

Open-Source Software Practice

8. HTML & CSS

Instructor: Jaemin Jo (조재민, jmjo@skku.edu)
Interactive Data Computing Lab (*IDCLab*),
College of Computing and Informatics,
Sungkyunkwan University

Graphical User Interface (GUI)

- The *menu* and *stat* commands we developed are command-line tools.
 - Command-line interface or CLI
- Most modern user interfaces are graphical!
 - Graphical user interface or GUI
 - WIMP: Windows + Icons + Menu + Pointer
- Benefits:
 - Simplicity
 - Learnable
 - Easy interaction
 - ...

Email Password

Address

Address 2

City State Zip

☐ Check me out

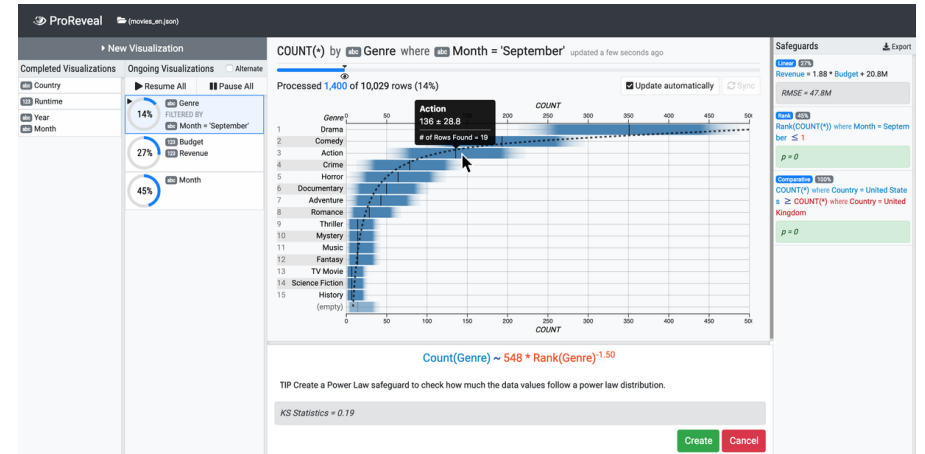
[Sign in](#)

How to Develop?

- As I explained before, there is no default programming language for GUI.
 - As if there is no default language for writing a novel.
- But, there are popular combinations of PL & framework/library
 - JavaScript & React (Web)
 - TypeScript & React (Web)
 - Python & PyQt5
 - Dart & Flutter
 - C# & Universal Windows Platform

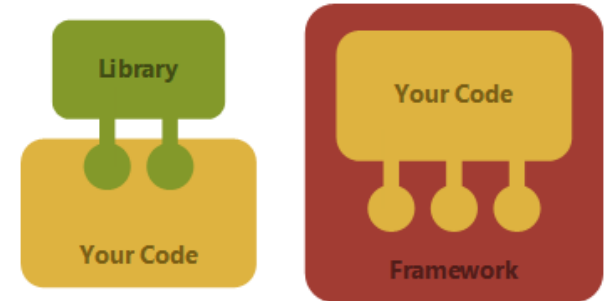
Why Web?

- We will develop our interfaces on the Web!
- Benefits:
 - Cross-platform (desktop and mobile)
 - Accessible (no installation)
 - Easy to update
 - Versatile
- Dev stack
 - JavaScript
 - HTML
 - CSS

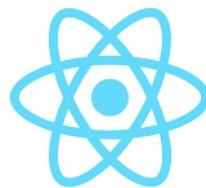


Web Frameworks

- In most modern development scenarios, we use a **framework**.
- **Framework**: the skeleton on which your application is built.
 - Your code is run by the framework.
 - Sometimes, the code borrows functions from a library.



- Web frameworks:
 - React
 - Angular
 - Vue
 - Svelte
 - ...



Web Frameworks

- But, using a framework entirely changes your development pattern, and this is quite confusing for novices.
- So, we first start with **vanilla web development** with no support; we will use HTML, CSS, and JavaScript only.

