

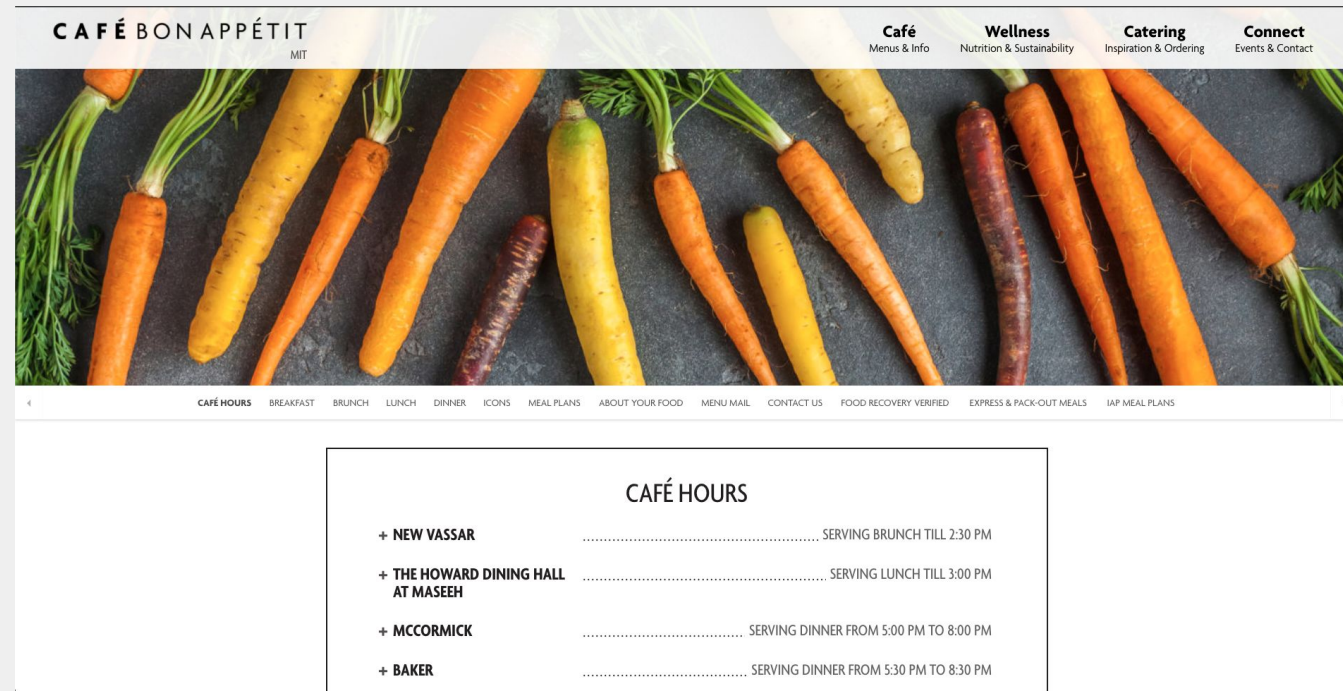
Intro to HTML/CSS



Mark Tabor
Andy Jiang

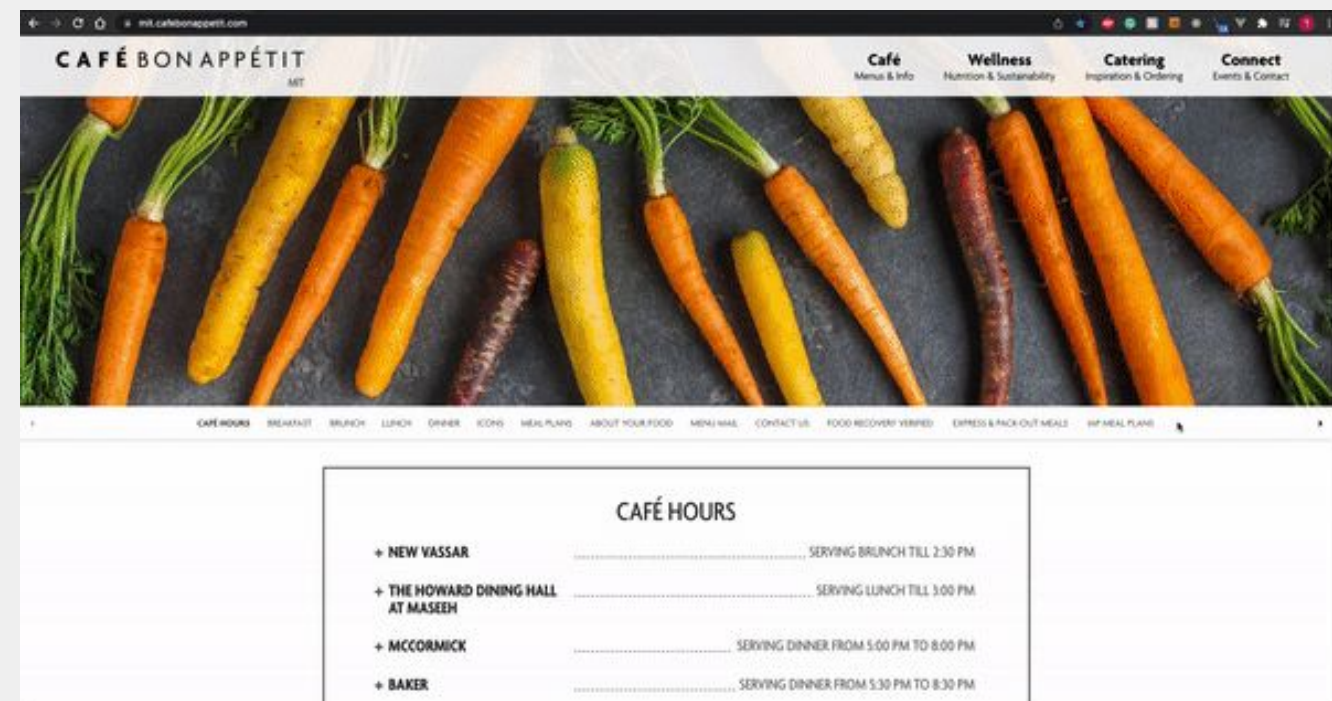
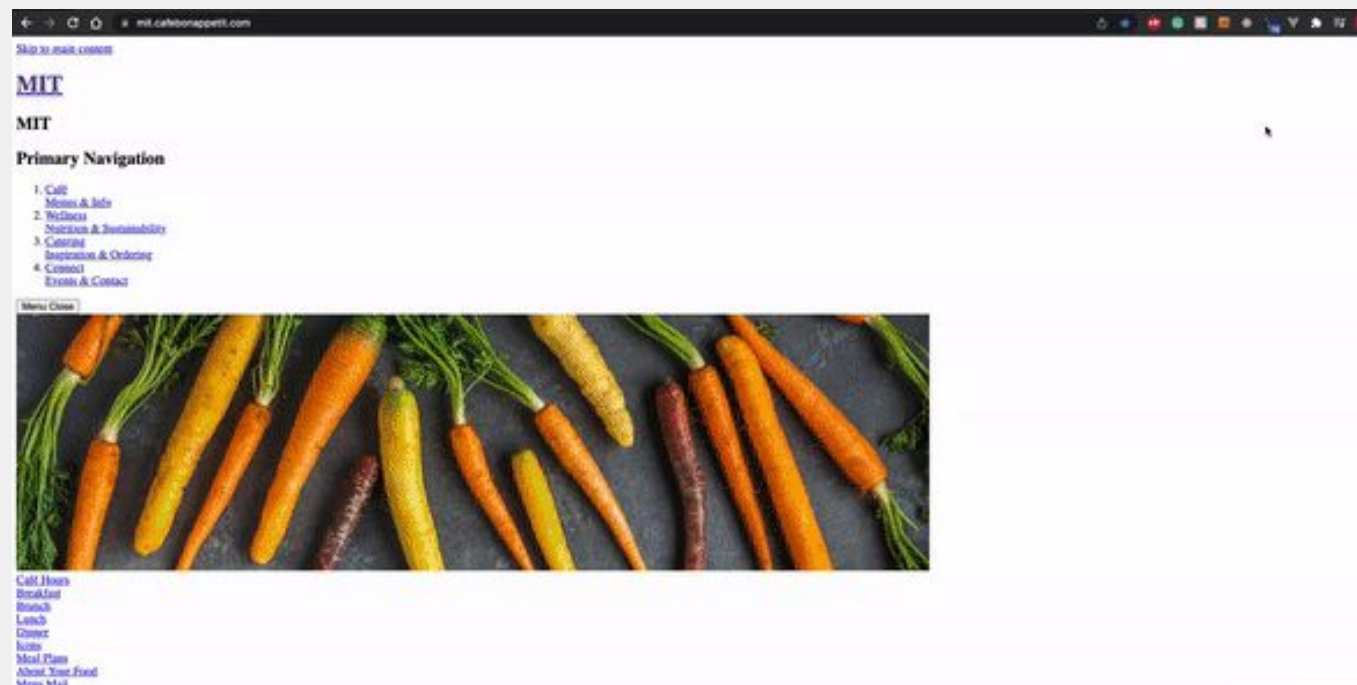


Why does this matter??



html

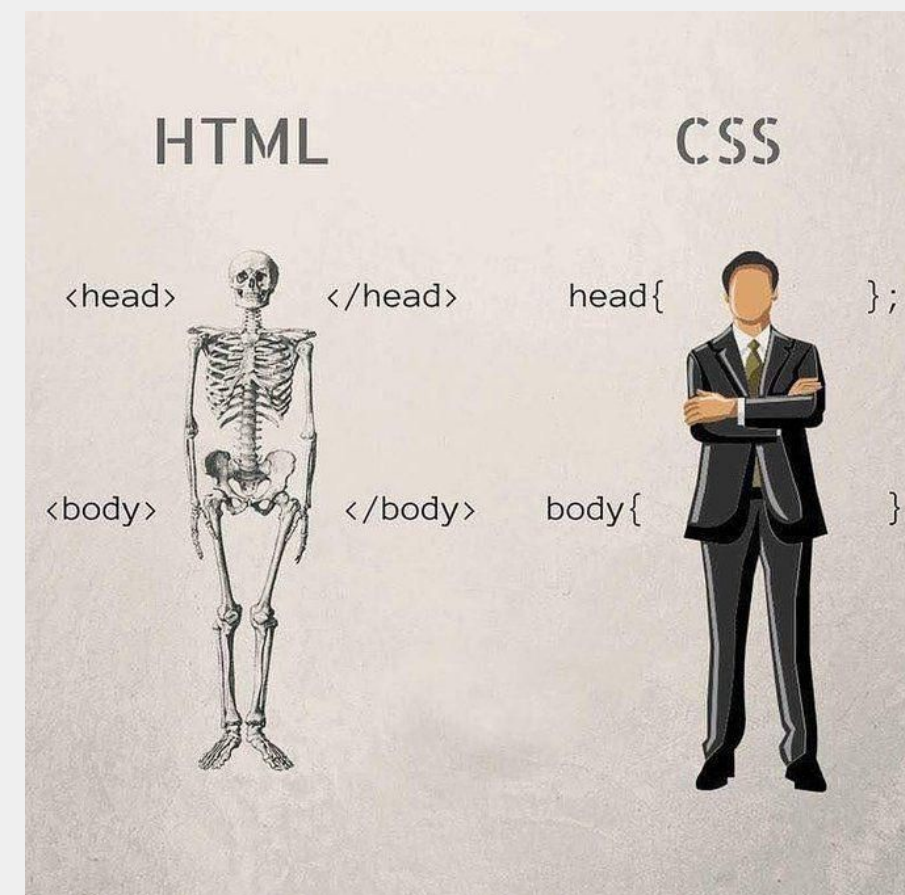
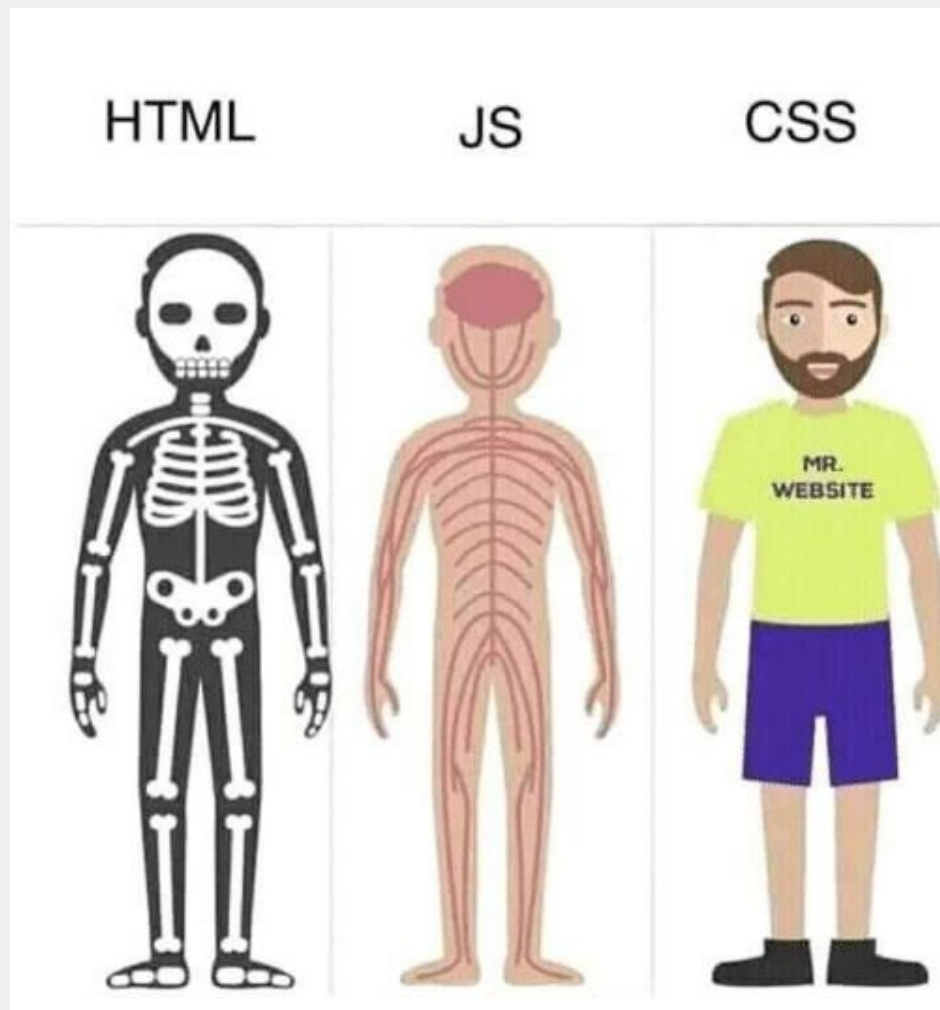
CSS



HTML

Hypertext Markup Language

The language your web browser uses to describe the content and structure of web pages



HTML = Nested Boxes

ARCHIVE
WHAT IF?
BLAG
STORE
ABOUT



A *WEB*COMIC OF ROMANCE,
SARCASM, MATH, AND LANGUAGE.

THING EXPLAINER IS AVAILABLE AT: [AMAZON](#), [BARNES & NOBLE](#), [INDIE BOUND](#),
[HUDSON](#)

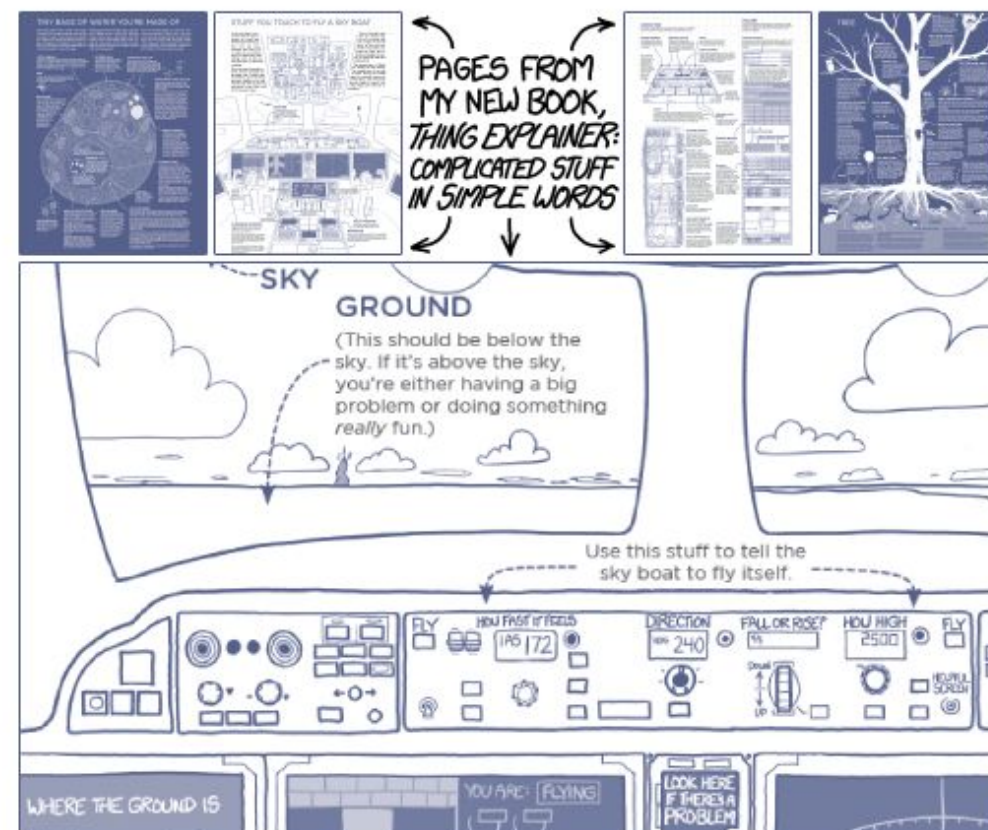
TAGS

|< < PREV RANDOM NEXT > >|

<DIV>Q: HOW DO YOU ANNOY A WEB DEVELOPER?

|< < PREV RANDOM NEXT > >|

PERMANENT LINK TO THIS COMIC: [HTTP://XKCD.COM/1144/](http://xkcd.com/1144/)
IMAGE URL (FOR HOTLINKING/EMBEDDING): [HTTP://IMGS.XKCD.COM/COMICS/TAGS.PNG](http://imgs.xkcd.com/comics/tags.png)



ARCHIVE
WHAT IF?
BLAG
STORE
ABOUT



A *WEBCOMIC OF ROMANCE,*
SARCASM, MATH, AND LANGUAGE.

