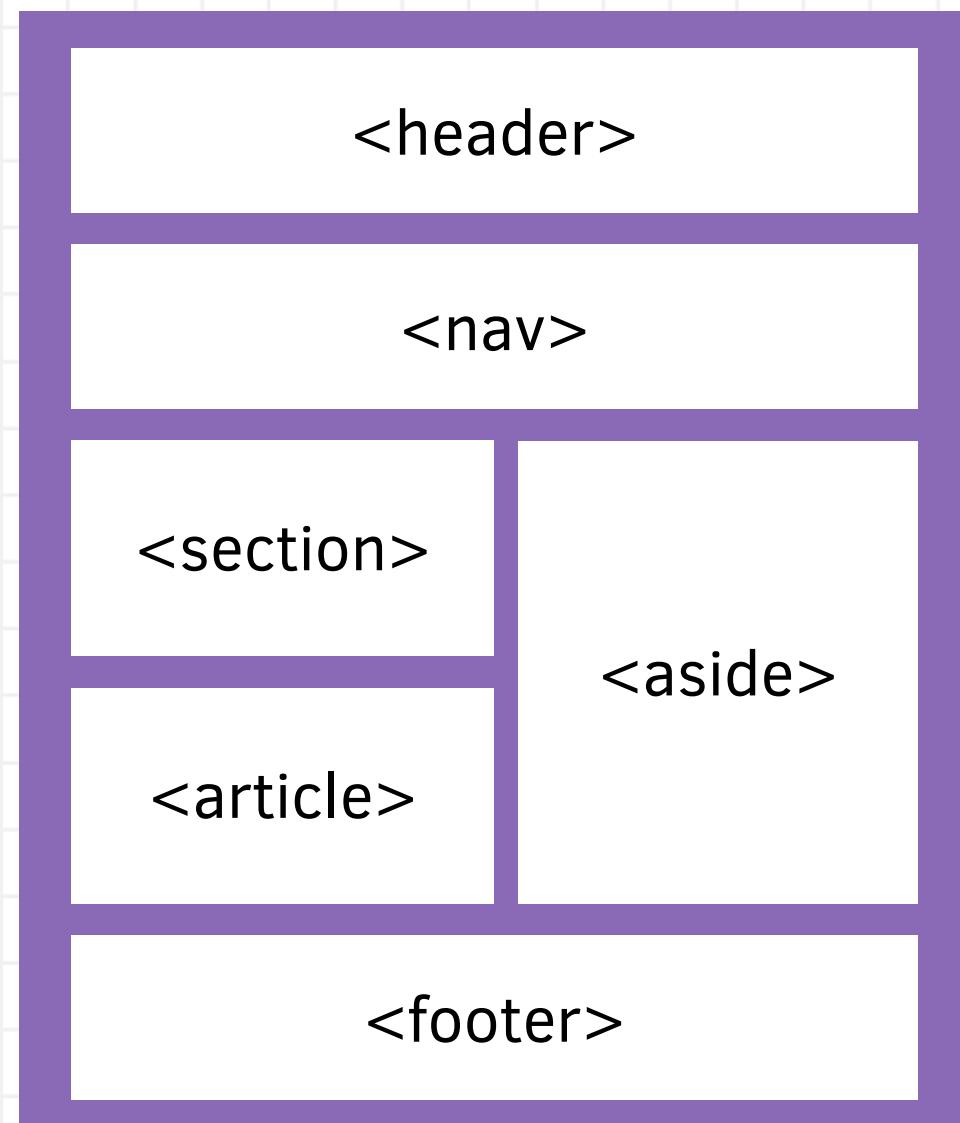
Pseudo Classes

- **:hover** <!-- เอาเม้าส์上去 -->
- **:focus** <!-- เอาเม้าส์ไปไว้ที่ input -->

Element wrapper

Semantic Tags

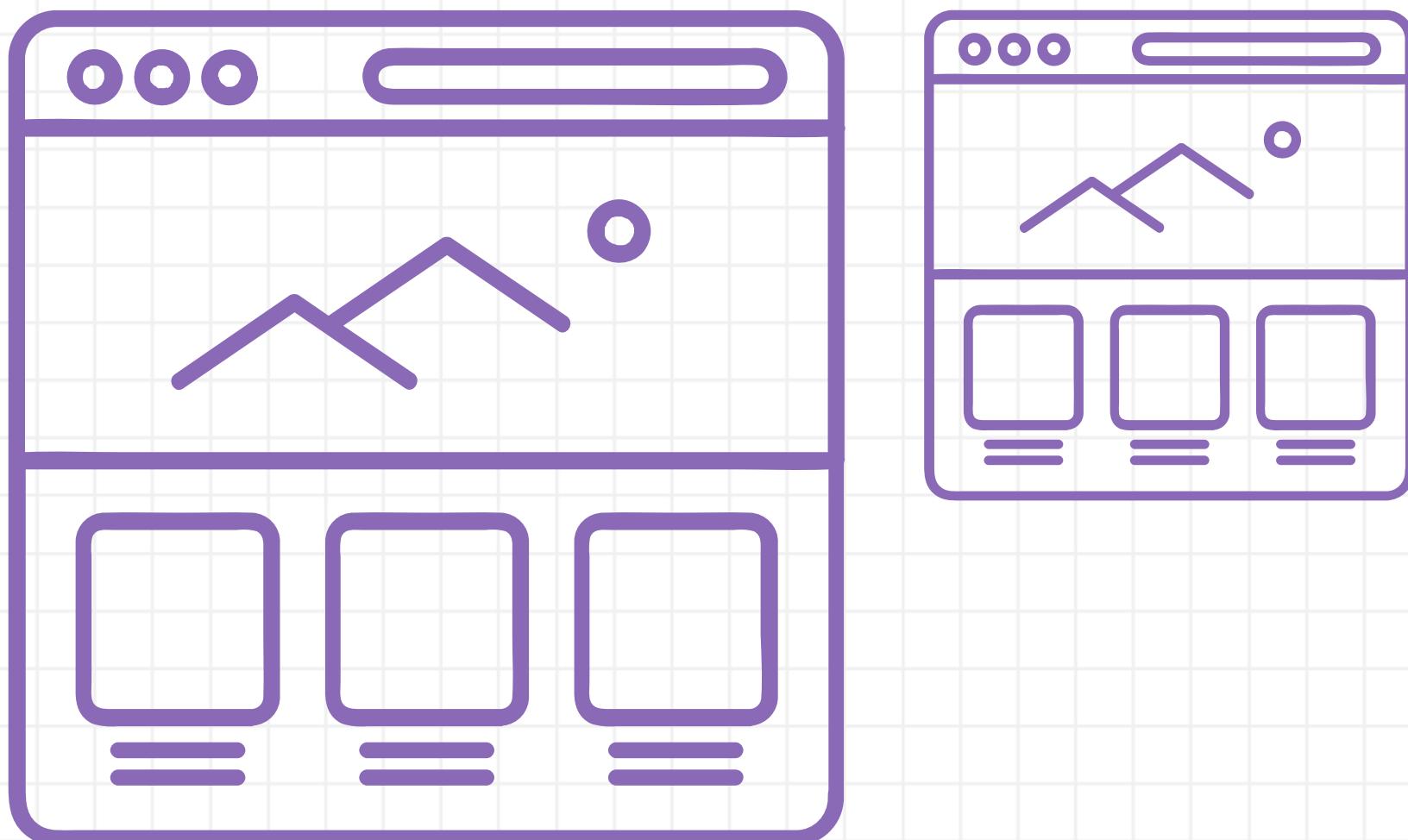
- **<header>**
- **<footer>**
- **<section>**
- **<article>**
- **<nav>**



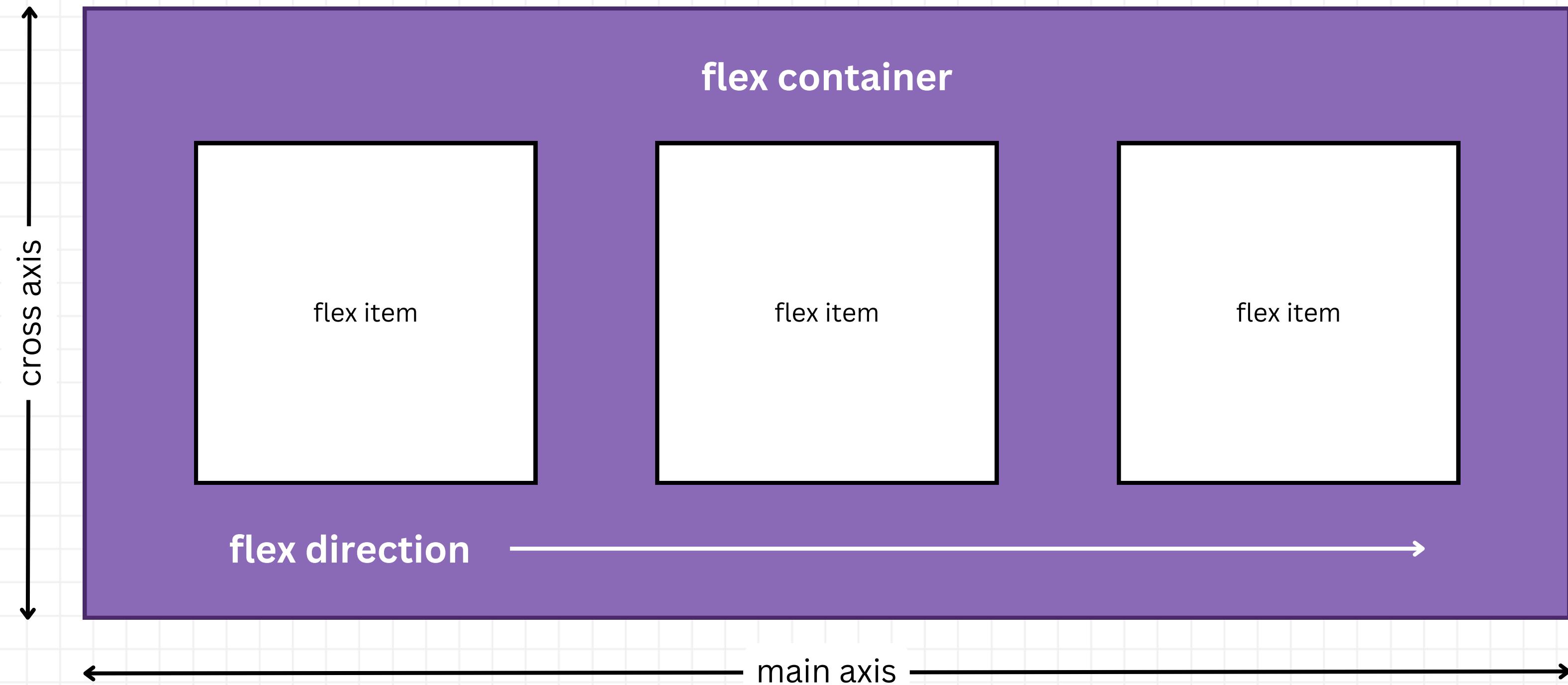
Advance Topic

Website Layout

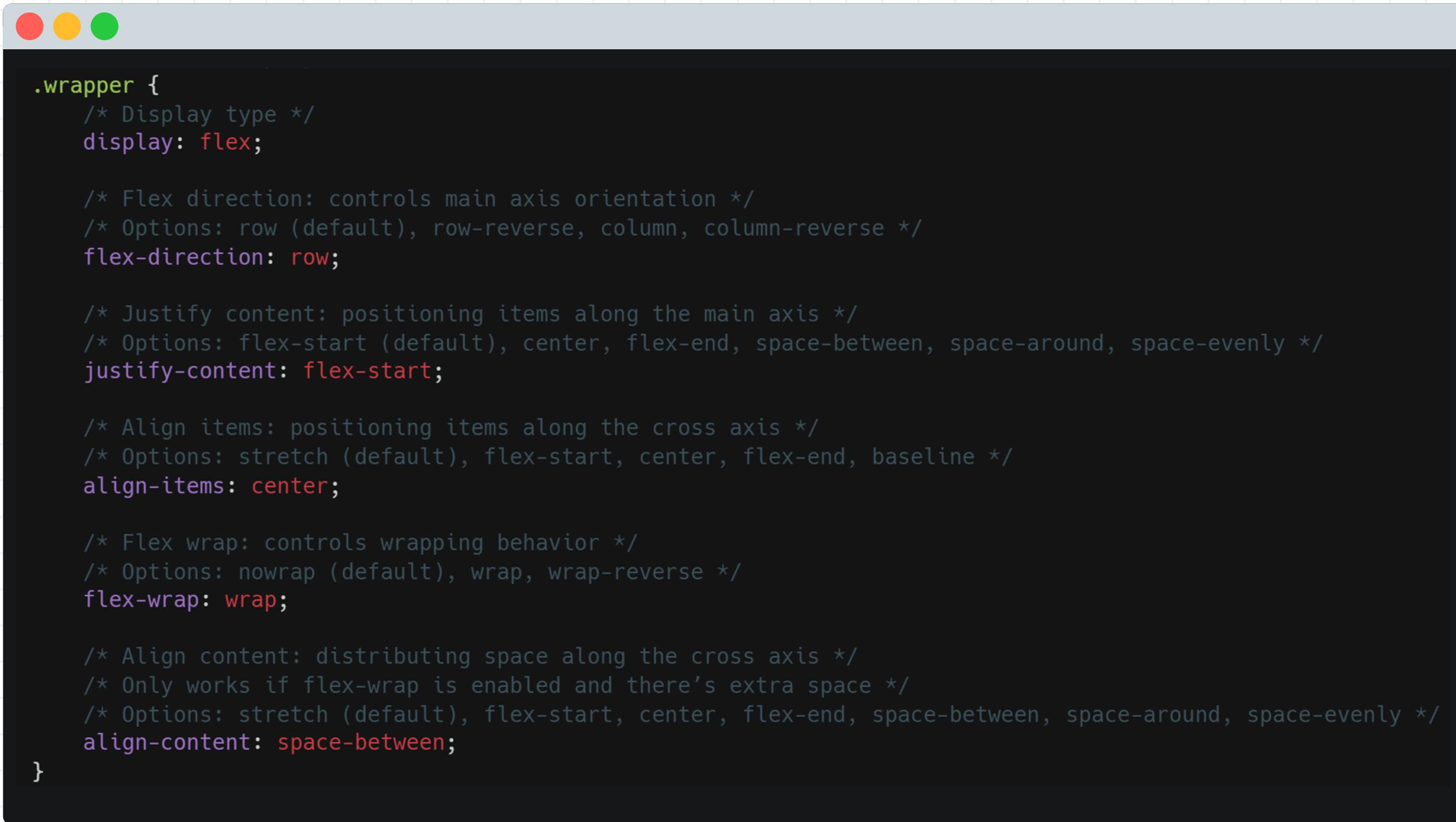
- Grid
- Flex 😊



Flexbox



Flexbox Container Default Value



The image shows a window with a dark background and light gray borders, resembling a Mac OS X application. In the top-left corner, there are three colored window control buttons: red, yellow, and green. The main area contains the following CSS code:

```
.wrapper {
    /* Display type */
    display: flex;

    /* Flex direction: controls main axis orientation */
    /* Options: row (default), row-reverse, column, column-reverse */
    flex-direction: row;

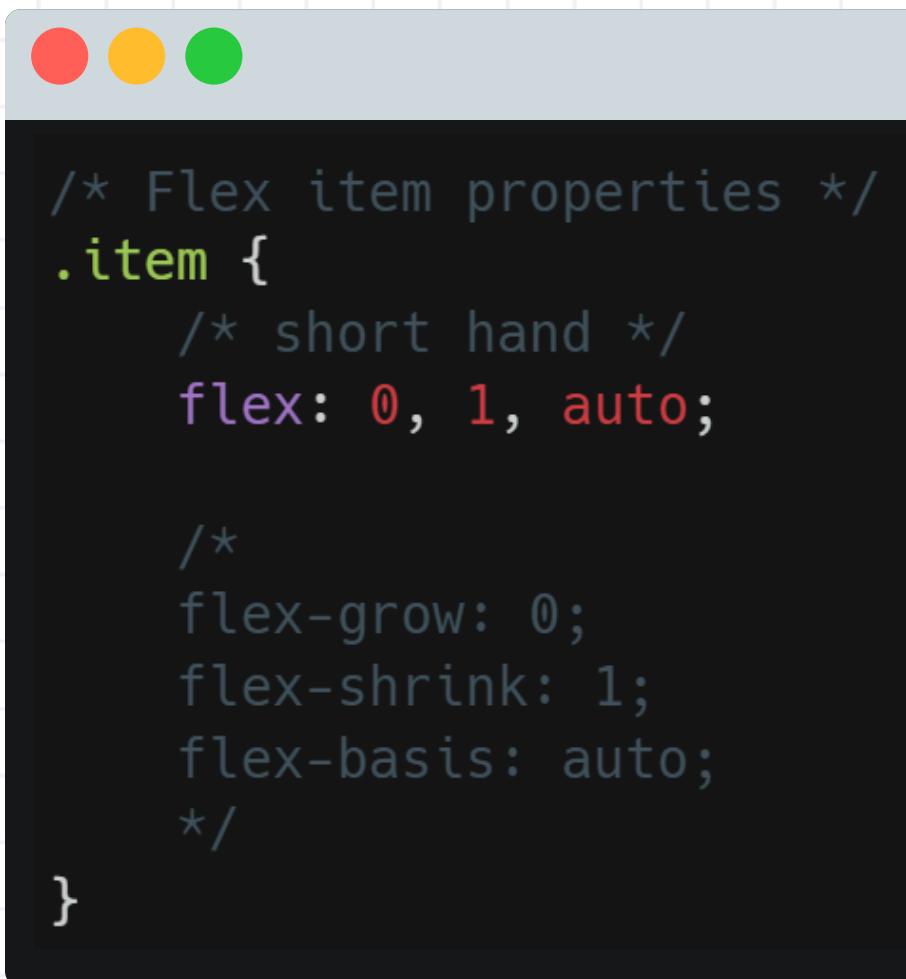
    /* Justify content: positioning items along the main axis */
    /* Options: flex-start (default), center, flex-end, space-between, space-around, space-evenly */
    justify-content: flex-start;

    /* Align items: positioning items along the cross axis */
    /* Options: stretch (default), flex-start, center, flex-end, baseline */
    align-items: center;

    /* Flex wrap: controls wrapping behavior */
    /* Options: nowrap (default), wrap, wrap-reverse */
    flex-wrap: wrap;

    /* Align content: distributing space along the cross axis */
    /* Only works if flex-wrap is enabled and there's extra space */
    /* Options: stretch (default), flex-start, center, flex-end, space-between, space-around, space-evenly */
    align-content: space-between;
}
```