The Big Picture

- **HTML** for the structure of a web page
- **CSS** for the style
- **JavaScript** for the interaction

검색

검색어 [검색이 더욱 편리해졌습니다]

성균관대학교
SUNGKYUNKWAN UNIVERSITY

- ENG
- [링크ID LOGIN](#)
- [링크ID](#)
- [대학정보](#)
- [입학](#)
- [교육](#)
- [연구](#)
- [산학](#)
- [국제화](#)
- [대학생활](#)

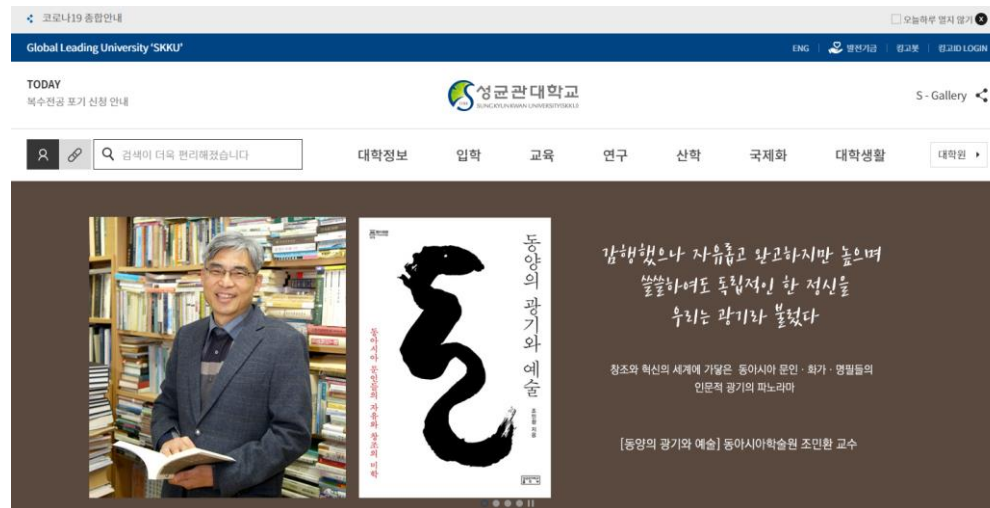
대학원 발전기금

검색

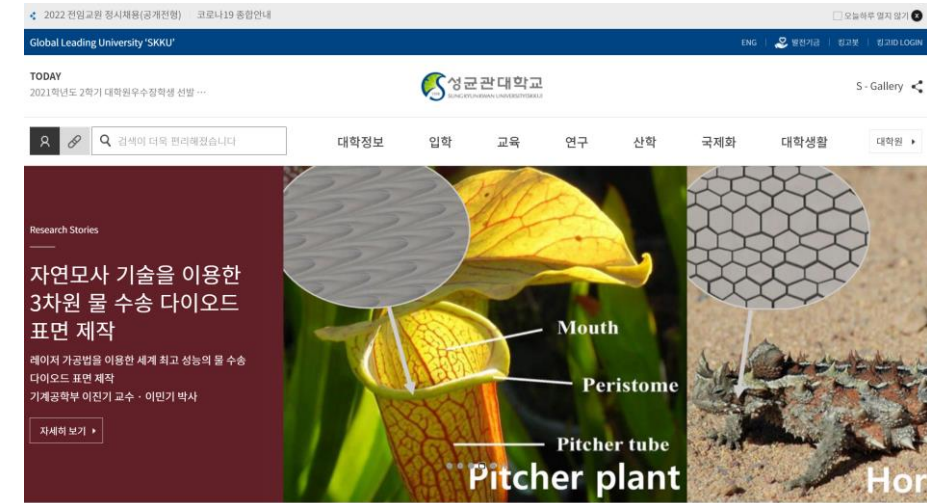
검색어 [검색어를 입력하세요]



HTML Only



HTML + CSS



HTML + CSS + JS (slide animation)

HyperText Markup Language or HTML^{IDCLab}



- **HyperText Markup Language or HTML** is the standard markup language for documents designed to be displayed in a web browser.
- .docx files are parsed and rendered by MS Word.
- .html files are parsed and rendered by web browsers.
- Human-readable
- Not a programming language (markup language)

