

# Web

---

## Web

HTML-Hyper Text Markup Language

Get Started

Elements

Elements Structure

Elements Attributes

Practice 1: `<img>` and its attributes

Boolean Attributes

About Quotes Around Attribute Values

Nesting Element

Void Element

HTML Structure

Whitespace

Special Characters

Comments

Head-Metadata

Title `<title></title>`

Metadata `<meta>`

Icon

Applying CSS and JavaScript

The primary language tag

Text Fundamentals

Headings and Paragraphs `<h></h>` `<p></p>`

Lists

Emphasis `<em></em>`

Strong `<strong></strong>`

Italic `<i></i>`, bold `<b></b>` and underline `<u></u>`

Hyperlinks

Anatomy of a link `<a href="">Text</a>`

Block level links

Image Links

Adding Supporting information with the `title` attribute

Email links `mailto:`

URLs

Link Best Practices

Use clear link wording

Linking to non-HTML resources-----Leave clear signposts

[?] Use the download attribute when linking to a download

Advanced Text Formatting

Description lists

Quotations

Blockquotes `<blockquote>`

Inline quotations `<q>`

Citations `<cite>`

Abbreviations

Marking up contact details `<address>`

Superscript `<sup>` and subscript `<sub>`

Code

Marking up Times and Date `<time>`

Document and Website Structure

- Basic Sections
- Non-semantic wrappers `<div>` `<span>`
- Line breaks and horizontal rules `<br>` `<hr>`

Debug HTML

- Errors
- Validation

Project1: Marking Up a Letter

Project2: Structure a Web Page

Multimedia and embedding

- Images `<img>`
  - Attribute: `<src>` `<alt>`
  - `width` and `height`
  - `title`
- Annotation with `<figure>` `<figcaption>`
- CSS Background Images
- Problem: media assets and licensing
  - License types:
  - Searching for permissively-licensed images
- Video and Audio `<video>` `<audio>`
  - `<video>`
    - `<video>` Elements
    - Enhancement
    - Other `<video>`
    - `<audio>` Element
    - Javascript about audio
    - 显示音轨文本WebVTT
- Other embedded technologies `object` `iframe`
  - History
  - Embedding Web
  - Plugin technologies
    - `<iframe>` (along with `<canvas>` `<video>` etc)
  - Security Problem of `<iframe>`
  - `<embed>` `<object>`

---

## HTML-Hyper Text Markup Language<sup>1</sup>

A markup language that tells web browsers how to structure the web pages you visit.

1. Feature:

1. Not Case-sensitive
  1. Example: `<title>` `<Title>` `<TITLE>`
  2. Better write all tags in lowercase

# Get Started

## Elements

Based on [learning-site](#) project.

### Elements Structure

```
1 | <p> //Opening tag  
2 |     My Cat is very cute. //Content  
3 | </p> //Closing tag
```

### Elements Attributes

Elements have attributes

```
1 | <p class="editor-note"> My Cat is very grumpy</p>
```

- `class`:
  - An identifying name
  - Used to target the element with style information
- Attribute Structure:
  - `space` between it and the element name
  - The attribute name, followed by `=`
  - An attribute value, wrapped with `""`

### Practice 1: `<img>` and its attributes

`src`: a required attribute that specifies the location of the image

`alt`: specifies a text description of the image.

`width`: specifies the width of the image with the unit being pixels

`height`: specifies the height of the image with the unit being pixels

```
1 | <img  
2 |     src = "https://static.zerochan.net/Paimon.full.3141703.png"  
3 |     alt = "Paimon: a lovely character in Genshin"  
4 |     width = 288px  
5 |     height = 512px  
6 |   />
```

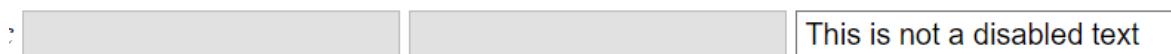


## Boolean Attributes

Attributes that written without values.

- Only have one value, which is generally the same as the attribute name

```
1 <input type="text" disabled = "disabled" />
2 <input type="text" disabled />
3 <input type="text" />
```



- `disabled`: To disable the form input elements to the user can't make entries
  - Typically have a grayed-out appearance

## About Quotes Around Attribute Values

### 1. Omitting

- Only permitted in certain circumstances
- May break your markup in other circumstances

```
1 <a href=https://developer.mozilla.org/en-US/docs/Learn/HTML/Introduction_to_HTML/Getting_started#omitting_quotes_around_attribute_values>
2   This is the tutor for this page
3 </a>
```

- `a`: Anchor
- `href`: hyper reference

- Wrong usage:

```

1 | <a title="The Learning Site>
2 |   This is the tutor for this page
3 | </a>
```

```

<a href="https://developer.mozilla.org/
  title="The Learning Site">
  This is the tutor for this page
</a>
```

## 2. Single/Double

- Feel free to choose which one you prefer.
- But don't mixing of quotes
- Using &quot; to nesting

```

1 | <a href="https://developer.mozilla.org/" title="An
  &quot;interesting&quot; reference">A Example Link</a>
```

## Nesting Element

```

1 | <p>
2 |   My cat is <strong>very</strong> grumpy
3 | </p>
```

This is a NESTING test

## Void Element

```

1 | 
```



Paimon: A lovely character in Genshin

- `alt`: for
  - `/`: There is no requirement to use it, but it is a valid syntax for making HTML to be valid XML
- 

## HTML Structure

```
1 <!doctype html>
2 <html lang="en-US">
3   <head>
4     <meta charset="utf-8" />
5     <meta name="viewport" content="width=device-width">
6     <title>My test page</title>
7   </head>
8   <body>
9     <p>This is my page</p>
10    </body>
11 </html>
```

- `!doctype html`:

The doctype. When HTML was young (1991-1992), doctypes were meant to act as links to a set of rules that the HTML page had to follow to be considered good HTML. Doctypes used to look something like this:

```
1 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

More recently, the doctype is a historical artifact that needs to be included for everything else to work right. `<!DOCTYPE html>` is the shortest string of characters that counts as a valid doctype. That is all you need to know!

- `<html>`:
    - Function: Wraps all the content on the page
    - Sometimes known as the root element
  - `<head>`:
    - Function: Act as a container for everything you want to include on the HTML page
      - Not content, but includes keywords and a page description that would appear in search results, CSS to style content, character set declarations, and more.
  - `<meta charset="utf-8">`: metadata <sup>2</sup>
  - `<title>`
  - `<body>`
- 

## Whitespace

No matter how much whitespace you use inside HTML element content (which can include one or more space characters, but also line breaks), the HTML parser reduces each sequence of whitespace to a single space when rendering the code

- For readability

**Note:** Accessing the [innerHTML](#) of elements from JavaScript will keep all the whitespace intact. This may return unexpected results if the whitespace is trimmed by the browser.(Example as following)

```
1 const nowhitespace = document.getElementById("nowhitespace").innerHTML;
2 console.log(nowhitespace);
3 // "Dogs are silly."
4
5 const whitespace = document.getElementById("whitespace").innerHTML;
6 console.log(whitespace);
7 // "Dogs
8 //     are
9 //         silly."
```

## Special Characters

Special character	Character reference equivalent
<	&lt;
>	&gt;
"	&quot;
'	&apos;
&	&amp;

```
1 |
```

- Character reference: special codes that represent characters.
- [For More](#)

## Comments

```
1 | <!-- This is a comment -->
```

## Head-Metadata

The content in `<head></head/>`

## Title <title></title>

```
1 | <title>ThisIsATitle</title>
```

- The metadata that represent the title of the overall HTML document(not the content)
  - As the suggested bookmark name
  - As the search results

## Metadata <meta>

```
1 | <!-- Character Encoding -->
2 | <meta charset="utf-8" />
```

- `utf-8`: a universal character set that include pretty much any character from any human languages

```
1 | <meta name="author" content="Yujie" />
2 | <meta name="Description" content="A Website to introduce Paimon~" />
```

- **Specifying An author:** Useful to be able to understand who wrote the page(for contact)
- **Specifying a description:** Used on search engine result pages
  - Sitelink: Configurable
- **Specifying keywords:** provide keywords for search engines to determine the relevance of that page for different search terms to determine the relevance of that page for different search terms
  - Ignored by search engines. Spammers will filling it



- Proprietary creations designed to provide certain sites with specific information they can use
  - Example: Open Graph Data(metadata protocol), Facebook and WDN docs

When you link to MDN Web Dos on Facebook, the link appears along with an image and description

```
1 | <meta  
2 |   property="og:image"  
3 |   content="https://developer.mozilla.org/mdn-social-share.png" />  
4 | <meta  
5 |   property="og:description"  
6 |   content="The Mozilla Developer Network (MDN) provides  
7 | information about open web technologies including HTML, CSS, and APIs  
8 | for both websites  
9 | and HTML Apps." />  
9 | <meta property="og:title" content="Mozilla Developer Network" />
```



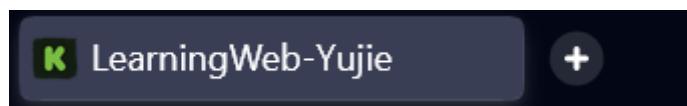
## Icon

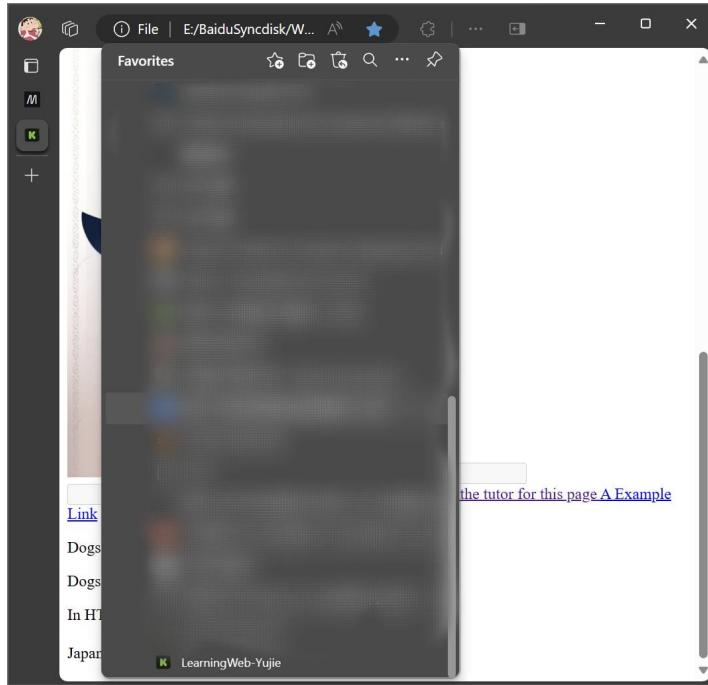
Favicon-favorites icon:

- Concept: use the custom icon in the "favorites" or "bookmark" lists
- Feature:
  - With `.ico` format (most also support `.gif` `.png` ...)

### 1. Local Icon