THING EXPLAINER IS AVAILABLE AT: [AMAZON](#), [BARNES & NOBLE](#), [INDIE BOUND](#),
[HUDSON](#)

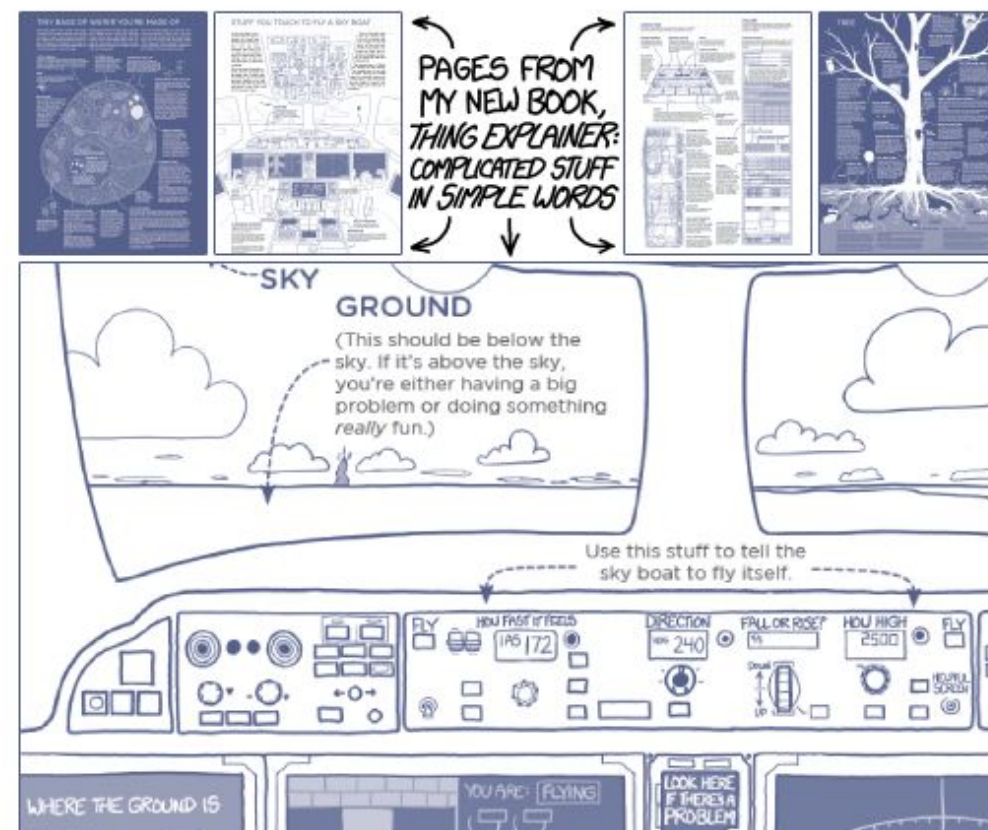
TAGS

|< < PREV RANDOM NEXT > >|

<DIV>Q: HOW DO YOU ANNOY A WEB DEVELOPER?

|< < PREV RANDOM NEXT > >|

PERMANENT LINK TO THIS COMIC: [HTTP://XKCD.COM/1144/](http://xkcd.com/1144/)
IMAGE URL (FOR HOTLINKING/EMBEDDING): [HTTP://IMGS.XKCD.COM/COMICS/TAGS.PNG](http://imgs.xkcd.com/comics/tags.png)



ARCHIVE
WHAT IF?
BLAG
STORE
ABOUT



A *WEBCOMIC OF ROMANCE,
SARCASM, MATH, AND LANGUAGE.*

THING EXPLAINER IS AVAILABLE AT: [AMAZON](#), [BARNES & NOBLE](#), [INDIE BOUND](#),
[HUDSON](#)

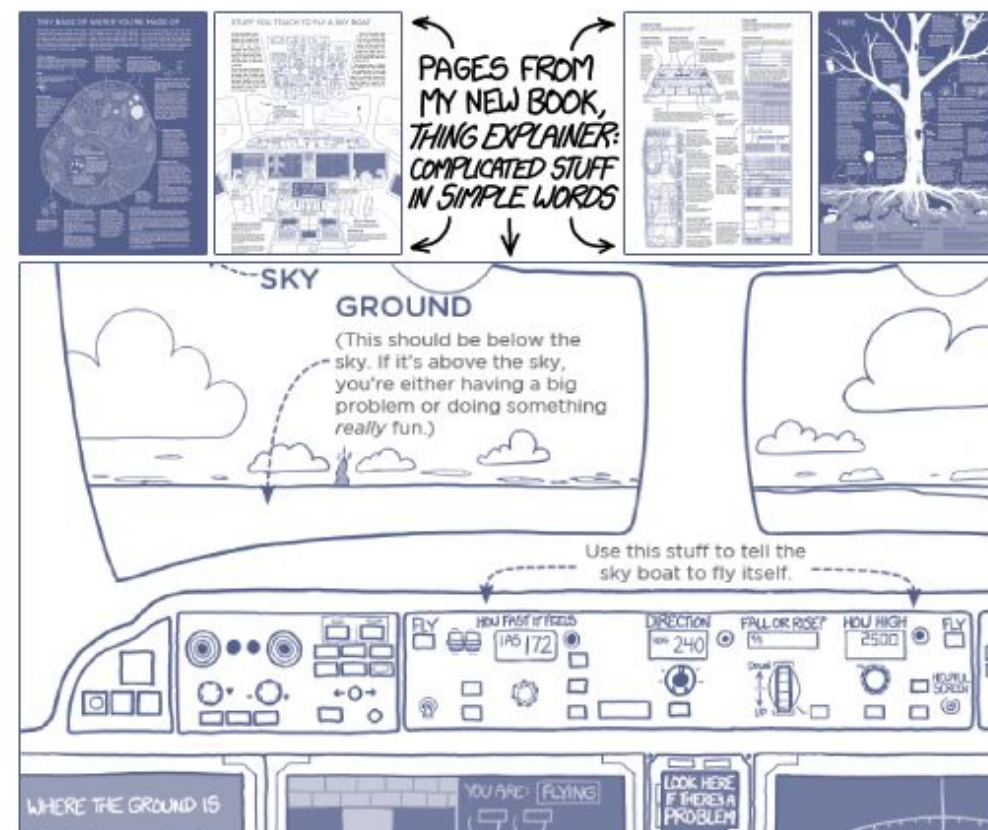
TAGS

|< < PREV RANDOM NEXT > >|

<DIV>Q: HOW DO YOU ANNOY A WEB DEVELOPER?

|< < PREV RANDOM NEXT > >|

PERMANENT LINK TO THIS COMIC: [HTTP://XKCD.COM/1144/](http://xkcd.com/1144/)
IMAGE URL (FOR HOTLINKING/EMBEDDING): [HTTP://IMGS.XKCD.COM/COMICS/TAGS.PNG](http://imgs.xkcd.com/comics/tags.png)



ARCHIVE
WHAT IF?
BLAG
STORE
ABOUT



A *WEB*COMIC OF ROMANCE,
SARCASM, MATH, AND LANGUAGE.

THING EXPLAINER IS AVAILABLE AT: [AMAZON](#), [BARNES & NOBLE](#), [INDIE BOUND](#),
[HUDSON](#)

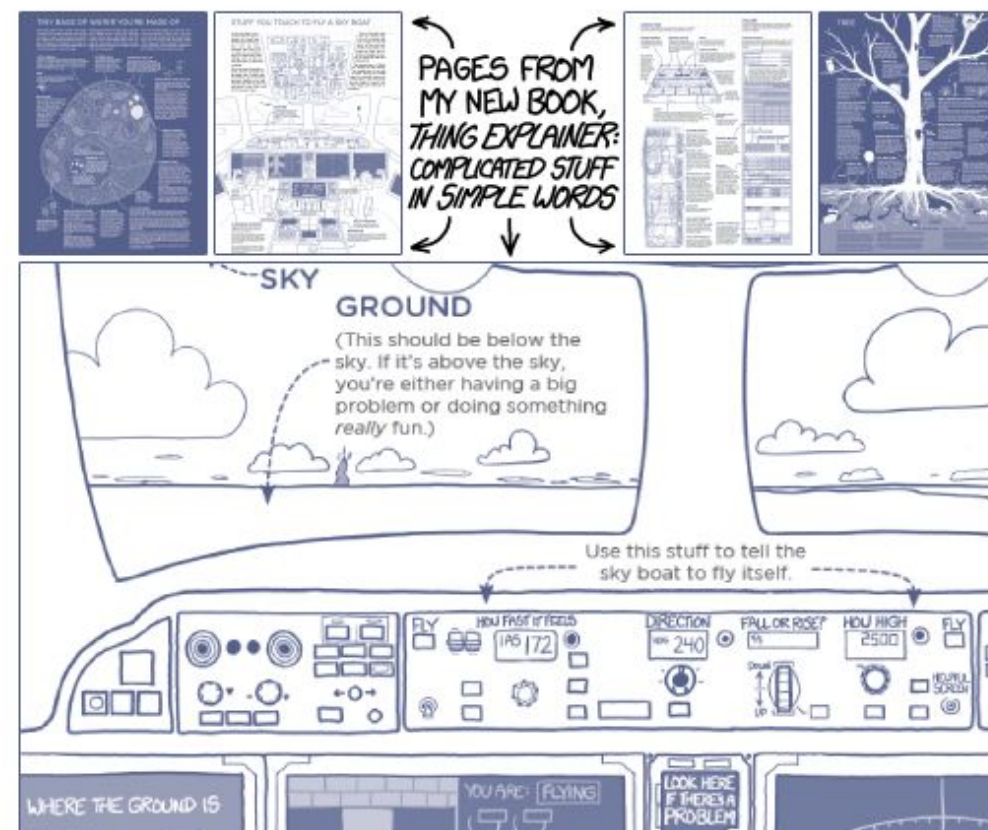
TAGS

|< < PREV RANDOM NEXT > |>

<DIV>Q: HOW DO YOU ANNOY A WEB DEVELOPER?

|< < PREV RANDOM NEXT > |>

PERMANENT LINK TO THIS COMIC: [HTTP://XKCD.COM/1144/](http://xkcd.com/1144/)
IMAGE URL (FOR HOTLINKING/EMBEDDING): [HTTP://IMGS.XKCD.COM/COMICS/TAGS.PNG](http://imgs.xkcd.com/comics/tags.png)



ARCHIVE

WHAT IF?

BLAG

STORE

ABOUT



**A WEBCOMIC OF ROMANCE,
SARCASM, MATH, AND LANGUAGE.**

THING EXPLAINER IS AVAILABLE AT: [AMAZON](#), [BARNES & NOBLE](#), [INDIE BOUND](#), [HUDSON](#)

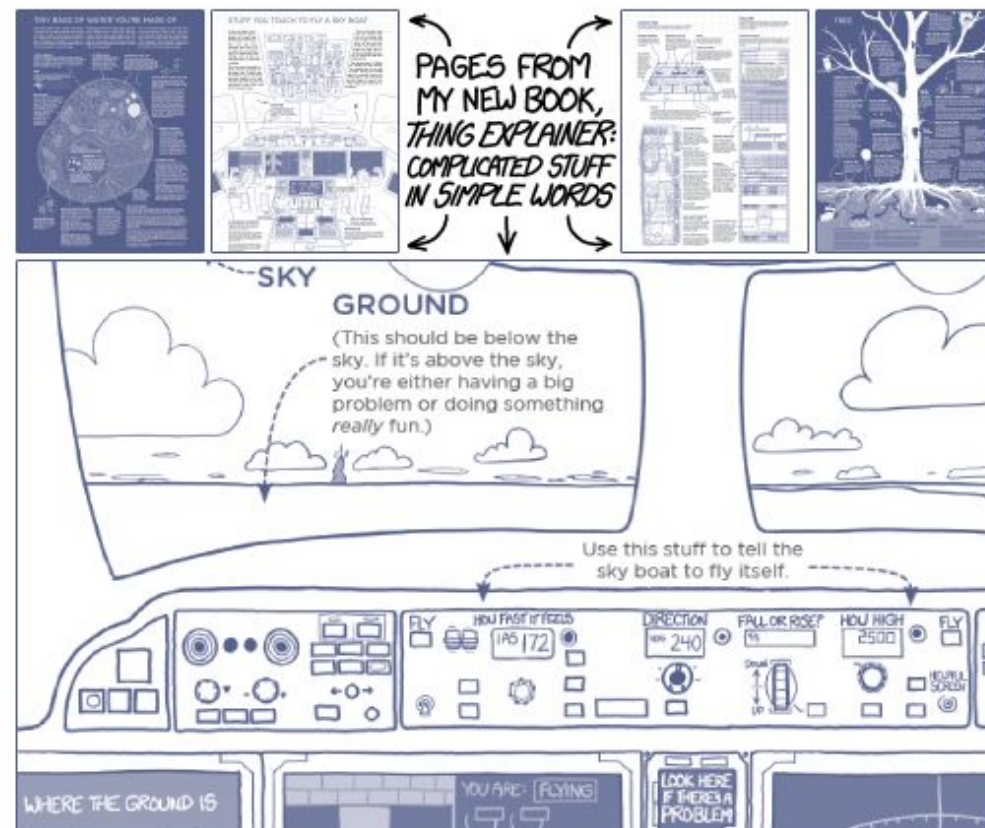
TAGS

[|<](#)
[< P_{REV}](#)
[R_{ANDOM}](#)
[N_{EXT} >](#)
[>|](#)

<DIV> Q: HOW DO YOU ANNOY A WEB DEVELOPER?

[|<](#)
[< PREV](#)
[RANDOM](#)
[NEXT >](#)
[>|](#)

PERMANENT LINK TO THIS COMIC: [HTTP://XKCD.COM/1144/](http://xkcd.com/1144/)
 IMAGE URL (FOR HOTLINKING/EMBEDDING): [HTTP://IMGS.XKCD.COM/COMICS/TAGS.PNG](http://imgs.xkcd.com/comics/tags.png)



ARCHIVE

WHAT IF?

BLAG

STORE

ABOUT

xkcd

A WEBCOMIC OF ROMANCE,
SARCASM, MATH, AND LANGUAGE.

THING EXPLAINER IS AVAILABLE AT: [AMAZON](#), [BARNES & NOBLE](#), [INDIE BOUND](#),
[HUDSON](#)

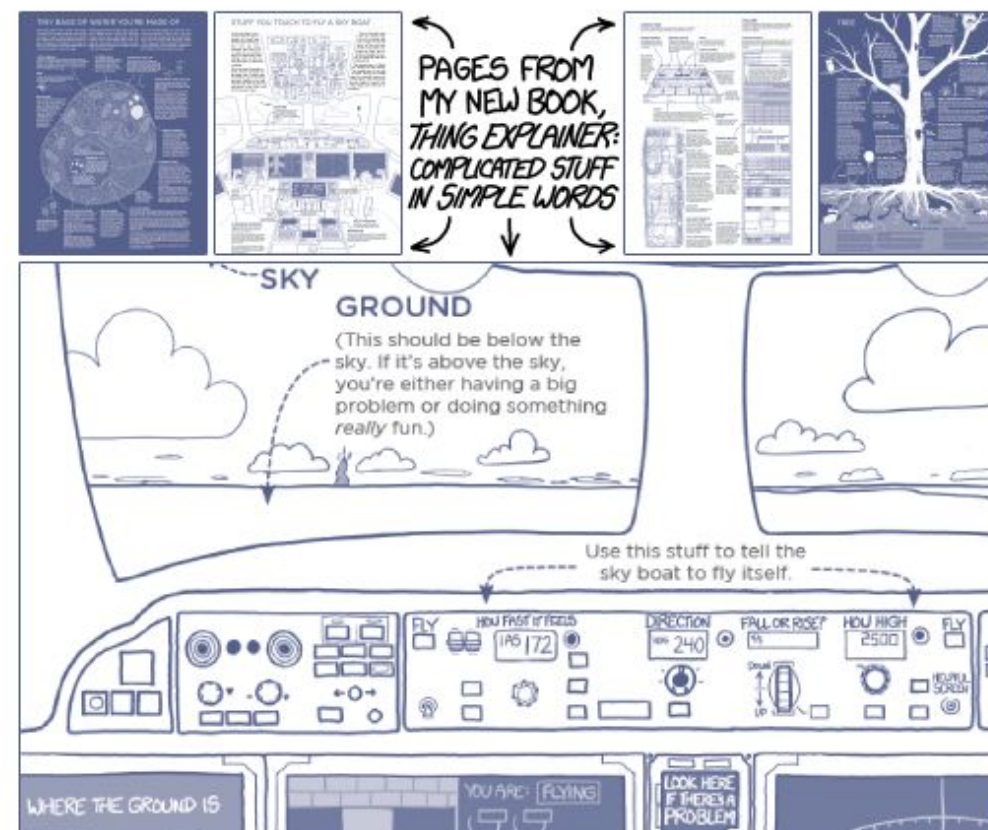
TAGS

|< < PREV RANDOM NEXT > >|

<DIV>Q: HOW DO YOU ANNOY A WEB DEVELOPER?

|< < PREV RANDOM NEXT > >|

PERMANENT LINK TO THIS COMIC: [HTTP://XKCD.COM/1144/](http://xkcd.com/1144/)
IMAGE URL (FOR HOTLINKING/EMBEDDING): [HTTP://IMGS.XKCD.COM/COMICS/TAGS.PNG](http://imgs.xkcd.com/comics/tags.png)



ARCHIVE

WHAT IF?

BLAG

STORE

ABOUT

xkcd

A WEBCOMIC OF ROMANCE,
SARCASM, MATH, AND LANGUAGE.

THING EXPLAINER IS AVAILABLE AT: [AMAZON](#), [BARNES & NOBLE](#), [INDIE BOUND](#),
[HUDSON](#)

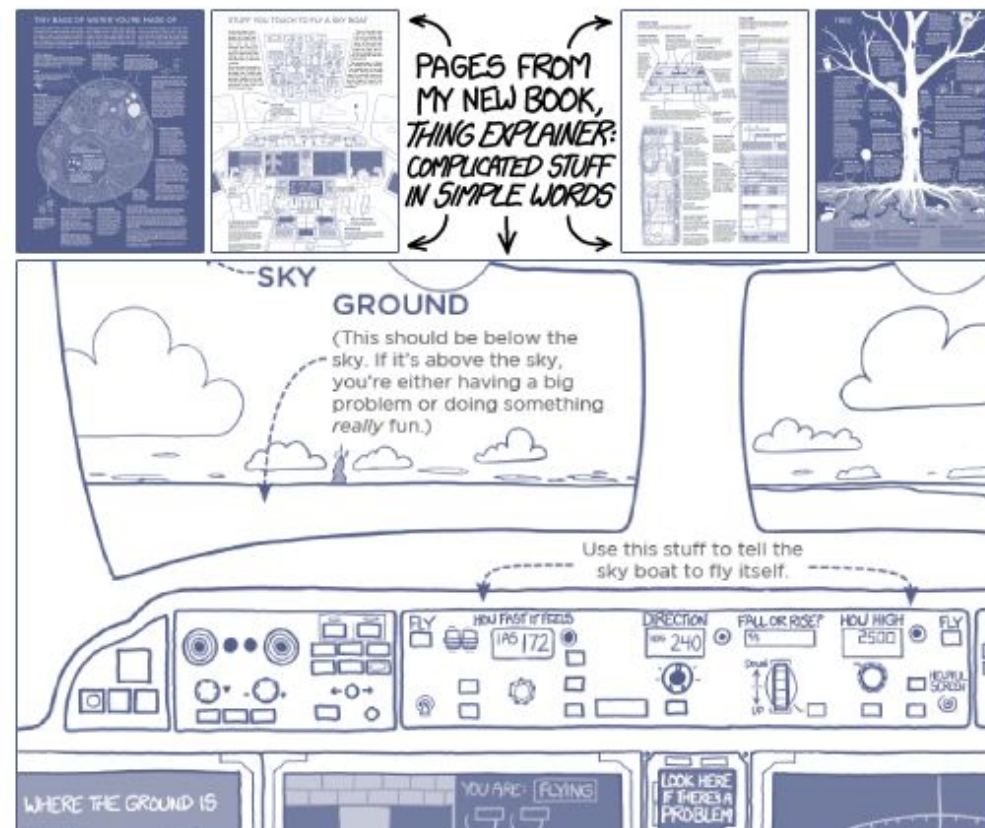
TAGS

|< < PREV RANDOM NEXT > >|

<DIV>Q: HOW DO YOU ANNOY A WEB DEVELOPER?

|< < PREV RANDOM NEXT > >|

PERMANENT LINK TO THIS COMIC: [HTTP://XKCD.COM/1144/](http://xkcd.com/1144/)
IMAGE URL (FOR HOTLINKING/EMBEDDING): [HTTP://IMGS.XKCD.COM/COMICS/TAGS.PNG](http://imgs.xkcd.com/comics/tags.png)



ARCHIVE

WHAT IF?