HTML in Practice

- Prepend “view-source:” to a url to see the HTML code.
 - view-source:<https://www.skku.edu/skku/index.do>

```
1
2
3
4 <!doctype html>
5 <html lang="ko">
6 <head>
7 <title>성균관대학교</title>
8 <link rel="shortcut icon" href="/_res/skku/img/common/favicon.png">
9 <meta name="viewport" content="width=device-width,initial-scale=1.0,minimum-scale=1.0,maximum-scale=1.0,user-scalable=no">
10 <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
11 <meta http-equiv="Content-Script-Type" content="text/javascript" />
12 <meta http-equiv="Content-Style-Type" content="text/css" />
13 <meta http-equiv="X-UA-Compatible" content="IE=edge" />
14 <!-- Global site tag (gtag.js) - Google Analytics -->
15 <script async src="https://www.googletagmanager.com/gtag/js?id=UA-53596226-1"></script>
16 <script>
17   window.dataLayer = window.dataLayer || [];
18   function gtag(){dataLayer.push(arguments);}
19   gtag('js', new Date());
20
21   gtag('config', 'UA-53596226-1');
22 </script><link rel="canonical" href="http://www.skku.edu/skku/index.do" />
23
24 <link rel="stylesheet" type="text/css" href="/_common/cms.css" />
25 <link rel="stylesheet" type="text/css" href="/_res/skku/_css/layout.css" />
26 <!--[if IE 8]>
27   <link rel="stylesheet" type="text/css" href="/_common/ie8.css" />
```

Tags

- An HTML document consists of **tags**.
- Tags are keywords that define how web browser will format and display the content.

Welcome to HTML

HTML is a markup language for the Web!



[Do you want to know more?](#)

```
<!DOCTYPE html>
<html>
  <head></head>
  <body>
    <section>
      <h1>Welcome to HTML</h1>
      <p>HTML is a markup language for the Web!</p>

      <div class="footer">
        <a href=".">Do you want to know more?</a>
      </div>
    </section>
  </body>
</html>
```

Facts about Tags

- Tags are keywords inside angle brackets <>.
 - <h4>, </h4>, <p>, , ...
- There are opening tags and closing tags. Closing tags always start with a slash /.

```
<!DOCTYPE html>
<html>
  <head></head>
  <body>
    <section>
      <h1>Welcome to HTML</h1>
      <p>HTML is a markup language for the Web!</p>

      <div class="footer">
        <a href=".">Do you want to know more?</a>
      </div>
    </section>
  </body>
</html>
```

Facts about Tags

- Some tags do not have a closing. is one of them.
- Tags can be nested. A tag can be inside other tag, forming a tree structure.
 - A parent tag <section> in the document has four child tags: <h1>, <p>, , and <div>.

```
<!DOCTYPE html>
<html>
  <head></head>
  <body>
    <section>
      <h1>Welcome to HTML</h1>
      <p>HTML is a markup language for the Web!</p>

      <div class="footer">
        <a href=".">Do you want to know more?</a>
      </div>
    </section>
  </body>
</html>
```

- An HTML document must start with a doctype declaration.
 - `<!DOCTYPE html>`
 - This means that this HTML document is written in HTML 5 (the most recent version).
- The root tag must be `<html>`.
- Inside `<html>`, there must be two tags `<head>` and `<body>`.
- `<head>` provides the metadata of the page, e.g., title, character set, styles, scripts, ...
- `<body>` provides the content of the page. Tags in `<body>` are rendered on the screen.

Starter Example

- Usually, the first page is named as “index.html”.

```
<!DOCTYPE html>
<html>
  <head>
    <title>Page Title</title>
    <style>
      /* CSS Code */
    </style>
  </head>
  <body>
    <!-- HTML Code -->
    <script>
      /* JS Code */
    </script>
  </body>
</html>
```

Heading

- <h1>: the main heading
- <h2>...<h6>: headings for sections

```
<h1>This is heading 1</h1>  
<h2>This is heading 2</h2>  
<h3>This is heading 3</h3>  
<h4>This is heading 4</h4>  
<h5>This is heading 5</h5>  
<h6>This is heading 6</h6>
```

This is heading 1

This is heading 2

This is heading 3

This is heading 4

This is heading 5

This is heading 6

Paragraph, Emphasis, and Hyperlink

- `<p>` creates a paragraph (has a bottom margin by default).
- `` and `` emphasize text.
- `<a>` creates a hyperlink.
 - Set the "href" attribute to designate the url.

This is a *sample* **paragraph**. Click [here](#) to go to Google.