```
1 | <link rel="icon" href="./images/favicon.ico" type = "image/x-icon" />
```





## 2. Third-party icon

```
1 <!-- third-generation iPad with high-resolution Retina display: -->
2 <link
3   rel="apple-touch-icon"
4   sizes="144x144"
5   href="https://developer.mozilla.org/static/img/favicon144.png" />
6 <!-- iPhone with high-resolution Retina display: -->
7 <link
8   rel="apple-touch-icon"
9   sizes="114x114"
10  href="https://developer.mozilla.org/static/img/favicon114.png" />
11 <!-- first- and second-generation iPad: -->
12 <link
13   rel="apple-touch-icon"
14   sizes="72x72"
15   href="https://developer.mozilla.org/static/img/favicon72.png" />
16 <!-- non-Retina iPhone, iPod Touch, and Android 2.1+ devices: -->
17 <link
18   rel="apple-touch-icon"
19   href="https://developer.mozilla.org/static/img/favicon57.png" />
20 <!-- basic favicon -->
21 <link
22   rel="icon"
23   href="https://developer.mozilla.org/static/img/favicon32.png" />
24
```

## Applying CSS and JavaScript

```
1 | <link rel="stylesheet" href="styles/styles.css" />
```

- `rel="stylesheet"`: It is the document's stylesheet
- `href` : contains the path to the stylesheet file

```
1 <script src="scripts/scripts.js" defer></script>
2 <!-- Most Reliable way to Apply HTML-->
```

- `defer`: instructs the browser to load the JavaScript **after** the page has finished parsing the HTML
  - Don't get error resulting from JavaScript trying to access an HTML element that doesn't exist on the page yet

## The primary language tag

- Be indexed more effectively by search engine
- Useful to people with visual impairments using screen readers(不同语言同字不同音)

### 1. For whole page

```
1 <html lang="en-US">
2 ...
3 </html>
```

### 2. For part of page to recognized as different languages

```
1 <p>Japanese example: <span lang="ja">ご飯が熱い。</span>.</p>
```

# Text Fundamentals

## Headings and Paragraphs `<h></h>` `<p></p>`

Advantages of heading:

- Let the browser know which is heading
- Make the Web page more readability
- Search engines indexing the page
- Good for the screen reader
- Make CSS/JavaScript can target these elements
- Heading elements are semantic elements

```
1 <h1>Heading1</h1>
2 <h2>Heading2</h2>
3 <h3>Heading3</h3>
4 <h4>Heading4</h4>
5 <h5>Heading5</h5>
6 <h6>Heading6</h6>
7 <h7>Heading7</h7>      <!--Invalid-->
8 <p>Paragraph</p>
```

```
Heading1  
Heading2  
Heading3  
Heading4  
Heading5  
Heading6  
Heading7  
Paragraph
```

- If we do not use this text structure, the text becomes hard to be organized and read

```
1 This is line1.<br>  
2 This is line2.  
3 This is line3.  
4 This is line4.
```

This is line1.  
This is line2. This is line3. This is line4.

- Make any element look like a top level heading:

But `<span>` element Has no semantics, it is not preferred.

```
1 <span style="font-size: 32px; margin: 21px 0; display: block;">  
2     I do not have any semantic!  
3 </span>
```

## Lists

### 1. Unordered list `<ul></ul>`

```
1 <ul>  
2     <li>milk</li>  
3     <li>eggs</li>  
4     <li>bread</li>  
5     <li>hummus</li>  
6 </ul>
```

### 2. Ordered list `<ol></ol>`

```
1 <ol>  
2     <li>milk</li>  
3     <li>eggs</li>  
4     <li>bread</li>  
5     <li>hummus</li>  
6 </ol>
```

### 3. Nested lists

```
1 <ul>
2   <li>Colors</li>
3   <ol>
4     <li>Red</li>
5     <li>white</li>
6     <ul>
7       <li>Not the original color</li>
8     </ul>
9     <li>Yellow</li>
10    <li>Green</li>
11  </ol>
12 </ul>
```

- Colors
  - 1. Red
  - 2. White
    - Not the original color
  - 3. Yellow
  - 4. Green

## Practice

```
1 <h1>Quick hummus recipe</h1>
2
3 <p>This recipe makes quick, tasty hummus, with no messing. It has been
adapted from a number of different recipes that I have read over the
years.</p>
4
5 <p>Hummus is a delicious thick paste used heavily in Greek and Middle
Eastern dishes. It is very tasty with salad, grilled meats and pitta
breads.</p>
6
7 <h2>Ingredients</h2>
8
9 <ul>
10   <li>1 can (400g) of chick peas (garbanzo beans)
11   <li>175g of tahini</li>
12   <li>6 sundried tomatoes</li>
13   <li>Half a red pepper</li>
14   <li>A pinch of cayenne pepper</li>
15   <li>1 clove of garlic</li>
16   <li>A dash of olive oil</li>
17 </ul>
18 <h2>Instructions</h2>
19 <ol>
20   <li>Remove the skin from the garlic, and chop coarsely
21   <li>Remove all the seeds and stalk from the pepper, and chop
coarsely
22   <li>Add all the ingredients into a food processor
23   <li>Process all the ingredients into a paste
```

```
24     <li>If you want a coarse "chunky" hummus, process it for a short  
time  
25     <li>If you want a smooth hummus, process it for a longer time  
26 </ol>  
27 <p>For a different flavor, you could try blending in a small measure of  
lemon and coriander, chili pepper, lime and chipotle, harissa and mint,  
or spinach and feta cheese. Experiment and see what works for you.</p>  
28  
29 <h2>Storage</h2>  
30  
31 <p>Refrigerate the finished hummus in a sealed container. You should be  
able to use it for about a week after you've made it. If it starts to  
become fizzy, you should definitely discard it.</p>  
32  
33 <p>Hummus is suitable for freezing; you should thaw it and use it  
within a couple of months.</p>
```

## Live output

# Quick hummus recipe

This recipe makes quick, tasty hummus, with no messing. It has been adapted from a number of different recipes that I have read over the years.

Hummus is a delicious thick paste used heavily in Greek and Middle Eastern dishes. It is very tasty with salad, grilled meats and pitta breads.

## Ingredients

- 1 can (400g) of chick peas (garbanzo beans)
- 175g of tahini
- 6 sundried tomatoes
- Half a red pepper
- A pinch of cayenne pepper
- 1 clove of garlic
- A dash of olive oil

## Instructions

1. Remove the skin from the garlic, and chop coarsely
2. Remove all the seeds and stalk from the pepper, and chop coarsely
3. Add all the ingredients into a food processor
4. Process all the ingredients into a paste
5. If you want a coarse "chunky" hummus, process it for a short time
6. If you want a smooth hummus, process it for a longer time

## Emphasis <em></em>

Emphasis in Italics, which is recognized by screen readers(can be speak in a different tone of voice)

```
1 | <p>I am <em>glad</em> you weren't <em>late</em></p>
```

I am *glad* you weren't *late*

## Strong <strong></strong>

Strong in bold, which is recognized by screen readers(can be speak in a different tone of voice)

```
1 | <p> Do <strong>not</strong> be late</p>
```

**Do not be late**

## Italic <i></i>, bold <b></b> and underline <u></u>

```
1 | <p>
2 |   <i>Italic</i>
3 |   <b>Bold</b>
4 |   <u>Underline</u>
5 | </p>>
```

*Italic* **Bold** Underline

[Practice1: text fundamentals](#)

## Hyperlinks

Hyperlinks are really important feature of Web

### Anatomy of a link <a href="">Text</a>

```
1 | <a href="https://developer.mozilla.org/en-US/docs/Learn/HTML/Introduction_to_HTML/Creating_hyperlinks">
2 |   Link to Tutor
3 | </a>
```

## Block level links

Almost any content can be made into a link even block-level elements

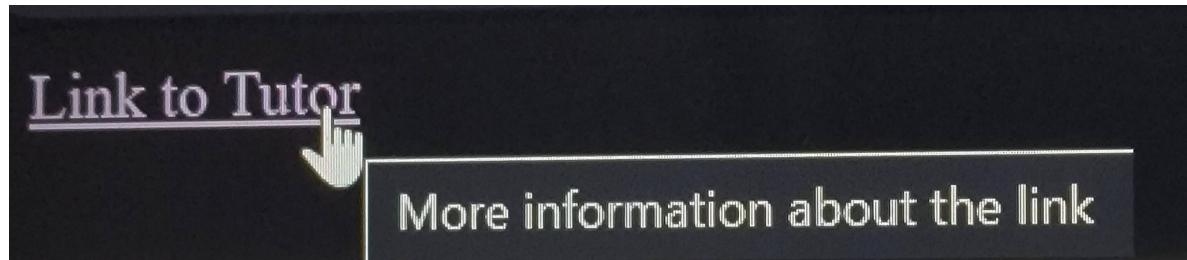
```
1 <a href="https://developer.mozilla.org/en-US/docs/Learn/HTML/Introduction_to_HTML/Creating_hyperlinks">
2   <h1>
3     A Block level links
4   </h1>
5   <ul>
6     <li>Even a list</li>
7   </ul>
8 </a>
```

## Image Links

```
1 <a href="https://developer.mozilla.org/en-US/docs/Learn/HTML/Introduction_to_HTML/Creating_hyperlinks">
2   
3 </a>
```

## Adding Supporting information with the `title` attribute

```
1 <a
2   href="https://developer.mozilla.org/en-US/docs/Learn/HTML/Introduction_to_HTML/Creating_hyperlinks"
3   title="More information about the link">
4   Link to Tutor
5 </a>
```



## Email links `mailto:`

1. Base email link:

```
1 <a href="mailto:liu619666@163.com">
2   Send email to Yujie~
3 </a>
```

2. Email link with more details:

```
1 <a
2   href="mailto:liu619666@163.com"
3   cc="name2@rapidtables.com"
4   &bcc="name3@rapidtables.com"
5   &subject="The subject of the email"
6   &body="The body of the email">
7   Send mail with cc/bcc/subject/body
8 </a>
```

If do not use `""`, use `%20` to replace the space

## URLs

- `URL` : Uniform Resource Locator
  - A string of text that defines where something is located on the Web
- Same directory:
  - only use the `filename`
- Moving down into subdirectories:
  - `thisFileFolder/filename`
- Moving back into parent directories:
  - `../FileFolder/filename`
  - `../../../../FileFolder/filename`
- Document Fragments

```
1 | <h2 id="mail">E-mail</h2>
```

```
1 | <p>jump to
2 |   <a href="index.html#mail">
3 |     E-mail
4 |   </a>
5 | </p>
```

[jump to E-mail](#)

[Link to Tutor](#)



## E-mail

I do not have any semantic!

*This is the first test with italic*

My Cat is very grumpy

This is a **NESTING** test

## Link Best Practices

### Use clear link wording

- Screen reader users like jumping around from link to link on the page, and reading links out of context
- Search engines use link text to index target files, so it is a good idea to **include keywords** in your link text to effectively describe what is being linked to
- Visual readers skim over the page rather than reading every word, and their eyes will be drawn to page features that stand out, like links.

**It is important to make descriptive link text useful**

Tips:

- Don't repeat the ugly URL as part of the link text
- Don't say "link" or "links to". They are noise even for people using screen reader(screen reader will tell them there's a link)
- Keep the link text as short as possible(Because the screen header need to interpret the entire link text)
- Don't use too much same link text on different links

- Good link text:

```
1 <p>
2   <a href="https://www.mozilla.org/firefox/">
3     Download Firebox
4   </a>
5 </p>
```

- Bad link text:

```
1 <p>
2   <a href="https://www.mozilla.org/firefox/">
3     click here
4   </a>
5     to download Firefox
6 </p>
```

### Linking to non-HTML resources----Leave clear signposts

When linking to a resource that will be downloaded, streamed, or has another potentially unexpected effect,

Should add clear wording to reduce any confusion

```
1 <p>
2   <a href="https://www.example.com/large-report.pdf">
3     Download the sales report (PDF, 10MB)
4   </a>
5 </p>
6
7 <p>
8   <a href="https://www.example.com/video-stream/" target="_blank">
9     Watch the video (stream opens in separate tab, HD quality)
10  </a>
11 </p>
12
```

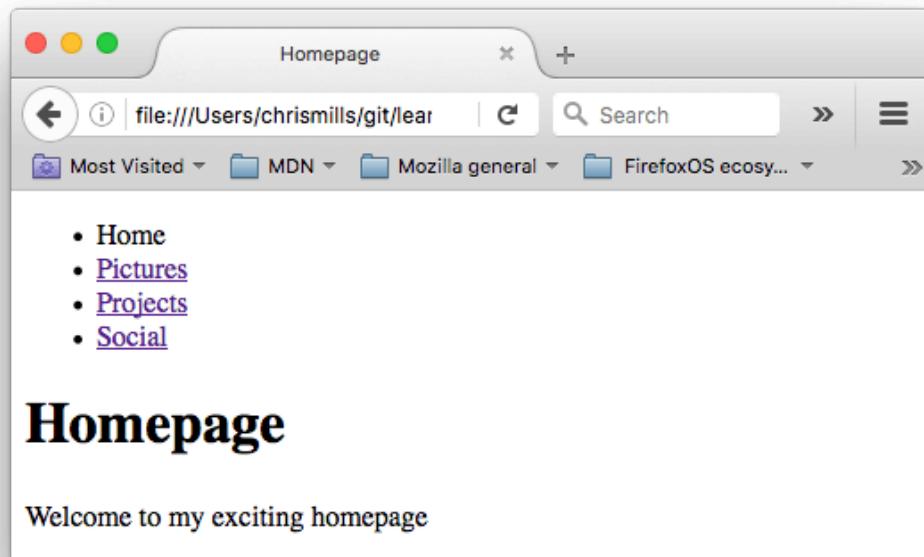
## [?] Use the download attribute when linking to a download

Provide a default save filename

```
1 <a
2   href="https://download.mozilla.org/?product=firefox-latest-
3   ss1&os=win64&lang=en-US"
4   download="firefox-latest-64bit-installer.exe">
5   Download Latest Firefox for Windows (64-bit) (English, US)
6 </a>
```

Practice: Creating a navigation menu

- [index.html](#) ↗
- [projects.html](#) ↗
- [pictures.html](#) ↗
- [social.html](#) ↗



## Advanced Text Formatting

### Description lists

To mark up a set of items and their associated description,  
such as terms and definitions, or questions and answers

```
1 <dl>
2   <dt>soliloquy</dt>
3   <dd>In drama, where a character speaks to themselves...</dd>
4   <dt>Monologue</dt>
5   <dd>
6     In drama, where a character speaks their thoughts out loud to share
7     them
8   </dd>
9   <dt>Aside</dt>
10  <dd>In drama, where a character shares a comment only with the audience
    for...</dd>
11 </dl>
```

- `dd` : description definition
- `dt` : description term
- `dl` : description list

#### Soliloquy

In drama, where a character speaks to themselves...

#### Monologue

In drama, where a character speaks their thoughts out loud to share them

#### Aside

In drama, where a character shares a comment only with the audience for...

- Multiple descriptions for one term

```
1 <dl>
2   <dt>Apple</dt>
3   <dd>It is red</dd>
4   <dd>It is fruit, not vegetables.</dd>
5 </dl>
```

Apple  
It is red  
It is fruit, not vegetables.

## Quotations

Marking up a block or inline quotation

### Blockquotes `<blockquote>`

```
1 <blockquote
2   cite="https://developer.mozilla.org/en-
3   US/docs/Web/HTML/Element/blockquote">
4     <p>
5       The <strong>HTML <code>&lt;blockquote&gt;</code> Element </strong>
6       (or <em>HTML Block Quotation Element</em>)
7       indicates that the enclosed text is an extended quotation
8   </p>
9 </blockquote>
```

- If a section of block level content is quoted from somewhere else, you should wrap it inside `<blockquote>`
- `cite`: include the URL pointing to the source of the quote
- Browser default styling will render this as an indented paragraph

The **HTML `<blockquote>` Element** (or *HTML Block Quotation Element*) indicates that the enclosed text is an extended quotation

The **HTML `<blockquote>` Element** (or *HTML Block Quotation Element*) indicates that the enclosed text is an extended quotation

Soliloquy  
In drama, where a character speaks to themselves...  
Monologue  
In drama, where a character speaks their thoughts out loud to share them  
Aside  
In drama, where a character shares a comment only with the audience for...

### Inline quotations `<q>`

```
1 <q
2   cite="https://developer.mozilla.org/en-US/docs/Web/HTML/Element/q">
3     intended for short quotations that don't require paragraph breaks
4   </q>
```

- Browser default styling will render this as normal text put in quotes to indicate a quotation

Here is a inline quotations

“ intended for short quotations that don't require paragraph breaks ”

### Citations `<cite>`

1. The content of the `cite` attribute is not used by browsers, screen readers, etc
2. There is no way to get the browser to display the contents of `cite` just in HTML
3. So, there is a `<cite>` element: contain the title of the resource being quoted.

```
1 <a href="/en-US/docs/Web/HTML/Element/blockquote">
2   <ctie>MDN blockquote page</ctie>
3 </a>
```

- Italic font by default????remaining

According to [MDN blockquote page](#)

### Abbreviations <sup>5</sup>

Wrap around an abbreviation or acronym <sup>5</sup>, provide a full expansion of the term in plain text on first use

```
1 <p>
2   We use <abbr>HTML</abbr>, Hypertext Markup Language, to structure our Web
3   documents<br>
4   I think <abbr title="Reverend">Rev.</abbr> Green did it in the kitchen
5   with the chainsaw.
6 </p>
```

We use HTML, Hypertext Markup Language, to structure our Web documents  
I think Rev. Green did it in the kitchen with the chainsaw.

According | [Reverend](#) | [blockquote page](#)

`<acronym>` should not be used because it has been replaced by `<abbreviations>`

### Marking up contact details `<address>`

Provide contact information for the document contained with the nearest `<article>`  
`<body>`

Used in the footer of a site

```
1 <address>
2   <p>
3     Chiris Mills<br />
4     Manchester<br />
5     The Grim North<br />
6     UK
7   </p>
8   <ul>
9     <li>Tel: 01234 567 890</li>
10    <li>Email: me@grim-north.co.uk</li>
11  </ul>
12 </address>
```

```
1 <address>
2   Page written by <a href="../authors/chris-mills">Chris Mills</a>
3 </address>
```

*Page written by Chris Mills*

*Chiris Mills  
Manchester  
The Grim North  
UK*

- *Tel: 01234 567 890*
- *Email: me@grim-north.co.uk*

## Superscript `<sup>` and subscript `<sub>`

```
1 <p>
2   x<sup>2</sup>
3   x<sub>2</sub>
4 </p>
```

$$x^2 \ x_2$$

## Code

- `<code>`: For marking up generic pieces of code
- `<pre>`: (Have format with retaining whitespace) For retaining whitespace (generally code blocks) — if you use indentation or excess whitespace inside your text, browsers will ignore it and you will not see it on your rendered page. If you wrap the text in `<pre></pre>` tags however, your whitespace will be rendered identically to how you see it in your text editor.<sup>6</sup>
- `<var>`: For specifically marking up variable names

- `<kbd>`: For marking up keyboard (and other types of ) input entered into the computer
- `<samp>`: For marking up the output of a computer program

```

1 <pre><code>
2   const para = document.querySelector('p');
3
4   para.onclick = function(){
5     alert('Owww, stop poking me!');
6   }
7 </code></pre>
8
9 <p>
10    You shouldn't use presentational elements like <code>&lt;font&gt;</code>
11    and <code>&lt;center&gt;</code>
12 </p>
13 <p>
14    In the above JavaScript example, <var>para</var>
15    represents a paragraph element
16 </p>
17
18 <p>
19    Select all the text with <kbd>Ctrl</kbd>/<kbd>Cmd</kbd> + <kbd>A</kbd>.
20 </p>
21
22 <pre>
23 $ <kbd>ping mozilla.org</kbd>
24 <samp>PING mozilla.org (63.245.215.20): 56 data bytes
25 64 bytes from 63.245.215.20: icmp_seq=0 ttl=40 time=158.233 </samp>
26 </pre>

```

```

const para = document.querySelector('p');

para.onclick = function(){
  alert('Owww, stop poking me!');
}

```

You shouldn't use presentational elements like `<font>` and `<center>`

In the above JavaScript example, *para* represents a paragraph element

Select all the text with `Ctrl/Cmd + A`.