Flexbox item Default Value



```
/* Flex item properties */
.item {
    /* short hand */
    flex: 0, 1, auto;

    /*
    flex-grow: 0;
    flex-shrink: 1;
    flex-basis: auto;
    */
}
```

Properties	default value	behavior
flex-grow	0	ไม่ขยายขนาด
flex-shrink	1	สามารถย่อขนาดลงได้
flex-basis	auto	ขนาดตาม content ด้านใน

Flexbox Froggy

FLEXBOX FROGGY

Level 5 of 24

Now use `align-items` to help the frogs get to the bottom of the pond. This CSS property aligns items vertically and accepts the following values:

- `flex-start`: Items align to the top of the container.
- `flex-end`: Items align to the bottom of the container.
- `center`: Items align at the vertical center of the container.
- `baseline`: Items display at the baseline of the container.
- `stretch`: Items are stretched to fit the container.

```
1 #pond {  
2   display: flex;  
3   align-items: flex-end  
4 }  
5  
6  
7  
8  
9  
10
```

Submit

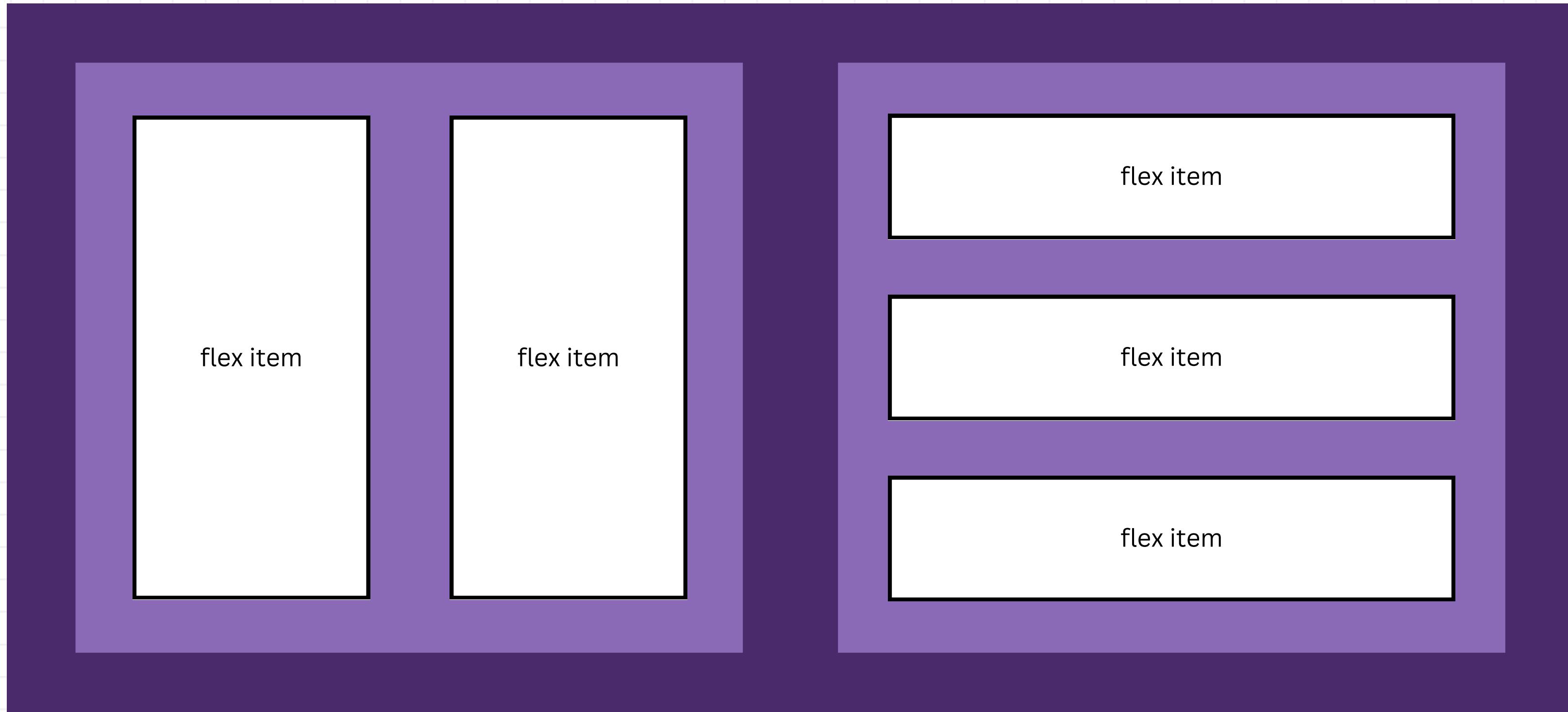
Flexbox Froggy

A game for learning CSS flexbox

 flexboxfroggy.com



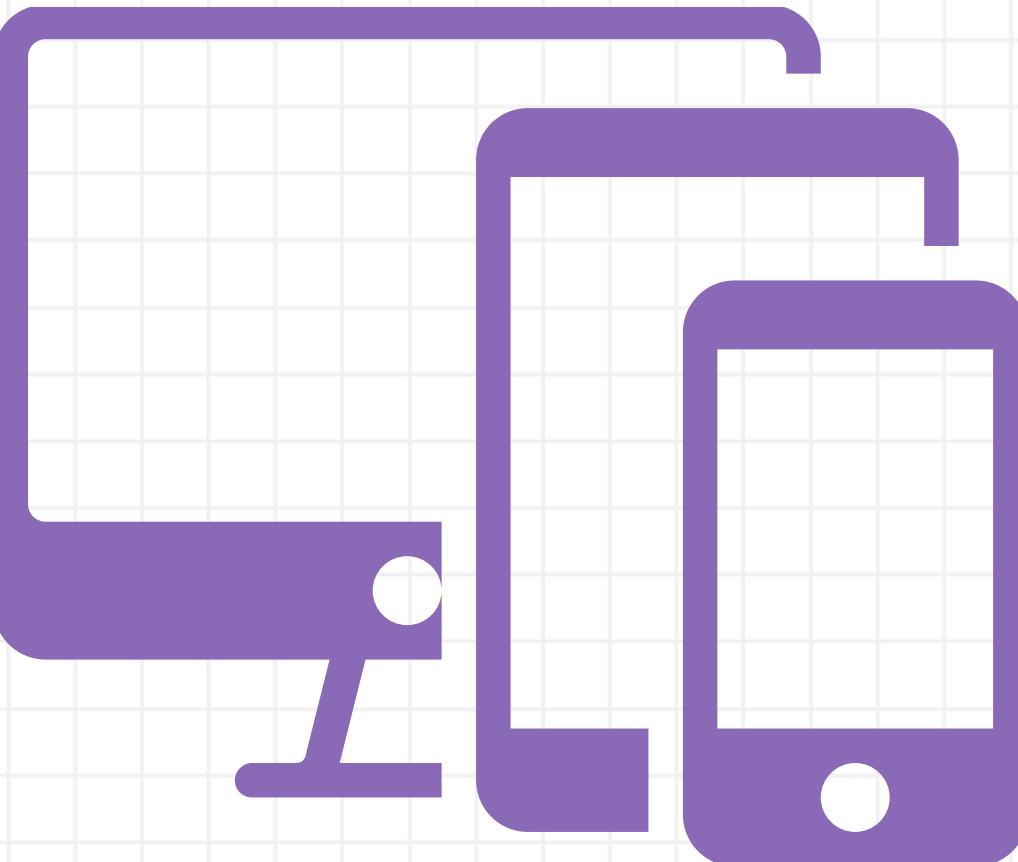
Exercise



Advance Topic

Responsive Web Design

- **@media**
- **max-width**
- **min-width**

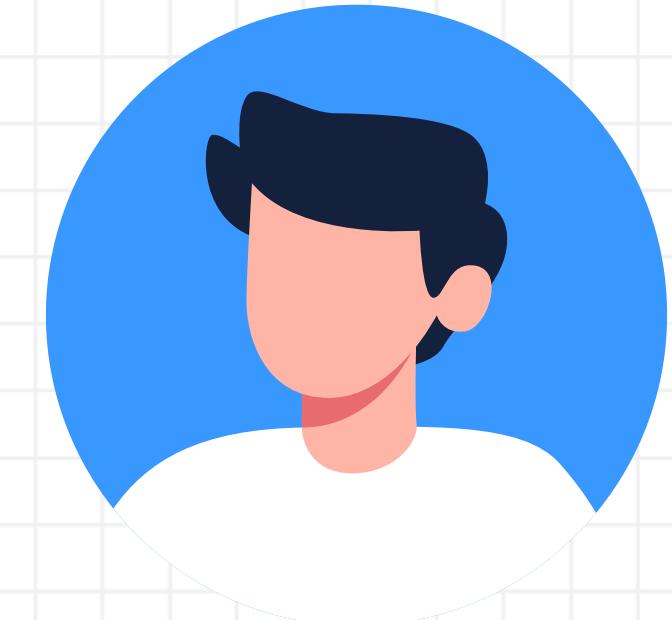


Mini Project #1

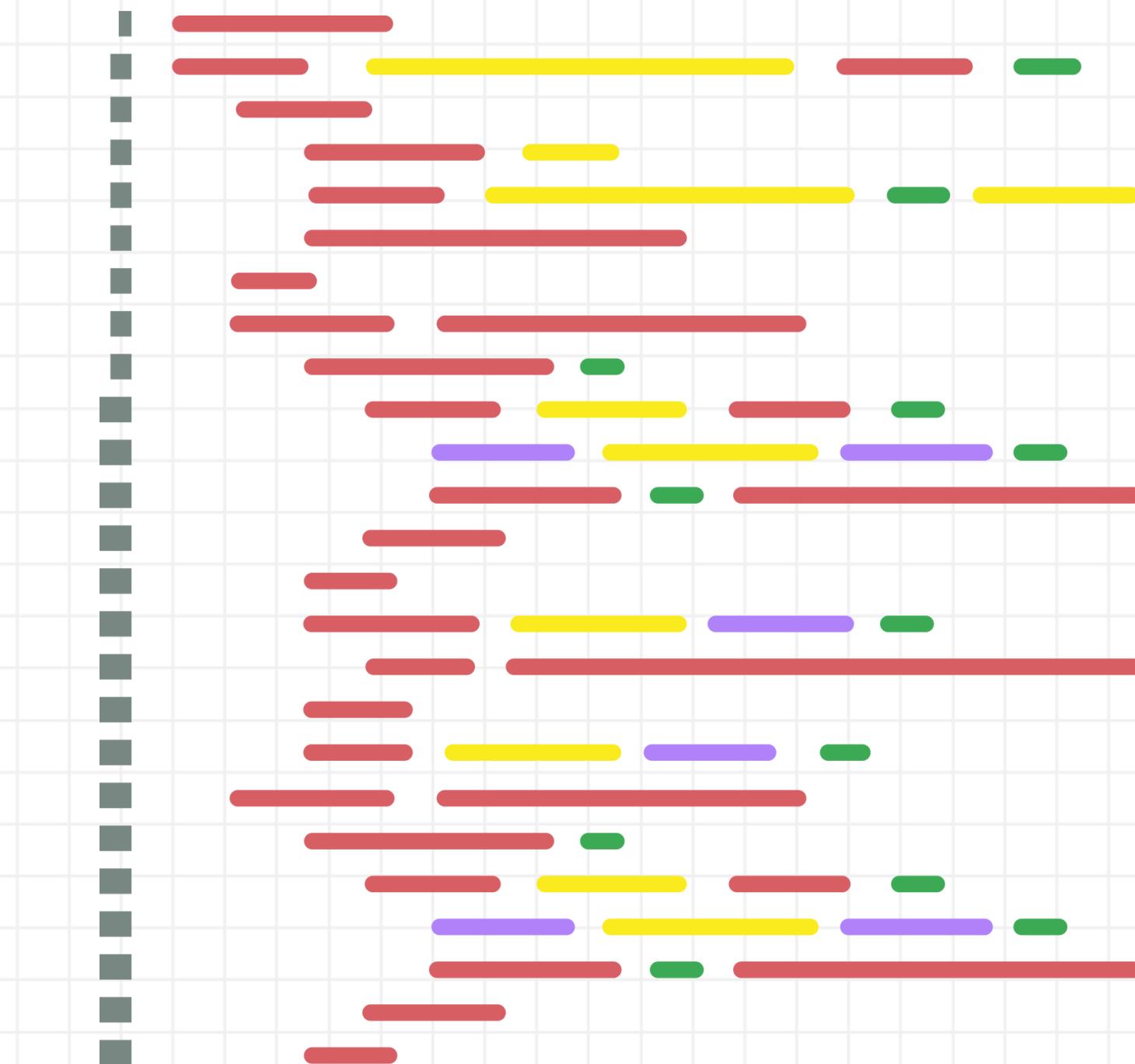
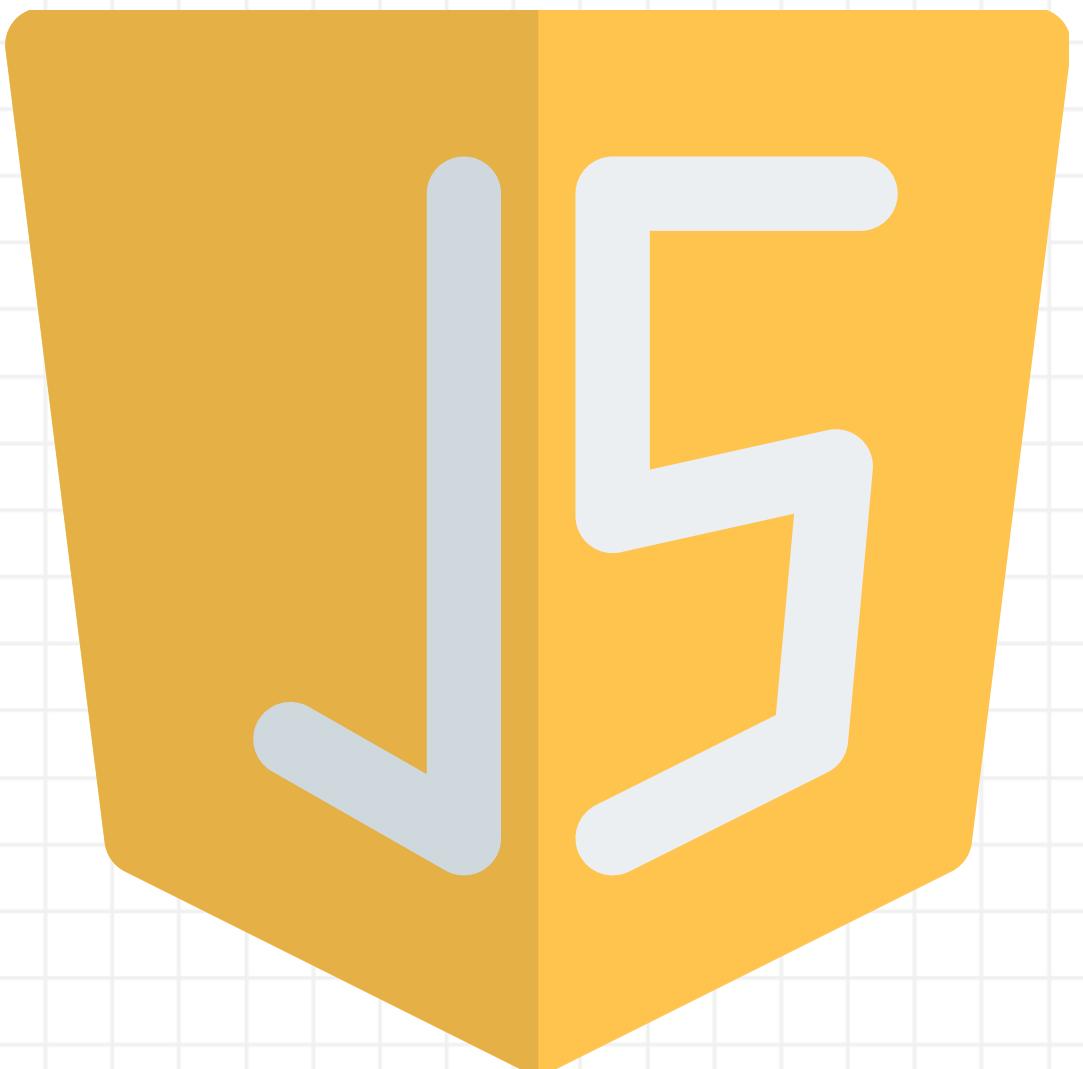
About me Website (45 minutes)