Second paragraph!

```
<p>
  This is a <em>sample</em> <strong>paragraph</strong>.
  Click <a href="www.google.com">here</a> to go to Google.
</p>
<p>
  Second paragraph!
</p>
```


List

- `` and `` create unordered and ordered lists respective.
- `` specifies a list item.
- Lists can be nested.

```
<ul>
  <li>Item 1</li>
  <li>Item 2</li>
  <li>
    List Item
    <ol>
      <li>Item 3a</li>
      <li>Item 3b</li>
      <li>Item 3c</li>
    </ol>
  </li>
</ul>
```

- Item 1
- Item 2
- List Item
 - 1. Item 3a
 - 2. Item 3b
 - 3. Item 3c


Image


- `` puts an image in an HTML document.
- You can set the size of the image using the *width* and *height* attributes.

```
<h1>Here is my cat</h1>  

```

☐ Name

 cat.jpg

 index.html

Here is my cat



Table

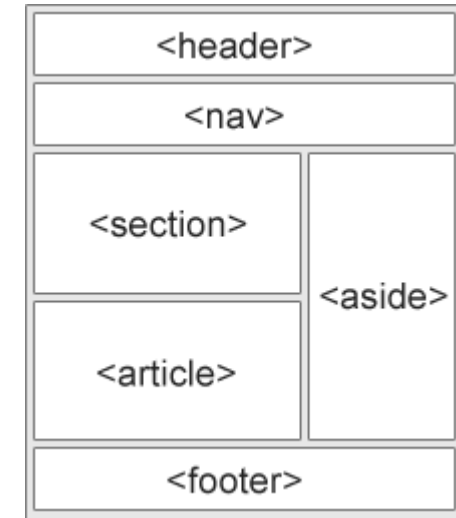
- `<table>` creates a table.
- `<tr>` represents a row.
- `<th>` and `<td>` represent a cell (`<th>` for table header).

```
<table>
  <tr>
    <th>th 1</th>
    <th>th 2</th>
  </tr>
  <tr>
    <td>cell 1</td>
    <td>cell 2</td>
  </tr>
  <tr>
    <td>cell 3</td>
    <td>cell 4</td>
  </tr>
</table>
```

| th 1 | th 2 |
|--------|--------|
| cell 1 | cell 2 |
| cell 3 | cell 4 |

Grouping

- `<div>` represents a group or block of tags. By default, it does not affect the rendering (it is transparent).
- Later, you will apply CSS rules to layout the elements in `<div>` (e.g., flexbox)
- `<header>`, `<nav>`, `<section>`, ...: Same as `<div>`, but define the semantic of tags.



Tags You Must Know

- `Click me` creates a hyperlink.
- `<body>` and `<head>`
- `text` emphasizes text (*italic* by default).
- `<h1>` to `<h6>` for titles
- `` includes an image.
- `` and `` for ordered and unordered lists
- `` represents a list item in `` or ``.
- `<p>` for paragraphs
- `` defines text with strong importance (**bold** by default).

```

<!DOCTYPE html>
<html>
  <head>
    <title>My Profile</title>
  </head>
  <body>
    <h1>My Profile</h1>
    <p>
      Hello, I am a software engineer. I love to develop <strong>ope
n-source projects</strong>.
      I also like cats; they are <em>awesome</em>!
    </p>

    
    <h2>Useful Links</h2>
    <ul>
      <li><a href="https://www.google.com">Google</a></li>
      <li><a href="https://www.github.com">Github</a></li>
      <li><a href="http://eportal.skku.edu/">Portal</a></li>
    </ul>
    <h2>My Schedule</h2>
    <table>
      <tr><th>Monday</th><th>Tuesday</th></tr>
      <tr><td>Busy</td><td>Available</td></tr>
    </table>
    <h3>Todos</h3>
    <ol>
      <li>Finish a project</li>
      <li>Get a haircut</li>
    </ol>
  </body>
</html>

```

My Profile

Hello, I am a software engineer. I love to develop **open-source projects**.
 I also like cats; they are *awesome*!