```

$ ping mozilla.org
PING mozilla.org (63.245.215.20): 56 data bytes
64 bytes from 63.245.215.20: icmp_seq=0 ttl=40 time=158.233

```

## Marking up Times and Date `<time>`

Allow you to attach an unambiguous, machine-readable time/date for automatically grab the dates of all events in a page and insert them into a calendar

```

1 <!-- Standard simple date -->
2 <time datetime="2016-01-20">20 January 2016</time>
3 <!-- Just year and month -->
4 <time datetime="2016-01">January 2016</time>
5 <!-- Just month and day -->
6 <time datetime="01-20">20 January</time>
7 <!-- Just time, hours and minutes -->
8 <time datetime="19:30">19:30</time>
9 <!-- You can do seconds and milliseconds too! -->
10 <time datetime="19:30:01.856">19:30:01.856</time>
11 <!-- Date and time -->
12 <time datetime="2016-01-20T19:30">7.30pm, 20 January 2016</time>
13 <!-- Date and time with timezone offset -->
14 <time datetime="2016-01-20T19:30+01:00">
15   7.30pm, 20 January 2016 is 8.30pm in France
16 </time>
17 <!-- calling out a specific week number -->
18 <time datetime="2016-W04">The fourth week of 2016</time>

```

## Document and Website Structure

### Basic Sections

- Header `<header>`
  - Usually a big strip <sup>7</sup> across the top with a big heading/logo/tagline <sup>8</sup>
- Navigation bar: `<nav>`
  - Links to the site's main sections
  - Usually represented by menu buttons, or tabs
- Main content `<main>` with `<article>` `<section>` `<div>`
  - `<main>`
    - Use `<main>` only once per page
    - Put it directly inside `<body>`
    - Ideally this shouldn't be nested within other elements
  - `<article>`
    - encloses a block of related content that makes sense on its own without the rest of the page (e.g., a single blog post)
  - `<section>`
    - Similar to `<article>`, but more for grouping together a single part of the page that constitutes one single piece of functionality  
(e.g., a mini map, or a set of article headlines and summaries), or a theme
    - Can break `<article>`s up into different `<section>`s, or `<section>`s up into different `<article>`s, depending on the context
- Sidebar `<aside>`
  - Often placed inside `<main>`
  - Some peripheral info, links, quotes, ads, etc
  - Usually, this is contextual to what is contained in the main content
  - Like a secondary navigation system
- Footer `<footer>`

- o Generally contains fine print, copyright notices, or contact info.
- o Not critical or secondary to the website.
- o Sometimes used for SEO purposes, by providing links for quick access to popular content

The screenshot shows a website layout with the following structure:

- Header:** A dark grey header bar with the word "Header" in a large, bold, white serif font.
- Navigation Bar:** A pink navigation bar below the header containing links for "HOME", "OUR TEAM", "PROJECTS", and "CONTACT". It also includes a search input field with placeholder "Search query" and a "Go!" button.
- Main Content Area:** A white rectangular area containing:
  - Article heading:** "Article heading" in bold black font.
  - Text:** A paragraph of placeholder text (Lorem ipsum).
  - Section Headers:** "subsection" and "Another subsection" in bold black font.
  - Text:** Two more paragraphs of placeholder text.
- Sidebar:** A pink sidebar on the right side with the title "Related" in bold black font. It lists five items from a bulleted list.
- Footer:** A dark grey footer bar with the text "©Copyright 2050 by nobody. All rights reversed."

```

1  <!DOCTYPE html>
2  <html lang="en-US">
3      <head>
4          <meta charset="utf-8" />
5          <meta name="viewport" content="width=device-width" />
6
7          <title>A simple website-Yujie</title>
8
9          <link
10             href="https://fonts.googleapis.com/css?
11             family=Open+Sans+Condensed:300|Sonsie+One"
12             rel="stylesheet"/>
13          <link rel="stylesheet" href="style.css">
14      </head>
15      <body>
16          <header>
17              <h1>Header</h1>
18          </header>
19
20          <nav>
21              <ul>
22                  <li><a href="#">Home</a></li>
23                  <li><a href="#">Our team</a></li>
24                  <li><a href="#">Projects</a></li>
25                  <li><a href="#">Contact</a></li>
26              </ul>
27
28          <form>
29              <input type="search" name="q" placeholder="Search query" />

```

```
29             <input type="submit" value="Go!" />
30         </form>
31     </nav>
32
33     <main>
34         <article>
35             <h2>Article heading</h2>
36             <p>
37                 Lorem ipsum dolor sit amet, consectetur adipisicing
38                 elit. Donec a diam
39                 lectus. Set sit amet ipsum mauris. Maecenas congue ligula
39                 viverra nec consectetur ant hendrerit. Donec et mollis dolor.
40             Praesent
41                 et diam eget libero egestas mattis sit amet vitae augue. Nam
42             tincidunt
43                 congue enim, ut porta lorem lacinia consectetur.
44             </p>
45             <h3>Subsection</h3>
46             <p>
47                 Donec ut librero sed accu vehicula ultricies a non
48             tortor. Lorem ipsum
49                 dolor sit amet, consectetur adipisicing elit. Aenean ut
50             gravida lorem.
51                 Ut turpis felis, pulvinar a semper sed, adipiscing id
52             dolor.
53             </p>
54             <p>
55                 Pelientesque auctor nisi id magna consequat sagittis.
56             Curabitur
57                 dapibus, enim sit amet elit pharetra tincidunt feugiat nist
58             imperdiet.
59             Ut convallis libero in urna ultrices accumsan. Donec sed odio
60             eros.
61             </p>
62             <h3>Another subsection</h3>
63             <p>
64                 Donec viverra mi quis quam pulvinar at malesuada arcu
65             rhoncus. Cum
66                 soclis natoque penatibus et manis dis parturient montes, nascetur
67                 ridiculus mus. In rutrum accumsan ultricies. Mauris vitae nisi at
68             sem
69                 facilisis semper ac in est.
70             </p>
71             <p>
72                 Vivamus fermentum semper porta. Nunc diam velit,
73             adipiscing ut
74                 tristique vitae sagittis vel odio. Maecenas convallis ullamcorper
75             ultricied. Curabitur ornare, ligula semper consectetur sagittis,
76             nisi
77                 diam iaculis velit, is fringille sem nunc vet mi.
78             </p>
79             </article>
80
81             <aside>
82                 <h2>Related</h2>
```

```

71             <ul>
72                 <li><a href="#">Oh I do like to be beside the seadie</a>
73             </li>
74                 <li><a href="#">Oh I do like to be beside the seadie</a>
75             </li>
76                 <li><a href="#">Although in the North of England</a>
77             </li>
78         </ul>
79     </aside>
80 </main>
81
82     <footer>
83         <p>Copyright 2050 by nobody. All rights reversed.</p>
84     </footer>
85
86 </body>
87 </html>

```

**Header**

- Home
- Our team
- Projects
- Contact

Search query

**Article heading**

Donec viverra mi quis quam pulvinar at malesuada arcu rhoncus. Cum sociis natoque penatibus et manis dis parturient montes, nascetur ridiculus mus. In rutrum accumsan ultricies. Mauris vitae nisi at sem facilisis semper ac in est.

Vivamus fermentum semper porta. Nunc diam velit, adipiscing ut tristique vitae sagittis vel odio. Maecenas convallis ullamcorper ultricies. Curabitur ornare, ligula semper consectetur sagittis, nisi diam iaculis velit, is fringilla sem nunc vel mi.

**Related**

- Oh I do like to be beside the seadie
- Oh I do like to be beside the seadie
- Although in the North of England
- It never stops raining
- Oh well

©Copyright 2050 by nobody. All rights reversed.

## Non-semantic wrappers `<div>` `<span>`

Just group a set of elements together to affect them as a single entity with some CSS

- `<div>`:
  - Block
  - `<div class=""></div>`
- `<span>`:
  - Inline
  - `<span class="..."></span>`
  - Provide extra direction for the director of the play, Not supposed to have extra semantic meaning  
对舞台剧导演提供额外指引；没有具体语义
- `class`: for the screen readers users

## Line breaks and horizontal rules `<br>` `<hr>`

- `<br>`: line breaks

```
1 <p>
2     There once was a man named o'dell<br/>
3     who loved to write HTML<br/>
4     But his structure was bad, his semantics were sad<br/>
5     And his markup didn't read very well.
6 </p>
```

- `<hr>`: the thematic break

```
1 <p>
2     Ron was backed into a corner by the marauding metherbeasts. Scared,
3     but determined to protect his friends, his wan and prepared to do battel,
4     hoping that his distress call has made it through.
5 </p>
6 <hr/>
7 <p>
8     Meanwhile, Harry was sitting at home, staring at his royalty
9     statement and pondering when the next spin off series would come out,
10    when an enchanted distress letter flew through his window and landed in
11    his lap. He read it hazily and sighed; "better get back to work then", he
12    mused.
13 </p>
```

---

## Debug HTML

### Errors

- Syntax errors ✘
  - Parse permissively mode
    - Reason: For HTML initially, Allowing people to get their content published was more important than the strict syntax
  - Still display
    - Sometimes it is a problem
  - Have built-in rules to state how to interpret incorrectly written markup
- Logic errors

### Validation

- For a huge, complex HTML document
- Best strategy: Run the HTML Page through the [The W3C Markup Validation Service](#):
  - Create and maintained by the **W3C**
  - This webpage takes an HTML document as a input, goes through it and gives a report to tell you what is wrong with your HTML
- Example:

```
1 <!DOCTYPE html>
```

```
2 <html lang="en-US">
3   <head>
4     <meta charset="utf-8">
5
6     <title>HTML debugging examples</title>
7
8
9   </head>
10
11 <body>
12   <h1>HTML debugging examples</h1>
13
14   <p>What causes errors in HTML?
15
16   <ul>
17     <li>Unclosed elements: If an element is <strong>not closed properly, then its effect can spread to areas you didn't intend
18
19     <li>Badly nested elements: Nesting elements properly is also very important for code behaving correctly. <strong>strong <em>strong emphasised? </strong> what is this?</em>
20
21     <li>Unclosed attributes: Another common source of HTML problems. Let's look at an example: <a href="https://www.mozilla.org/">link to Mozilla homepage</a>
22   </ul>
23 </body>
24 </html>
```

1. **Error** End tag `li` implied, but there were open elements.

From line 19, column 7; to line 19, column 10

nd<--> <li>Badly

2. **Error** Unclosed element `strong`.

From line 17, column 47; to line 17, column 54

lement is <strong>not cl

3. **Error** End tag `strong` violates nesting rules.

From line 19, column 149; to line 19, column 157

mphasised?</strong> what

4. **Error** End tag `li` implied, but there were open elements.

From line 21, column 7; to line 21, column 10

m><--> <li>Unclos

5. **Error** End of file reached when inside an attribute value.

Ignoring tag.

At line 24, column 8

</body><--></html><-->

6. **Error** End of file seen and there were open elements.

From line 21, column 98; to line 24, column 8

example: <a href="https://www.mozilla.org/>link to Mozilla  
homepage</a><--> </ul><--> </body><--></html><-->

7. **Error** Unclosed element `ul`.

From line 16, column 5; to line 16, column 8

TML?<--> <ul><-->

# Project1: Marking Up a Letter

Requirement: [标记信件 - 学习 Web 开发 | MDN \(mozilla.org\)](#)

```
1  <!DOCTYPE html>
2  <html lang="en-US">
3      <head>
4          <meta charset="utf-8">
5          <meta name="viewport" content="width=device-width">
6          <meta name="author" content="Dr. Eleanor Gaye">
7          <link href="styles/styles.css" rel="stylesheet" >
8          <title>A Letter</title>
9      </head>
10     <body>
11         <div class="sender-column">
12             Dr. Eleanor Gaye<br>
13             Awesome Science faculty<br>
14             <address>
15                 University of Awesome<br>
16                 Bobtown, CA 99999,<br>
17                 USA<br>
18             </address>
19             <!-- strong -->
20             <strong>
21                 Tel: 123-456-7890<br>
22                 Email: no_reply@example.com<br>
23             </strong>
24             20 January 2016<br>
25         </div>
26         <p>Miss Eileen Dover</p>
27         <address>
28             4321 cliff Top Edge<br>
29             Dover, CT9 XXX<br>
30             UK<br>
31         </address>
32
33         <!-- h1 -->
34         <h1>Re: Eileen Dover university application</h1>
35
36         <p>Dear Eileen,</p>
37         <p>
38             Thank you for your recent application to join us at the
University
39             of Awesome's science faculty to study as part of your
40             <abbr title="">PhD</abbr> next year. I will answer your
questions
41             one by one, in the following sections.
42         </p>
43         <!-- h2 -->
44         <h2>Starting dates</h2>
45         <p>
46             We are happy to accommodate you starting your study with us
47             at any time, however it would suit us better if you could
```