BLAG

STORE

ABOUT

xkcd

A WEBCOMIC OF ROMANCE,
SARCASM, MATH, AND LANGUAGE.

THING EXPLAINER IS AVAILABLE AT: [AMAZON](#), [BARNES & NOBLE](#), [INDIE BOUND](#),
[HUDSON](#)

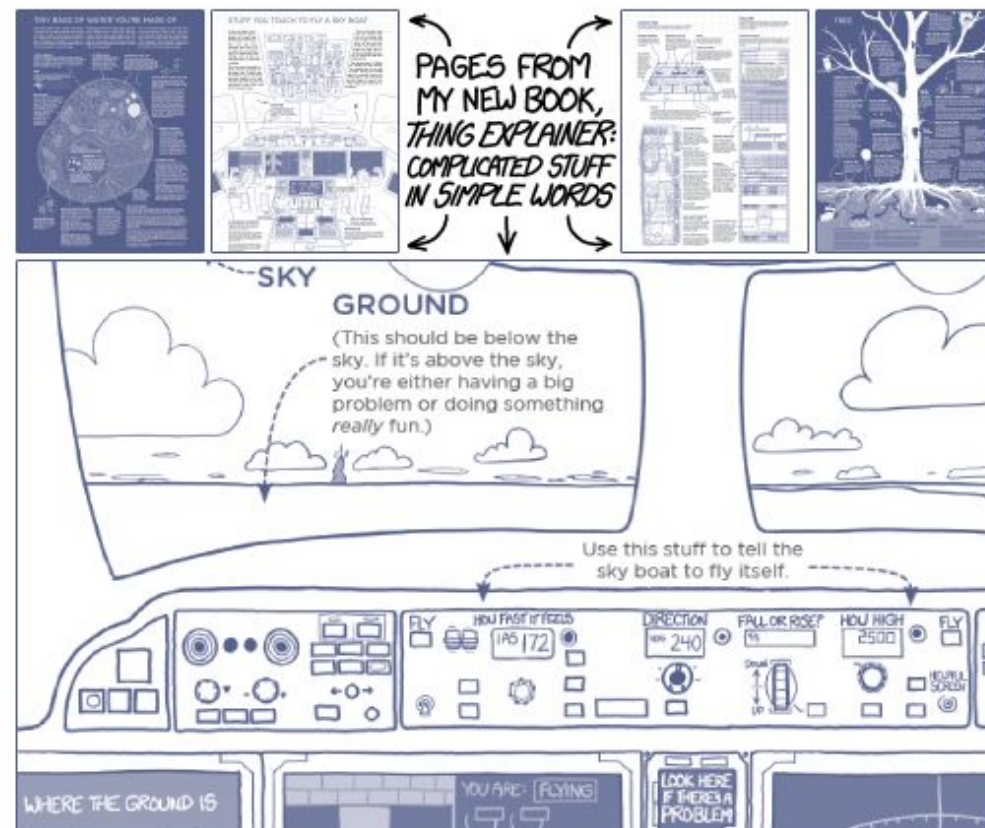
TAGS

IS < PREV RANDOM NEXT >

<DIV>Q: HOW DO YOU ANNOY A WEB DEVELOPER?

|< < PREV RANDOM NEXT > >|

PERMANENT LINK TO THIS COMIC: [HTTP://XKCD.COM/1144/](http://xkcd.com/1144/)
IMAGE URL (FOR HOTLINKING/EMBEDDING): [HTTP://IMGS.XKCD.COM/COMICS/TAGS.PNG](http://imgs.xkcd.com/comics/tags.png)

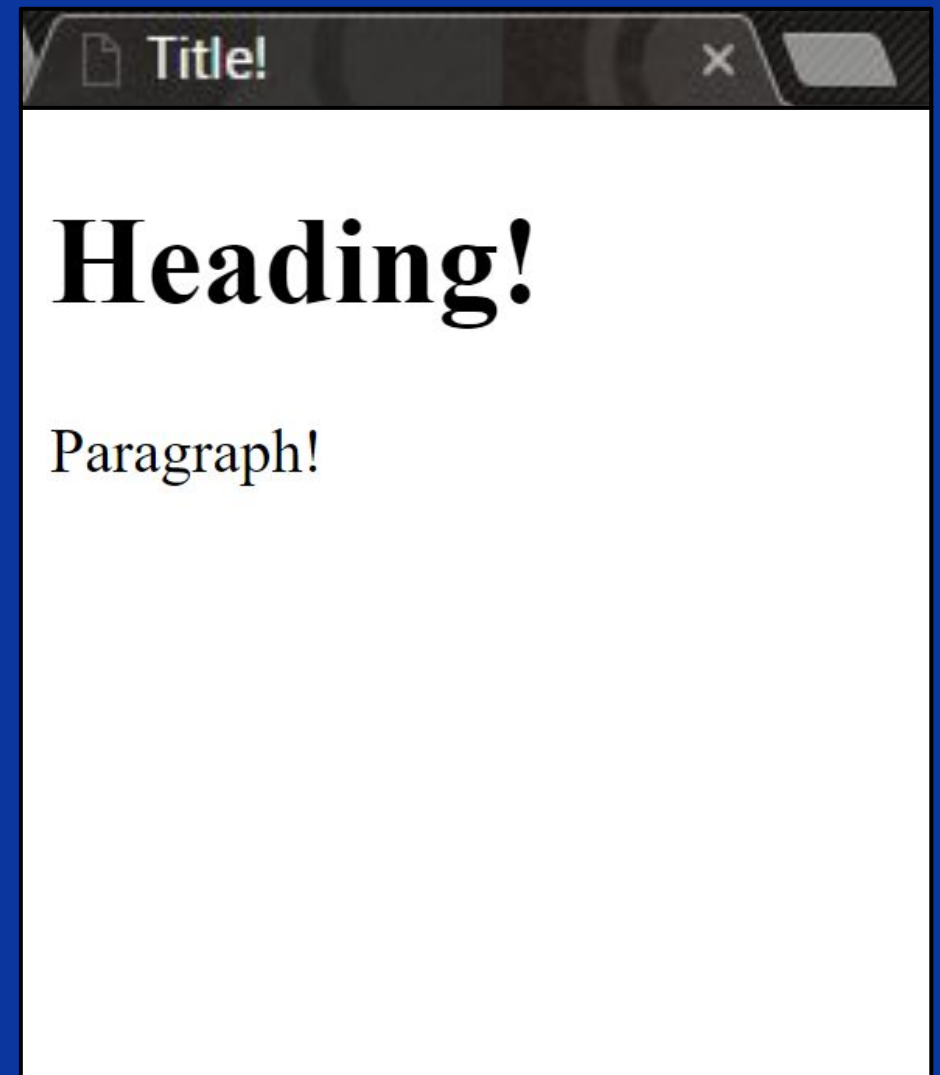


hello.html

```
<!DOCTYPE html>
<html>
  <head>
    <title>Title!</title>
  </head>
  <body>
    <h1>Heading!</h1>
    <p>Paragraph!</p>
  </body>
</html>
```

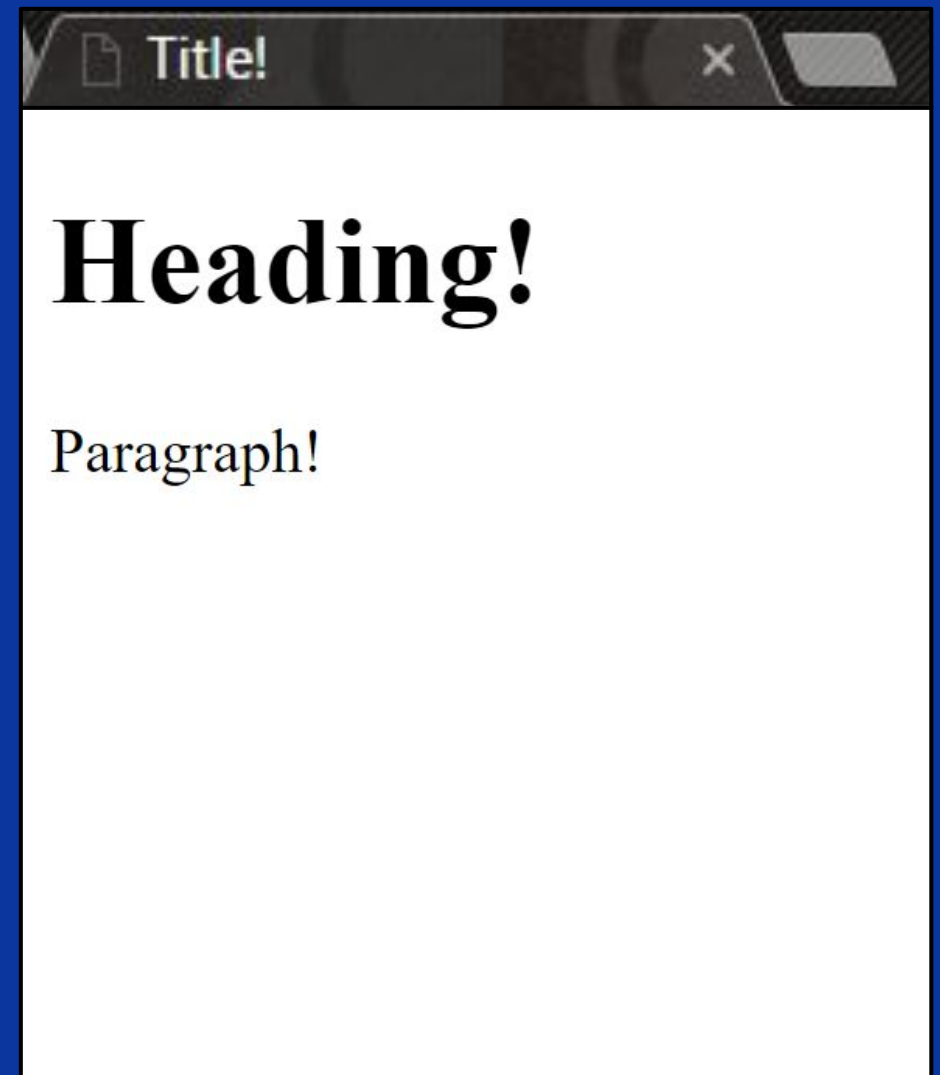

hello.html

```
<!DOCTYPE html>
<html>
  <head>
    <title>Title!</title>
  </head>
  <body>
    <h1>Heading!</h1>
    <p>Paragraph!</p>
  </body>
</html>
```



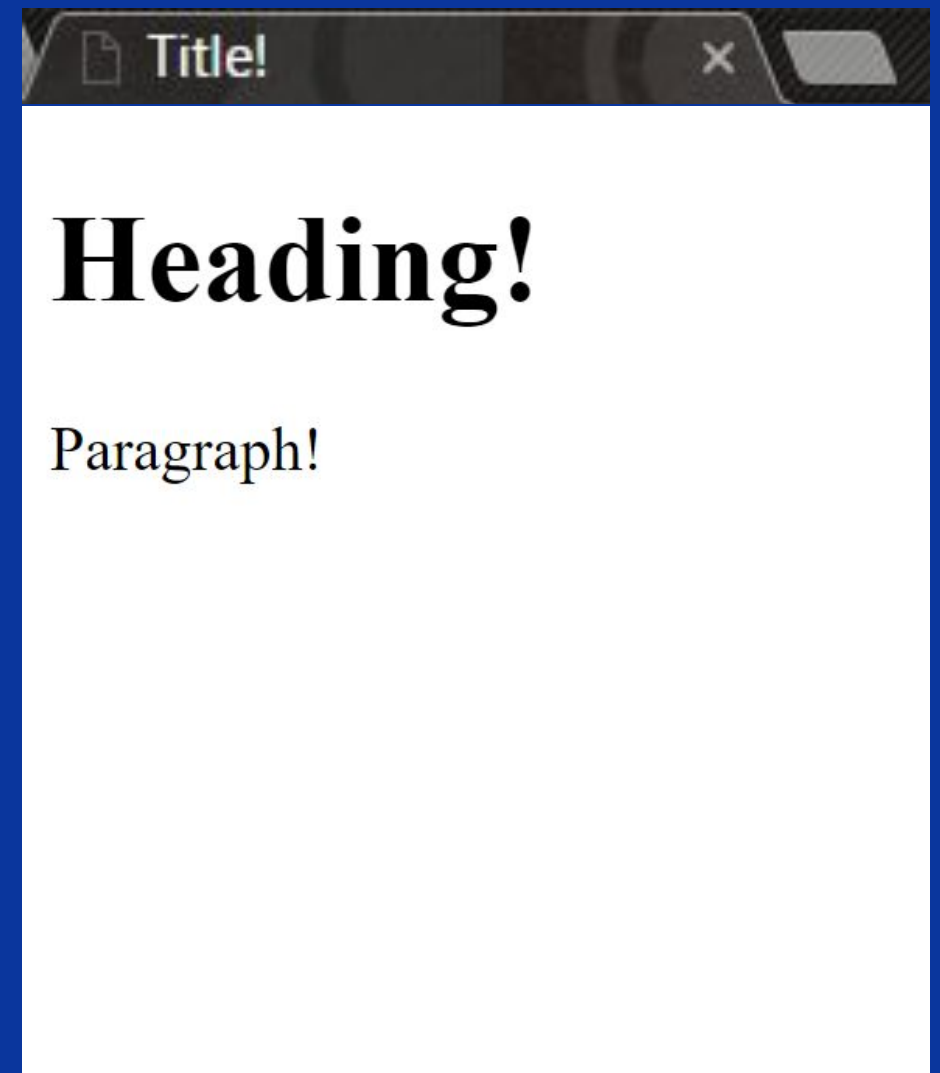
hello.html

```
<!DOCTYPE html>
<html>
  <head>
    <title>Title!</title>
  </head>
  <body>
    <h1>Heading!</h1>
    <p>Paragraph!</p>
  </body>
</html>
```

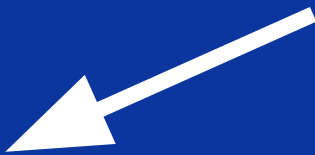


hello.html


```
<!DOCTYPE html>
<html>
  <head>
    <title>Title!</title>
  </head>
  <body>
    <h1>Heading!</h1>
    <p>Paragraph!</p>
  </body>
</html>
```



HTML Element

`<html>`  Opening tag

content goes here

`</html>`  Closing tag

hello.html

```
<!DOCTYPE html>
```

```
<html>
```

```
  <head>
```

```
    <title>Title!</title>
```

```
  </head>
```

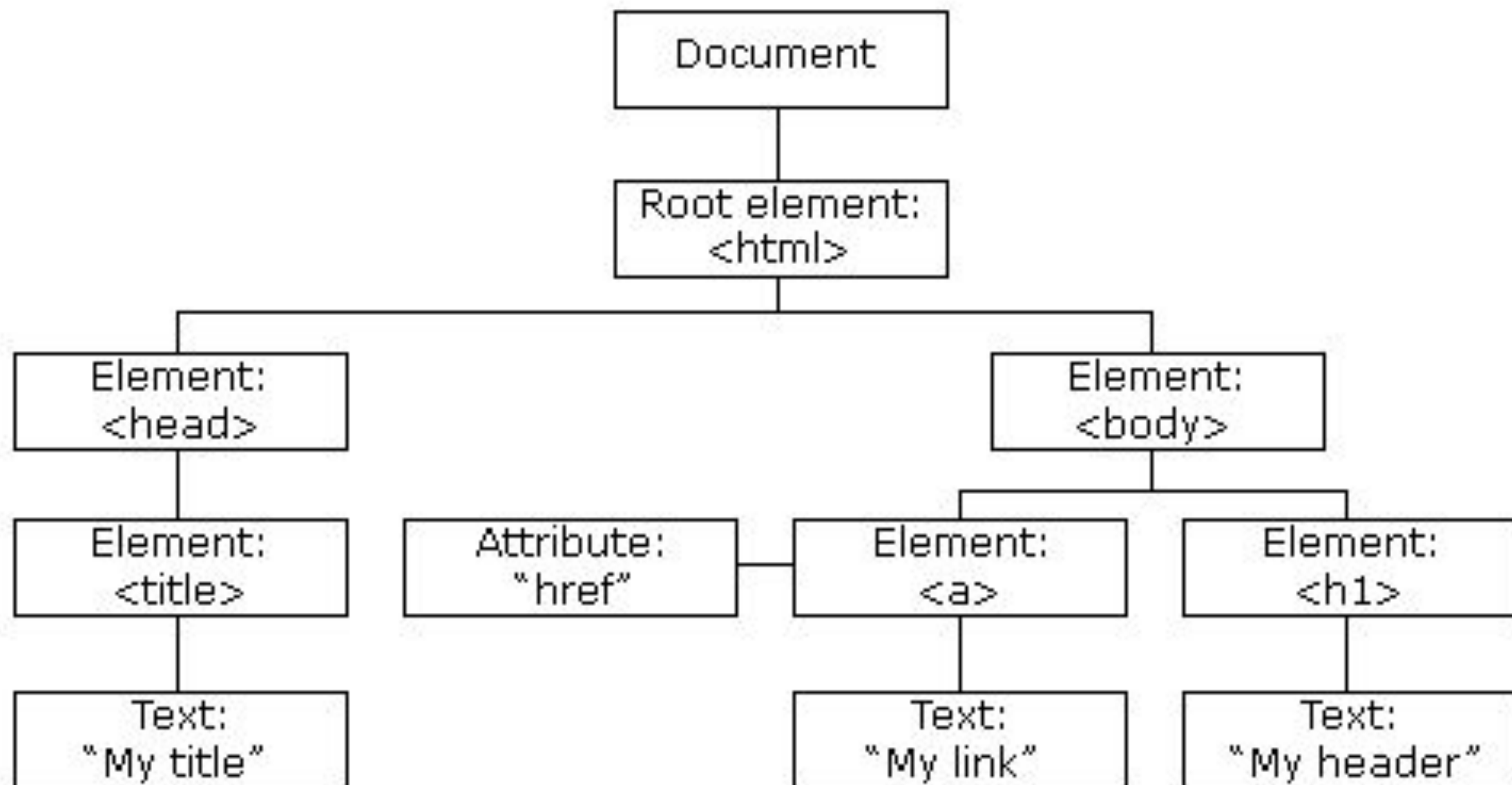
```
  <body>
```

```
    <h1>Heading!</h1>
```

```
    <p>Paragraph!</p>
```

```
  </body>
```

```
</html>
```