Useful Links

- [Google](https://www.google.com)
- [Github](https://www.github.com)
- [Portal](http://eportal.skku.edu/)

My Schedule

| Monday | Tuesday |
|--------|-----------|
| Busy | Available |

Todos

1. Finish a project
2. Get a haircut

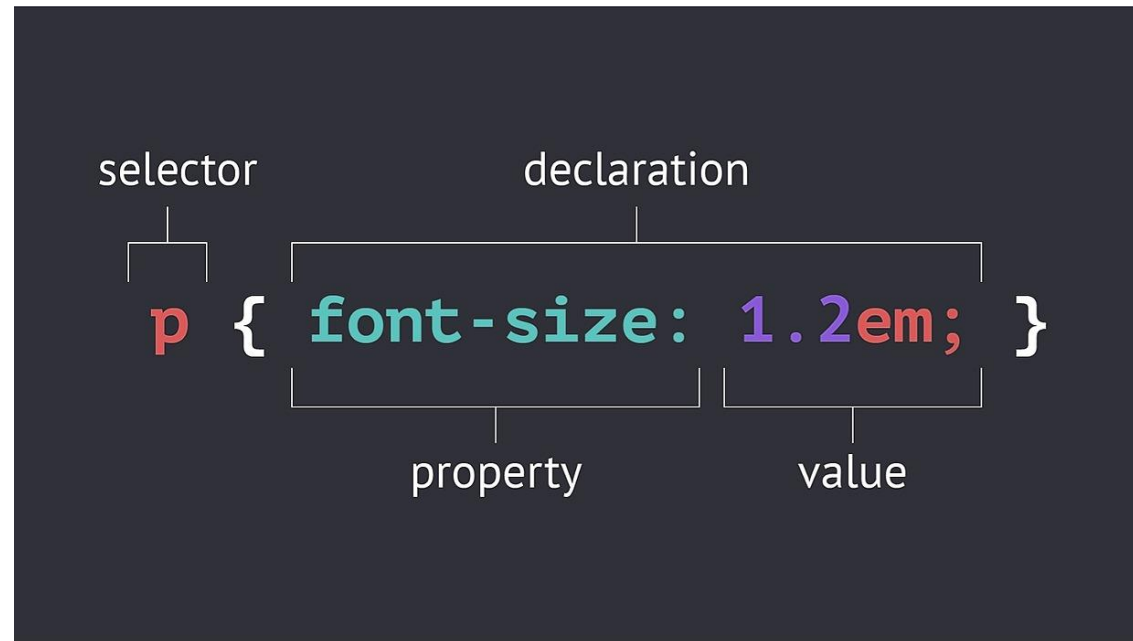
Why CSS?

- **Cascading Style Sheets or CSS** is a style sheet language used for describing the presentation of a document written in a markup language such as HTML.
- Without CSS, your HTML page will be rendered using your browser's default style.
 - Very ugly!

- Like HTML, a CSS file is a human-readable text file.

```
#courses li {  
  display: inline-block;  
  padding: .25em .4em;  
  font-size: 75%;  
  line-height: 1;  
  text-align: center;  
  white-space: nowrap;  
  vertical-align: baseline;  
  border-radius: .25rem;  
  background: #2196F3;  
  color: white;  
  font-weight: bold;  
}
```


- A CSS file consists of multiple CSS rules.
- Rule = Selector + (Property and Value)



CSS Selector

- The selector part specifies to which tags the following rule applies.
- There are many selector types, but only remember these three: by a tag name, by a class name, or by an id.
- Tag name selector *p*: select all `<p>` tags

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      p {
        color: red;
      }
    </style>
  </head>
  <body>
    <p>Change my color</p>
  </body>
</html>
```

Change my color

CSS Selector

- What if there are two `<p>` tags and you want to style only one of them?
- Give an id to the target tag.
- Id selector `#target`: select the element whose id is target.
- Id must be unique throughout a document (used for only one tag!)

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      #target {
        color: red;
      }
    </style>
  </head>
  <body>
    <p id="target">Change my color</p>
    <p>Not me!</p>
  </body>
</html>
```

Change my color

Not me!

CSS Selector

- What if there are three <p> tags and you want to style two of them?
- Giving an id is not a good idea since you cannot give the same id to two different tags. Use class instead!