```
48         start at the beginning of a semester; the start dates for
49         each one are as follows:
50
51     </p>
52     <!-- list -->
53     <!-- machine-readable dates -->
54     <ul>
55         <li>
56             First semester:
57             <time datetime="2016-09-09">9 September 2016</time>
58         </li>
59         <li>
60             Second semester:
61             <time datetime="2017-01-15">15 January 2017</time>
62         </li>
63         <li>
64             Third semester:
65             <time datetime="2017-05-02">2 May 2017</time>
66         </li>
67     </ul>
68     <p>
69         Please let me know if this is ok, and if so which start date
70         you would prefer.
71     </p>
72     <p>
73         You can find more information about <a
74             href="http://example.com">important university
75                 dates</a> on our website
76     </p>
77     <!-- h2 -->
78     <h2>Subjects of study</h2>
79     <p>
80         At the Awesome Science Faculty, we have a pretty <em>open-
81             minded</em>
82             research facility – as long as the subjects fall somewhere in
83                 the realm of science and technology. You seem like an
84             intelligent,
85             dedicated researcher, and just the kind of person we'd like to
86             have
87             on our team. Saying that, of the ideas you submitted we were
88             most
89                 intrigued by are as follows, in order of priority:
90
91             <ol>
92                 <li>
93                     Turning H2O into wine, and the health benefits
of Resveratrol (C14H12O3.)<br>
```

```
94      </li>
95      <li>
96          <abbr title="HyperText Markup Language">HTML</abbr> and
97          <abbr title="Cascading Style Sheets">CSS</abbr> constructs for
98              representing musical scores.
99      </li>
100     </ol>
101     <p>
102         So please can you provide more information on each of these
103         subjects,
104         including how long you'd expect the research to take, required
105         staff and
106         other resources, and anything else you think we'd need to know?
107     Thanks.
108     </p>
109     <!-- h2 -->
110     <h2>Exotic dance moves</h2>
111     <p>
112         Yes, you are right! As part of my post-doctorate work, I did
113         study exotic
114         tribal dances. To answer your question, my favourite dances are
115         as follows,
116         with definitions:
117     </p>
118     <dl>
119         <dt>Polynesian chicken dance</dt>
120         <dd>A little known but very influential dance dating back as
121         far as 300<abbr>BC</abbr>,
122         a whole village would dance around in a circle like
123         chickens,
124         to encourage their livestock to be "fruitful".
125     </dd>
126     <dt>Icelandic brownian shuffle</dt>
127     <dd>Before the Icelanders developed fire as a means of getting
128         warm,
129         they used to practice this dance, which involved huddling
130         close together
131         in a circle on the floor, and shuffling their bodies around
132         imperceptibly
133         tiny, very rapid movements. One of my fellow students used
134         to say that he thought t
135             his dance inspired modern styles such as Twerking.
136         </dd>
137         <dt>Arctic robot dance</dt>
138         <dd>
139             An interesting example of historic misinformation, English
140             explorers in the 1960s
141                 believed to have discovered a new dance style characterized
142                 by "robotic", stilted
143                     movements, being practiced by inhabitants of Northern
144                     Alaska and Canada. Later on
145                         however it was discovered that they were just moving like
146                         this because they were really cold.
147                     </dd>
148     </dl>
```

```
133      <p>
134          For more of my research, see my <a
135              href="http://example.com">exotic dance research page</a>
136          </p>
137          <p>
138              Yours sincerely,<br>
139              Dr Eleanor Gaye
140          </p>
141          <p>
142              University of Awesome motto:
143
144              <q>"Be awesome to each other."</q>
145              -- <cite>the memoirs of Bill S Preston</cite> ,
146              <abbr>Esq</abbr>
147          </p>
148      </body>
149  </html>
```

Dr. Eleanor Gaye  
Awesome Science faculty  
*University of Awesome*  
Bobtown, CA 99999,  
USA  
Tel: 123-456-7890  
Email: no\_reply@example.com  
20 January 2016

Miss Eileen Dover

4321 Cliff Top Edge  
Dover, CT9 XXX  
UK

## Re: Eileen Dover university application

Dear Eileen,

Thank you for your recent application to join us at the University of Awesome's science faculty to study as part of your PhD next year. I will answer your questions one by one, in the following sections.

### Starting dates

We are happy to accommodate you starting your study with us at any time, however it would suit us better if you could start at the beginning of a semester; the start dates for each one are as follows:

- First semester: 9 September 2016
- Second semester: 15 January 2017
- Third semester: 2 May 2017

Please let me know if this is ok, and if so which start date you would prefer.

You can find more information about [important university dates](#) on our website

### Subjects of study

At the Awesome Science Faculty, we have a pretty *open-minded* research facility — as long as the subjects fall somewhere in the realm of science and technology. You seem like an intelligent, dedicated researcher, and just the kind of person we'd like to have on our team. Saying that, of the ideas you submitted we were most intrigued by are as follows, in order of priority:

1. Turning H<sub>2</sub>O into wine, and the health benefits of Resveratrol (C<sub>14</sub>H<sub>12</sub>O<sub>3</sub>.)
2. Measuring the effect on performance of funk bassplayers at temperatures exceeding 30°C (86°F), when the audience size exponentially increases (effect of  $3 \times 10^3$  increasing to  $3 \times 10^4$ .)
3. HTML and CSS constructs for representing musical scores.

So please can you provide more information on each of these subjects, including how long you'd expect the research to take, required staff and other resources, and anything else you think we'd need to know? Thanks.

### Exotic dance moves

Yes, you are right! As part of my post-doctorate work, I did study exotic tribal dances. To answer your question, my favourite dances are as follows, with definitions:

#### Polynesian chicken dance

A little known but very influential dance dating back as far as 300BC, a whole village would dance around in a circle like chickens, to encourage their livestock to be "fruitful".

#### Icelandic brownian shuffle

Before the Icelanders developed fire as a means of getting warm, they used to practice this dance, which involved huddling close together in a circle on the floor, and shuffling their bodies around in imperceptibly tiny, very rapid movements. One of my fellow students used to say that he thought this dance inspired modern styles such as Twerking.

#### Arctic robot dance

An interesting example of historic misinformation, English explorers in the 1960s believed to have discovered a new dance style characterized by "robotic", stilted movements, being practiced by inhabitants of Northern Alaska and Canada. Later on however it was discovered that they were just moving like this because they were really cold.

For more of my research, see my [exotic dance research page](#)

Yours sincerely,  
Dr Eleanor Gaye

University of Awesome motto: ""Be awesome to each other." -- *the memoirs of Bill S Preston*, Esq

**Document checking completed. No errors or warnings to show.**

## Project2: Structure a Web Page

Requirement: [构建网页内容 - 学习 Web 开发 | MDN \(mozilla.org\)](#)

```
1  <!DOCTYPE html>
2  <html lang="en">
3      <head>
4          <meta charset="utf-8">
5          <title>Birdwatching</title>
6          <link href="https://fonts.googleapis.com/css?family=Roboto+Condensed:300%CCinzel+Decorative:700" rel="stylesheet">
7          <link href="styles/style.css" rel="stylesheet">
8          <!--[if lt IE 9]>
9              <script
10                 src="https://cdnjs.cloudflare.com/ajax/libs/html5shiv/3.7.3/html5shiv.js">
11             <![endif]-->
12         </head>
13
14     <body>
15         <header>
16             <h1>Birdwatching</h1>
17             
18         </header>
19
20         <nav>
21             <ul>
22                 <li><span>Home</span></li>
23                 <li><a href="#">Get started</a></li>
24                 <li><a href="#">Photos</a></li>
25                 <li><a href="#">Gear</a></li>
26                 <li><a href="#">Forum</a></li>
27             </ul>
28         </nav>
29
30         <main>
31             <article>
32                 <h2>Welcome</h2>
33
34                 <p>Welcome to our fake birdwatching site. If this were a real
35                 site, it would be the ideal place to come to learn more about birdwatching,
36                 whether you are a beginner looking to learn how to get into birding, or an
37                 expert wanting to share ideas, tips, and photos with other like-minded
38                 people.</p>
39
40                 <p>So don't waste time! Get what you need, then turn off that
41                 computer and get out into the great outdoors!</p>
42
43             </article>
44
45             <aside>
46                 <h2>Favourite photos</h2>
47
```

```
42         <a href="images/favorite-1.jpg"></a>
45             <a href="images/favorite-2.jpg">
48             </a>
49             <a href="images/favorite-3.jpg"></a>
52             <a href="images/favorite-4.jpg">
55         </aside>
56     </main>
57
58     <footer>
59         <p>This fake website example is CC0 – any part of this code may be
60         reused in any way you wish. Original example written by Chris Mills, 2016.
61     </p>
62
63         <p><a href="http://game-icons.net/lorc/originals/dove.html">Dove
64             icon</a> by Lorc.</p>
65
66     </footer>
67 </body>
68 </html>
```

[HOME](#)[GET STARTED](#)[PHOTOS](#)[GEAR](#)[FORUM](#)

## WELCOME

Welcome to our fake birdwatching site. If this were a real site, it would be the ideal place to come to learn more about birdwatching, whether you are a beginner looking to learn how to get into birding, or an expert wanting to share ideas, tips, and photos with other like-minded people.

So don't waste time! Get what you need, then turn off that computer and get out into the great outdoors!

## FAVOURITE PHOTOS



This fake website example is CC0 – any part of this code may be reused in any way you wish. Original example written by Chris Mills, 2016.

[Dove icon](#) by Lorc.

## Multimedia and embedding

Guide:

- Images: `<img>` `<figure>`
- Video and audio: `<video>` `<audio>`s
- Embedding other types:
  - `<iframe>`: for embedding other pages
  - `<embed>` `<object>`: for embed external resource such as PDF files
- Vector graphics
  - Don't distort/pixelate when zoomed in(remain smooth when scaled)
  - Example: [SVG](#)
- Responsive images-----[Responsive design\(CSS\)](#)
  - Concept: the image that work well on devices with widely differing screen sizes
  - Function: improve the performance across different devices

## Images <img>

- <img>
  - **Void element**
    - The element that cannot have any child content and cannot have an end tag
  - **Replaced element**
    - the content and size are defined by an external resource, not by the contents of the element itself

### Attribute: <src> <alt>

- <src> : Contains the relative/absolute URL that points to the image location
  - Relative URL:

```
1 | 
```



- Good for maintenance
  - When you move your site to a different domain, you won't need to update all your URLs to include the new domain

In more advanced setups, you might use a CDN<sup>9</sup> to deliver your images

- Absolute URL:

```
1 | 
```



- **Not recommend:**

- For simple setups, all the images have to be kept in the same server

Note:

**All the images used in your page need to be verify the publishing license!**

- Otherwise:
    - Cause the hotlinking problem(unethical)
      - Someone else would be paying the bandwidth costs for delivering the image when someone visits your page
      - No control over the image being removed or replaced(embarrassing!)
  - <alt> : The alternative that include the image description
    - Function: Used in situations where the image cannot be seen/displayed or takes a long time to render
    - Reason:
      1. For screen reader users who are visually impaired
      2. For the text-only browser users
      3. For search engines to utilize
      4. For the situation that the file/path name is wrong
      5. For the situation that the users have turned off images to reduce data transfer volume and distractions, which is especially common on mobile phones
    - Content:
      - What exactly should be written in the `alt` depends on why the image is there in the first place
      - Try turning off images in your browser to see how things look
    - For decoration:
      - `alt=""`
- For a screen reader should not waste time reading it.
- Better use CSS background images to replace HTML decorative images

- For content:

```
alt="IMAGEINFORMATION"
```

- Don't redundant<sup>10</sup> `alt`
  - If the image is described adequately by the main text body, then use `alt=""`
  - It will annoy the sighted user!
- For adding images for the link

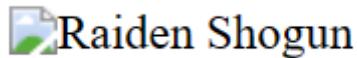
```
alt="LINK"
```

If you put an image inside `a` tags, to turn an image into a link, you still must provide [accessible link text](#). In such cases you may, either, write it inside the same `<a>` element, or inside the image's `alt` attribute – whichever works best in your case.

- For text:

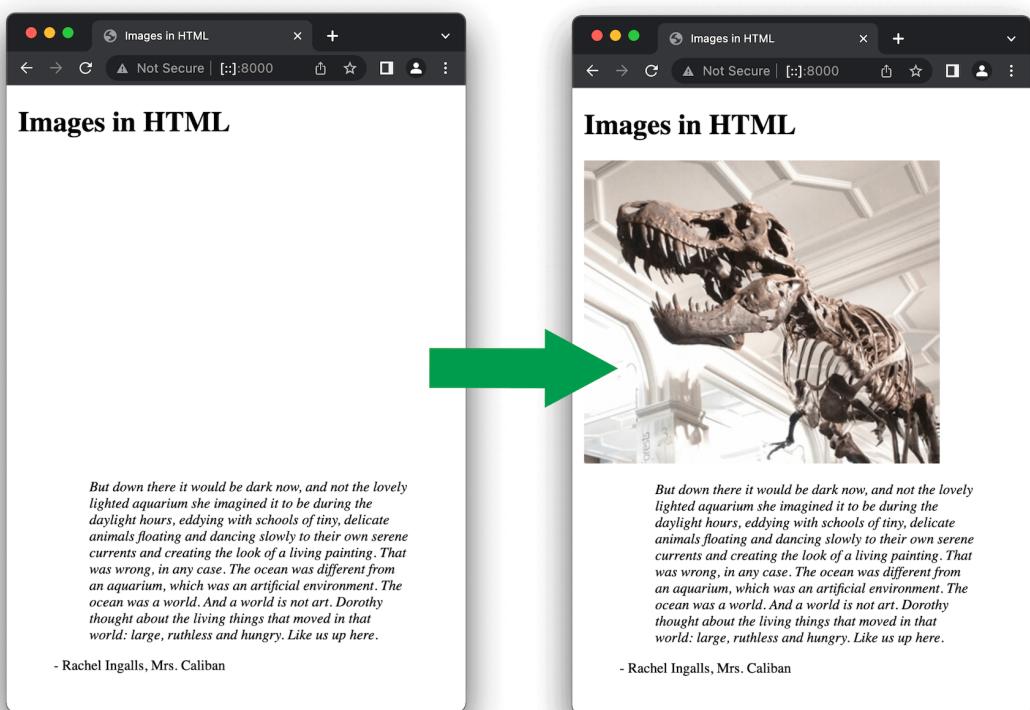
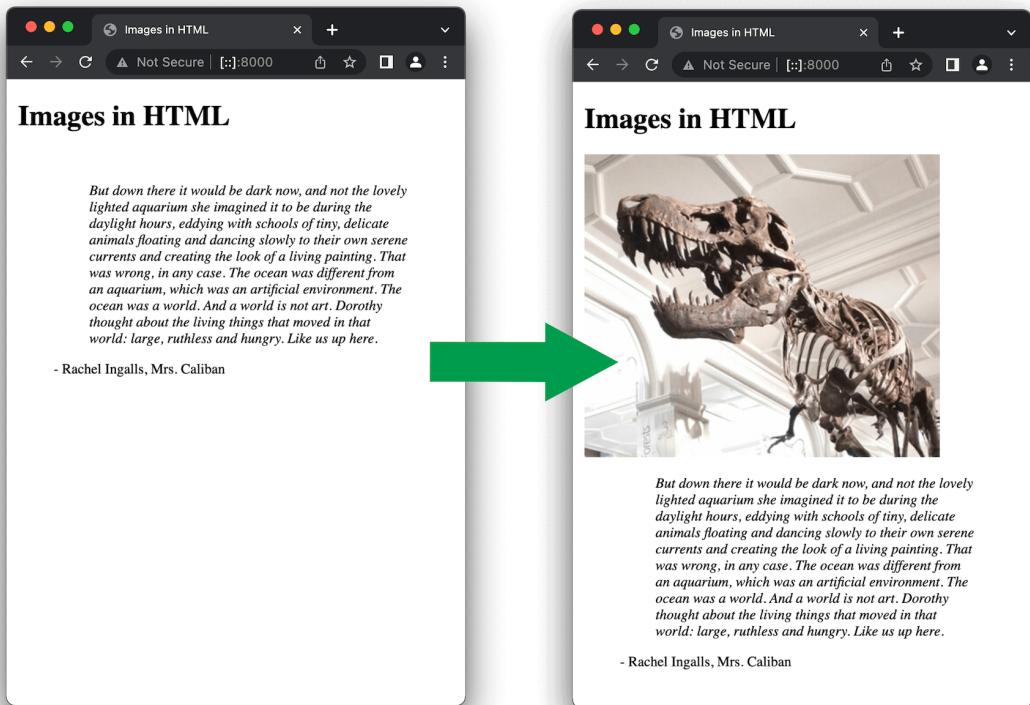
- Should not put text into images
  - If main heading needs a drop shadow, better use CSS rather use HTML images

- Example in the above code:



## width and height

- Reason:
  - HTML and images are separate resources, fetched by the browser as separate HTTP(s) requests
  - The browser starts the HTML when it receives the file.
  - But there a case often appears that the HTML is received by the browser faster than images.
  - So use `width` and `height` to reserve a space for the image

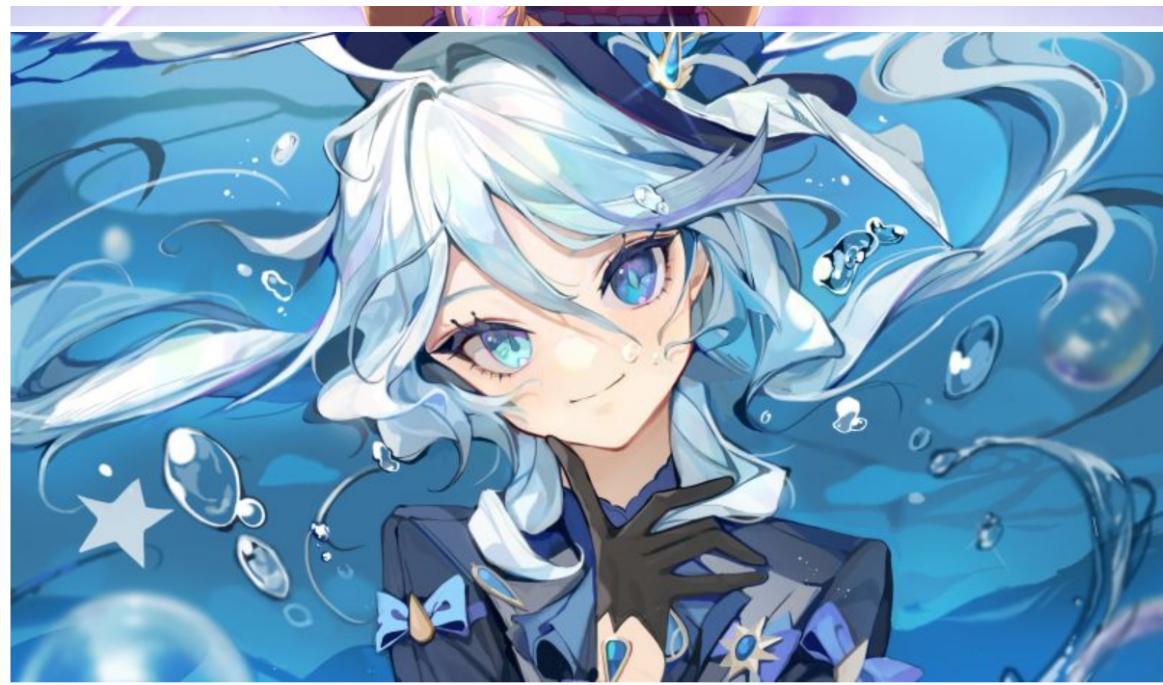


- Function:
  - Used for reserving space for image
  - **Not used for resizing images!**
- Better use CSS instead(aspect ratio)
- Or use an image editor to put the image at the correct size\*\*