Nesting

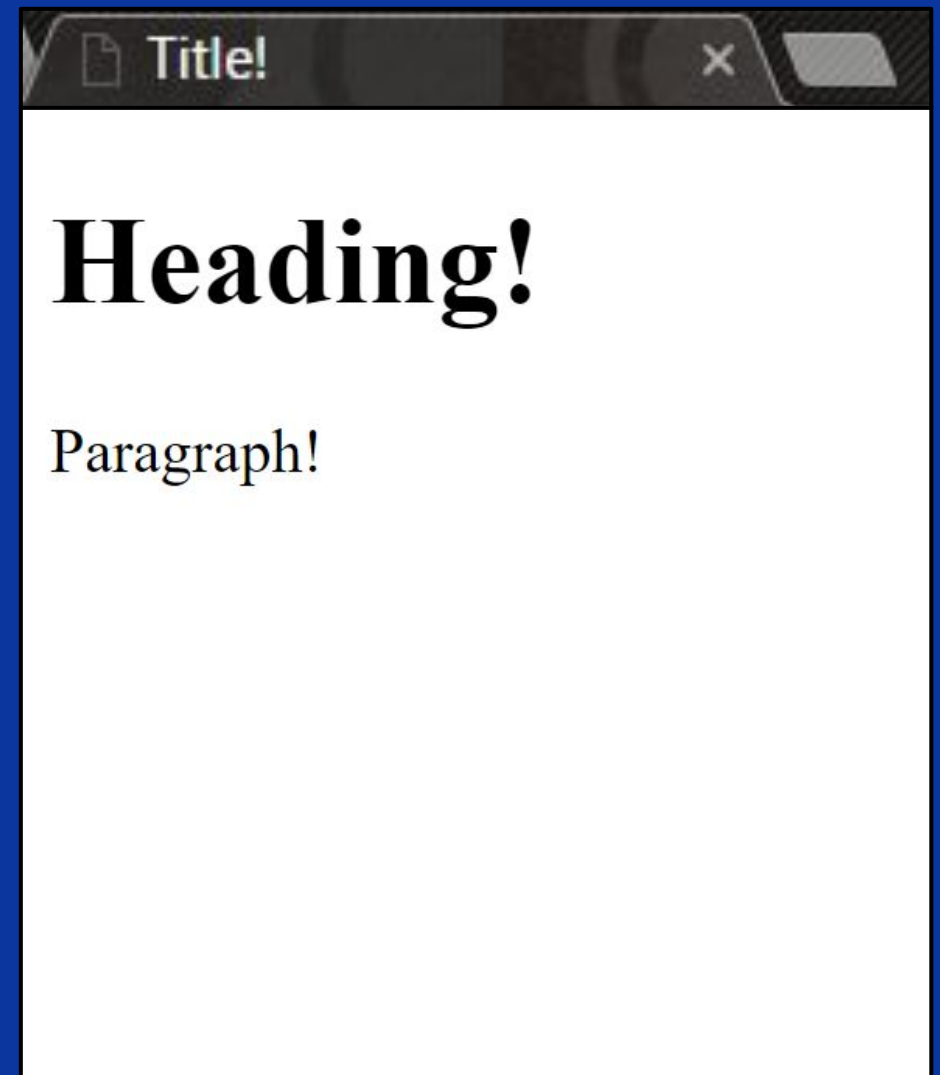
~~<a>

~~



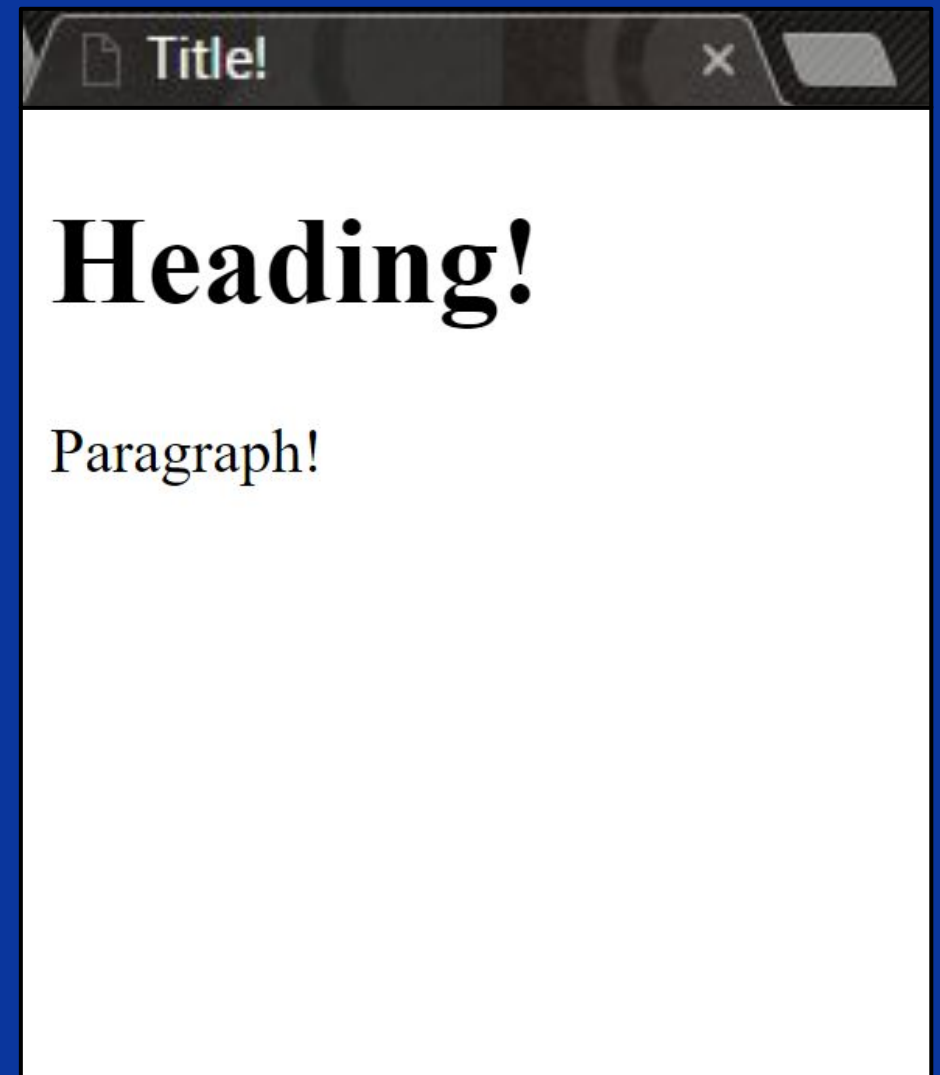
hello.html

```
<!DOCTYPE html>
<html>
  <head>
    <title>Title!</title>
  </head>
  <body>
    <h1>Heading!</h1>
    <p>Paragraph!</p>
  </body>
</html>
```



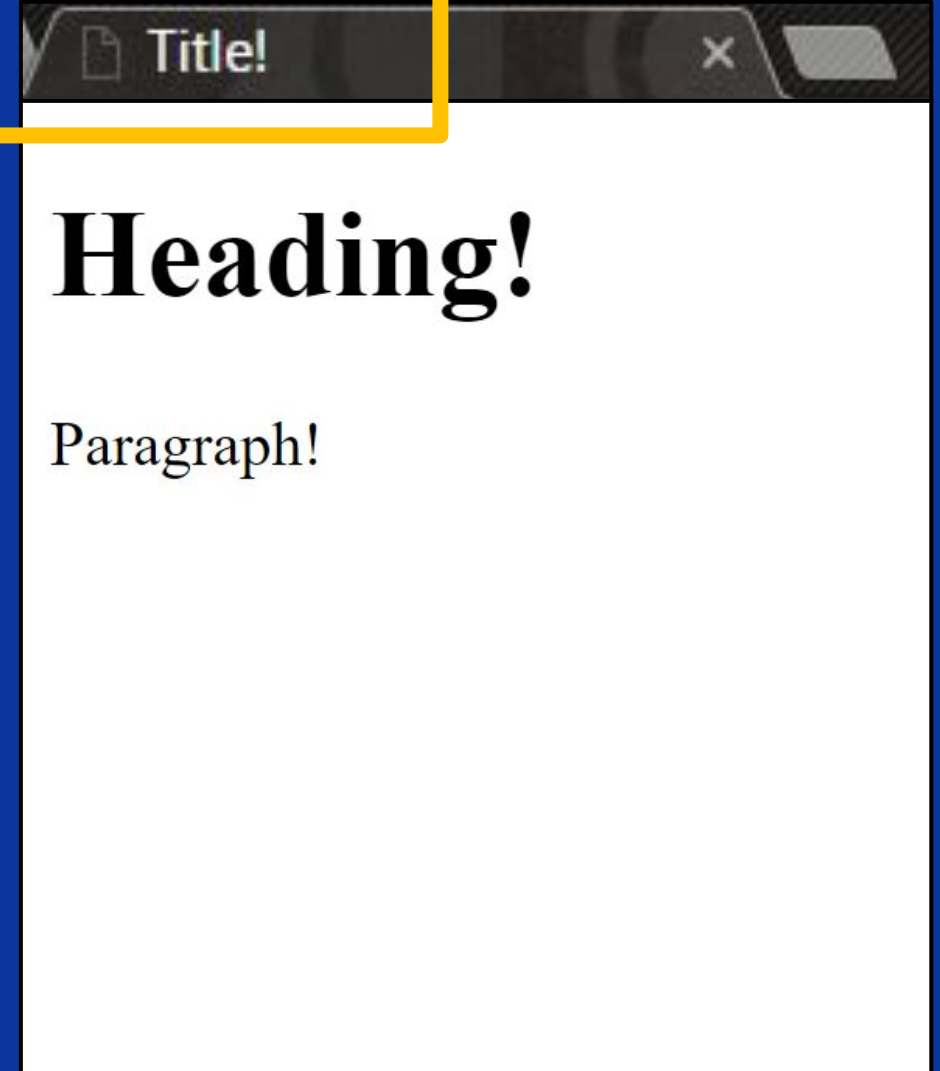
hello.html

```
<!DOCTYPE html>
<html>
  <head>
    <title>Title!</title>
  </head>
  <body>
    <h1>Heading!</h1>
    <p>Paragraph!</p>
  </body>
</html>
```



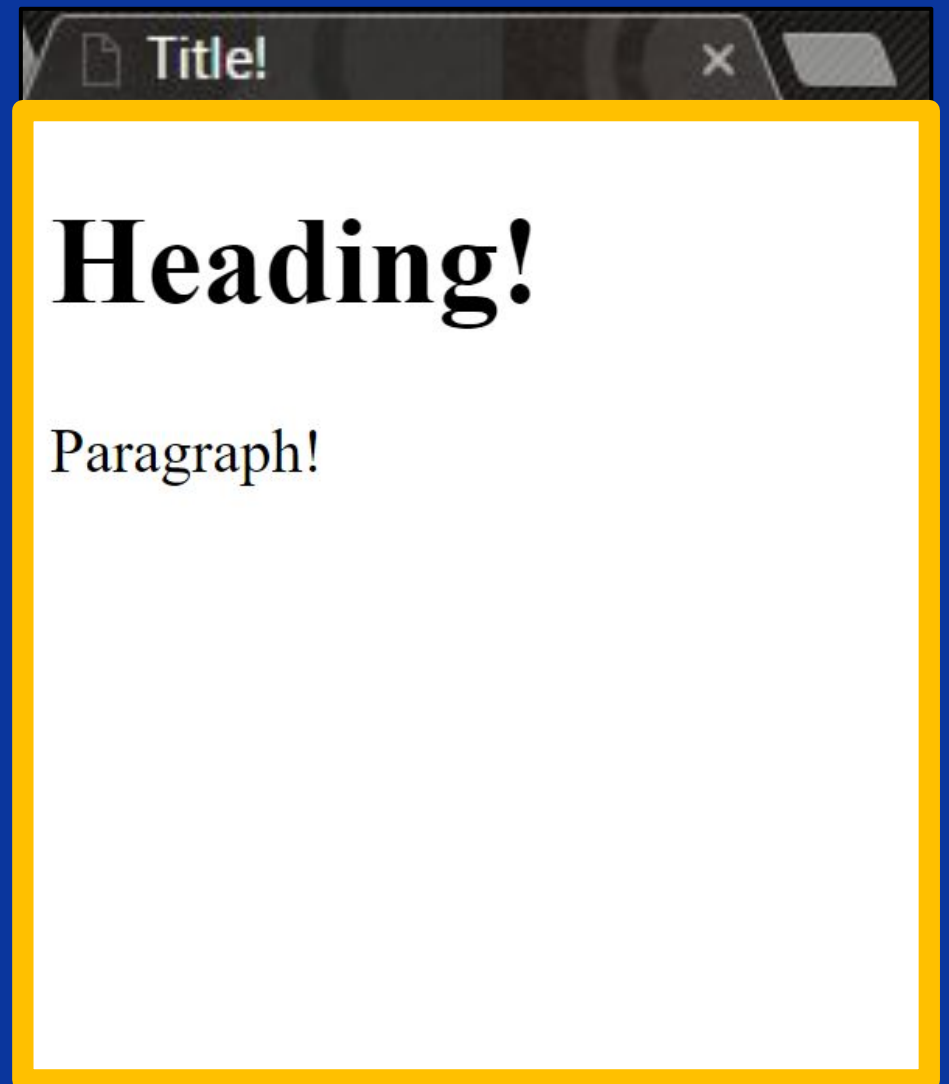
hello.html

```
<!DOCTYPE html>
<html>
  <head>
    <title>Title!</title>
  </head>
  <body>
    <h1>Heading!</h1>
    <p>Paragraph!</p>
  </body>
</html>
```



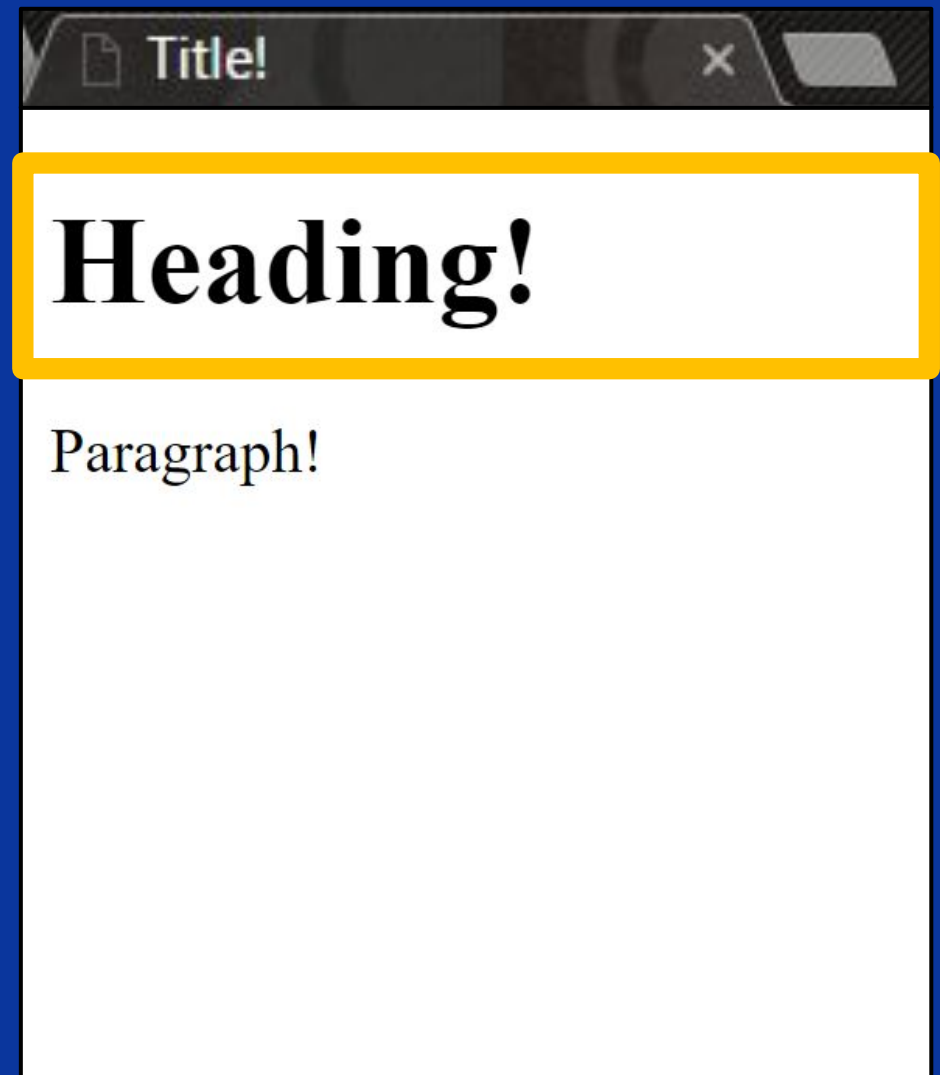
hello.html

```
<!DOCTYPE html>
<html>
  <head>
    <title>Title!</title>
  </head>
  <body>
    <h1>Heading!</h1>
    <p>Paragraph!</p>
  </body>
</html>
```



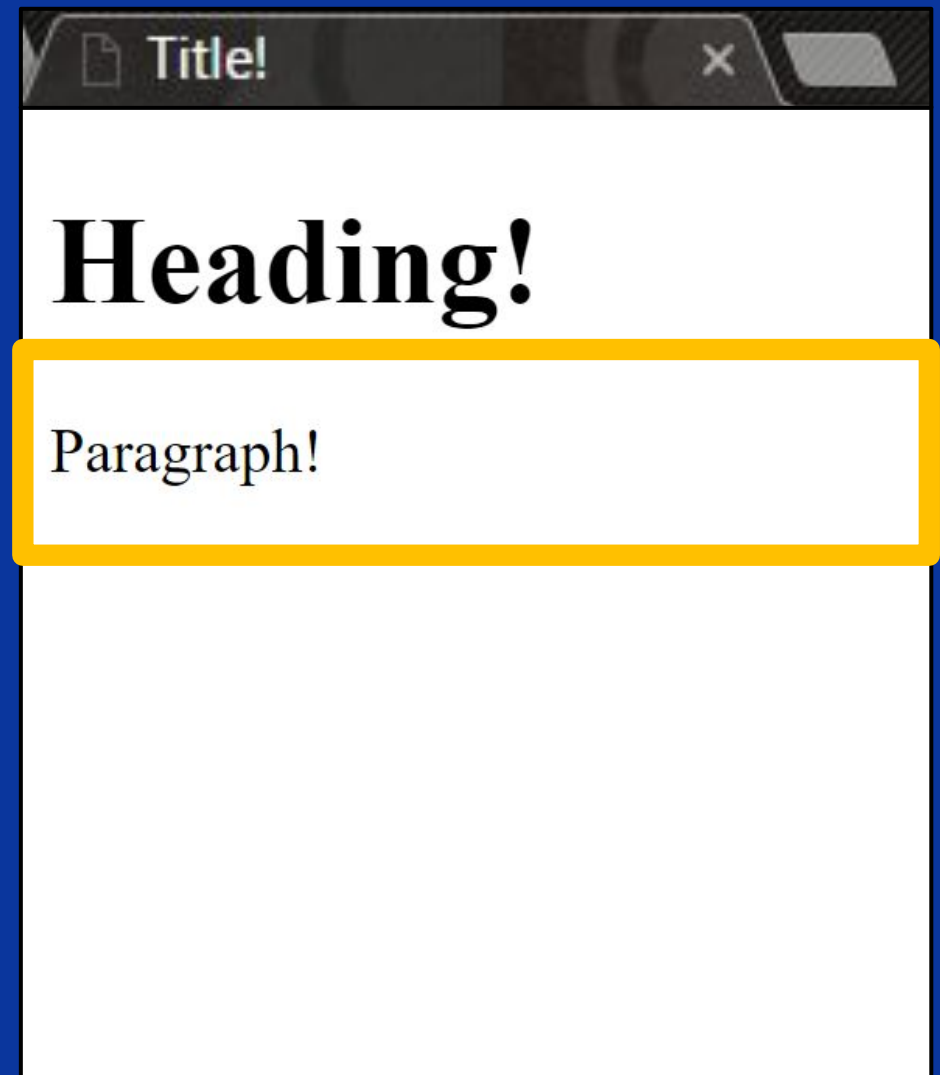
hello.html

```
<!DOCTYPE html>
<html>
  <head>
    <title>Title!</title>
  </head>
  <body>
    <h1>Heading!</h1>
    <p>Paragraph!</p>
  </body>
</html>
```