Objectives

- ทำเว็บไซต์อธิบายเกี่ยวกับตนเอง
- มีรูปหน้าตัวเอง 1 รูป
- ตกแต่งให้สวยงาม



Javascript



Make web page “interactive”



light/dark mode



Method to add Javascript

index.html

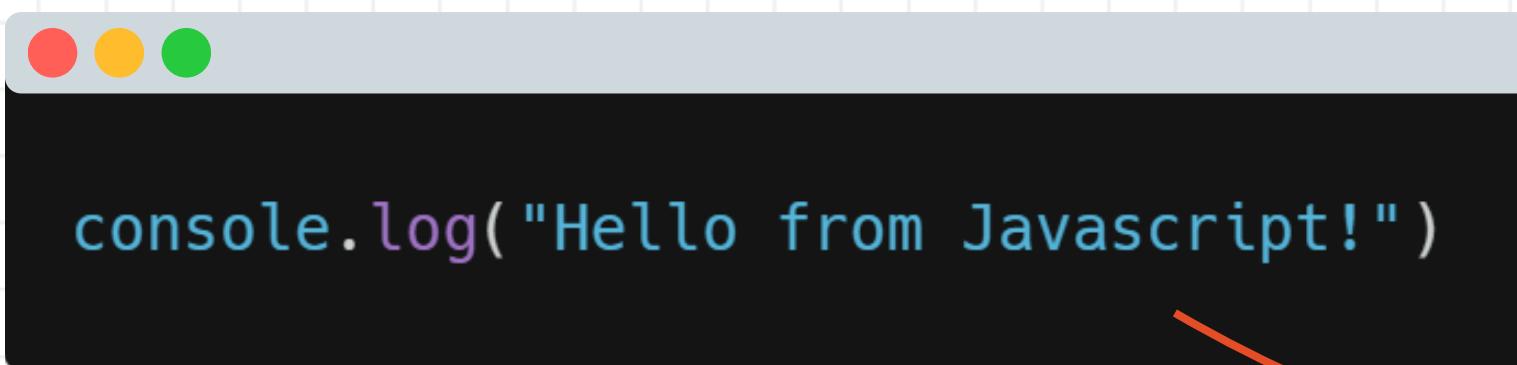
```
<body>
  <p>Hello, World!</p>
  <script src="script.js"></script>
</body>
```

script.js

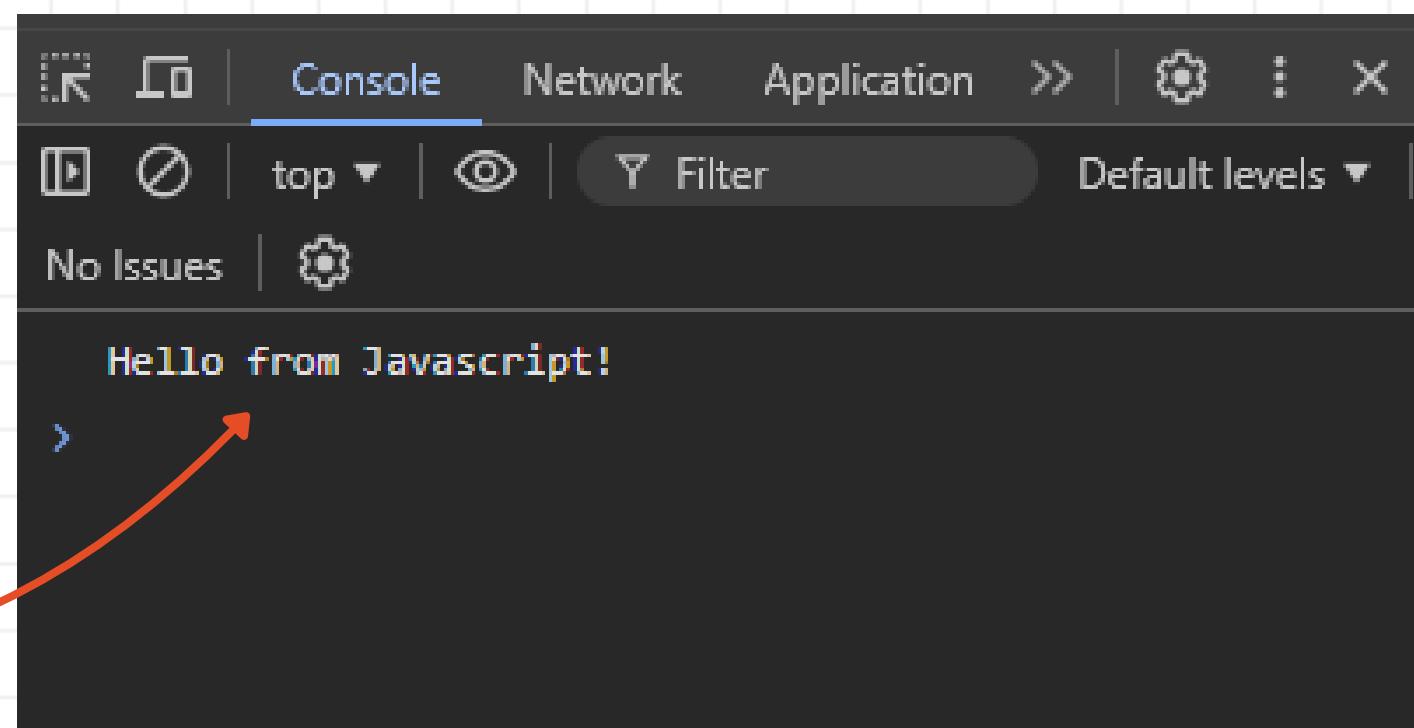
```
// write javascript here!
```

First Program

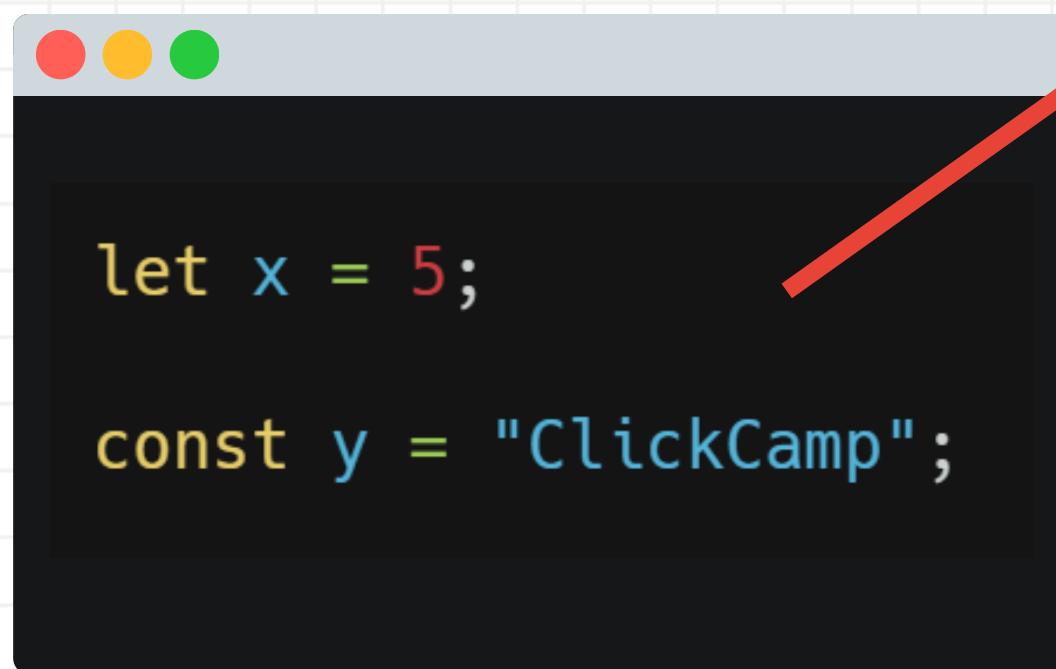
- console.log() เป็นคำสั่งที่แสดงผลออกมายัง Console ของ website เรานะ



```
console.log("Hello from Javascript!")
```



Variable Types



```
let x = 5;  
const y = "ClickCamp";
```

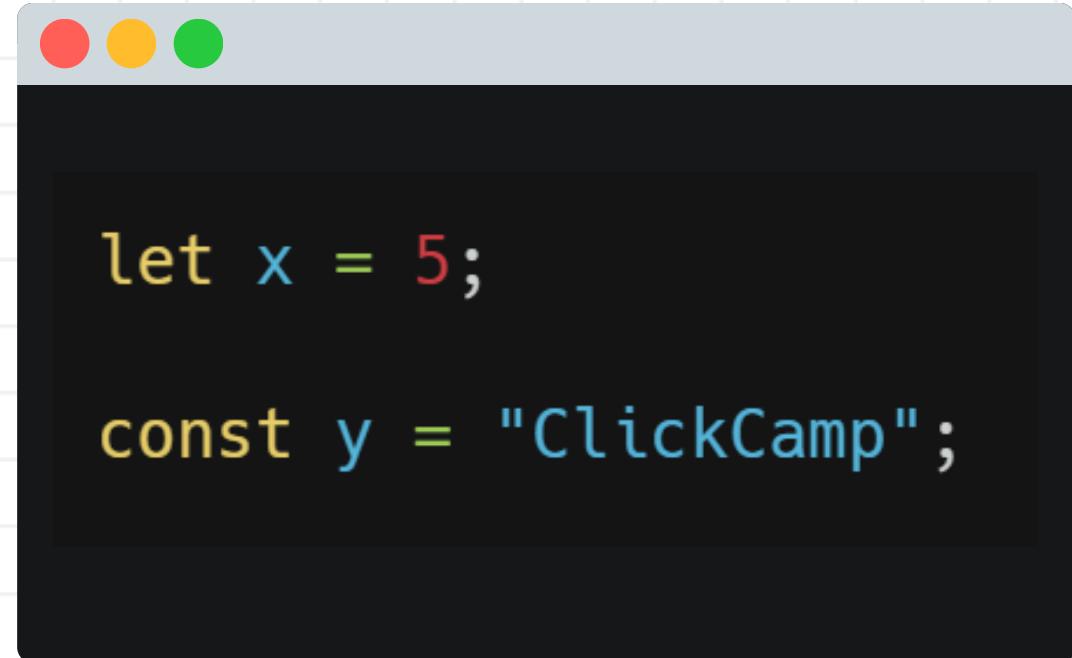
A screenshot of a browser's developer tools console. It shows two lines of JavaScript code: 'let x = 5;' and 'const y = "ClickCamp";'. The code is written in a monospaced font. A red arrow points from the text 'ตัวตั้ง' (base) below to the variable 'x' in the first line of code.

Number คือประเภทข้อมูลที่เป็นตัวเลข
String คือประเภทข้อมูลที่เป็นตัวอักษร

อื่นๆ

- Bigint = ตัวเลขที่เยอะมาก ๆ
- Boolean = **true / false**
- Undefined = ตัวแปรที่ไม่มีค่าเริ่มต้น
- Null = ตัวแปรที่ตั้งใจให้เป็นค่าว่าง
- Object = ตัวแปรที่สืบเนื่องกับสิ่งของ

let vs const



let

คือกำหนดให้ตัวแปรเปลี่ยนค่าได้

const

คือกำหนดให้ตัวแปรเปลี่ยนค่าไม่ได้

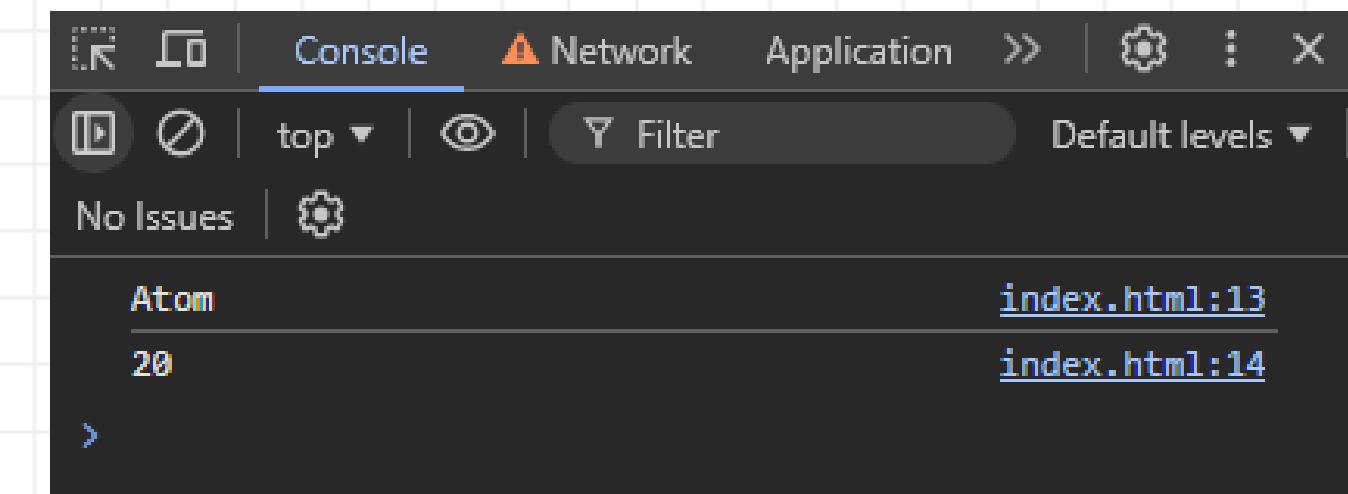
Print a Variable on Console

- เราสามารถนำตัวแปร (variable) มาใช้กับคำสั่ง console ได้



```
let name = "Atom"
let age = "20"

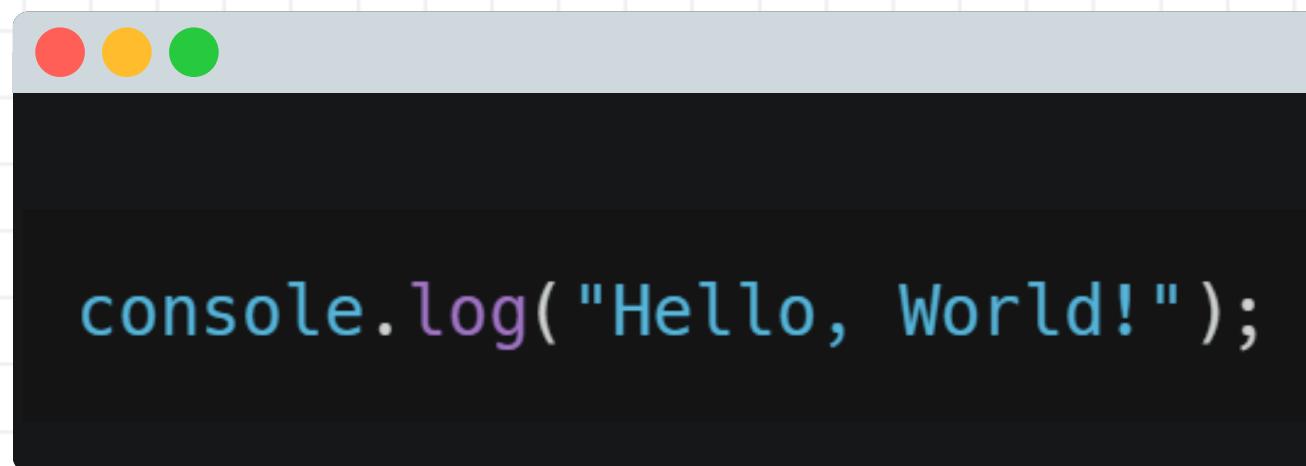
console.log(name)
console.log(age)
```



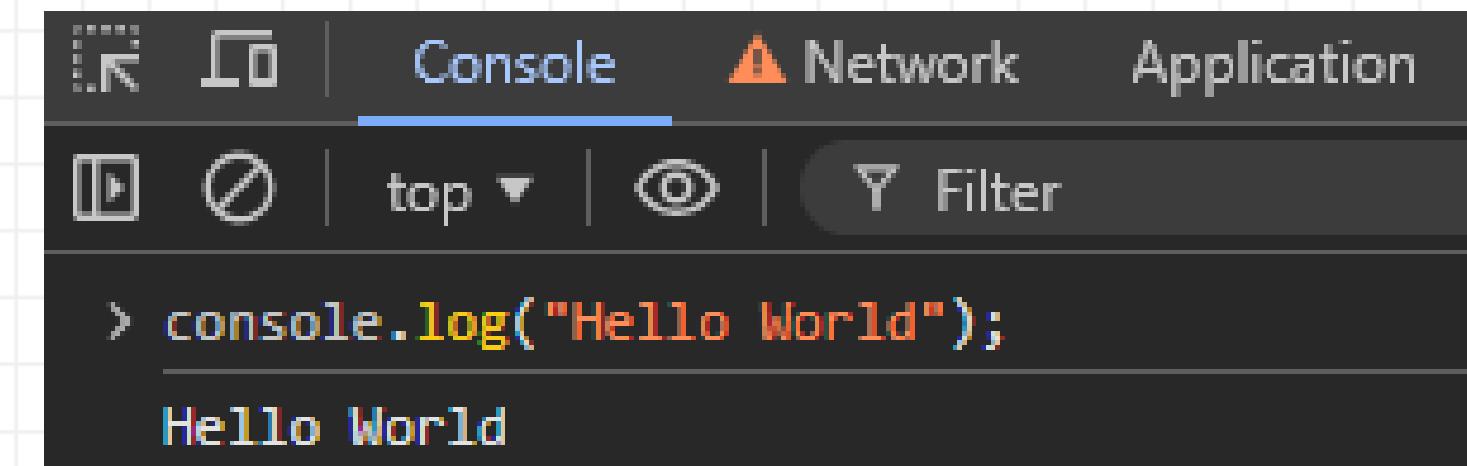
```
Console Network Application > ⚙️ ⓘ
top ⓘ Filter Default levels ⓘ
No Issues ⓘ
Atom index.html:13
20 index.html:14
>
```

Write on Console

- เราสามารถเขียน Javascript บน Console เลยก็ได้เหมือนกัน



```
console.log("Hello, World!");
```



```
Console Network Application
top ▾ Filter
> console.log("Hello World");
Hello World
```