```

1 

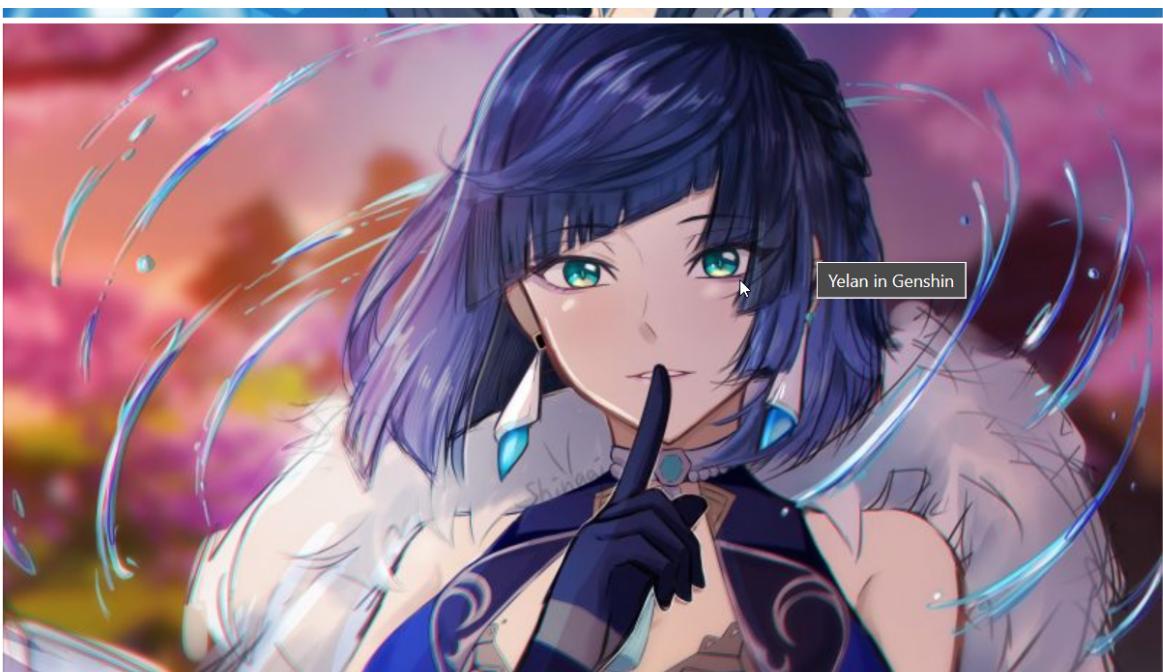
```



### title

- Not recommended for its accessibility problems
  - The screen reader support is very unpredictable
- Most browser won't show it unless the mouse hovering on the image

```
1 | 
```



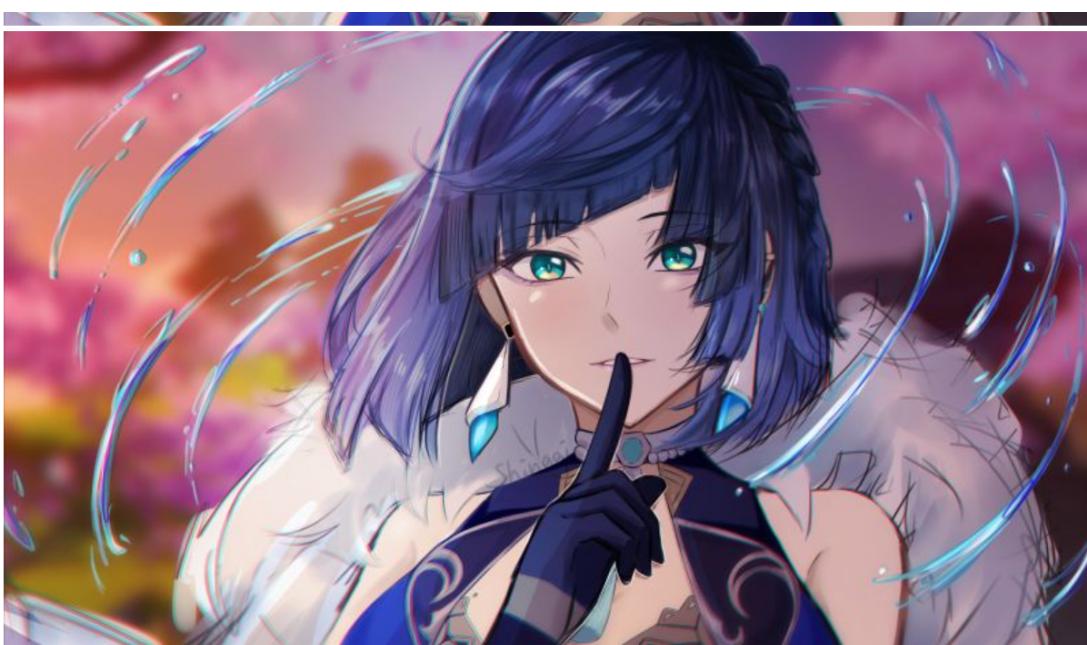
## Annotation with `<figure>` `<figcaption>`

- Wrong way:

```
1 <div class="figure">
2   
7   <p>Yelan in Genshin</p>
8 </div>
```

- There is nothing that semantically link the image to its caption

- Cause problems for screen reader



Yelan in Genshin

- Better solution:

```
1 <figure>
2   
7   <figcaption>
8     Yelan in Genshin Impact
9   </figcaption>
10 </figure>
```



Yelan in Genshin Impact

## CSS Background Images

Use CSS to replace HTML images, but only for decoration

### Problem: media assets and licensing

#### License types:

##### 1. All rights reserved:

- Obtain explicit, written permission from the copyright holder
- Pay a license fee to use them
- Limit your uses to those that would be considered fair use or fair dealing

Not required to include a copyright notice or license terms with their work

So the safest course is to assume it is protected by copyright with all rights reserved

##### 2. Permissive

- Do not need to pay a license fee or seek permission to use it
- Provide a link to the original source and credit its creator
- Indicate whether any changes were made to it
- Share any derivative works created using the image under the same license as the original
- Not share any derivative works at all
- Include a copy of the license along with any release that uses the image

Such as MIT, BSD, or a suitable CC

- Copyleft license:

- GNU GPL, Share alike
- Stipulate that derivative works need to be released under the same license as the original

##### 3. Public domain/CC0/No right reserved

- No copyright applies to it and it can be used without permission and without having to fulfill any licensing conditions
- Most effective way to place work in the public domain:

- license it under CC0
- Need to obtain proof that the image is in the public domain and keep the proof for your records

## Searching for permissively-licensed images

- Image search engine
    - Use tool "usage rights"
  - Image repositories sites
    - such as Flickr, ShutterStock, Pixabay
- 

## Video and Audio `<video>` `<audio>`

Three ways for video and audio:

1. Native HTML solution `<video>` `<audio>`
2. The availability of JavaScript APIs
3. The plugin-based technologies to influx the online videos and audio:
  - Example
    - Flash
    - Silverlight
  - Features
    - Both have security and accessibility issues
    - Now obsolete

OVPs(Online Video Providers): YouTube, Dailymotion

OAPs(Online Audio Providers): Soundcloud

`<video>`

`<video>` Elements

```

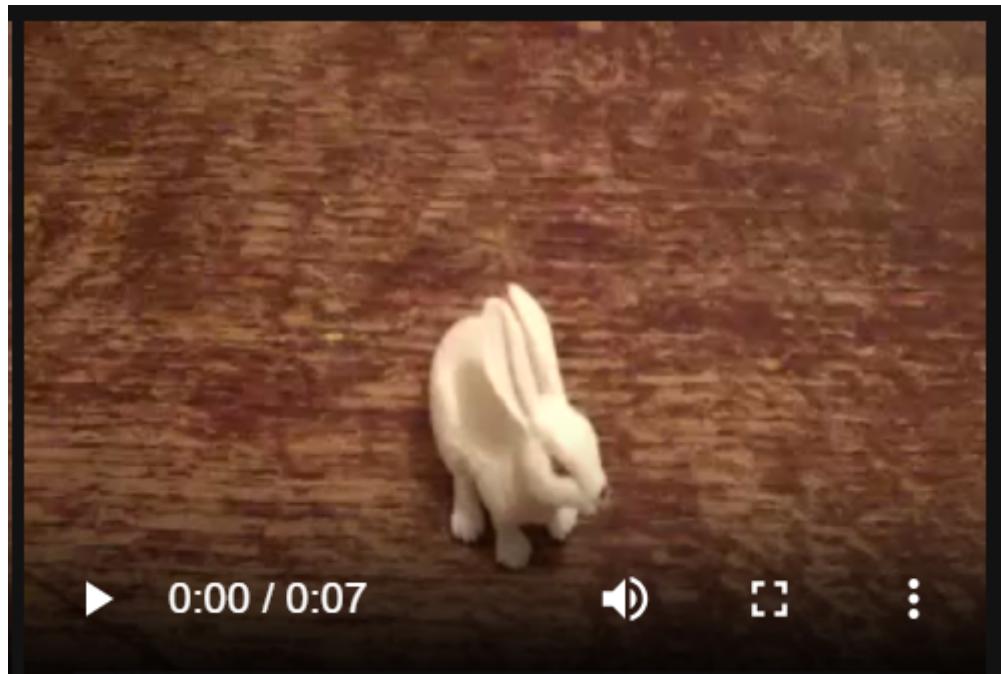
1 <video src="src/rabbit320.webm" controls>
2   <p>
3     Your broser doesn't support HTML Video.
4     Here is a <a href="src/rabbit320.webm">link to the video</a>
5     instead.
6   </p>

```

- `src`: Contain the path to the video location
- `controls`: Used to include the browser's own control interface
  - At least must include a way to start and stop the media and to adjust the volume
- Fallback content
  - Concept: the paragraph in the `<video>` tags
  - Function:
    - Provide a fallback for older browsers
    - Provided a direct link to the video file for accessing

## Enhancement

```
1 <video controls>
2     <source src="src/rabbit320.webm" type="video/webm">
3     <source src="src/rabbit320.mp4" type="video/mp4">
4     <p>
5         Your browser doesn't support HTML5 video.
6         Here is a <a href="src/rabbit320.mp4">link to the video</a>
7         instead
8     </p>
9     </video>
```



- 在这个例子当中，浏览器将会检查 `<source>` 标签，并且播放第一个与其自身 codec 相匹配的媒体。
- 你的视频应当包括 WebM 和 MP4 两种格式，这两种在目前已经足够支持大多数平台和浏览器。
- 每个 `<source>` 标签页含有一个 `type` 属性，这个属性是可选的，但是建议你添加上这个属性 — 它包含了视频文件的 [MIME types](#)，同时浏览器也会通过检查这个属性来迅速的跳过那些不支持的格式。如果你没有添加 `type` 属性，浏览器会尝试加载每一个文件，直到找到一个能正确播放的格式，这样会消耗掉大量的时间和资源。

## Other `<video>`

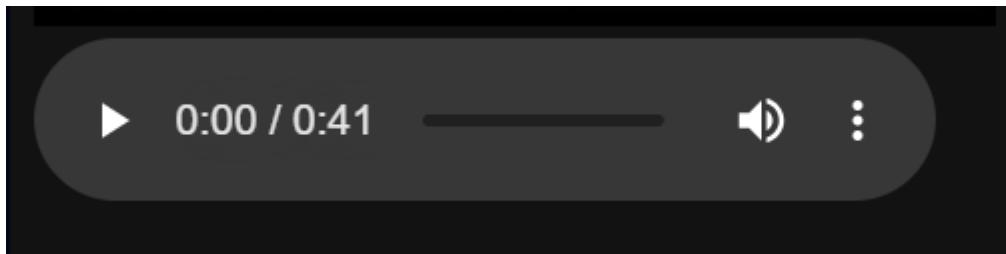
```
1 <video
2     controls
3     width="400"
4     height="400"
5     autoplay
6     loop
7     muted
8     poster="src/poster.png">
9     <source src="src/rabbit320.webm" type="video/webm">
10    <source src="src/rabbit320.mp4" type="video/mp4">
11    <p>
12        Your browser doesn't support HTML5 video.
13        Here is a <a href="src/rabbit320.mp4">link to the video</a>
```