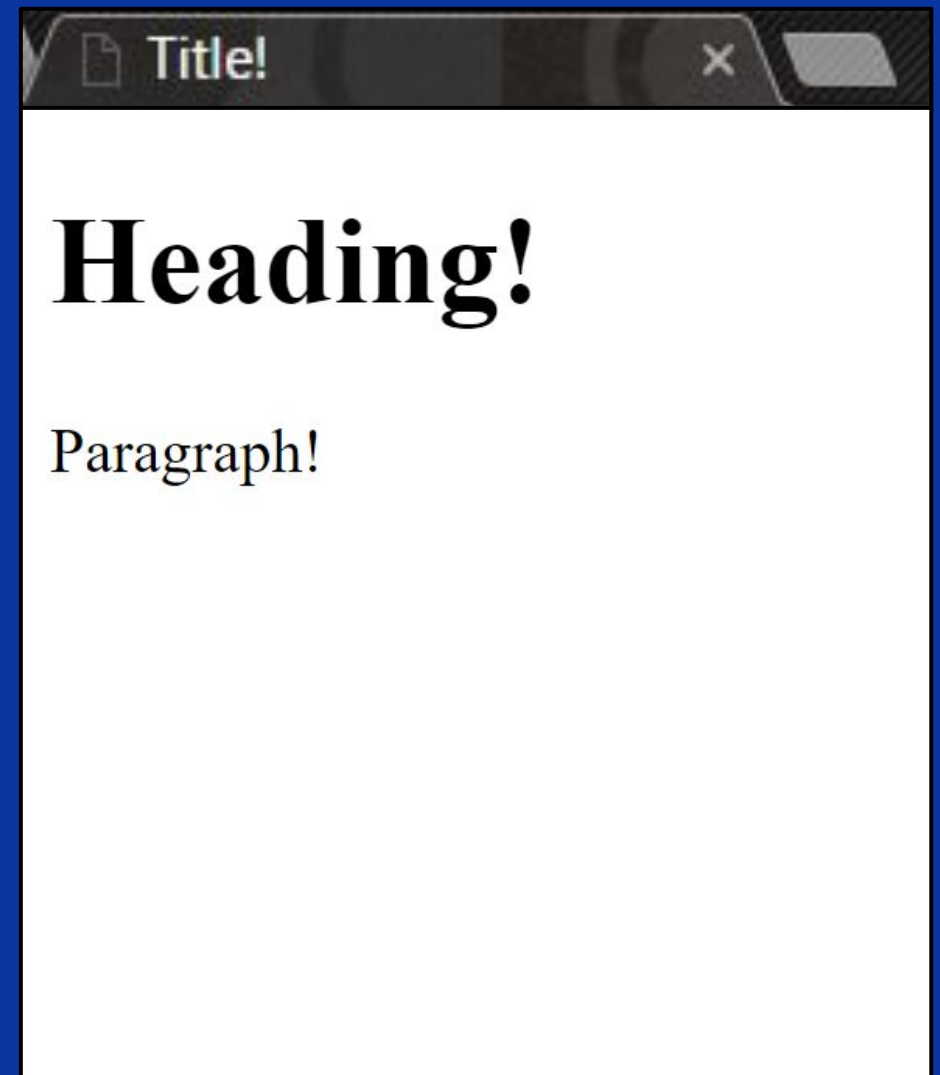
hello.html

```
<!DOCTYPE html>
<html>
  <head>
    <title>Title!</title>
  </head>
  <body>
    <h1>Heading!</h1>
    <p>Paragraph!</p>
  </body>
</html>
```



hello.html

```
<!DOCTYPE html>
<html>
  <head>
    <title>Title!</title>
  </head>
  <body>
    <h1>Heading!</h1>
    <p>Paragraph!</p>
  </body>
</html>
```



Basic HTML Tags

<html>

Root of HTML Document

<head>

Info about Document

<body>

Document Body

<h1>, <h2>, <h3>, ...

Header tags

<p>

Paragraph tag

<div>

Generic block section tag

Generic inline section tag

HTML Attributes

<tagname>



Opening tag

content goes here

</tagname>



Closing tag

HTML Attributes

attribute value

↓ ↓

<tagname abc="xyz">

Opening tag

content goes here

</tagname>

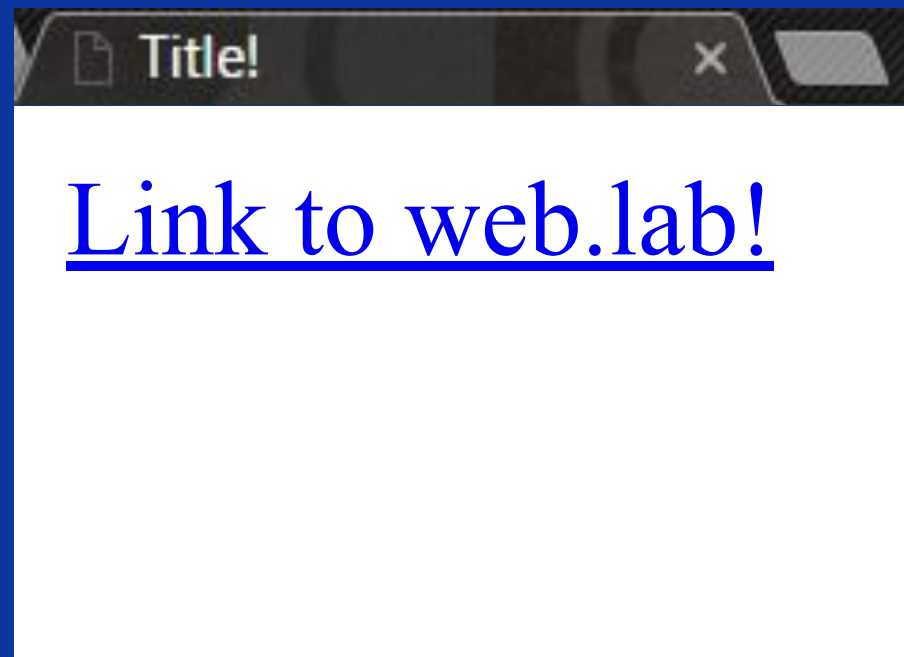
Closing tag

The diagram illustrates the structure of an HTML tag. It shows an opening tag <tagname abc="xyz"> and a closing tag </tagname>. Annotations with arrows identify the components: 'attribute' and 'value' point to 'abc' and 'xyz' respectively in the opening tag; 'Opening tag' points to the entire <tagname abc="xyz"> sequence; 'content goes here' is placed between the opening and closing tags; and 'Closing tag' points to the </tagname> sequence. The tag names and closing tag are highlighted in yellow, while attributes and values are in grey.

Inserting Links

<a>

Link



HTML Attributes

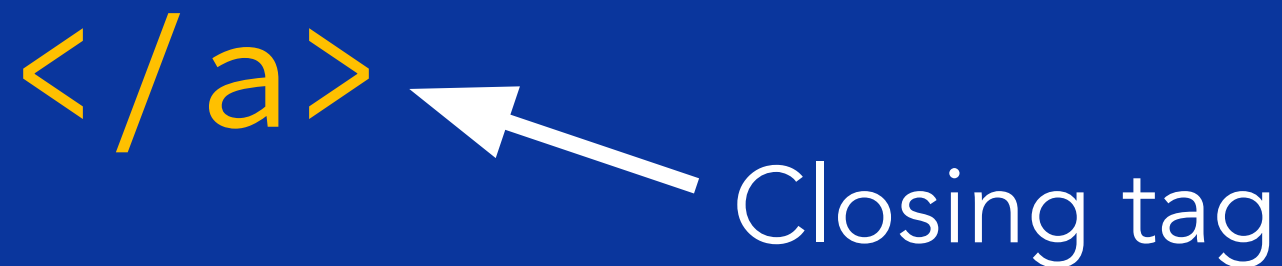
attribute value Opening tag

``

A diagram showing the opening tag ``. Three white arrows point to specific parts of the tag: one to `href` (labeled 'attribute'), one to `"link here"` (labeled 'value'), and one to the closing angle bracket `>` (labeled 'Opening tag').

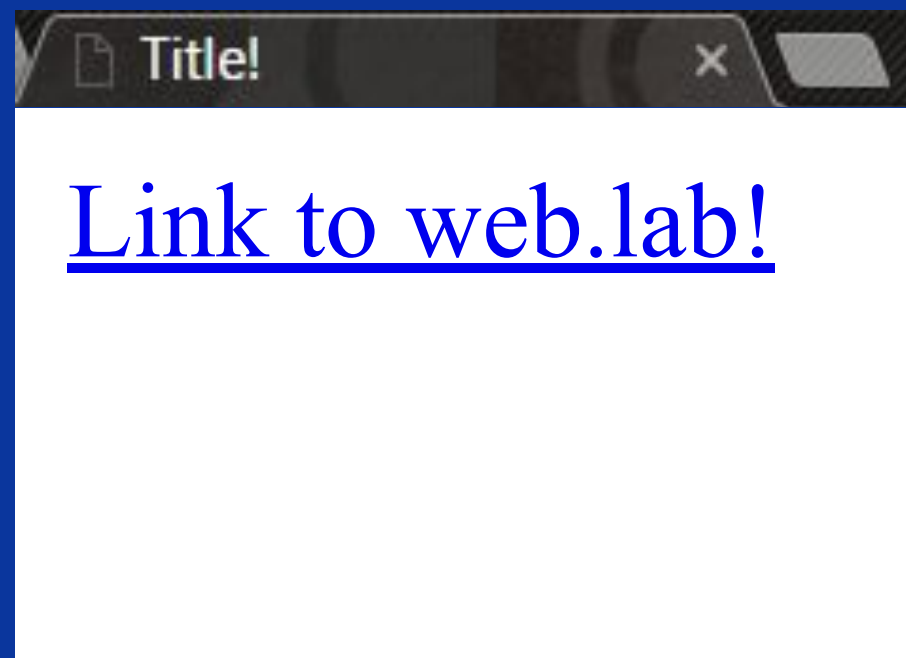
content goes here

`` Closing tag

A diagram showing the closing tag ``. A white arrow points to the tag (labeled 'Closing tag').

Inserting Links

```
<a href="http://web.lab.mit.edu">Link to web.lab!</a>
```



Inserting Images

Image

Inserting Images

```
</img>
```


Inserting Images

``

Inserting Images

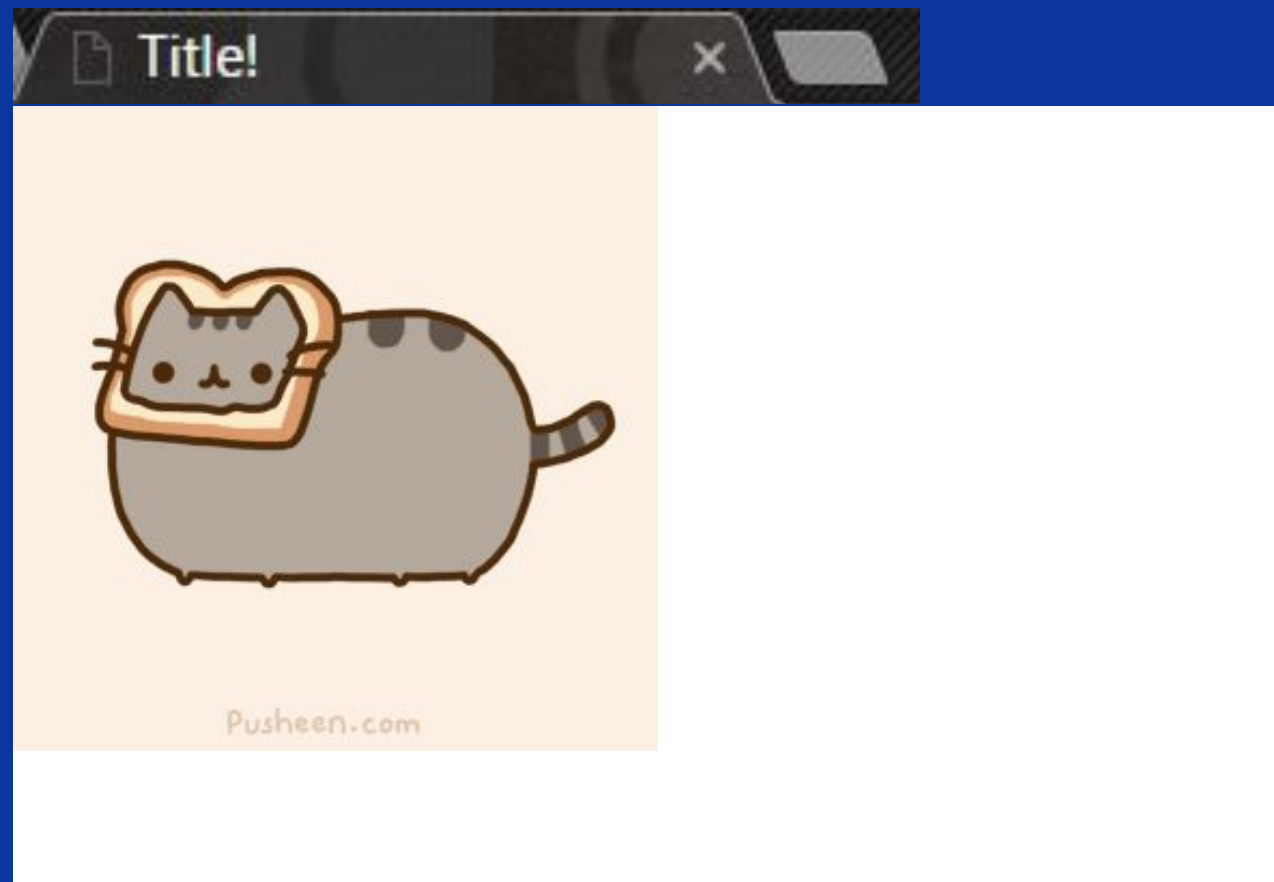
```

```

Inserting Images

```

```



Inserting Images

```

```

```
app/  
  hello.html  
  pusheen.gif
```

Inserting Images

~~~~



```
app/  
  hello.html  
    images/  
      pusheen.gif
```



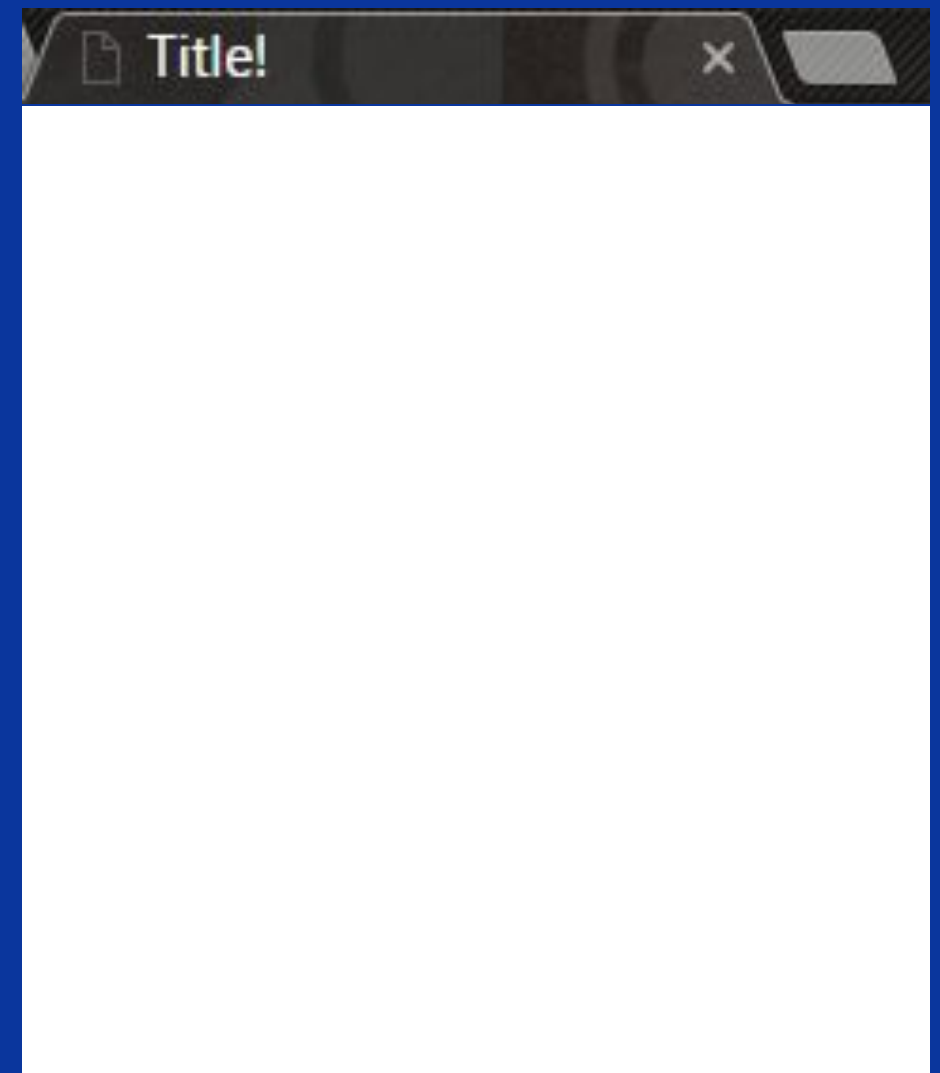
Questions?

# Lists

- `<ol>` Ordered List (1, 2, 3...)
- `<ul>` Unordered List (bullets)
- `<li>` List Item

hello.html

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



hello.html

```
<!DOCTYPE html>
<html>
  <head>
    <title>Title!</title>
  </head>
  <body>
    <ul>
      <li>Item 1</li>
    </ul>
  </body>
</html>
```

- 
- A screenshot of a web browser window. The title bar at the top shows a document icon, the text "Title!", a close button (X), and a maximize button. The main content area of the browser is white and contains a single bullet point.
- Item 1

hello.html

```
<!DOCTYPE html>
<html>
  <head>
    <title>Title!</title>
  </head>
  <body>
    <ul>
      <li>Item 1</li>
      <li>Item 2</li>
    </ul>
  </body>
</html>
```

- 
- Item 1
  - Item 2

# div & span

`<div>`

Block Section in Document

`<span>`

Inline Section in Document

# div & span