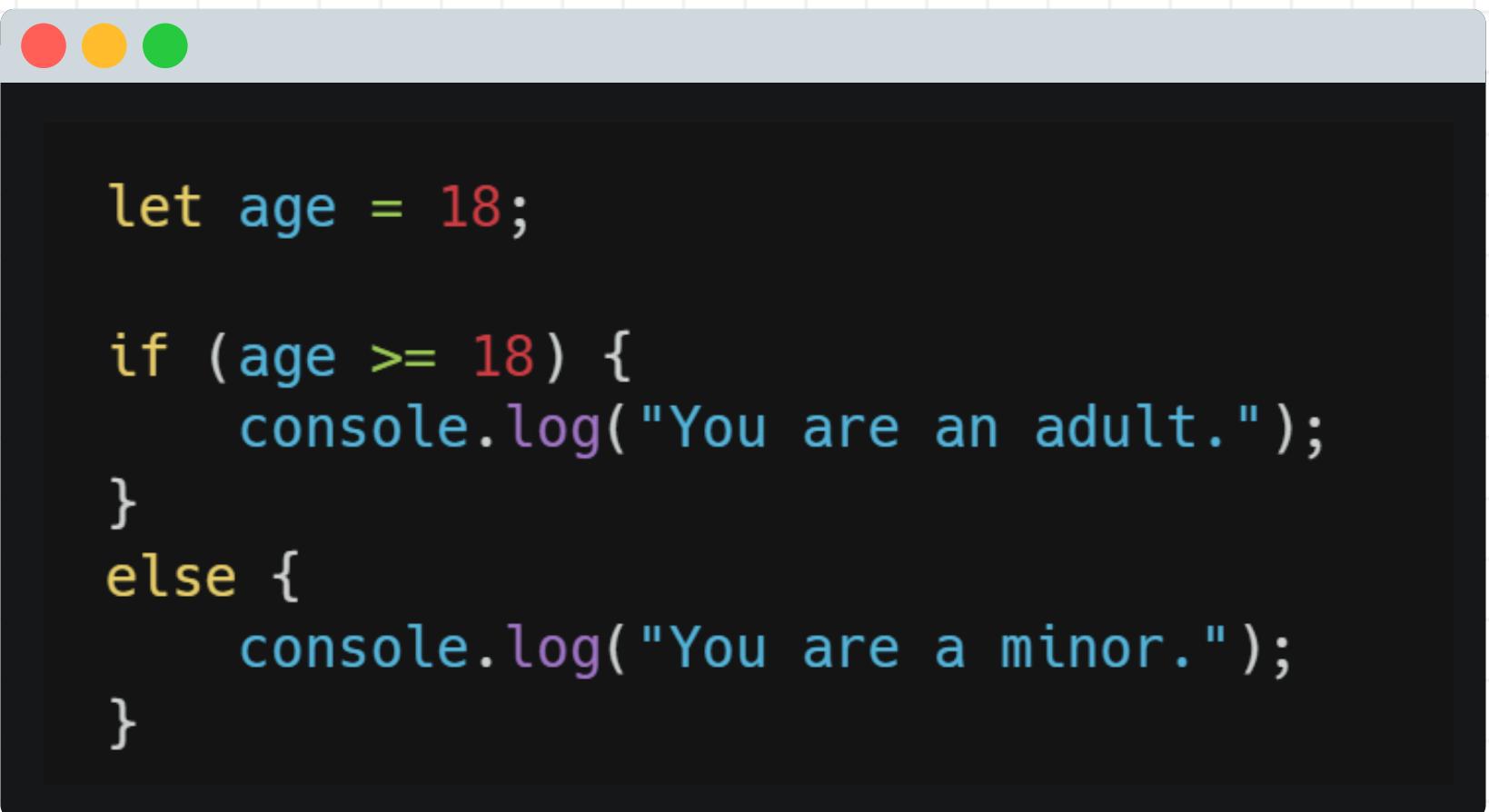
Basic Operator

```
let A = 5;
let B = 10;
let C = 2;
let D = 8;
let E = 1;
console.log(A+B*C/D-E)
```

+	บวก
-	ลบ
*	คูณ
/	หาร

If - else if - else

- การสร้างเงื่อนไขให้กับโปรแกรมเรา ถ้าเข้าเงื่อนไขแล้ว เราจะให้โปรแกรมทำอะไร



```
let age = 18;

if (age >= 18) {
    console.log("You are an adult.");
}
else {
    console.log("You are a minor.");
}
```

A screenshot of a terminal window with a dark background. The window has three colored window control buttons (red, yellow, green) at the top left. The terminal displays a piece of JavaScript code. The code defines a variable 'age' and uses an if-else statement to log different messages to the console based on the value of 'age'. The code is color-coded: 'let' and 'console' are blue, 'age' is orange, and the numbers and strings are black.

Basic comparison operators

- เครื่องหมายสำหรับเปรียบเทียบค่าต่างๆ ในโปรแกรม

OPERATORS	MEANING	EXAMPLE	RESULT
<	น้อยกว่า	4<2	False
>	มากกว่า	4>2	True
<=	น้อยกว่าเท่ากับ	4<=2	False
>=	มากกว่าเท่ากับ	4>=2	True
==	เท่ากับ	4==2	False
!=	ไม่เท่ากับ	4!=2	True
===	เท่ากับและข้อมูล ประเภทเดียวกัน	4==="4"	False
!==	ไม่เท่ากันหรือไม่เป็น ข้อมูลประเภทเดียวกัน	4!=="4"	True

prompt()

- เป็นคำสั่งที่จะรับค่าผ่านทางหน้าจอได้ และ เก็บค่าไว้ที่ตัวแปรที่เรากำหนดให้



For loops

- เป็นการสร้างเงื่อนไขในการกำคำสั่งซ้ำ ๆ ในโปรแกรมของเรา



```
for (initial; condition; update) {  
    // code  
}
```

Initial

Condition

update

กำหนดค่าเริ่มต้นสำหรับตัวแปรที่ใช้วนลูป

กำหนดเงื่อนไขสำหรับการวนลูป

เปลี่ยนแปลงค่าตัวแปรที่ใช้ในการวนลูป

For loops

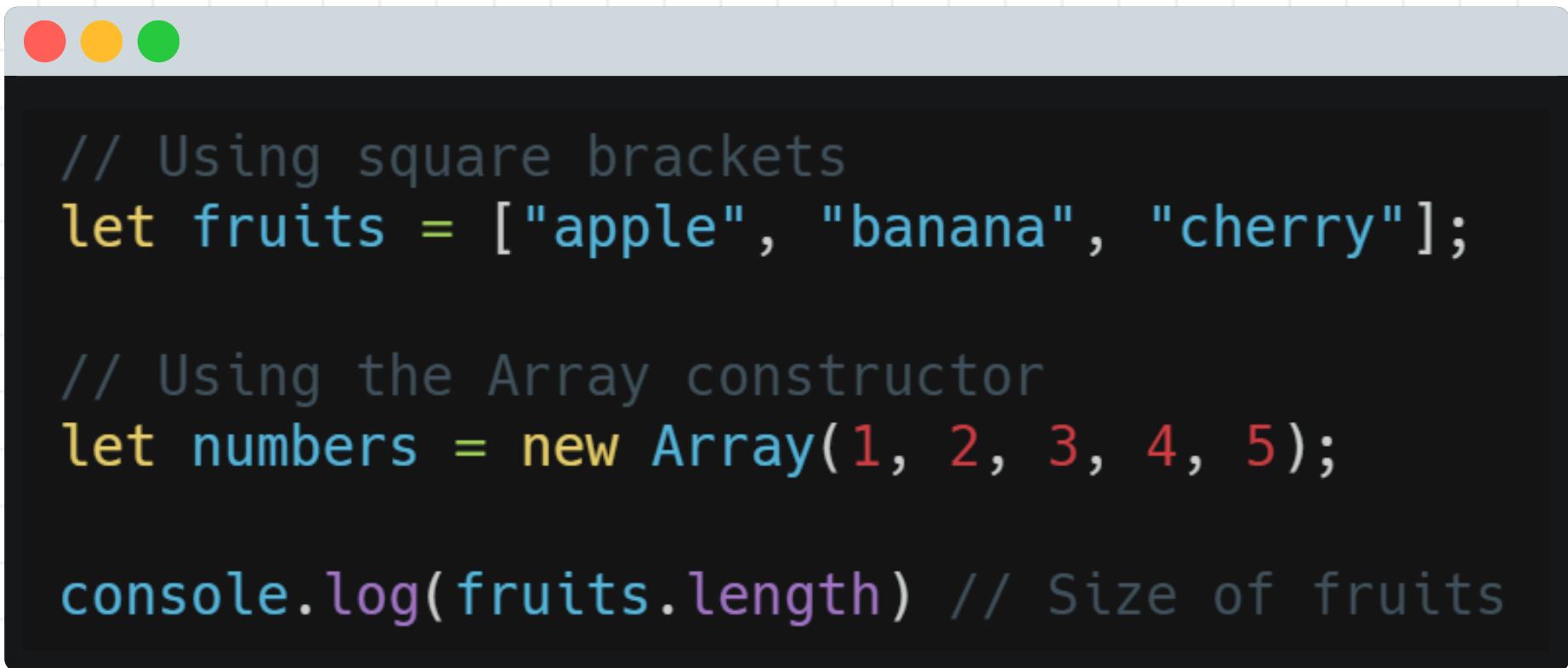
Example



```
for (let i = 0; i < 5; i++) {  
    console.log(i);  
}
```

Array

- คือตัวแปรที่เป็นชุดของข้อมูลที่สามารถเก็บข้อมูลไว้หลาย ๆ ตัวได้



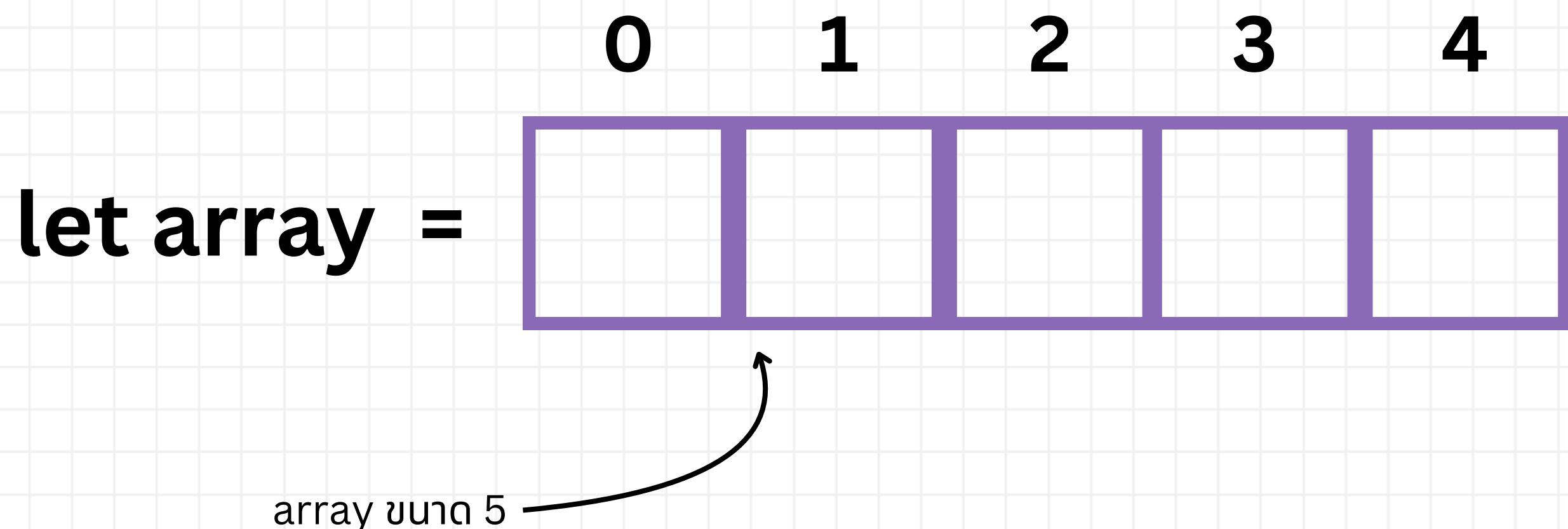
```
// Using square brackets
let fruits = ["apple", "banana", "cherry"];

// Using the Array constructor
let numbers = new Array(1, 2, 3, 4, 5);

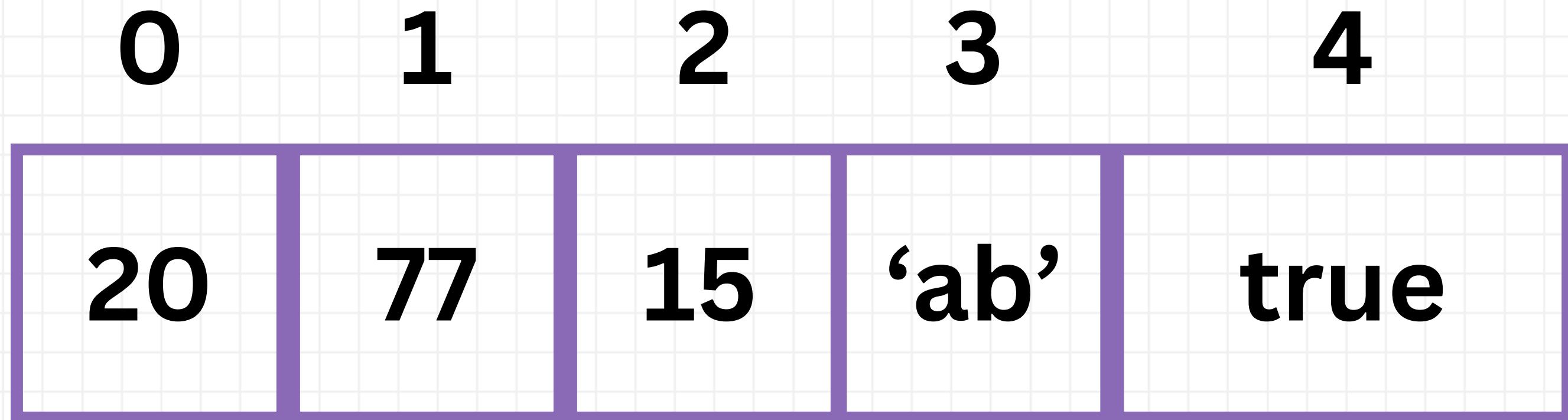
console.log(fruits.length) // Size of fruits
```

Array & index

- index คือลำดับของข้อมูลที่อยู่ภายใน array โดยเริ่มจาก 0 และ เราสามารถเรียกใช้ข้อมูลภายใน array ผ่าน index ได้



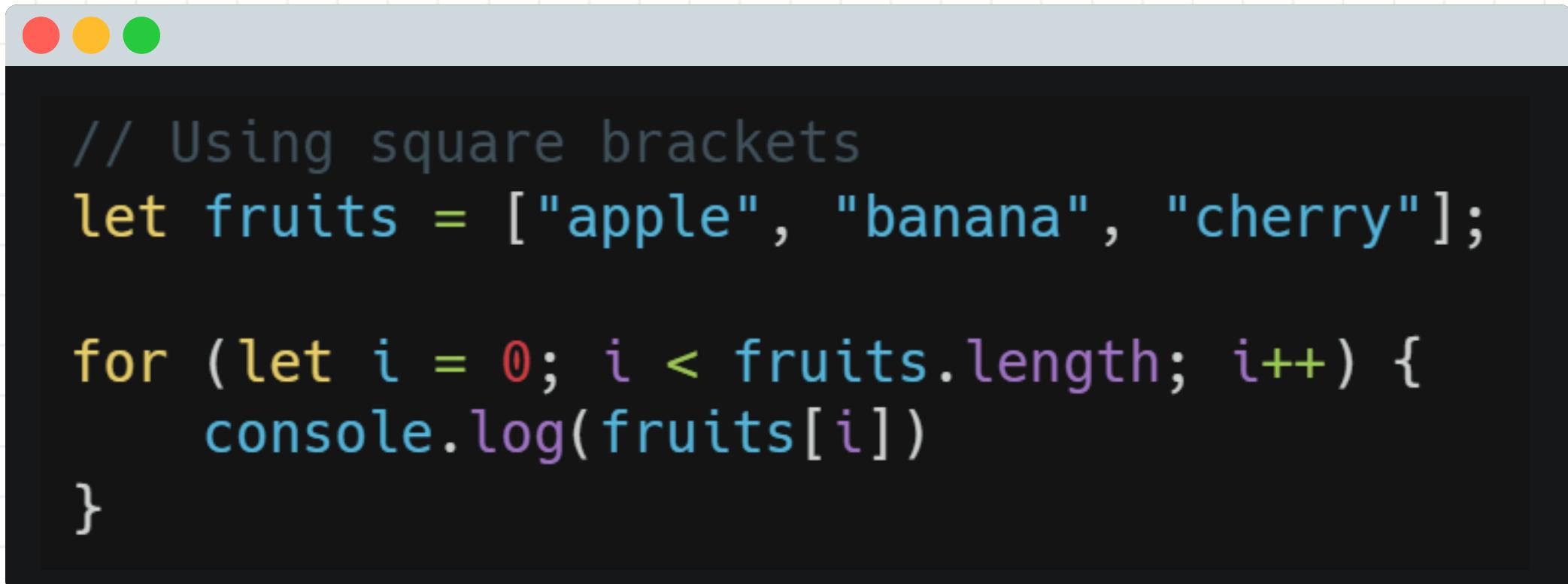
Array & index



array[3] → ‘ab’

Array & index

Example



A screenshot of a terminal window with a dark background and light-colored text. The window has three red, yellow, and green close buttons at the top left. The code inside the terminal is:

```
// Using square brackets
let fruits = ["apple", "banana", "cherry"];

for (let i = 0; i < fruits.length; i++) {
    console.log(fruits[i])
}
```