```
14     instead
15         </p>
16     </video>
```

- width height
- autoplay
- loop
- muted
- poster
- preload
  - none
  - auto
  - metadata

## <audio> Element

```
1 <audio controls>
2     <source src="src/viper.mp3" type="audio/mp3">
3     <source src="src/viper.ogg" type="audio/ogg">
4     <p>
5         Your browser doesn't support HTML5 audio.
6         Here is a <a href="src/viper.mp3">link to the video</a>
7         instead
8     </p>
9 </audio>
```



## Javascript about audio

### 1. 从选择媒体开始重新播放媒体

```
1 const mediaElem = document.getElementById("my-media-element");
2 mediaElem.load();
```

### 2. 音轨增删时间

```
1 const mediaElem = document.querySelector("video");
2 mediaElem.audioTracks.onaddtrack= function(event)
3     {audioTrackAdded(event.track);}
```

## 显示音轨文本WebVTT

- cues: 字符串
  - subtitles
  - captions
  - timed descriptions
- Example:

```
1 WEBVTT
2
3 1
4 00:00:22.230 ---> 00:00:24.606
5 第一段字幕
6
7 ...
```

- Add it in HTML

```
1 <video controls>
2   <source src="example.mp4" type="video/mp4" />
3   <source src="example.webm" type="video/webm" />
4   <track kind="subtitles" src="subtitles_en.vtt" srclang="en" />
5 </video>
```

## Other embedded technologies object iframe

### History

#### Embedding Web

- **Frame**: small part of a website stored in individual HTML pages
- **Frameset**: the master document where you specify the area on the screen that each frame filled
- Pros: better for download speeds
- Cons: this technology still have many problem
- **ABANDON!**

#### Plugin technologies

- <object> <embed>
- Such as Java Applets/ Flash
- Cons: this technology still have many problem
- **ABANDON!**

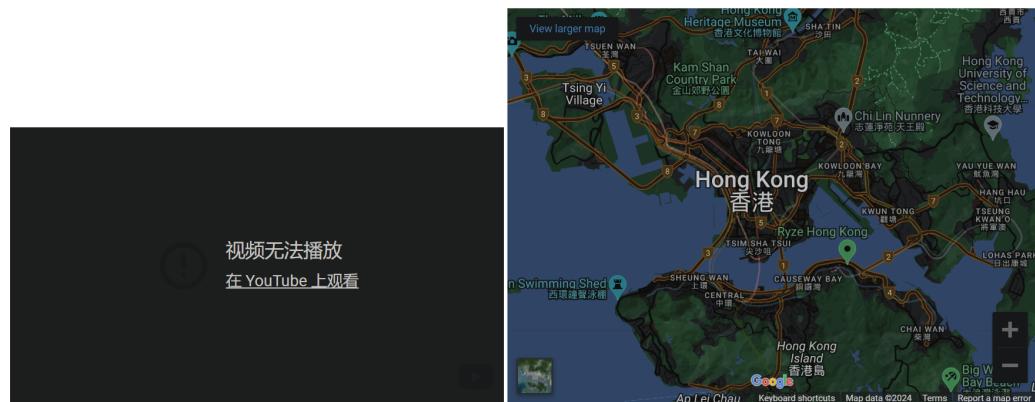
#### <iframe> (along with <canvas> <video> etc)

- Concept: embed an entire web document inside another one (like <img> )

### Small Practice

- **Youtube**: share-embed-copy

- GoogleMap: menu-embedMap-copy



**Live output**

```

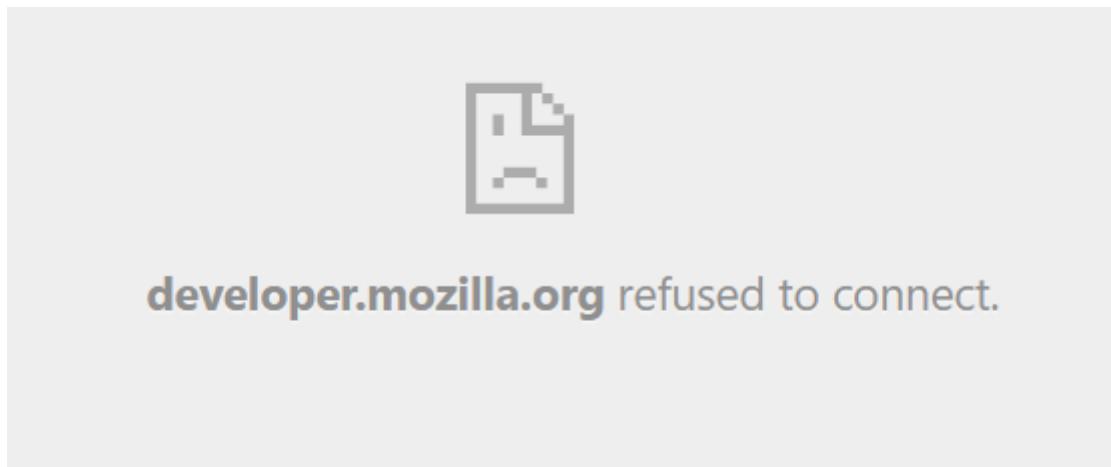
1 <head>
2   <style>
3     iframe{
4       border:none;

```

```

5         }
6     </style>
7 </head>
8 ...
9 <body>
10    <iframe
11        src="https://developer.mozilla.org/en-US/docs/Glossary"
12        width="100%"
13        height="500"
14        allowfullscreen
15        sandbox>
16    <p>
17        <a href="/en-US/docs/Glossary">
18            Fallback link for browsers that don't support iframes
19        </a>
20    </p>
21    </iframe>
22 </body>

```



- `border: none`: Browsers display the `<iframe>` with a surrounding border by default
- `allowfullscreen`: Using the **Fullscreen API**
- `src`:

Tips

Better set the `src` with JavaScript after the main content is done with loading

- More usable and decrease the load time
- `width / height`
- `sandbox`

Tips

Always use the `sandbox` attribute

- Concept: the container that only gives the permissions needed for doing its job
  - Without it, the content may be able to execute JavaScript, submit forms trigger popup windows etc.
- Should impose all available restrictions by **using the `sandbox` attribute with no parameters by default**
  - Add permissions back one by one inside the `sandbox=""`

- Never add `allow-scripts` and `allow-same-origin` to `sandbox` attribute
  - Otherwise the content could bypass the Same-origin policy that stops sites from executing scripts and use JavaScript to turn off sandboxing altogether

Note:

`sandbox` provides no protection if attackers can fool people into visiting malicious content directly

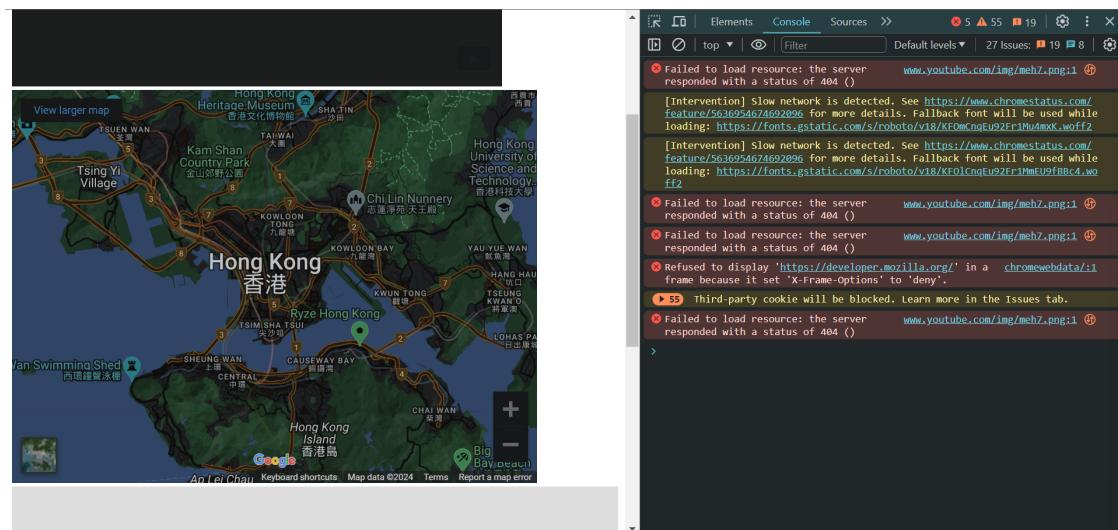
💡 Tips:

If there's any chance that certain content may be malicious, please serve it from a different domain to your main site

## Security Problem of `<iframe>`

- Problem: Hackers/Crackers might use `<iframe>` as a attack vector
  - Example: the iframe attack----Clickjacking
- Solution:
  - Need to get the permission and obey the license terms
  - Use HTTPS: the encrypted version of HTTP
    - Requirement: special security certificate to be installed
      - Many hosting providers offer HTTPS-enabled hosting, do not need to setup own certificate
      - If you do need to setup HTTPS support for your site on your own, using Let's Encrypt
    - Pros
      - Prevent embedded content from accessing content in your parent document, and vice versa
      - Reduce the chance that remote content has been tampered with in transit

Browser developer tools: `ctrl + shift + I`



Refused to display '<https://developer.mozilla.org/>' in a frame because it set 'X-Frame-Options' to 'deny'.

- For MDN page, it doesn't make sense to be embedded in other pages

- Unless you want to claim them as your own or attempt to steal data via clickjacking
- And if everybody embedded the MDN page, all the additional bandwidth would start to cost Mozilla a lot of money

💡 Tips:

### Only embed when necessary

Rule for web security: *You can never be too cautious. If you made it, double-check it anyway. If someone else made it, assume it's dangerous until proven otherwise.*

💡 Tips

Configure CSP<sup>11</sup> directives

- CSP<sup>11</sup> : Provides a set of HTTP Headers designed to improve the security of HTML document
- For `<iframe>`, can configure the server to send an appropriate `X-Frame-Options` header
  - Prevent other websites from embedding content in their webpages

`<embed>` `<object>`

- Concept: general purpose embedding tools for embedding external content(like PDFs)

Unlikely use these elements very much

💡 Tips

### Better link to PDFs, rather than embedding them in the page

---

---

---

---

---

---

---

---

---

---

---

---

1. 超文本标记语言 [P](#)

2. 元数据 [P](#)

3. apostrophe [P](#)

4. ampersand [P](#)

5. 缩写 [P](#) [P](#)

6. 对于保留空白 (通常是在代码块) --如果您在文本中使用缩进或多余的空白, 浏览器会忽略它, 您在渲染的页面上看不到它。但是, 如果用标记对文本进行包装, 那么空白处的呈现效果将与文本编辑器中的效果相同。 [P](#)

7. 条, 带 [P](#)

8. 口号 [P](#)

9. Content Delivery Network [P](#)

10. 多余, 重复 [P](#)

11. Content Security Policy [P](#) [P](#)