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      .targets {
        color: red;
      }
    </style>
  </head>
  <body>
    <p id="target" class="targets">Change my color</p>
    <p class="targets">Me too!</p>
    <p>Not me!</p>
  </body>
</html>
```

Change my color

Me too!

Not me!

- If you want to select all elements with a specific tag, use the tag name selector.
 - *p*
- If you want to select a unique element, give it an id and use the id selector.
 - e.g., global navigation, login button, ...
 - *#global-nav*
- If you want to select multiple elements, give them a class name and use the class selector.
 - e.g., active items, highlighted links, ...
 - *.active*

Combining Selectors

- **#wrapper.content** selects all tags that have a class name “content” under an element whose id is “wrapper”.
- **p.highlight** selects all p tags that have a class name “highlight”.
- **#wrapper.bordered** selects an element whose id is “wrapper” and has a class name “bordered”.

```
<!DOCTYPE html>
<html>
  <head></head>
  <body>
    <div id="wrapper" class="bordered">
      <p class="content highlight">Highlighted Content</p>
      <p class="content">Content</p>
    </div>
  </body>
</html>
```

Using CSS in HTML

- There are three ways to use CSS in your HTML document:
- **Using a <link> tag**
 - Create a separate CSS file.
 - Add a <link> tag to the <body> tag.
 - Below, I assumed that the HTML and CSS files are in the same directory.

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

Using CSS in HTML

- **Using a <style> tag**

- Embed CSS rules in a <style> tag.
- The <style> tag can be placed anywhere but usually in the <head> tag.

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      body {
        background: blue;
      }

      /* More css rules... */
    </style>
  </head>
  <body>
    <!-- ... -->
  </body>
</html>
```


Using CSS in HTML

- **Using the style attribute**

- You can embed CSS rules for each HTML element using the *style* attribute.
- *Inline styles*
- Inline styles are very strong. They even override rules set by an id selector.

```
<!DOCTYPE html>
<html>
  <head></head>
  <body>
    <p style="color: red; font-size: 12px">
      <!-- ... -->
    </p>
  </body>
</html>
```

```

<!DOCTYPE html>
<html>
  <head>
    <title>My Profile</title>
    <link rel="stylesheet" href="style.css" >
  </head>
  <body>
    <h1>My Profile</h1>

    <p>
      Hello, I am a software engineer. I love to develop
      <strong>open-source projects</strong>. <br />
      I also like cats; they are <em>awesome</em>!
    </p>

    
    <h2>Useful Links</h2>
    <ul>
      <li><a href="https://www.google.com">Google</a></li>
      <li><a href="https://www.github.com">Github</a></li>
      <li><a href="http://eportal.skku.edu/">Portal</a></li>
    </ul>
    <h2>My Schedule</h2>
    <table>
      <tr>
        <th>Monday</th>
        <th>Tuesday</th>
      </tr>
      <tr>
        <td>Busy</td>
        <td>Available</td>
      </tr>
    </table>
    <h3>Todos</h3>
    <ol>
      <li>Finish a project</li>
      <li>Get a haircut</li>
    </ol>
  </body>
</html>

```

My Profile

Hello, I am a software engineer. I love to develop **open-source projects**.
 I also like cats; they are *awesome*!



Useful Links

- [Google](https://www.google.com)
- [Github](https://www.github.com)
- [Portal](http://eportal.skku.edu/)


My Schedule

| Monday | Tuesday |
|--------|-----------|
| Busy | Available |

Todos

1. Finish a project
2. Get a haircut

 cat.jpg

 index.html

 style.css

Text Styling

- color: <color code>
- font-weight: normal/bold
- text-decoration: none/underline
- font-style: normal/italic
- font-size: <size>
- text-align: left/center/right
- What if I want to style only a part of text?

```
p {  
  color: orange; /* or #ff0000 */  
  font-weight: bold;  
  text-decoration: underline;  
  font-style: italic;  
  font-size: 15px;  
  text-align: center;  
}
```

Hello, I am a software engineer. I love to develop open-source projects.
I also like cats; they are awesome!

Text Styling

- Wrap the text with ``.
- `` does not affect the rendering, but you can apply CSS rules to a certain portion of text!

```
I also like <span class="cats">cats</span>;  
they are <em>awesome</em>!
```

Hello, I am a software engineer. I love to develop **open-source projects**.
I also like *cats*; they are *awesome*!

```
.cats {  
  color: orange; /* or #ff0000 */  
  font-weight: bold;  
  text-decoration: underline;  
  font-style: italic;  
  font-size: 15px;  
}
```

Defining a Size

- Sometimes, you need to define a size or length in CSS.
 - width, height, font-size, padding, margin, ...
- px: pixels
- em: a unit of measurement, where 1 em = current font size.
- rem: root em. Same measurement as em without inheritance
- %: percentages

Margin and Padding

- It seems that `<p>` and `` are too close.
- Let's give more space below `<p>`.
- Margin: outer spacing
- Padding: inner spacing



My Profile

Hello, I am a software engineer. I love to develop **open-source projects**.
I also like *cats*; they are *awesome*!