Function

- ชุดคำสั่งที่เราสามารถนำมาเรียกใช้ได้ โดยเราสามารถสร้างเอง หรือ ใช้อันที่มีอยู่แล้วก็ได้

```
function name (parameter)
{
    // code
}
```

name	ชื่อของฟังก์ชันที่เราสร้าง
parameter	ค่าที่ฟังก์ชันสามารถนำไปใช้ต่อได้
argument	ค่าที่เราใส่เมื่อเรียกใช้ฟังก์ชัน

```
name(argument)
```

argument = parameter

argument ที่เราใส่ในฟังก์ชันเมื่อเรียกใช้จะ
เท่ากับ parameter ในฟังก์ชัน

Function

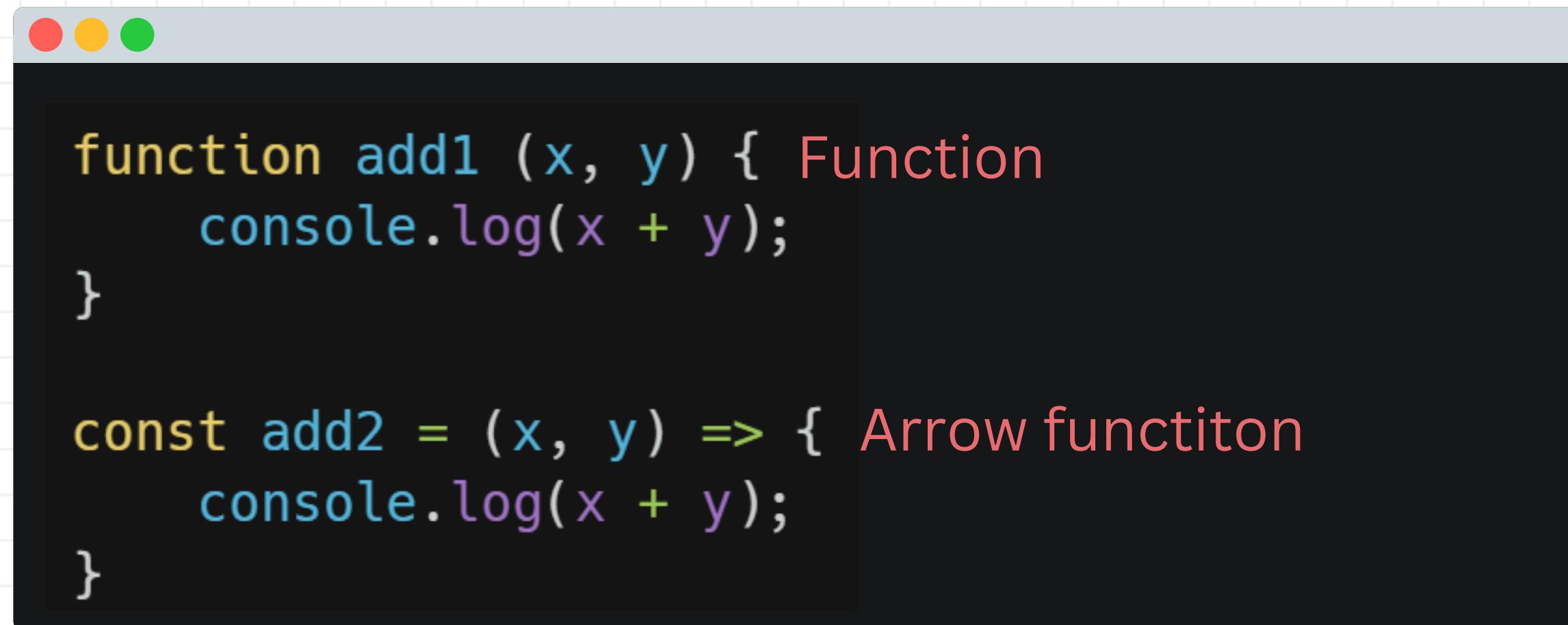
Example



```
function printName (name) {  
    console.log("My name is " + name)  
}
```

Arrow Function

Example



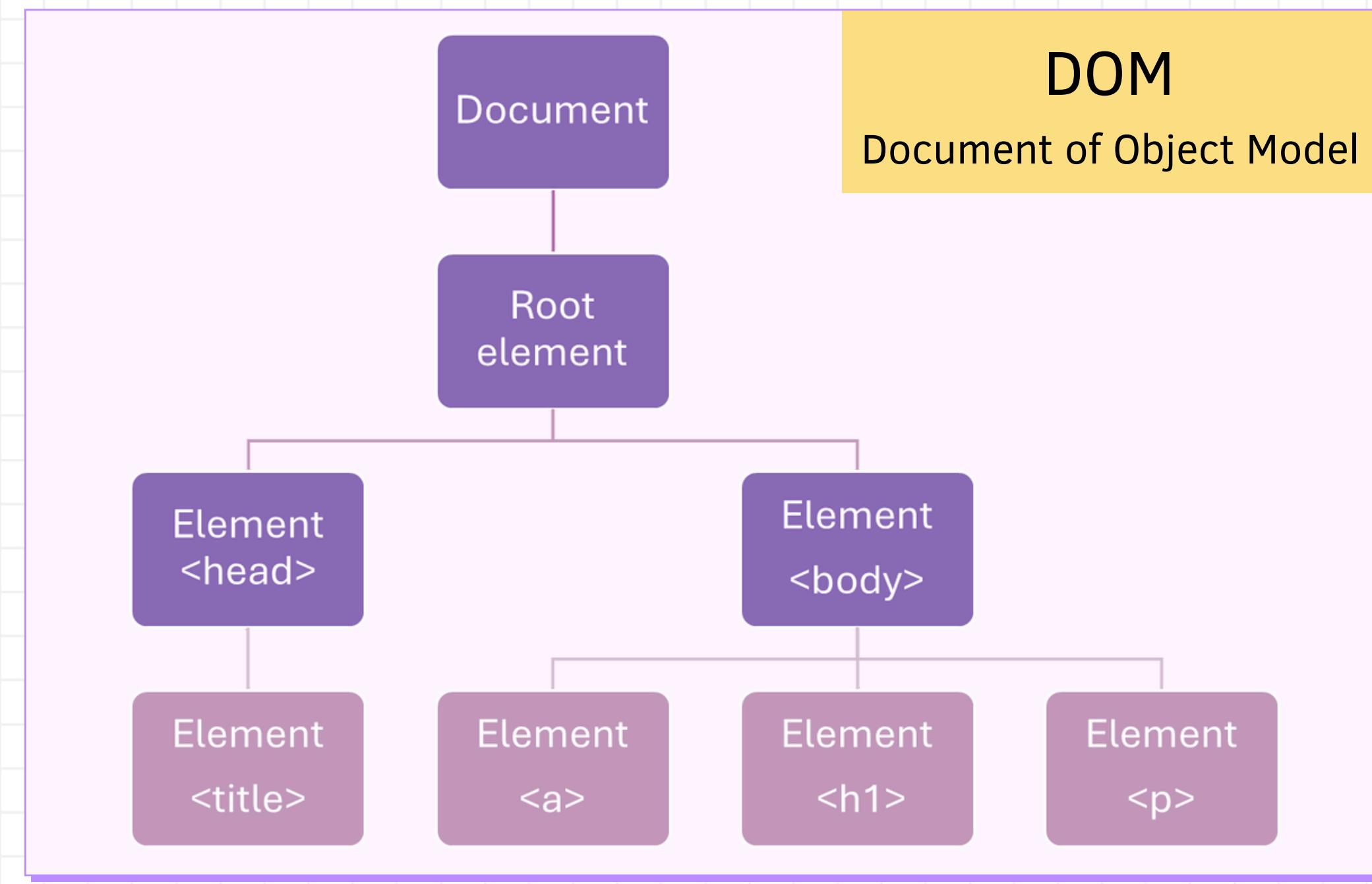
The image shows a screenshot of a macOS application window. The window has a dark gray background and three colored window controls (red, yellow, green) at the top-left corner. Inside the window, there are two blocks of code side-by-side, both using syntax highlighting.

```
function add1 (x, y) { Function
  console.log(x + y);
}

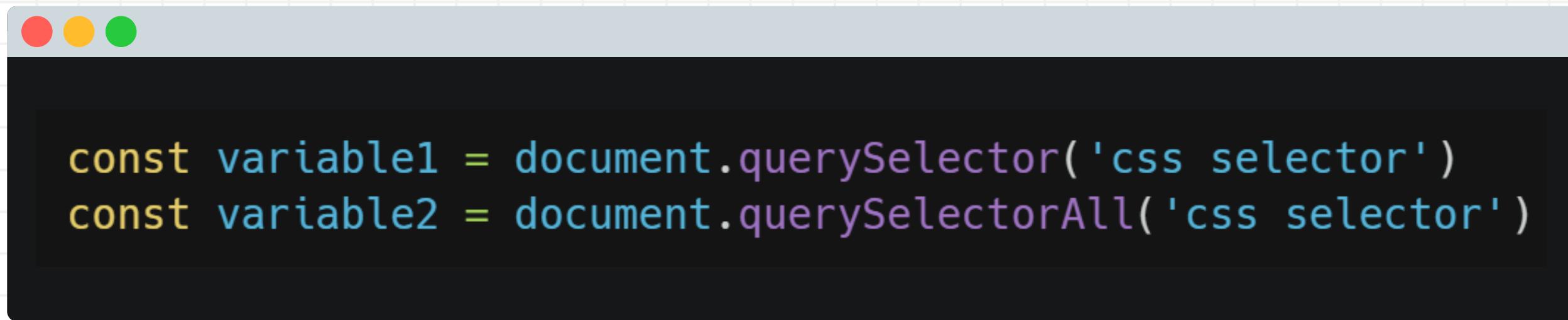
const add2 = (x, y) => { Arrow functiton
  console.log(x + y);
}
```

DOM

- โครงสร้างของ HTML สามารถเขียนได้ในลักษณะของแผนภาพต้นไม้ และ javascript สามารถเข้าถึงและแก้ไขค่าภายในต่างๆ ได้



DOM Manipulation



```
const variable1 = document.querySelector('css selector')
const variable2 = document.querySelectorAll('css selector')
```

document.querySelector()

document.querySelectorAll()

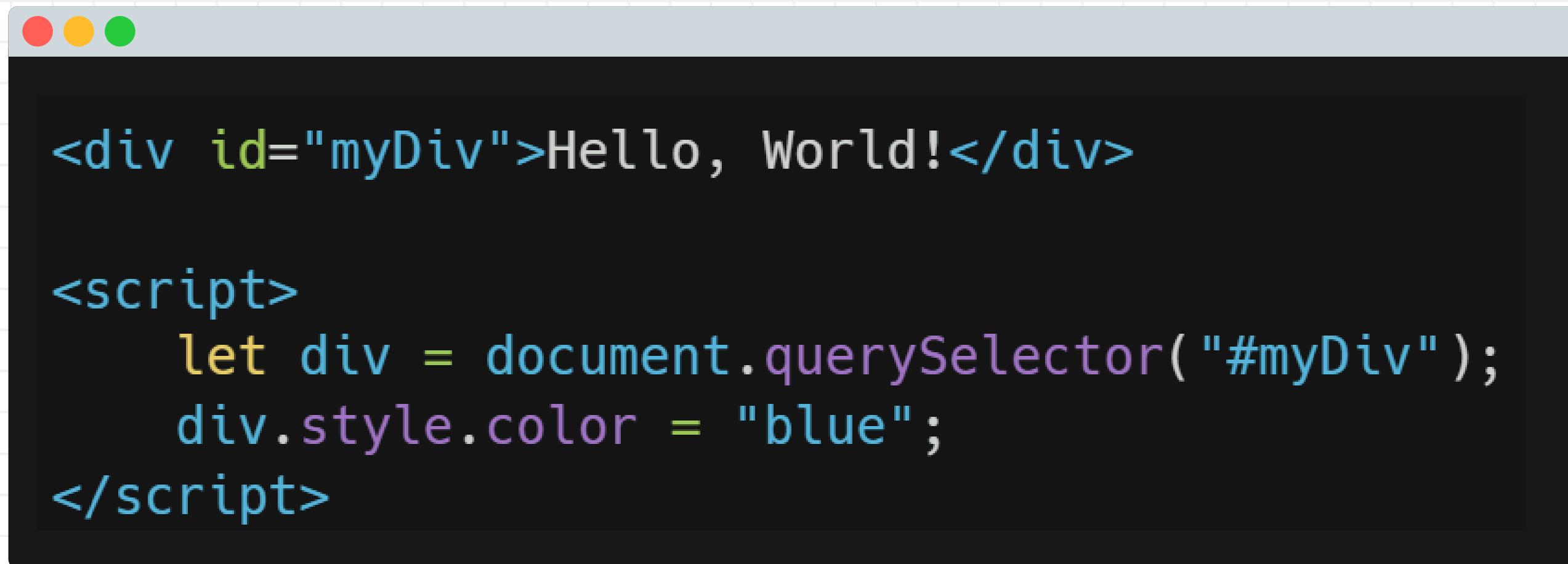
เลือกตัวแรกที่เจอ

เลือกทุกตัวที่เงื่อนไขตรง

.innerHTML



.style



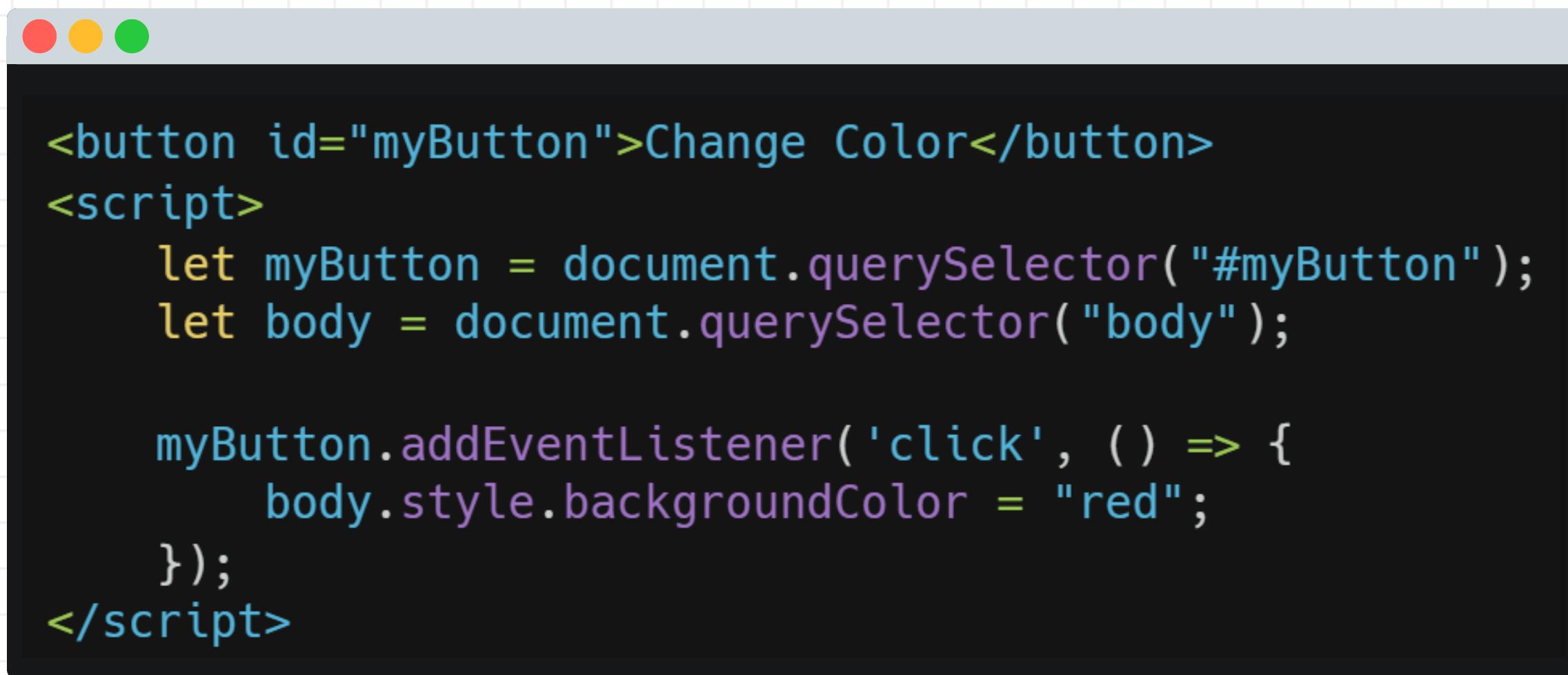
A screenshot of a window with a dark background and light gray header bar. The window contains the following code:

```
<div id="myDiv">Hello, World!</div>

<script>
  let div = document.querySelector("#myDiv");
  div.style.color = "blue";
</script>
```