```
<div id="div-0">  
  <h2>Div 1!</h2>  
  <p>Paragraph</p>  
</div>
```

```
<div id="div-1">  
  <h2>Div 2!</h2>  
  <p>Paragraph!!</p>  
</div>
```

```
<span id="span-0">This text is inside a span element.</span>  
<span id="span-1" span>This text is inside another span element.</span>
```

**Div 1!**

Paragraph

**Div 2!**

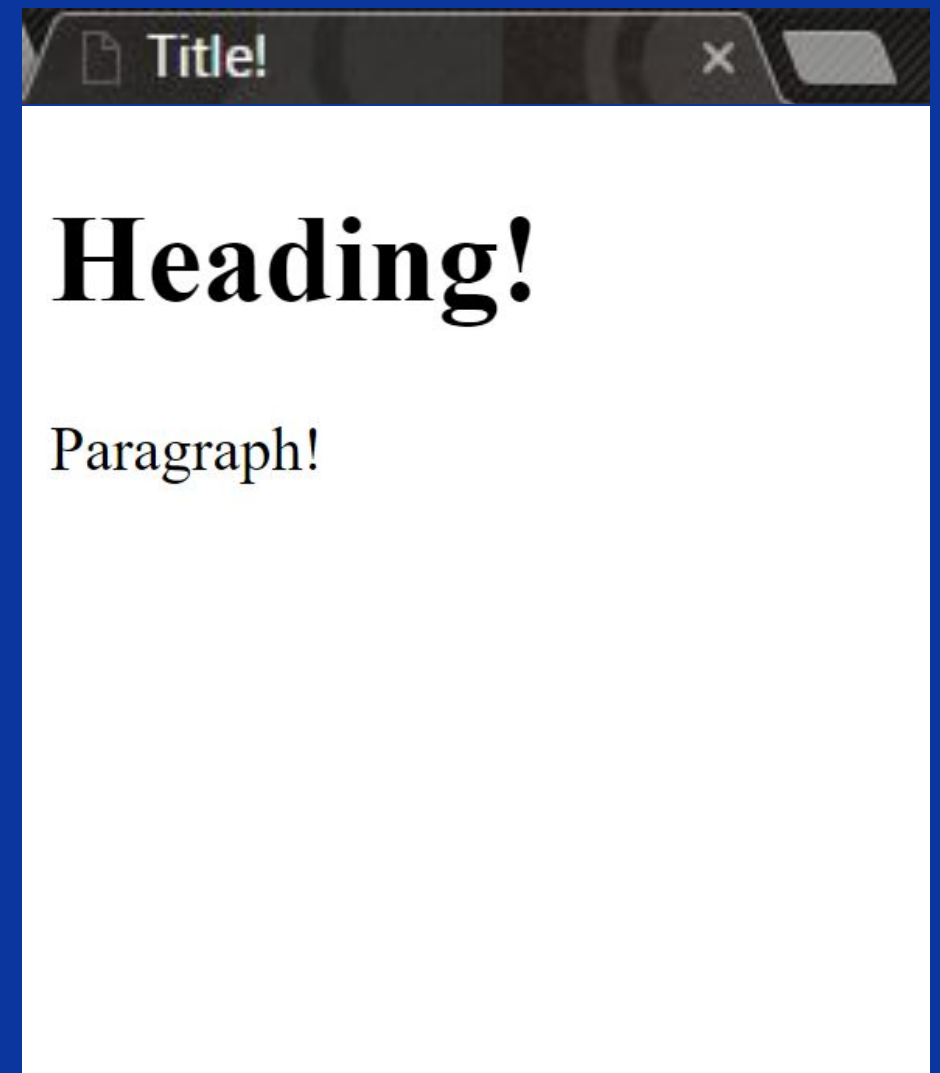
Paragraph!!

This text is inside a span element. This text is inside another span element.



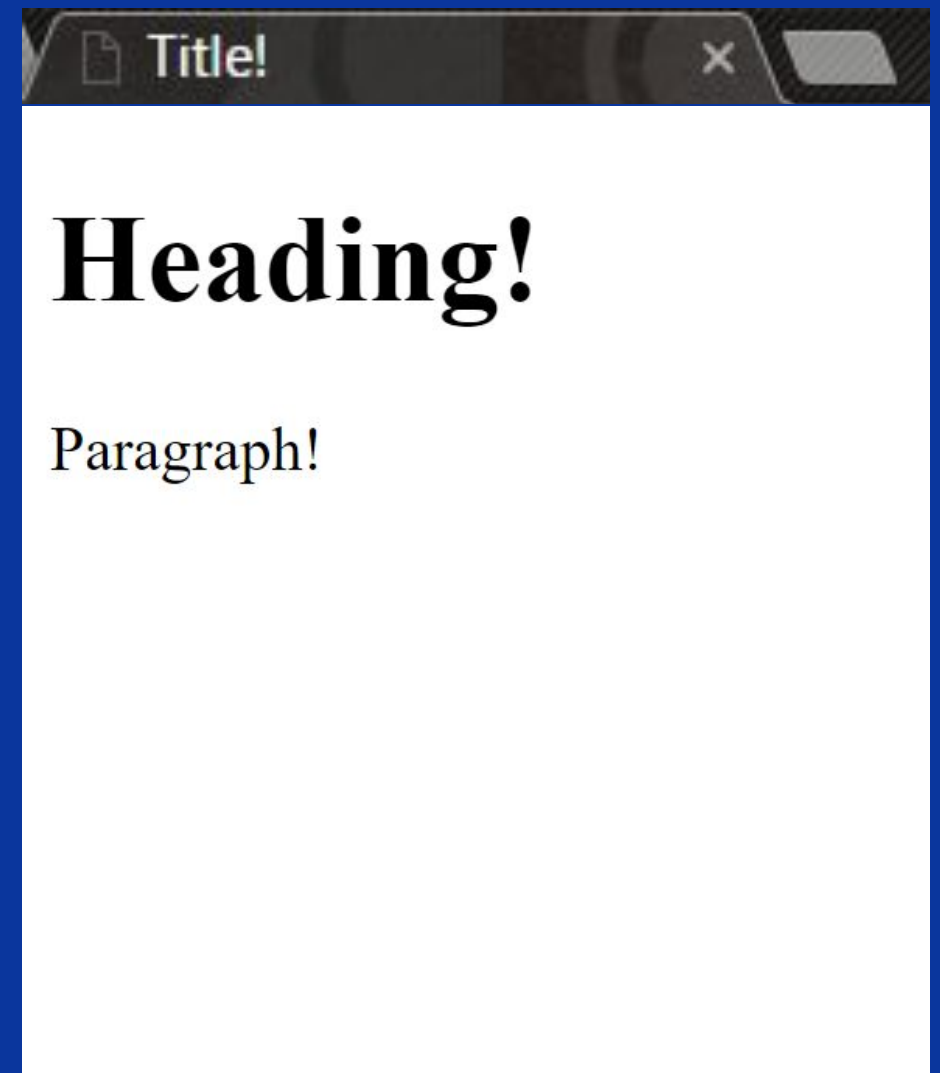
hello.html

```
<!DOCTYPE html>
<html>
  <head>
    <title>Title!</title>
  </head>
  <body>
    <h1>Heading!</h1>
    <p>Paragraph!</p>
  </body>
</html>
```



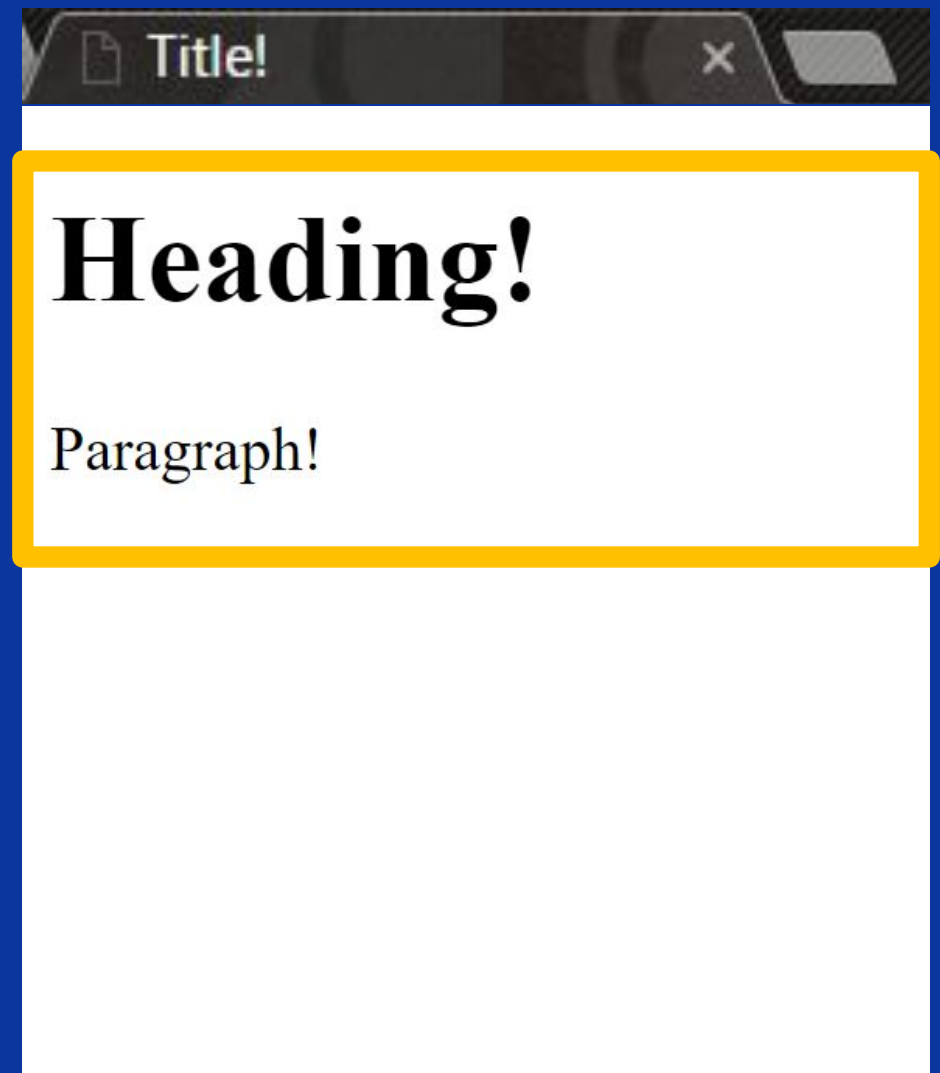
hello.html

```
<!DOCTYPE html>
<html>
  <head>
    <title>Title!</title>
  </head>
  <body>
    <div>
      <h1>Heading!</h1>
      <p>Paragraph!</p>
    </div>
  </body>
</html>
```



hello.html

```
<!DOCTYPE html>
<html>
  <head>
    <title>Title!</title>
  </head>
  <body>
    <div>
      <h1>Heading!</h1>
      <p>Paragraph!</p>
    </div>
  </body>
</html>
```



so many tags...

<ul></ul>

<span></span>

so many tags...

<nav></nav>

# General Advice

1. Online documentation (MDN Web Docs)
  - a. Always up to date!
2. Utilize Google to see what tag is best!

Why not...

“just use <div>”



Why not...

“just use <div>”

why could this be bad?

# <div>

## <div>: The Content Division element

Web technology for developers > HTML: HyperText Markup Language > HTML elements reference > <div>: The Content Division element

English (US)

Change language

### Jump to section

Attributes

Usage notes

Examples

Specifications

Browser compatibility

See also

### Related Topics

<blockquote>

<dd>

<div>

<dl>

<dt>

<figcaption>

<figure>

<hr>

<li>

The **HTML Content Division element** (`<div>`) is the generic container for flow content. It has no effect on the content or layout until styled in some way using [CSS](#) (e.g. styling is directly applied to it, or some kind of layout model like [Flexbox](#) is applied to its parent element).

HTML Demo: <div>

HTML

CSS