Margin and Padding

```
p {  
  border: 1px solid black;  
  padding: 20px;  
}
```

Hello, I am a software engineer. I love to develop **open-source projects**.
I also like cats, they are *awesome*!



```
p {  
  border: 1px solid black;  
  margin: 20px;  
}
```

Hello, I am a software engineer. I love to develop **open-source projects**.
I also like cats, they are *awesome*!



```
p {  
  margin-bottom: 40px;  
}
```

Hello, I am a software engineer. I love to develop **open-source projects**.
I also like cats, they are *awesome*!



Aligning to Center

- Let's put everything in `<body>` to `<div>` whose id is wrapper.
- `margin:auto` aligns the wrapper to the center.
 - Useful technique!

```
<body>
  <div id="wrapper">
    <h1>My Profile</h1>
    <!--...-->
  </div>
</body>
```

```
body {
  background: #eeeeee;
}

#wrapper {
  width: 720px;
  background: white;
  margin: auto;
}
```

My Profile

Hello, I am a software engineer. I love to develop **open-source** projects.
I also like *cats*; they are *awesome*!



Useful Links

- [Google](#)
- [Github](#)
- [Portal](#)

My Schedule

Monday Tuesday
Busy Available

Todos

1. Finish a project
2. Get a haircut

More Styling...

```
body {  
  background: #eeeeee;  
}  
  
#wrapper {  
  width: 720px;  
  background: white;  
  margin: auto;  
  padding: 10px;  
  border-radius: 10px;  
  border: 1px solid lightgray;  
  box-  
shadow: 0px 3px 3px rgba(0, 0, 0,  
  0.1);  
}
```

My Profile

Hello, I am a software engineer. I love to develop **open-source** projects.
I also like *cats*; they are *awesome*!



Useful Links

- [Google](#)
- [Github](#)
- [Portal](#)

My Schedule

Monday Tuesday
Busy Available

Todos

1. Finish a project
2. Get a haircut

My Profile

Hello, I am a software engineer. I love to develop **open-source** projects.
I also like *cats*; they are *awesome*!



Useful Links

- [Google](#)
- [Github](#)
- [Portal](#)

My Schedule

Monday Tuesday
Busy Available

Todos

1. Finish a project
2. Get a haircut

More Styling...

```
body {  
  font-family: "Segoe UI";  
}  
  
h1 {  
  text-align: center;  
  margin-top: 0px;  
}
```

My Profile

Hello, I am a software engineer. I love to develop **open-source projects**.
I also like cats; they are *awesome*!



My Profile

Hello, I am a software engineer. I love to develop **open-source projects**.
I also like cats; they are *awesome*!




More Styling...

```
img {  
  border-radius: 12px;  
}  
  
h2 {  
  border-left: 3px solid orange;  
  padding-left: 5px;  
}  
  
a {  
  color: steelblue;  
}  
  
a:visited {  
  color: darkblue;  
}
```

My Profile

Hello, I am a software engineer. I love to develop **open-source projects**.
I also like **cats**; they are *awesome*!



Useful Links

- [Google](#)
- [Github](#)
- [Portal](#)

My Schedule

| Monday | Tuesday |
|--------|-----------|
| Busy | Available |

Todos

1. Finish a project
2. Get a haircut

Final CSS File

```
body {
  background: #eeeeee;
  font-family: "Segoe UI";
}

p {
  margin-bottom: 40px;
}

#wrapper {
  width: 720px;
  background: white;
  margin: auto;
  padding: 10px;
  border-radius: 10px;
  border: 1px solid lightgray;
  box-shadow: 0px 3px 3px rgba(0, 0, 0, 0.1);
}

.cats {
  color: orange; /* or #ff0000 */
  font-weight: bold;
  text-decoration: underline;
  font-style: italic;
  font-size: 15px;
}
```

```
h1 {
  text-align: center;
  margin-top: 0px;
}

img {
  border-radius: 12px;
}

h2 {
  border-left: 3px solid orange;
  padding-left: 5px;
}

a {
  color: steelblue;
}

a:visited {
  color: darkblue;
}
```