addEventListener

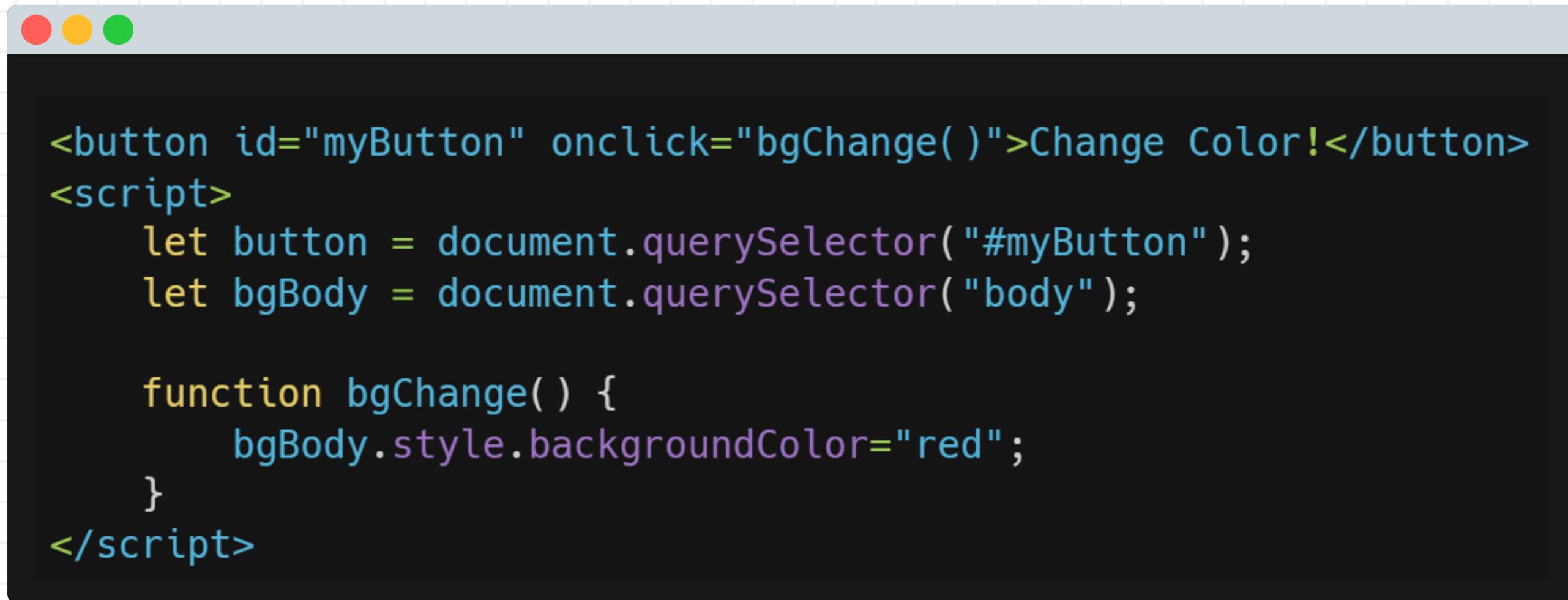
- เราสามารถทำให้การกำหนด動作ไปกับ element ใน HTML ส่งผลต่อการเปลี่ยนแปลงภายใน website ของเรา เช่น การ **click**



```
<button id="myButton">Change Color</button>
<script>
    let myButton = document.querySelector("#myButton");
    let body = document.querySelector("body");

    myButton.addEventListener('click', () => {
        body.style.backgroundColor = "red";
    });
</script>
```

onclick

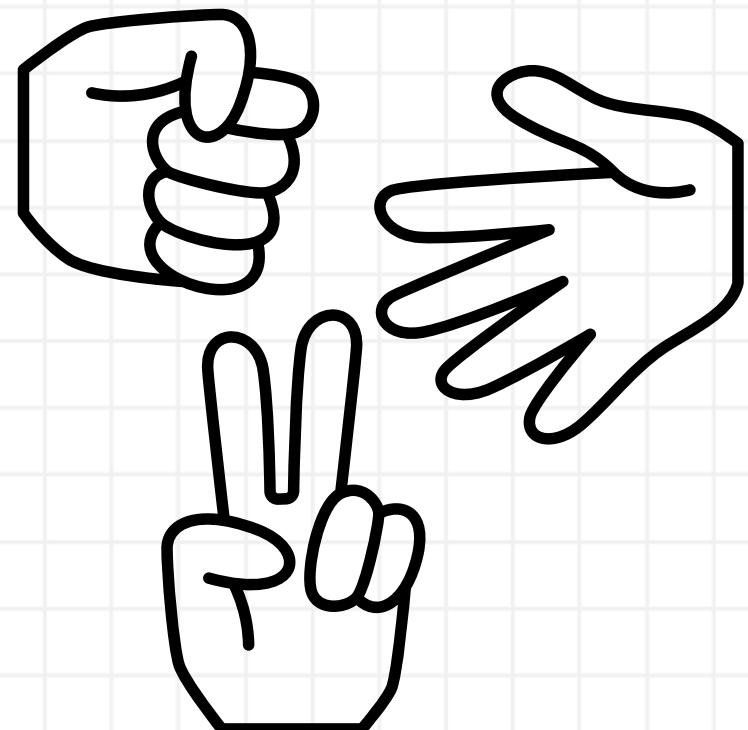


Mini Project #2

Utilize JS with our website (60 minutes)

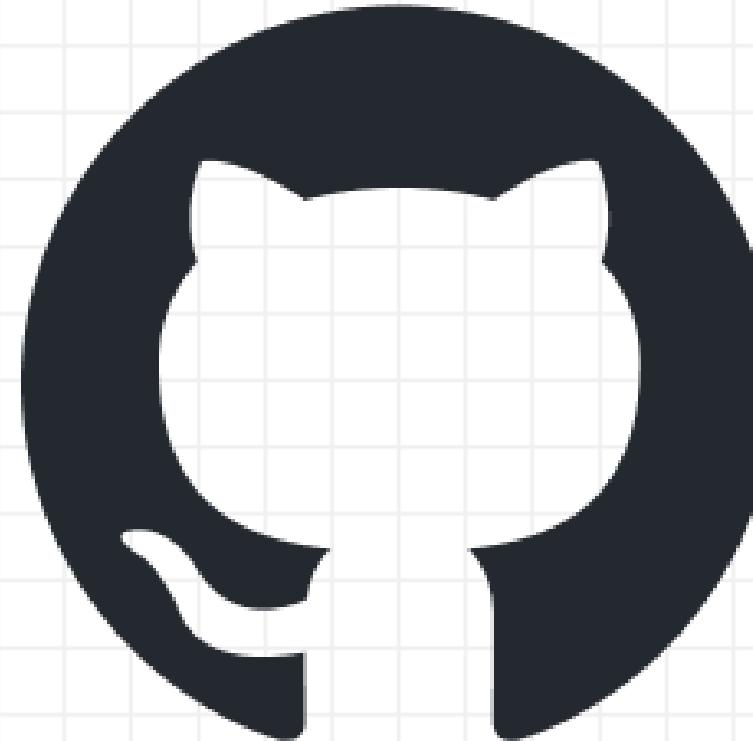
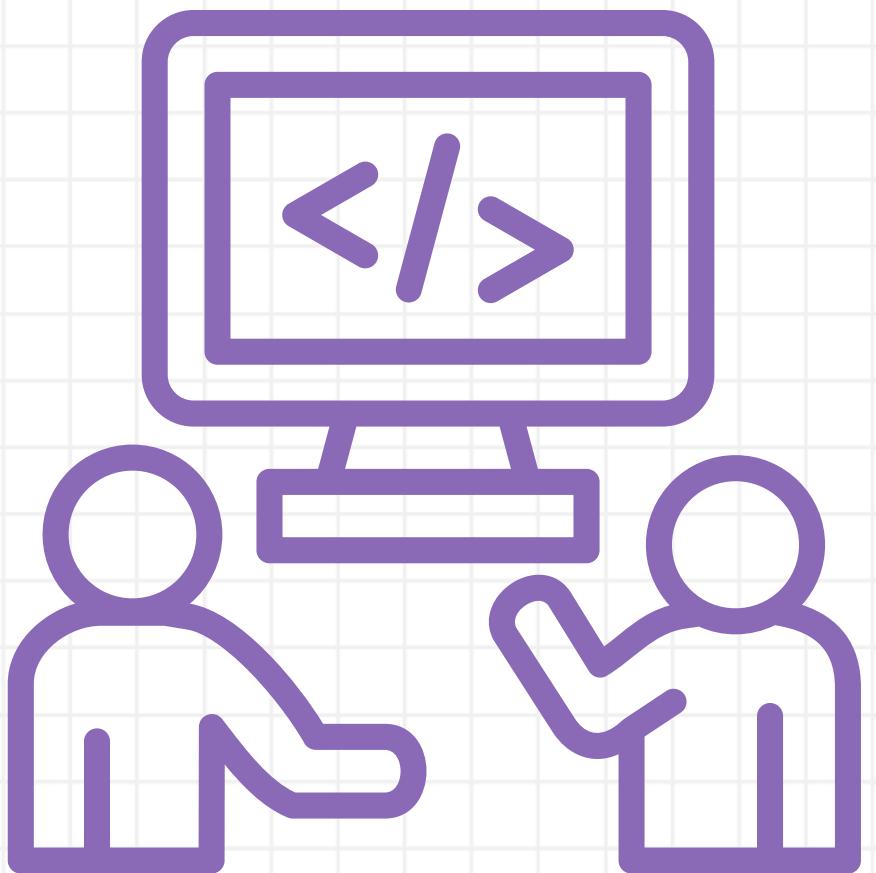
Objectives

- ทำเว็บไซต์เกมเป่ายิ่งจุบ
- คนเล่นสามารถเลือกปุ่มได้ คือ ค้อน
 กรรไกร และ กระดาษ
- มีการแสดงข้อความผลแพ้-ชนะ



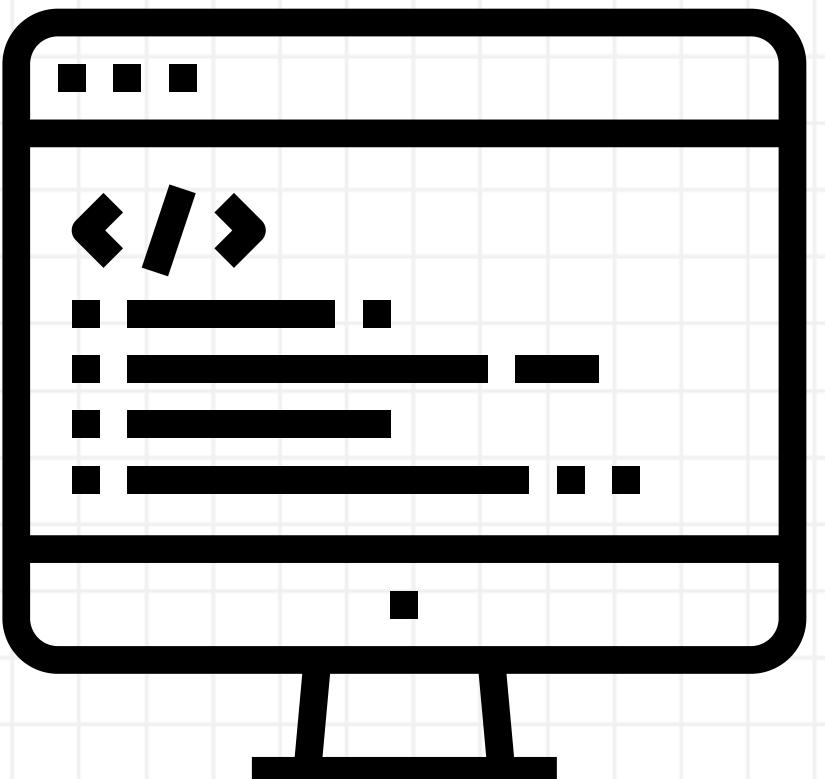
What is GitHub ?

- GitHub เป็นเครื่องมือที่ใช้ในการจัดเก็บและจัดการโค้ดผ่านแพลตฟอร์มออนไลน์ เพื่อเอาไว้ใช้ในการทำงานร่วมกับภายในทีม
- การใช้ GitHub ทำให้เราสามารถควบคุมเวอร์ชันของโค้ดเราได้ว่าอย่างใดเวอร์ชันไหน

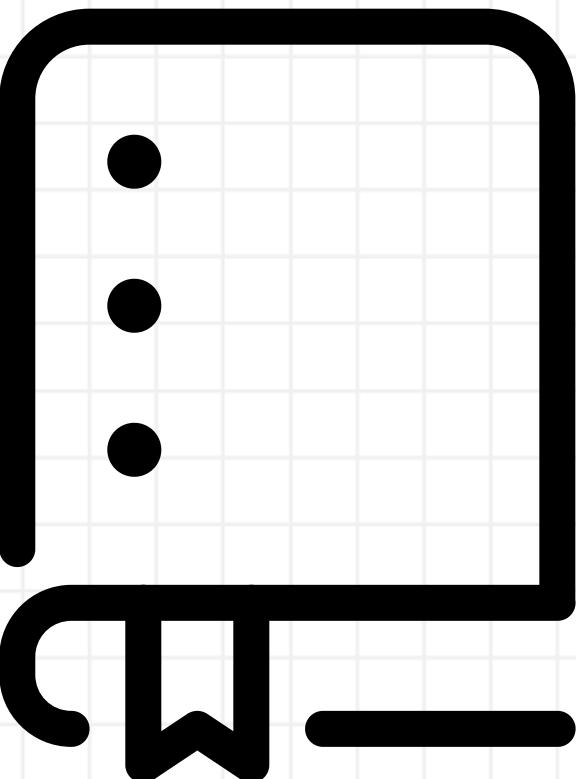


Repositories

- Repository คือพื้นที่ที่เราเอาไว้จัดเก็บโค้ดของเรา โดยเรามาระบุเปิดสาธารณะให้คนอื่นเห็น หรือ ทำเป็นแบบส่วนตัวก็ได้



/

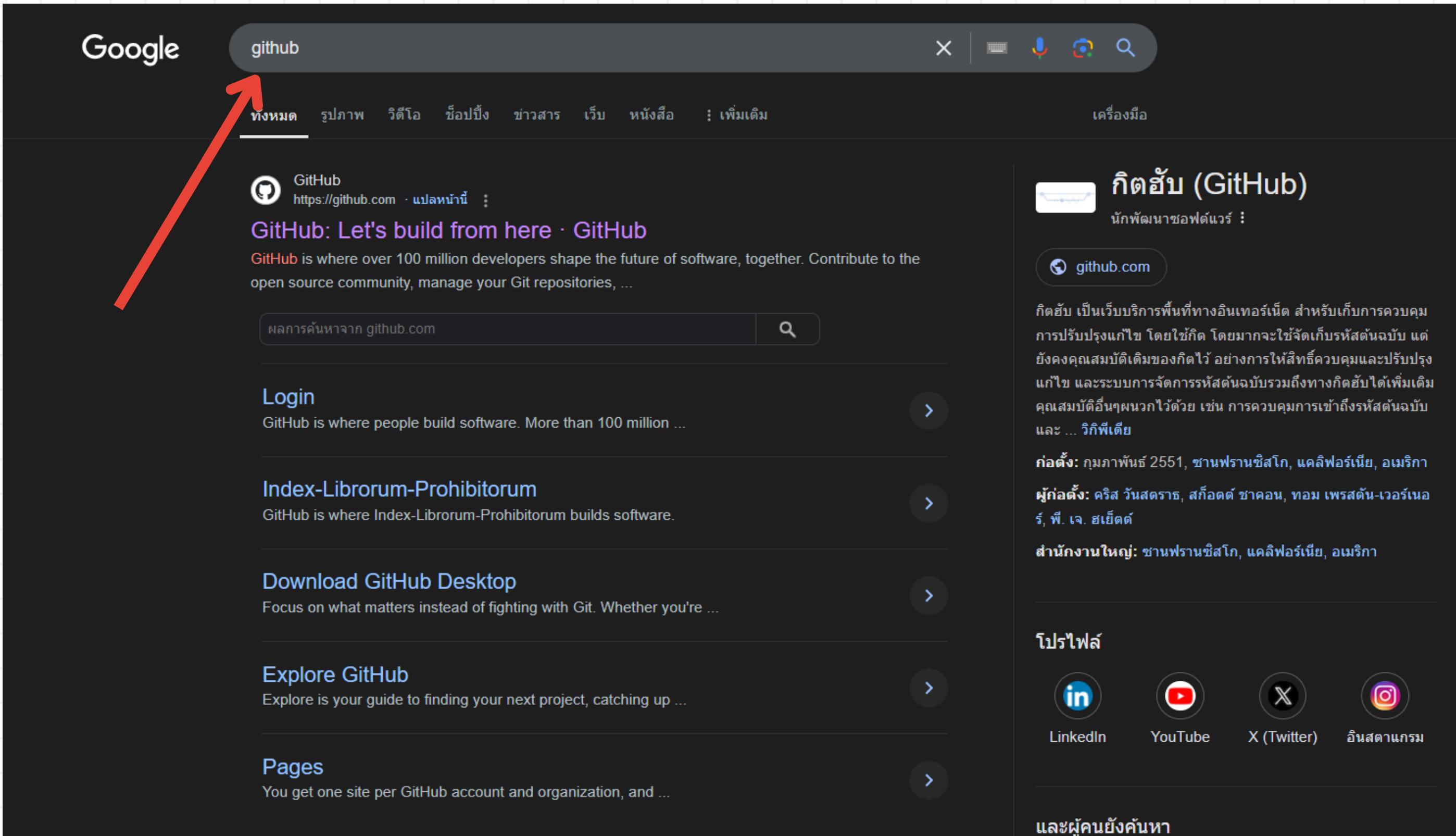


How to use Github ?