```
1 <div class="warning">
2   
4   <p>Beware of the leopard</p>
5 </div>
6
```

Output

As a "pure" container, the `<div>` element does not inherently represent anything. Instead, it's used to group content so it can be easily styled using the `class` or `id` attributes, marking a section of a document as being written in a different language (using the `lang` attribute), and

# <div>

As a "pure" container, the `<div>` element does not inherently represent anything. I

## Usage notes

- The `<div>` element should be used only when no other **semantic** element (such as `<article>` or `<nav>`) is appropriate.

## See also

- Semantic sectioning elements: `<section>`, `<article>`, `<nav>`, `<header>`, `<footer>`

there might be  
better options...

Why Semantics?

a brief aside...

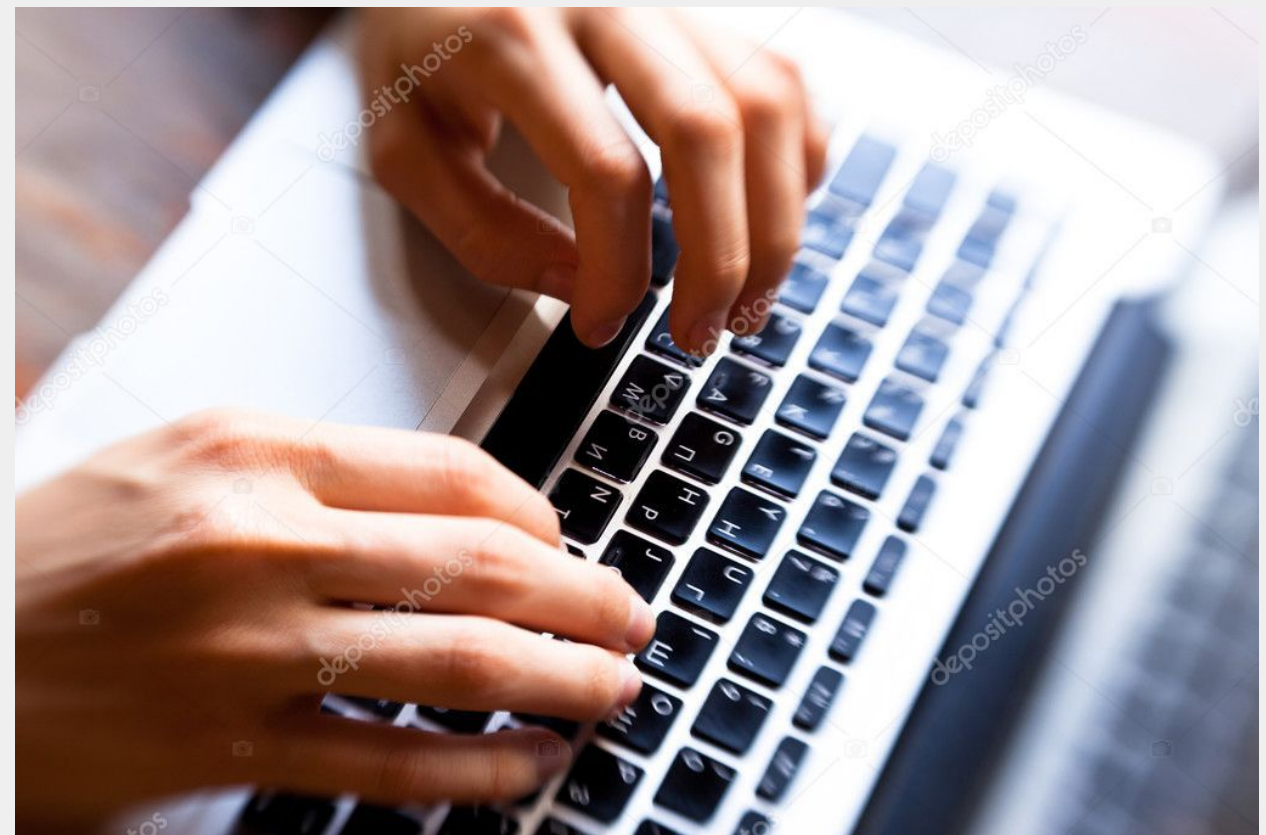
# Why Semantics?

## Screen readers



## **KB Navigation**

(why can you tab through an order or sign up screen?)

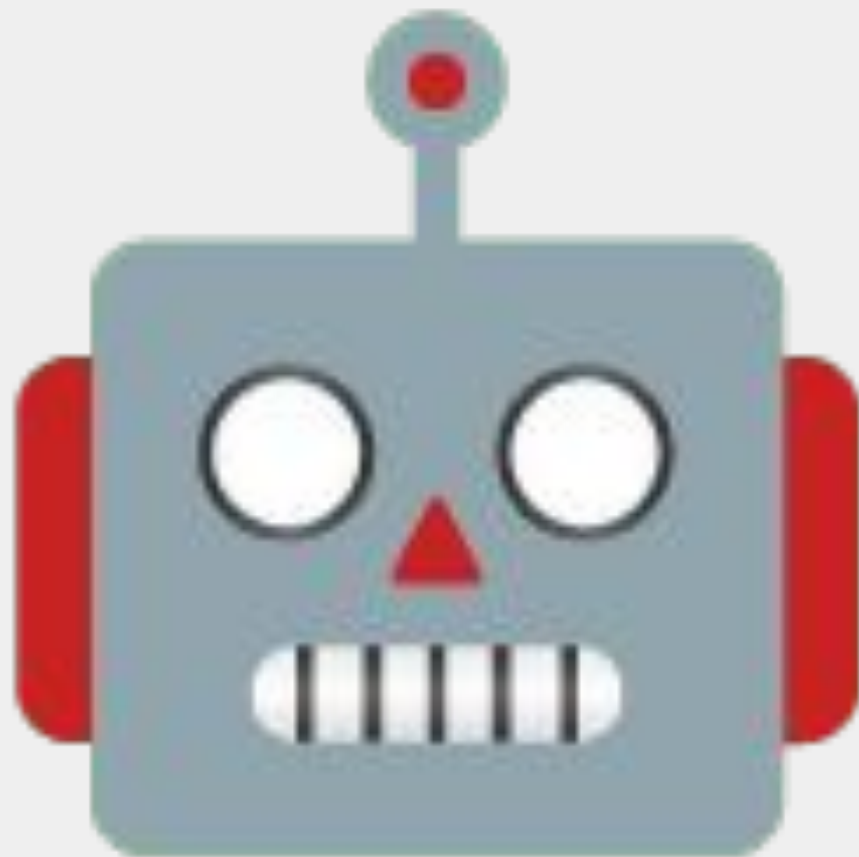


a brief aside...

# Why Semantics?

**Machine readable/friendly to web crawlers**

(make your site more discoverable)



**Human readable**

(easier to maintain in the future)





Don't just use `<div>`.

MDN is your friend!



# HTML



## Hypertext Markup Language

The language your web browser uses to describe the content and structure of web pages

CSS

# CSS

## Cascading Style Sheets

The **rules** that tell your web browser **how stuff looks**

CSS = A list of descriptions

used for markup languages, such as  
HTML

# Holidays Leaving You Short on ZZZ's?

What you eat plays a role in the quality of your sleep.

[7 eating tips to support healthy sleep →](#)

**WELLNESS  
BON APPÉTIT**  
➤➤➤

[CAFÉ HOURS](#) [BREAKFAST](#) [BRUNCH](#) [LUNCH](#) [DINNER](#) [ICONS](#) [MEAL PLANS](#) [ABOUT YOUR FOOD](#) [MENU MAIL](#) [CONTACT US](#) [FOOD RECOVERY VERIFIED](#) [EXPRESS & PACK-OUT MEALS](#) [IAP MEAL PLANS](#)

## CAFÉ HOURS

+ **NEW VASSAR** ..... SERVING DINNER FROM 5:00 PM TO 8:00 PM

+ **THE HOWARD DINING HALL** ..... **CLOSED**  
**AT MACEEM**

[Skip to main content](#)

**MIT**

**MIT**

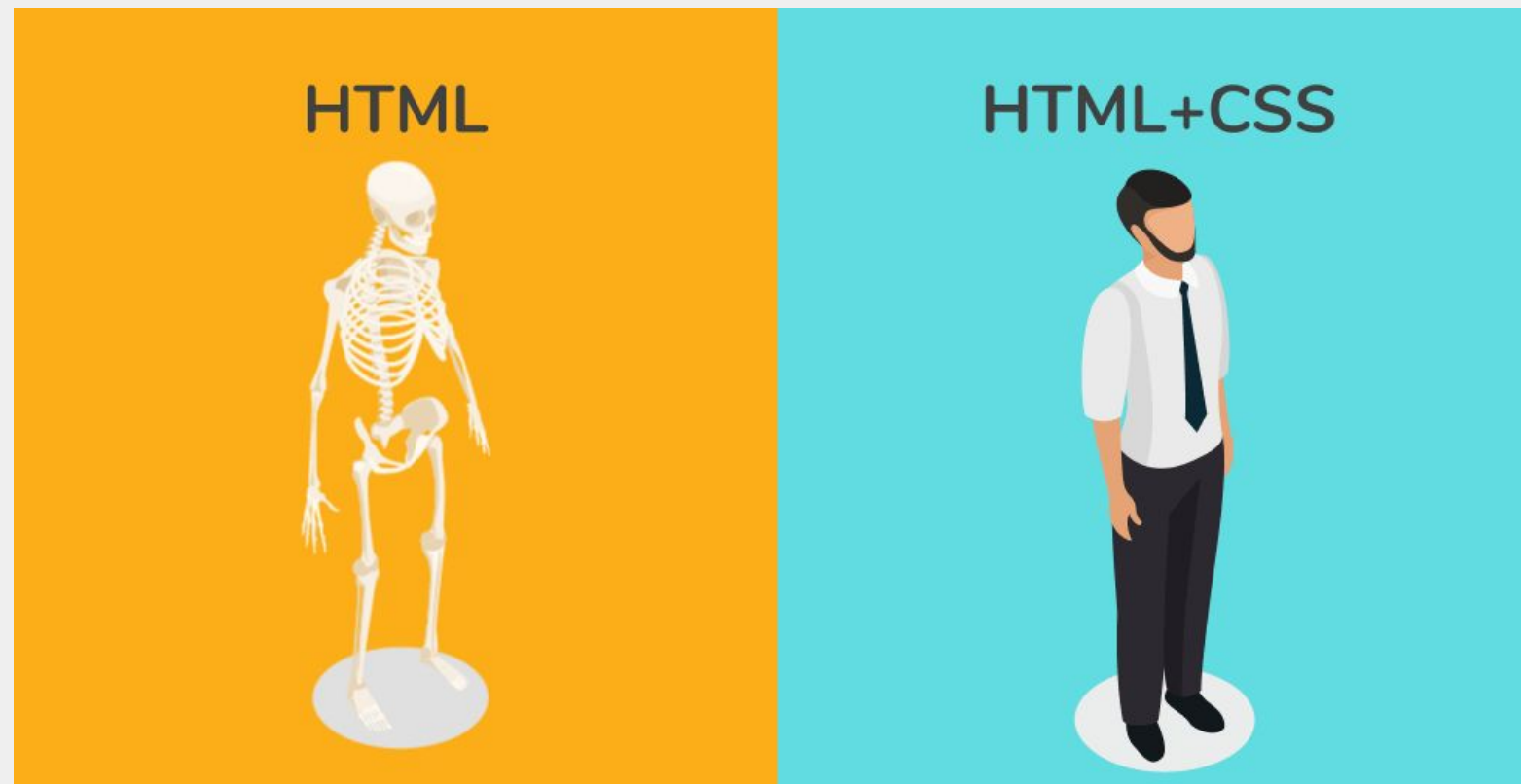
**Primary Navigation**

- 1. [Café](#)  
[Menus & Info](#)
- 2. [Wellness](#)  
[Nutrition & Sustainability](#)
- 3. [Catering](#)  
[Inspiration & Ordering](#)
- 4. [Connect](#)  
[Events & Contact](#)

Menu Close



**WELLNESS**  
**BON APPÉTIT**  
>>>

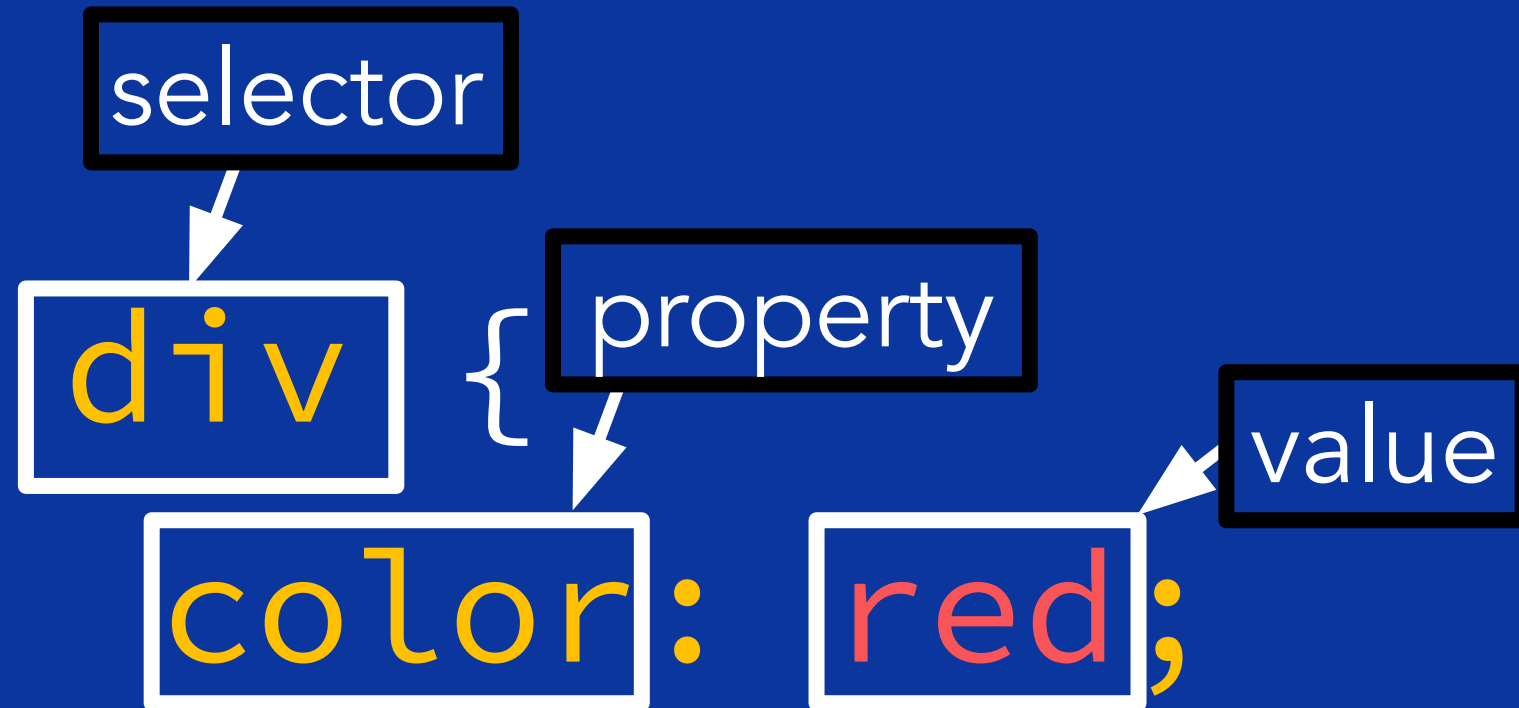


CSS = A list of descriptions

aka making things look pretty



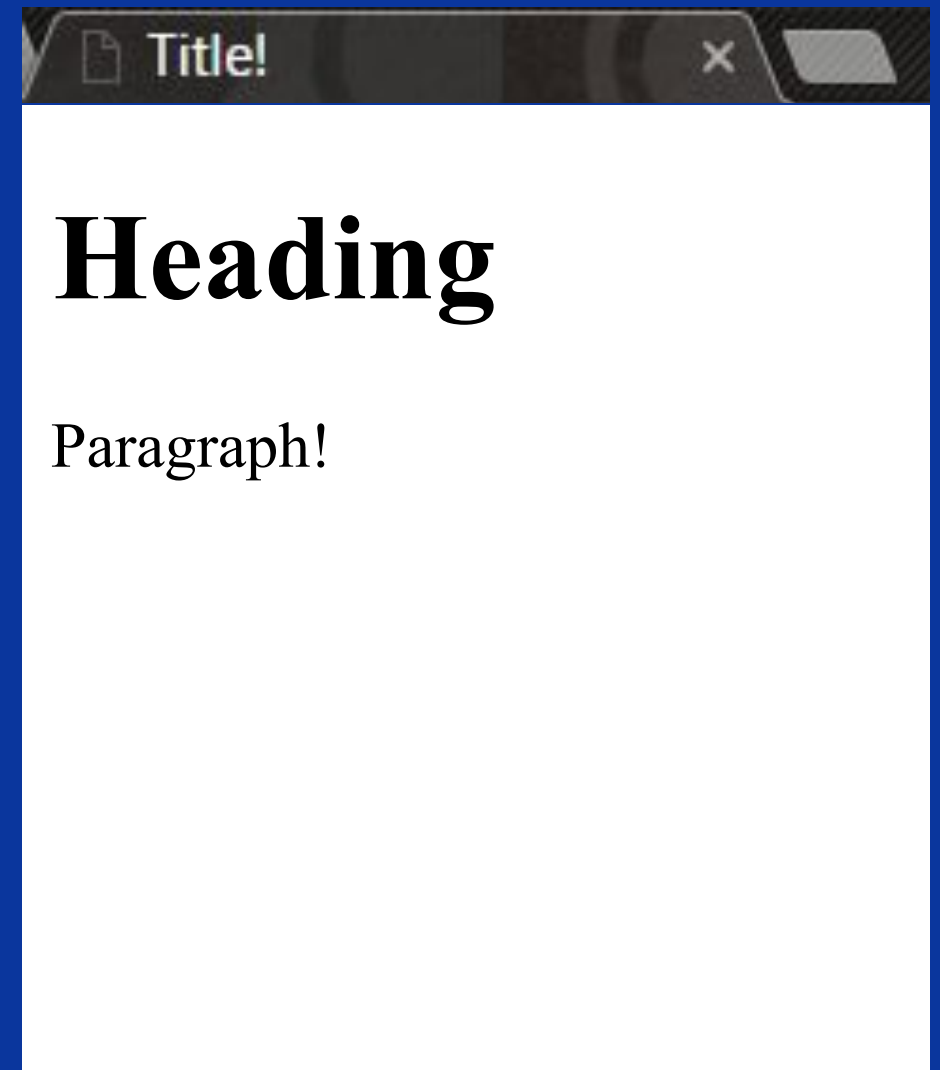
# CSS ruleset (rule)



```
font-family: Arial;  
font-size: 24pt;  
}
```

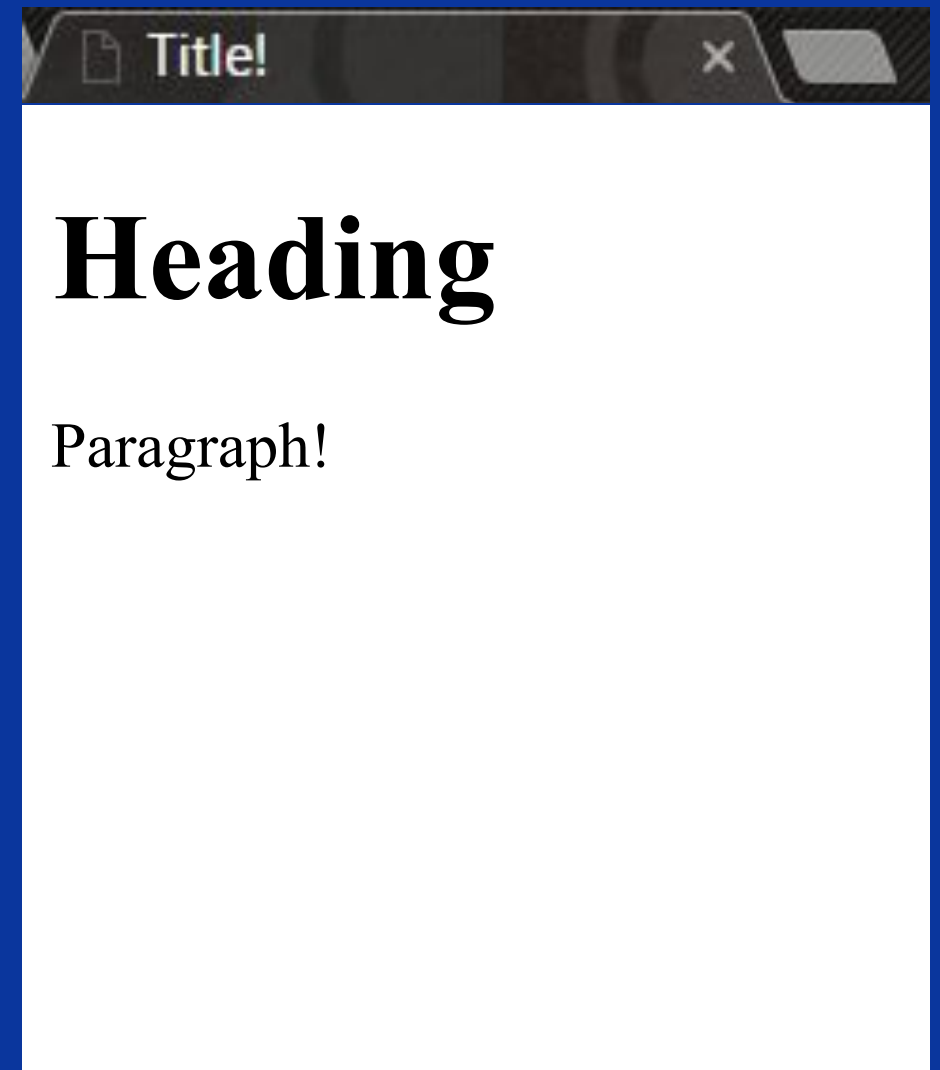
hello.html

```
<!DOCTYPE html>
<html>
  <head>
    <title>Title!</title>
  </head>
  <body>
    <h1>Heading!</h1>
    <div>Paragraph!</div>
  </body>
</html>
```



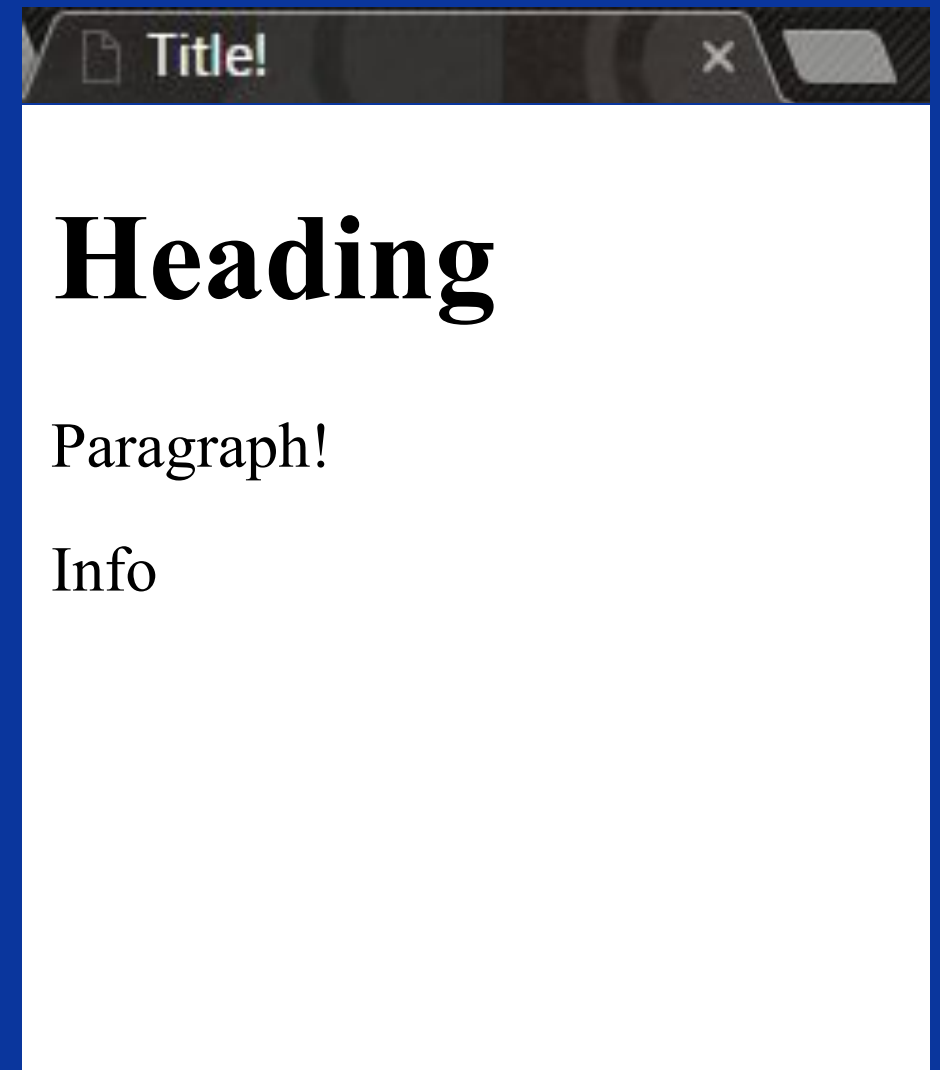
hello.html