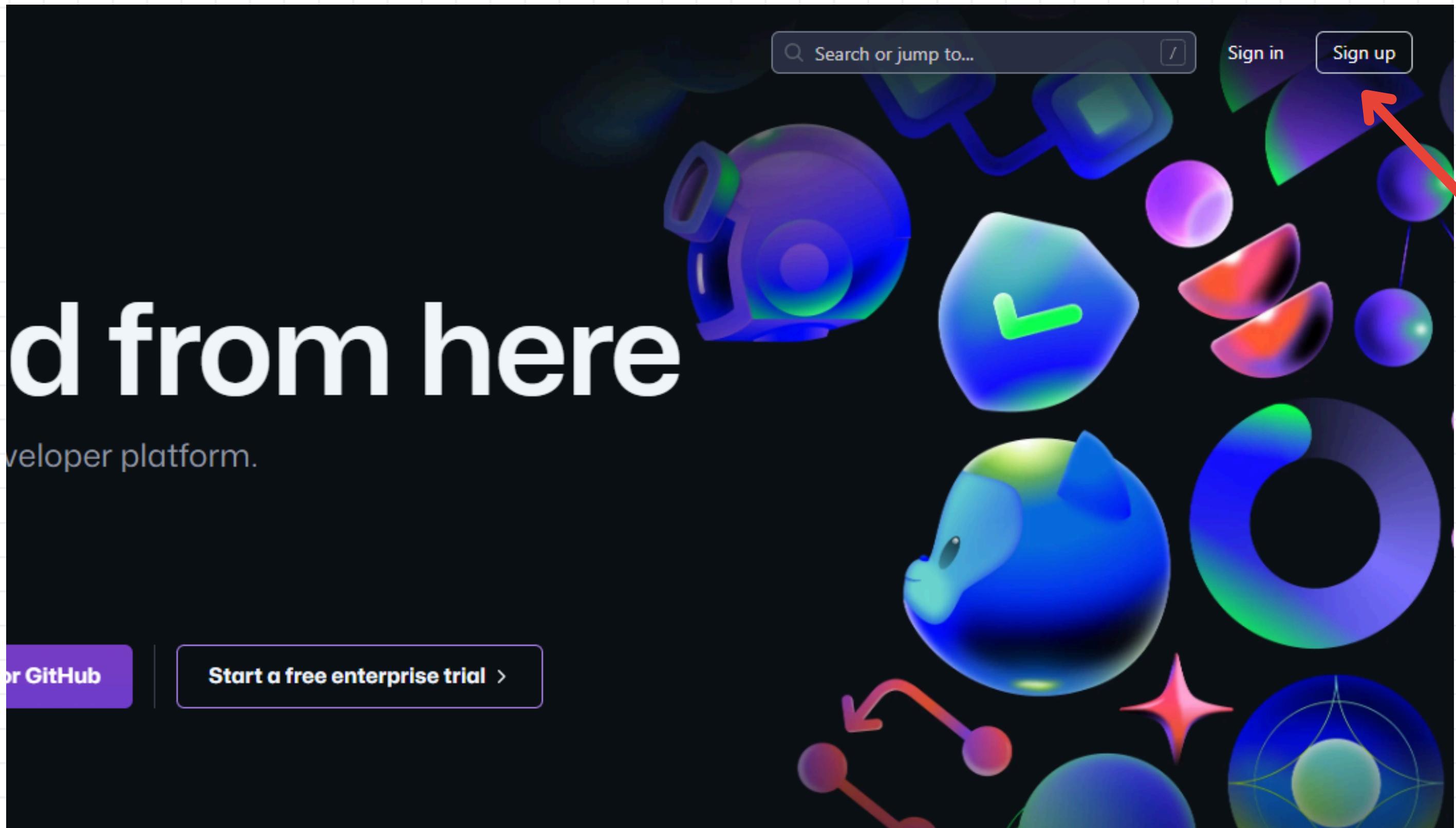
สร้าง repository และ upload ไฟล์



Step 1 : ให้น้อง ๆ search ในว่า “github”



Step 2 : เข้าไปในเว็บไซต์ GitHub และกดที่ sign up (หรือ sign in ถ้ามี account)

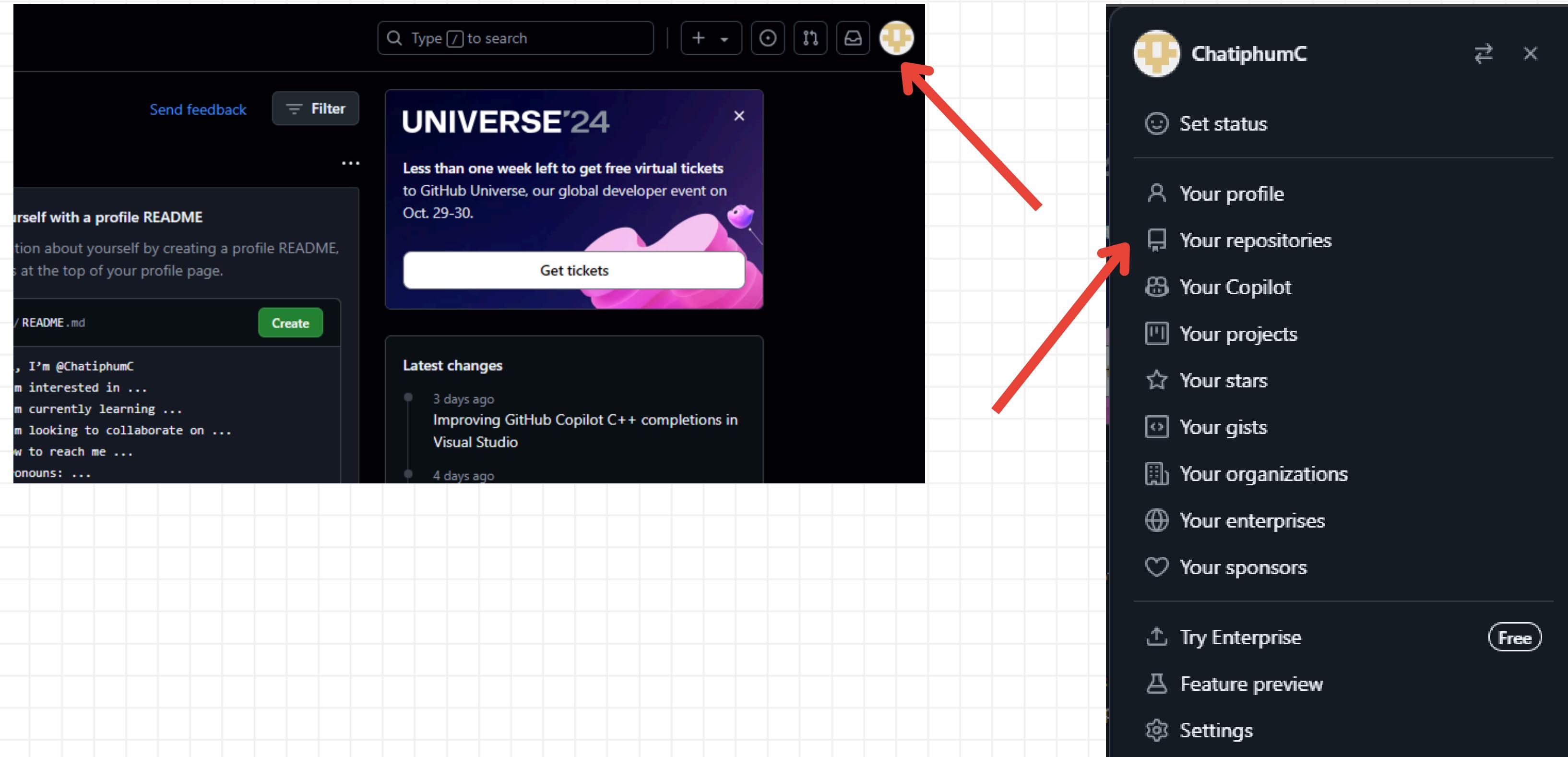


Step 3 : หลังจากเราสมัครเรียบร้อยแล้ว และเข้าเว็บไซต์ GitHub ก็จะขึ้นมาเป็นหน้านี้ 😊

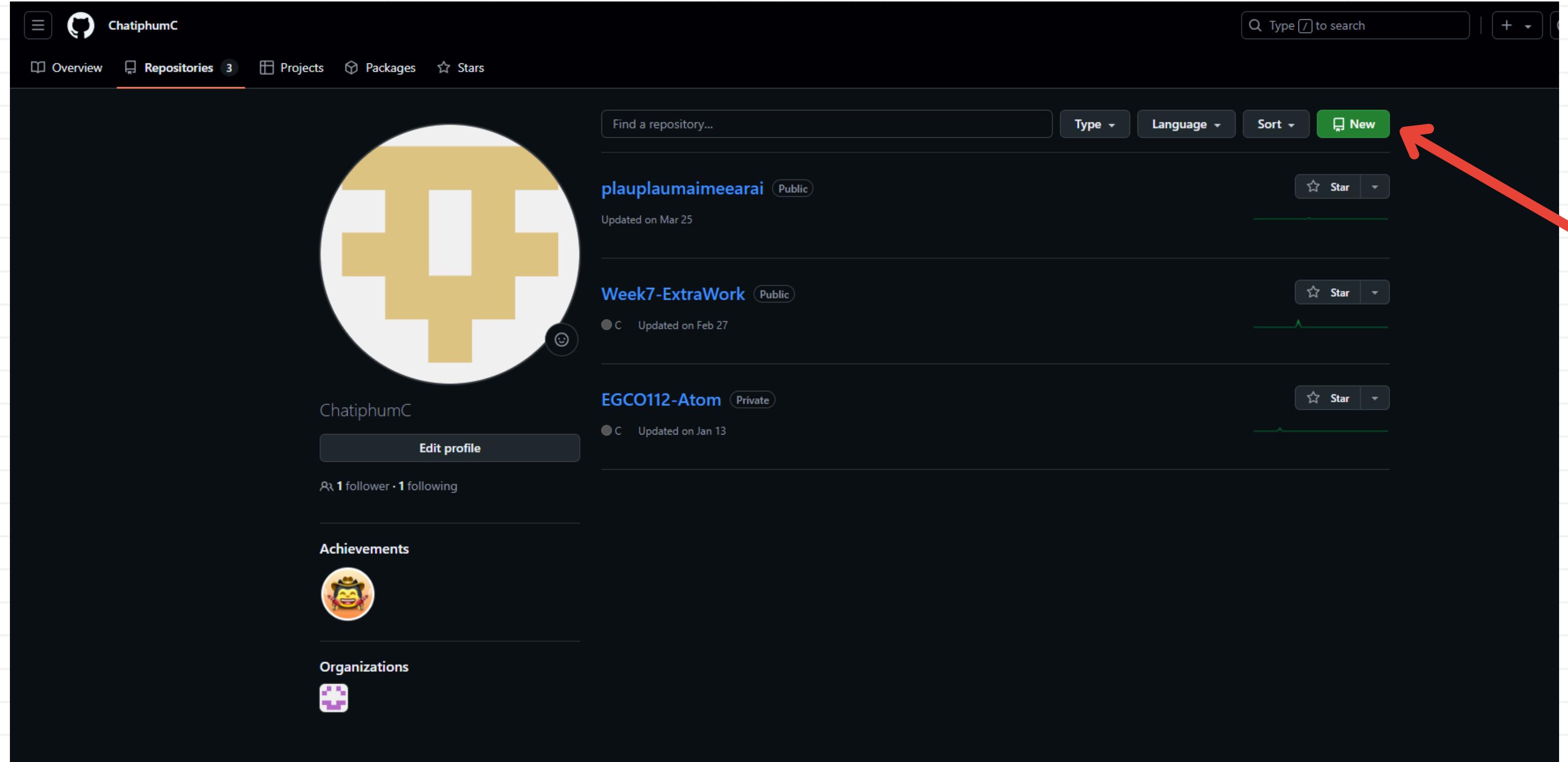
The screenshot shows the GitHub Home page with a dark theme. On the left sidebar, there's a list of 'Top repositories' belonging to the user 'ChatiphumC'. The main area features several cards:

- Start a new repository for ChatiphumC**: A card for creating a new repository. It includes a text input for the repository name, a radio button for 'Public' (selected) or 'Private', and a 'Create a new repository' button.
- Introduce yourself with a profile README**: A card for creating a profile README. It shows a preview of the file content, including a list of 8 items with icons like a person, a brain, and a lightning bolt.
- Latest changes**: A card listing recent GitHub improvements, such as 'Improving GitHub Copilot C++ completions in Visual Studio' and 'Secret scanning supports delegated bypass for push protection on file uploads (GA)'.
- Explore repositories**: A card showing two repository cards: 'cvxpy / cvxpy' and 'apache / jena'.
- Get AI-based coding suggestions**: A card about GitHub Copilot, stating it suggests entire functions in real time.
- Follow this exercise to try the GitHub flow**: A card about GitHub's 'Hello World' tutorial.
- UNIVERSE'24**: A promotional card for GitHub Universe 2024, encouraging users to get free virtual tickets.
- Use tools of the trade**: A card for writing code in a web browser using the github.dev editor.
- Get started on GitHub**: A video thumbnail showing a person using GitHub.

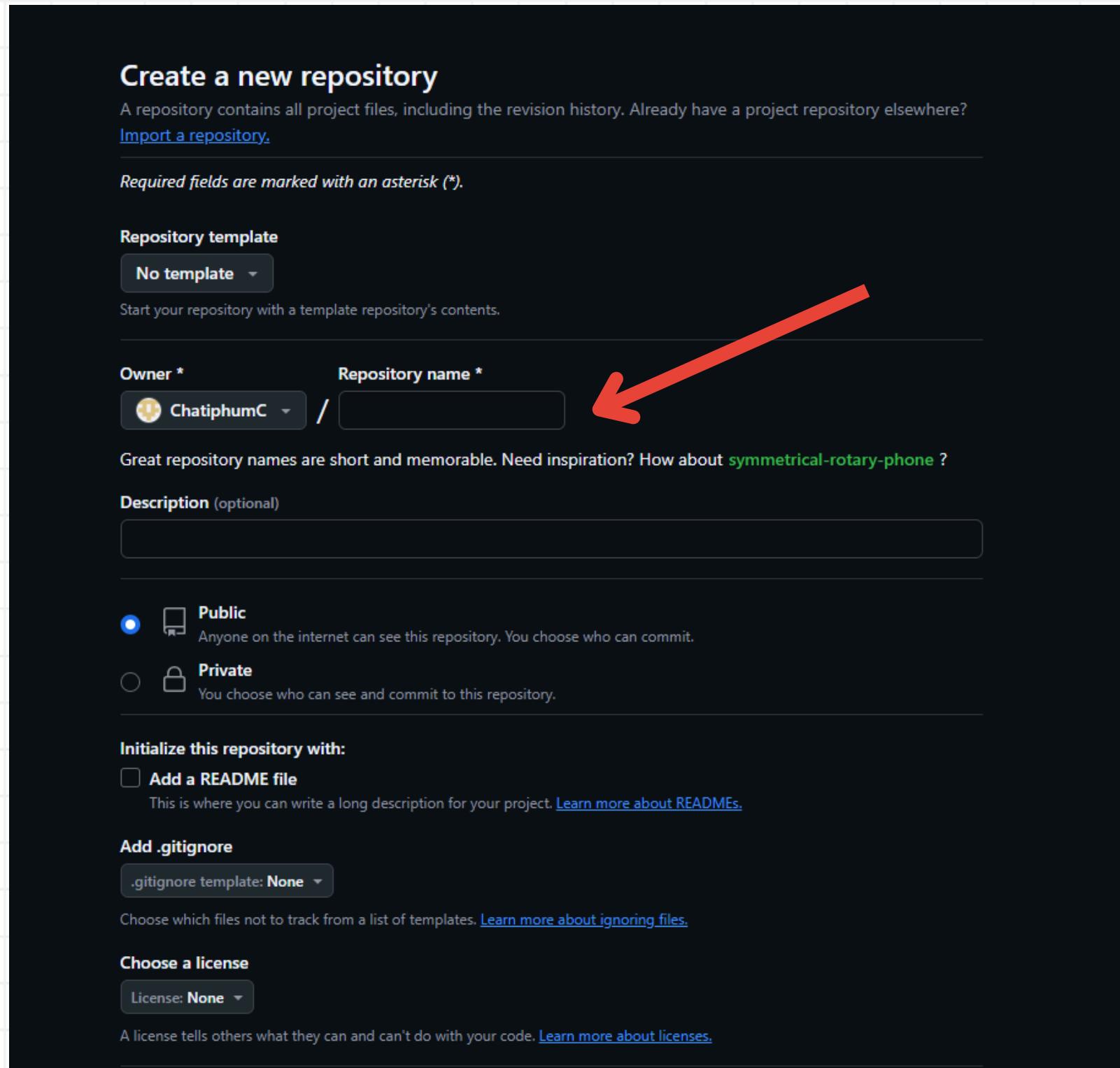
Step 4 : គេងចាកប៉ុន្តែង ឬ កត់ថ្មីបំនុរោយក្រុមហ៊ុន និង ផ្លូវការរបស់ខ្លួន



Step 5 : បង្កាប់ចំណាំក្នុងមាតិថត ដើម្បីក្រឡាកដបូន “new” ដើម្បីសរុប repository ໃម្ពឺ



Step 6 : เรายสามารถสร้าง repository ด้วยการระบุชื่อ “Repository name”



Step 7 : ເຮັດວຽກໃນຫຼັກ repo ຂອງເຮົາ ໃຫ້ບ້ອງ ທັງລອນກົດ “uploading an existing file”