```
<h1>Heading!</h1>  
<div>Paragraph!</div>
```



hello.html

```
<h1>Heading!</h1>  
<div>Paragraph!</div>  
<div>Info</div>
```

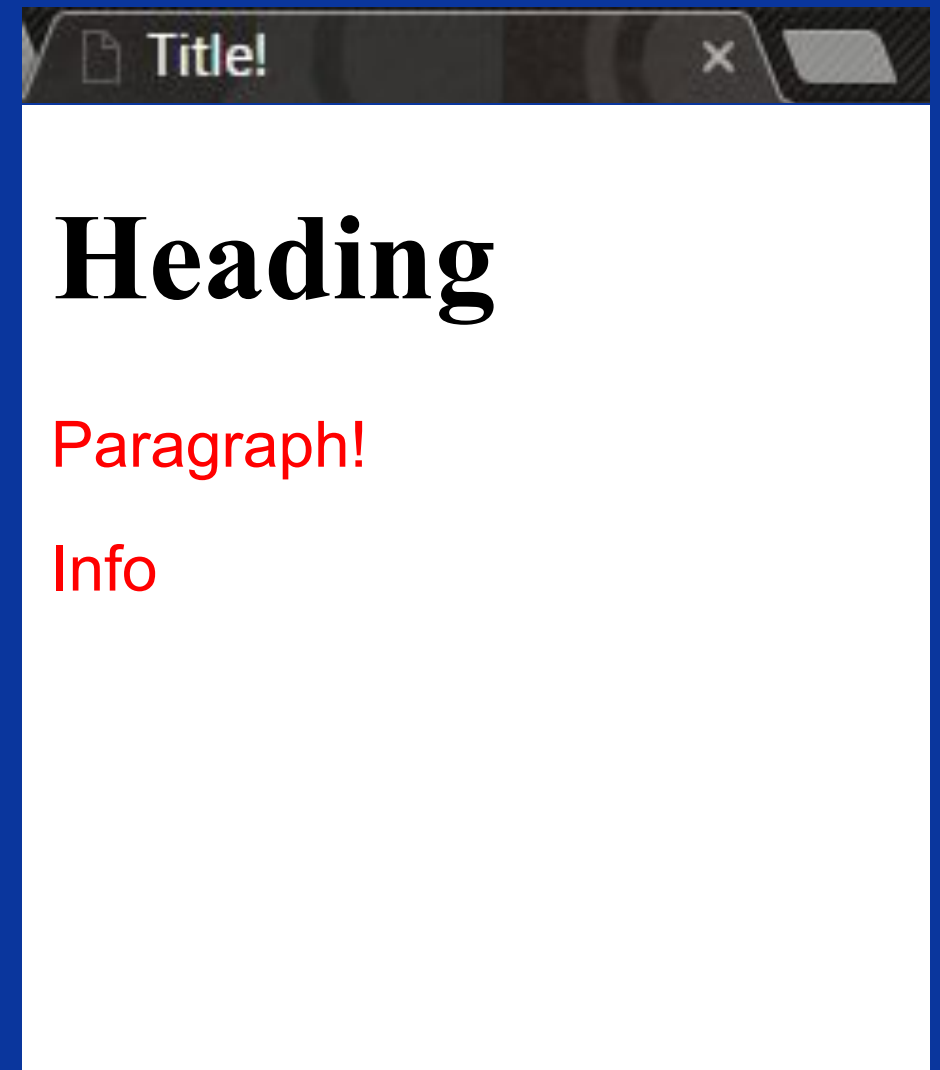


hello.html

```
<h1>Heading!</h1>
<div>Paragraph!</div>
<div>Info</div>
```

style.css

```
div {
  color: red;
  font-family: Arial;
  font-size: 24pt;
}
```

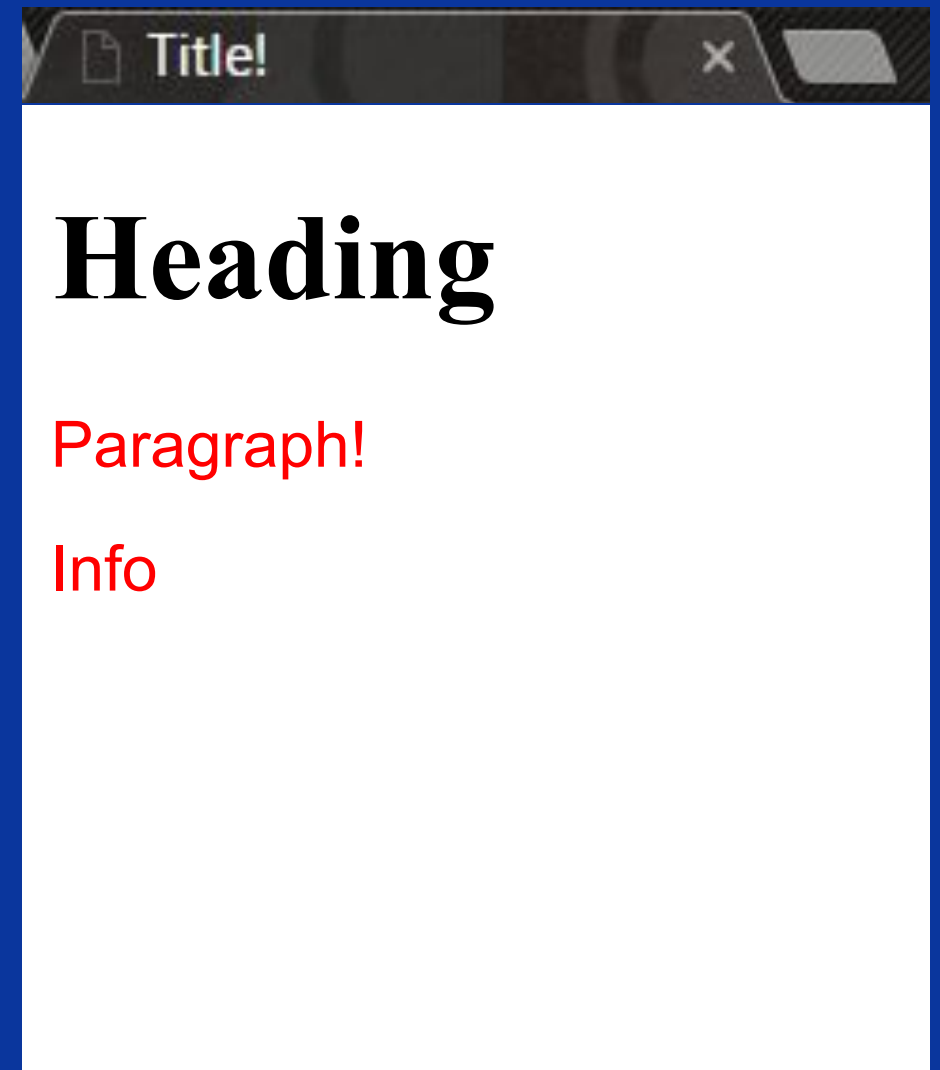


hello.html

```
<h1>Heading!</h1>  
<div>Paragraph!</div>  
<div class="info">Info</div>
```

style.css

```
div {  
  color: red;  
  font-family: Arial;  
  font-size: 24pt;  
}
```

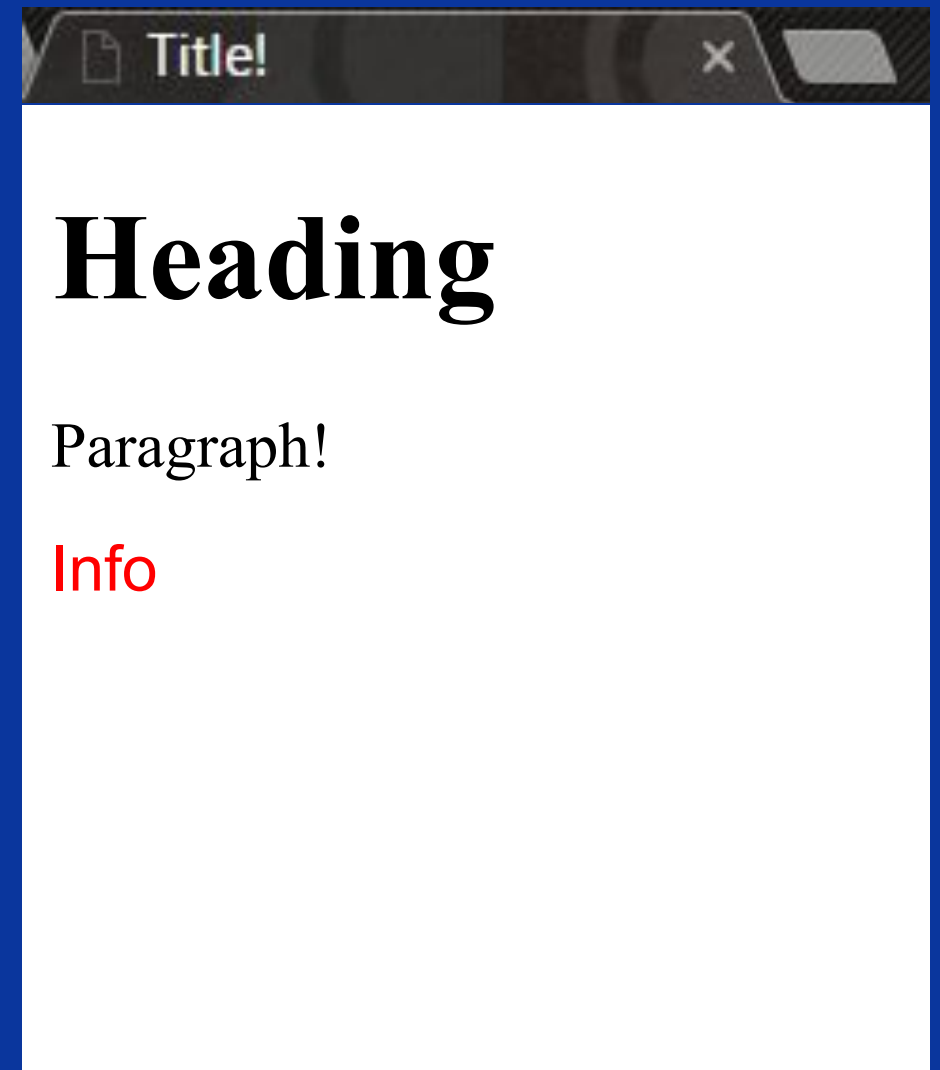


hello.html

```
<h1>Heading!</h1>  
<div>Paragraph!</div>  
<div class="info">Info</div>
```

style.css

```
.info {  
  color: red;  
  font-family: Arial;  
  font-size: 24pt;  
}
```

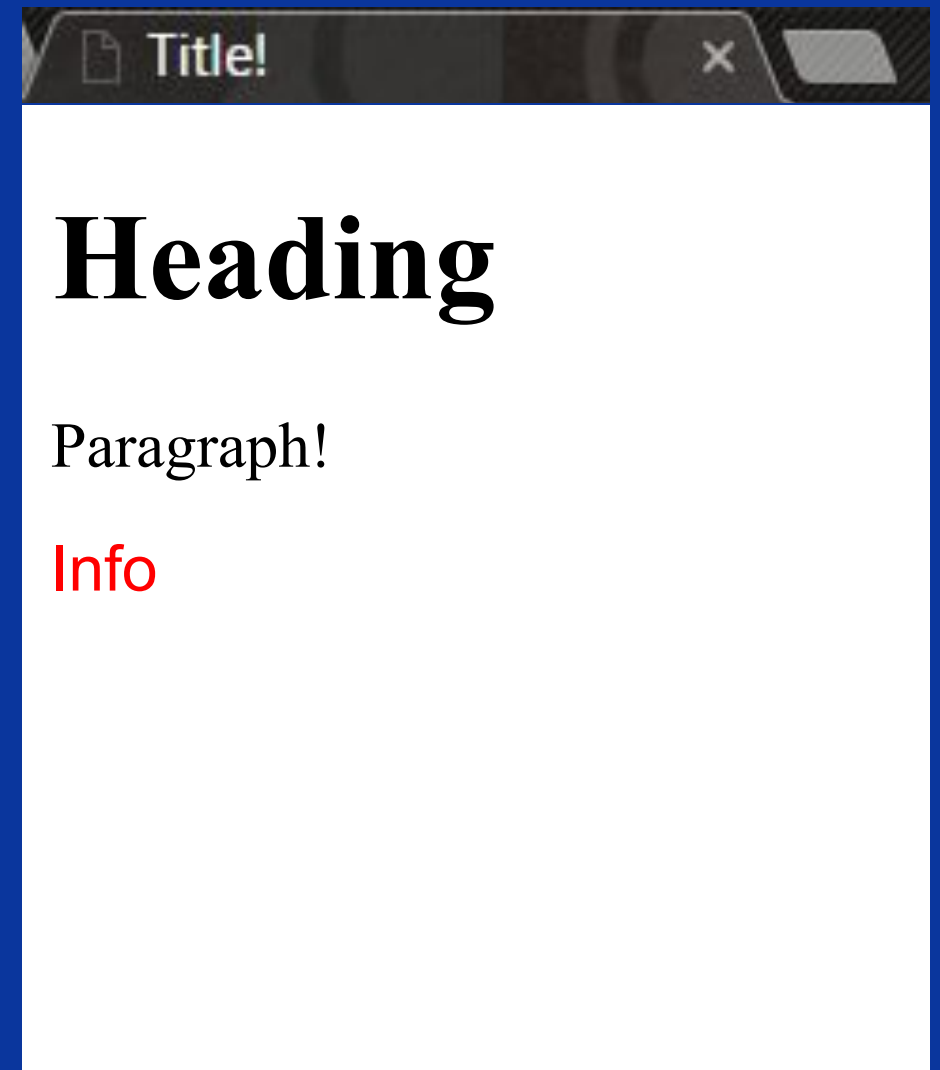


hello.html

```
<h1>Heading!</h1>  
<div>Paragraph!</div>  
<div id="unique">Info</div>
```

style.css

```
#unique {  
  color: red;  
  font-family: Arial;  
  font-size: 24pt;  
}
```





# ID vs Class

**ID**

**Class**

# ID vs Class

## ID

An element can have only one ID

## Class

An element **can have multiple classes**

# ID vs Class

## ID

An element can have only one ID

```
<div id="element-id">
```

## Class

An element can have multiple classes

```
<div class="class1  
class2 class3">
```

Ex:

```
<div class="list-item  
container">
```

# ID vs Class

## **ID**

An element can have only one ID

IDs must be unique in any given HTML document

## **Class**

An element can have multiple classes

Can use the same class on multiple elements

# ID vs Class

## **ID**

An element can have only one ID

IDs must be unique in any given HTML document

## **Class**

An element can have multiple classes

Can use the same class on multiple elements

(They can be reused!)

# ID vs Class

## ID

An element can have only one ID

IDs must be unique in any given HTML document

```
#id {  
    ...  
}
```

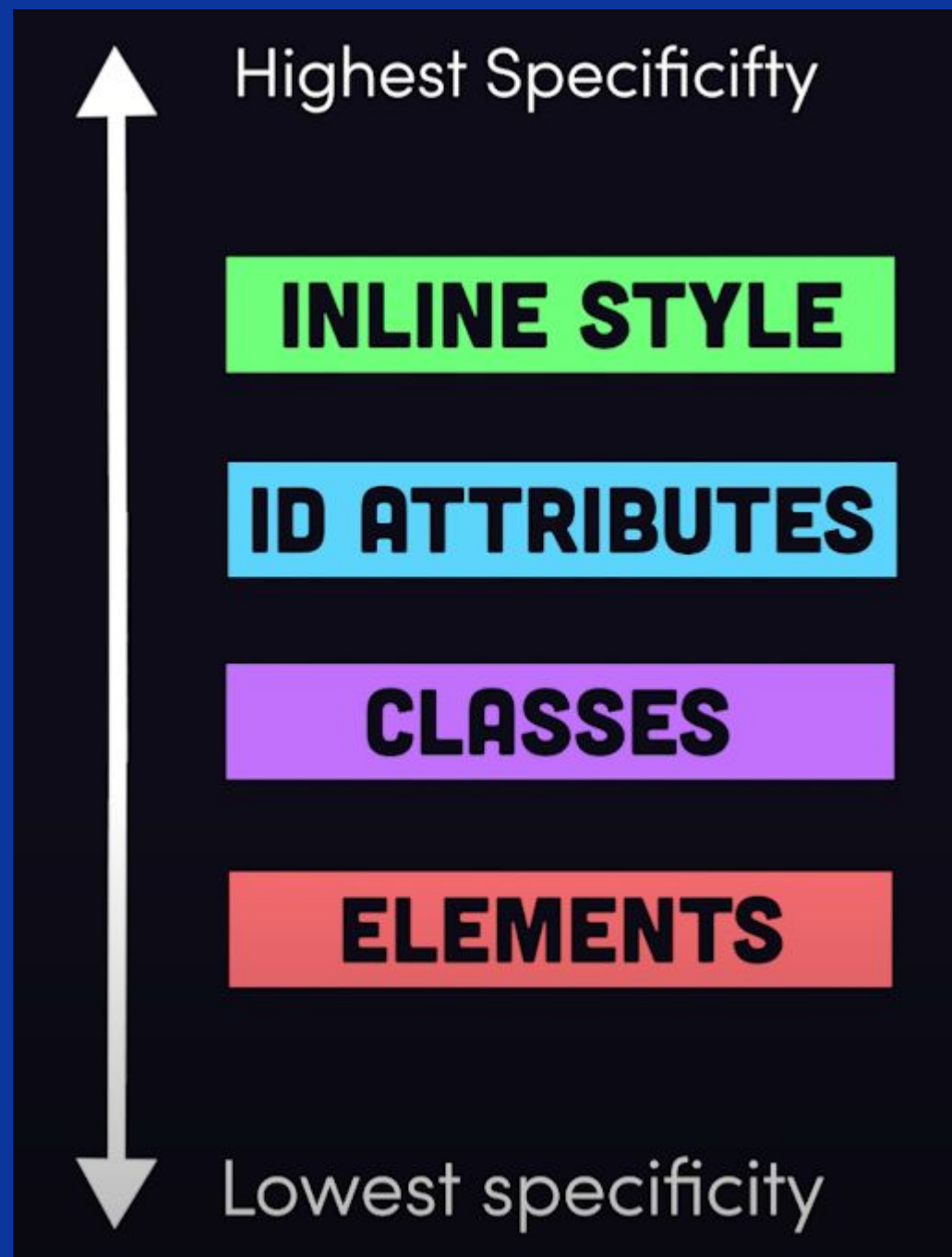
## Class

An element can have multiple classes

Can use the same class on multiple elements

```
.classname {  
    ...  
}
```

# CSS Hierarchy



(ignore)

#unique {...}

.info {...}

div {...}

read more about **specificity** of selectors [here](#)

Only use classes for  
CSS styling!





# Important Concept!

Tags such as `<h1>` have some default styling while `<div>` and `<span>` do not.

```
h1 {  
  display: block;  
  font-size: 2em;  
  margin-block-start: 0.67em;  
  margin-block-end: 0.67em;  
  margin-inline-start: 0px;  
  margin-inline-end: 0px;  
  font-weight: bold;  
}
```

```
div {  
  display: block;  
}
```

# Important Concept!

Tags such as `<h1>` have some default styling while `<div>` and `<span>` do not.

Any custom applied CSS rules will overwrite it!

## Important Concept!

Tags such as `<h1>` and `<p>` are basically the same as `<div>` except they have some default styling

**NO THIS IS BAD**

So, you can **build a webpage** using `<p>` and `<h1>`, and then **override the styling** for those elements by creating CSS rules for `p` and `h1`...

... or you could **build your webpage (almost) entirely out of divs**, create **CSS rules for various classes**, and **apply classes to divs** on the webpage!

(often a combination of both)

# Combining HTML and CSS

hello.html

```
<!DOCTYPE html>
<html>
  <head>
    <title>Title!</title>
  </head>
  <body>
    <h1>Heading!</h1>
    <p>Paragraph!</p>
  </body>
</html>
```

hello.html

```
<!DOCTYPE html>
<html>
  <head>
    <title>Title!</title>
    <link rel="stylesheet"
          href="style.css" />
  </head>
  <body>
    <h1>Heading!</h1>
    <p>Paragraph!</p>
  </body>
</html>
```

hello.html

```
<!DOCTYPE html>
<html>
  <head>
    <title>Title!</title>
    <link rel="stylesheet"
          href="style.css" />
  </head>
  <body>
    <h1 class="my-class">Heading!</h1>
    <p>Paragraph!</p>
  </body>
</html>
```

HTML/CSS in your browser...



# In Conclusion:

**HTML = nested boxes**  
**CSS = a list of descriptions**

A description in the CSS  
affects a box in the HTML

**Do Not Eat Food in the Lecture Hall or MIT will get mad at us!**

Lunch Break!  
until 1:00



We'll be getting started on Workshop 0 after lunch, so please try to get your terminal and Visual Studio Code setup!

[HERE](#) (or check Piazza)

Need help? Ask a staff member :)