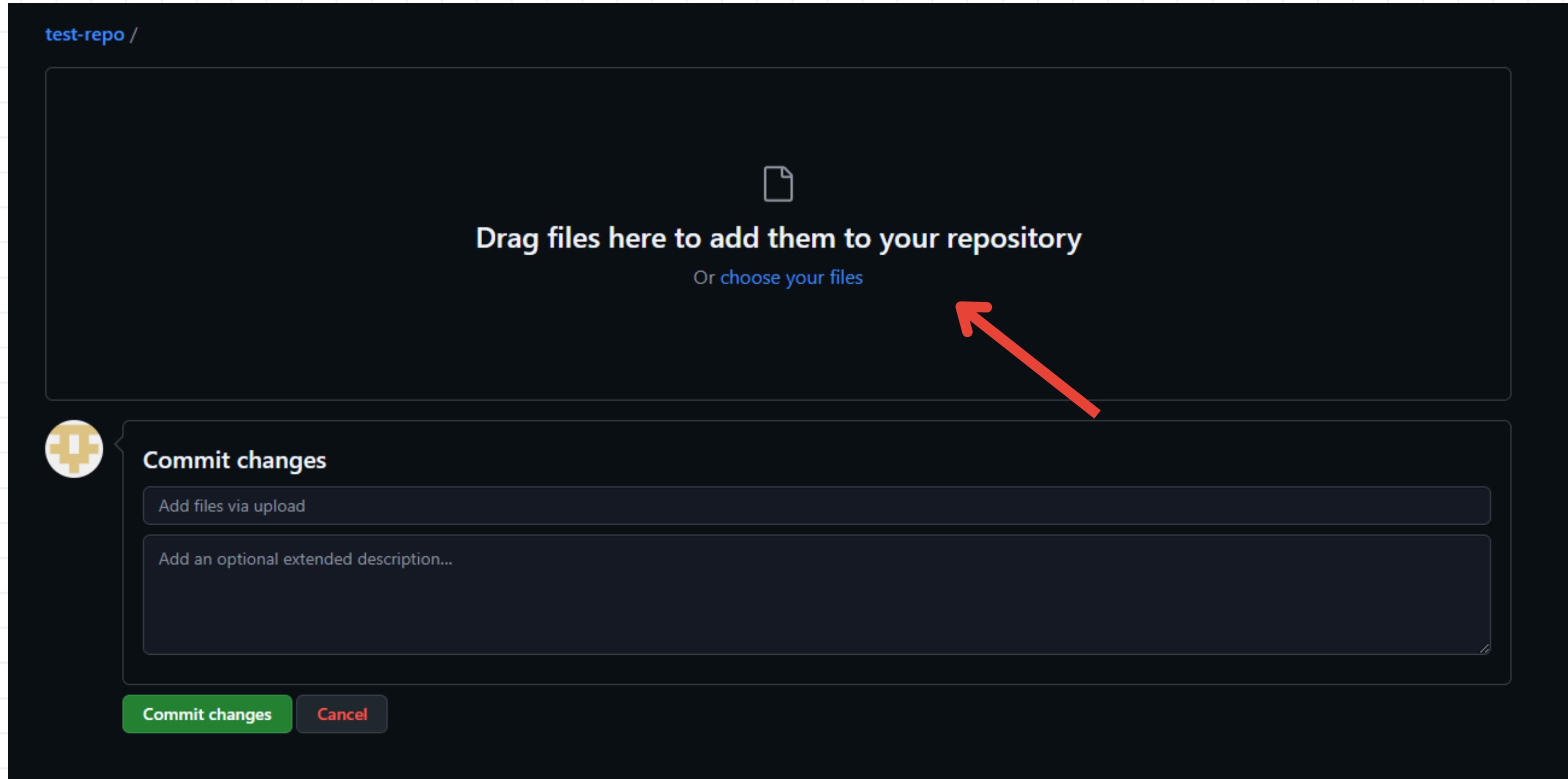
The screenshot shows a GitHub repository page for 'test-repo'. At the top, there's a 'Public' badge. Below it, there are buttons for 'Pin', 'Unwatch' (with 1 watch), 'Fork' (with 0 forks), and 'Star' (with 0 stars). On the left, there's a section for 'Set up GitHub Copilot' with a button to 'Get started with GitHub Copilot'. To the right, there's a section for 'Add collaborators' with a 'Invite collaborators' button. Below these sections, there's a 'Quick setup' section with options for 'Set up in Desktop' (selected), 'HTTPS', and 'SSH', and a URL 'https://github.com/ChatiphumC/test-repo.git'. A red arrow points from the text 'uploading an existing file' in the 'Quick setup' section towards the bottom of the page.

Quick setup — if you've done this kind of thing before

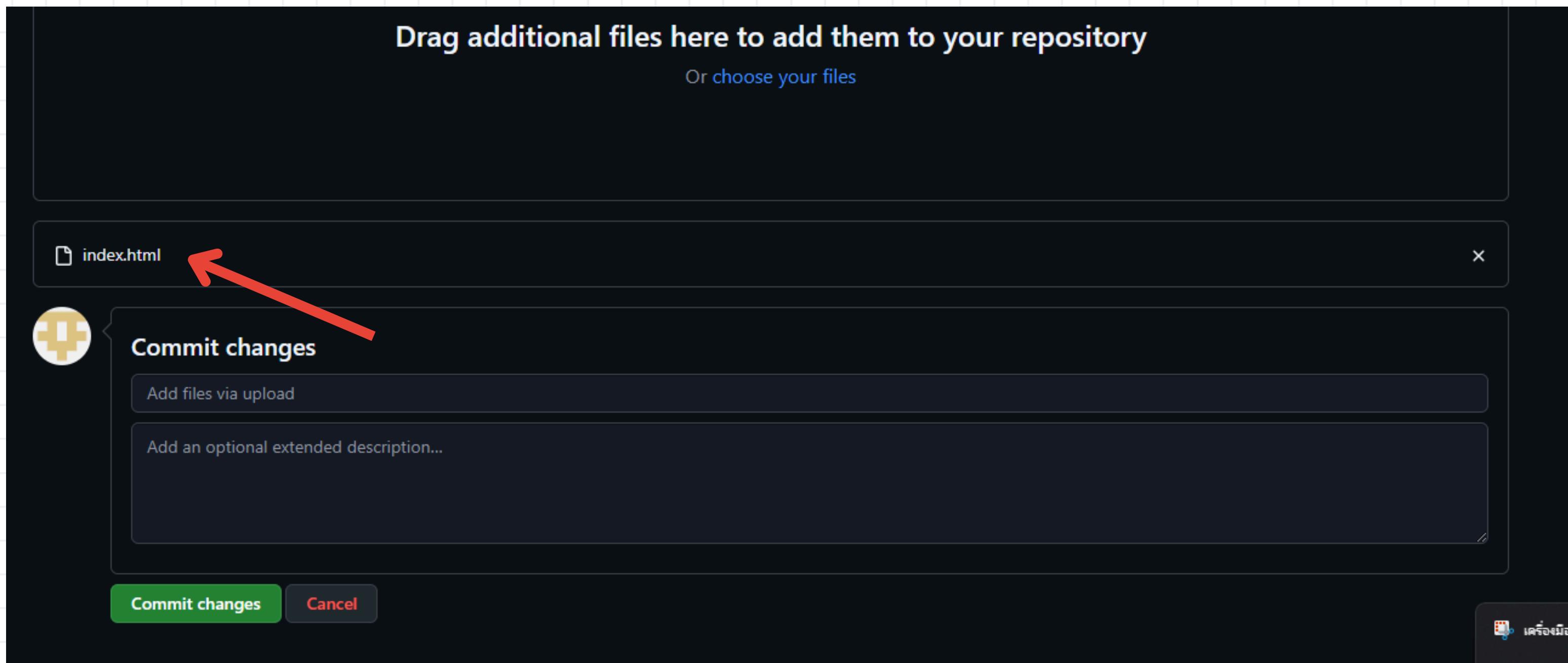
Set up in Desktop or HTTPS <https://github.com/ChatiphumC/test-repo.git>

Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

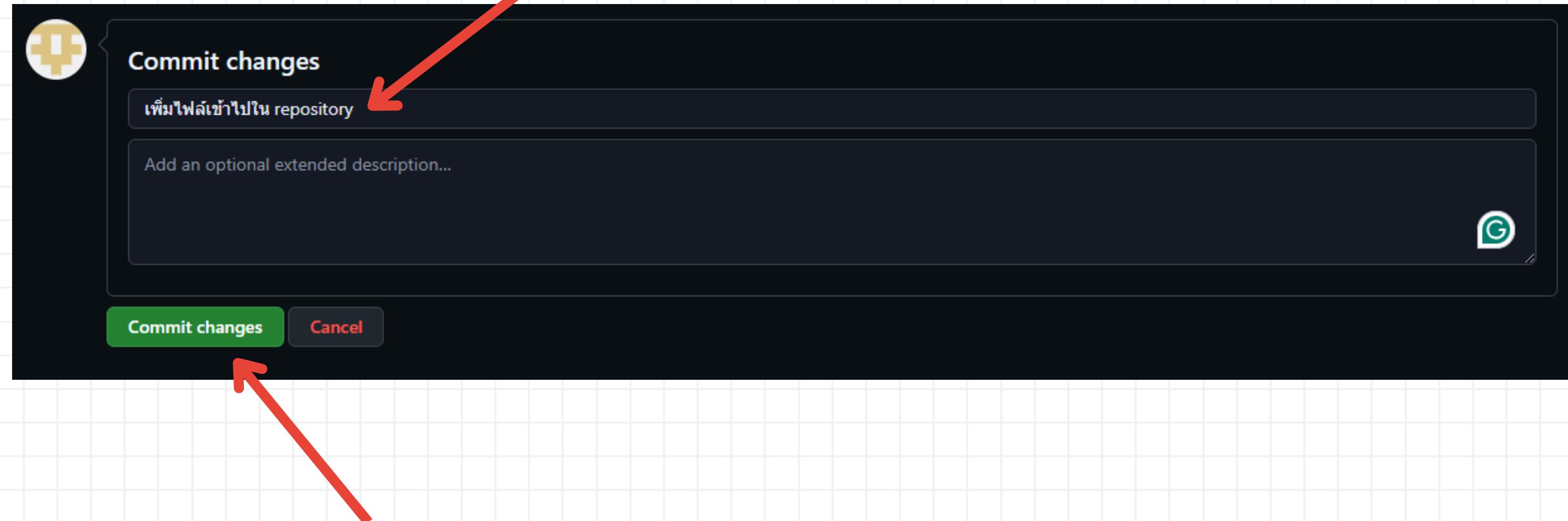
Step 8 : ให้บันทุณ ๆ ลากไฟล์เข้ามาในกล่อง (หรือกดเลือกไฟล์ก็ได้)



Step 9 : ไฟล์ที่เราเลือกจะปรากฏ



Step 10 : เราสามารถใส่ข้อความแบบกับการเอาไฟล์เข้า repo ได้ จากนั้นทำการกดปุ่ม “commit” เพื่อส่งไฟล์เข้า repo ได้



Step 11 : หลังจากเราทำการ commit ไปแล้ว ใน repo เราจะจะแสดงไฟล์ที่เรา upload ขึ้นไป

The screenshot shows a GitHub repository page for a public repository named 'test-repo'. The repository has 1 branch and 0 tags. A recent commit was made by 'ChatiphumC' 3 minutes ago, adding an 'index.html' file. The commit message is 'เพิ่มไฟล์เข้าไปใน repository'. The repository has 1 commit, 0 forks, 0 stars, and 1 person watching it. There are sections for Activity, Releases, and Packages, all of which are currently empty.

test-repo Public

main 1 Branch Tags

Go to file Add file Code

About

No description, website, or topics provided.

Activity

0 stars

1 watching

0 forks

Releases

No releases published

Create a new release

Packages

No packages published

Publish your first package

ChatiphumC เพิ่มไฟล์เข้าไปใน repository

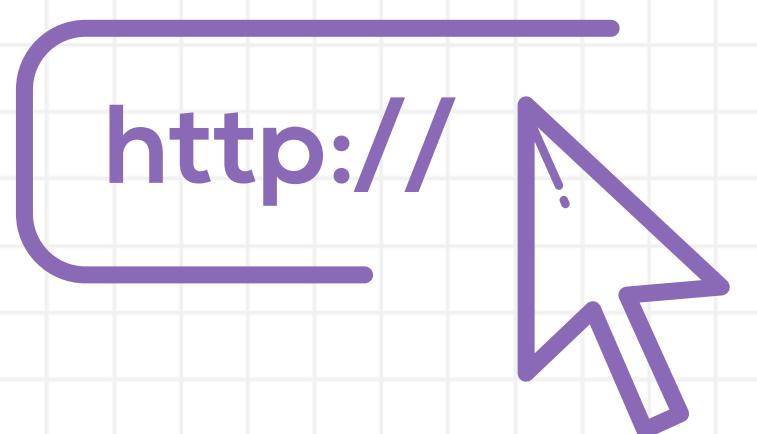
index.html เพิ่มไฟล์เข้าไปใน repository 3 minutes ago

README

Add a README

Add a README

Host a website on Github



GitHub Pages

- GitHub Pages เป็น service ของ GitHub ที่ทำให้เราสามารถ host เว็บไซต์ผ่าน repository ของเราได้ 😍
- GitHub Pages ไม่มีค่าใช้จ่าย อย่าง host เว็บก็ host ได้เลยย
- host ได้ 1 เว็บไซต์ (Personal website) ต่อ 1 account GitHub ของเรา เท่านั้น



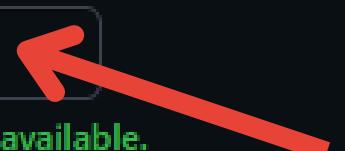
Step 1 : ให้บ้อง ๆ สร้าง repo ขึ้นมา โดยกำหนดชื่อให้มี “username.github.io” ตามหลัง

Create a new repository
A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Required fields are marked with an asterisk ().*

Repository template
No template ▾

Start your repository with a template repository's contents.

Owner * ChatiphumC / **Repository name *** ChatiphumC.github.io 

ChatiphumC.github.io is available.

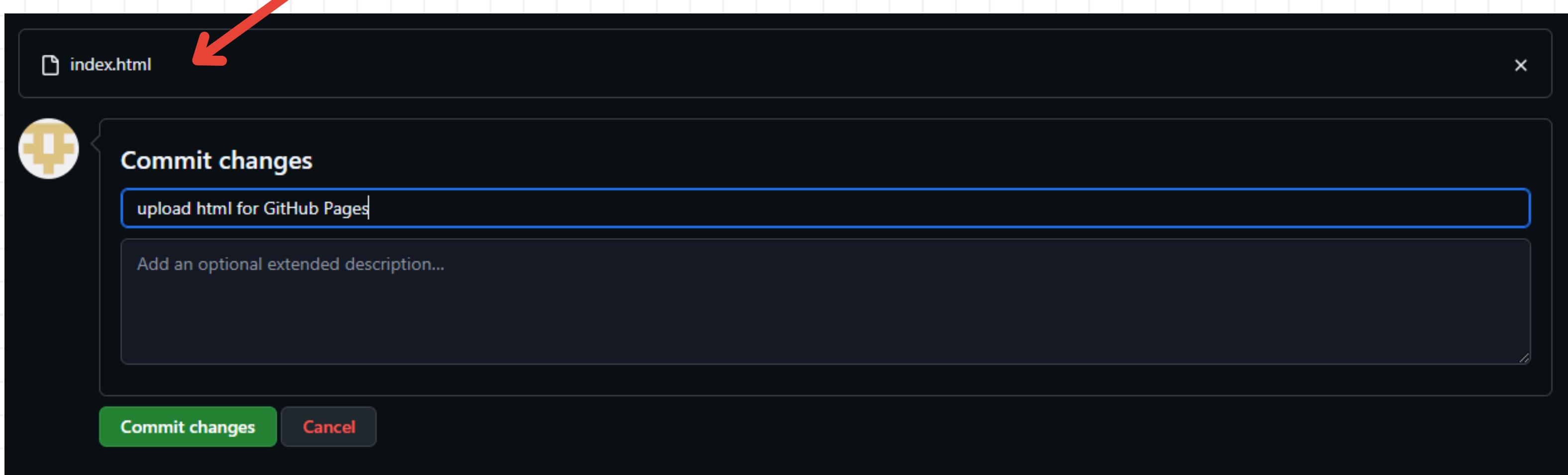
Great repository names are short and memorable. Need inspiration? How about [congenial-disco](#) ?

Description (optional)

Public
Anyone on the internet can see this repository. You choose who can commit.

ใช้เป็น username ของ account GitHub เรานะ

Step 2 : จากนั้น upload ไฟล์ index.html ที่เราต้องการขึ้นไปบน repository



Step 3 : สุดท้ายเปิดดูตามลิ้งค์ <https://username.github.io>



This is a Website Host with GitHub Pages 🙏

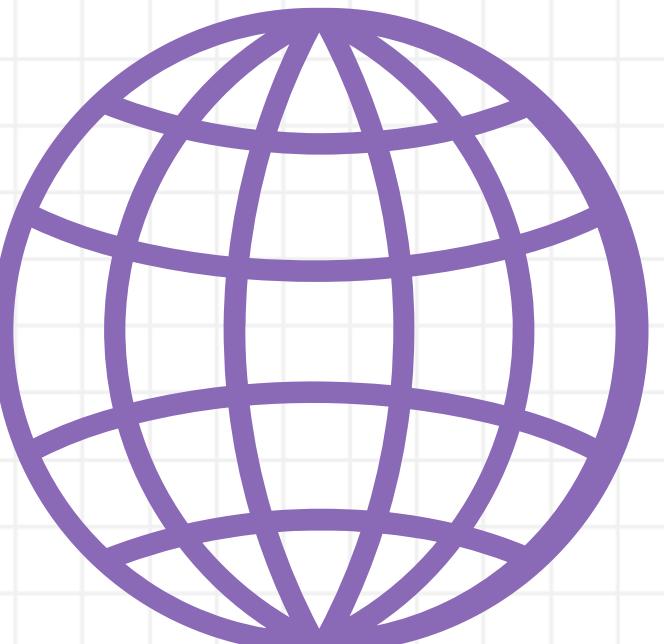
ตัวอย่างเว็บไซต์ของพี่ : <https://chatiphumc.github.io>

Mini Project #3

Host your website (GitHub Pages) (60 minutes)

Objectives

- สร้าง website portfolio ต่อจาก website about me
- ตกแต่งให้สวยงาม
- Host ด้วย Github Pages



A wide-angle photograph of a tropical sunset. The sky is filled with large, billowing clouds colored in shades of orange, pink, and purple. The sun is low on the horizon, casting a warm glow over the ocean. In the foreground, a sandy beach curves along the left side, with the ocean waves crashing onto it. A single palm tree is visible on the far left edge.

fin