Selector Specificity

- When two rules conflict, the more specific rule is applied.
- Determined by selectors: *#id* > *.class* > *tag name*
- To override, add !important to the style declaration.
- `<p id="main" class="content welcome">abc</p>`
- `#main { color: red; }`
- `.content { color: blue; }`
- `p { color: green !important; }`

Selector Specificity

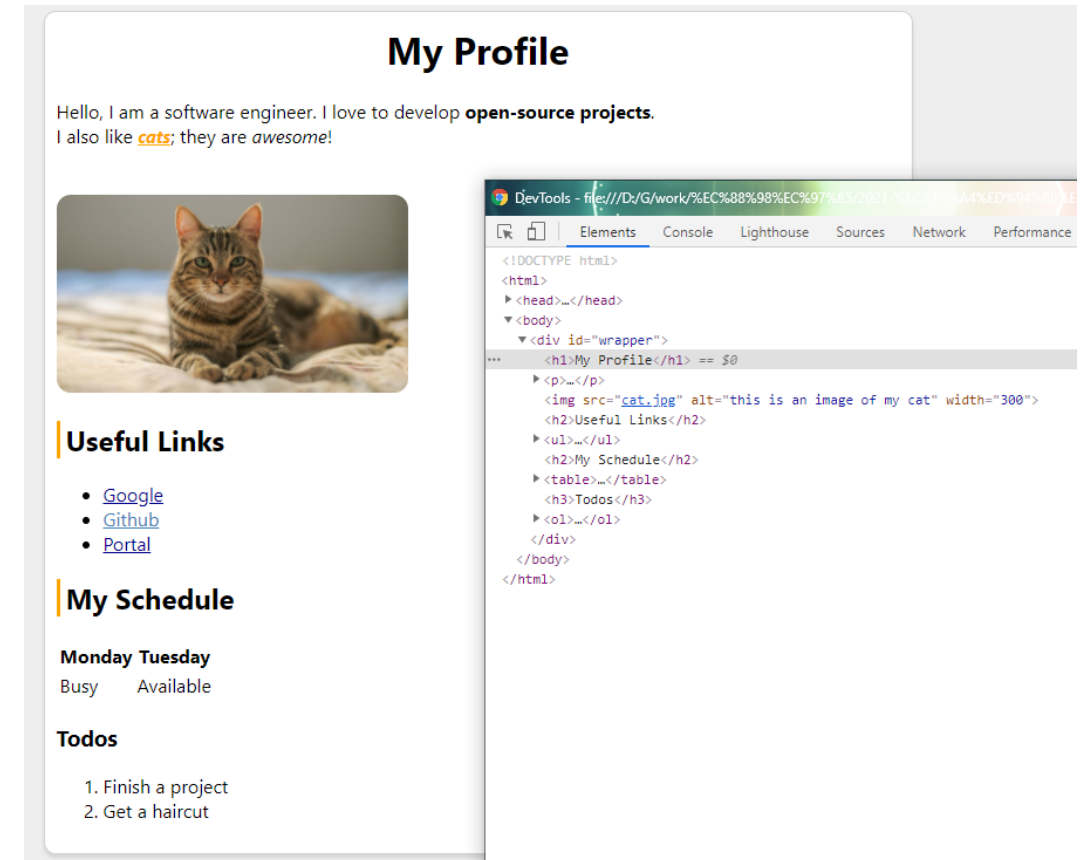
```
<p id="main" class="content welcome">abc</p>
```

```
/* 1 */ #main { color: red; font-size: 12px;}  
/* 2 */ .content { color: orange; }  
/* 3 */ p { color: blue; }  
/* 4 */ p.content.welcome { color: green; }  
/* 5 */ p { color: purple !important; font-size: 10px;}
```

- 1 vs 2 -> 1 wins.
- 2 vs 3 -> 2 wins.
- 3 vs 4 -> 4 wins.
- 1 vs 4 -> 1 wins.
- 1 vs 5 -> draw (color is set to purple, but font-size is set to 12px).

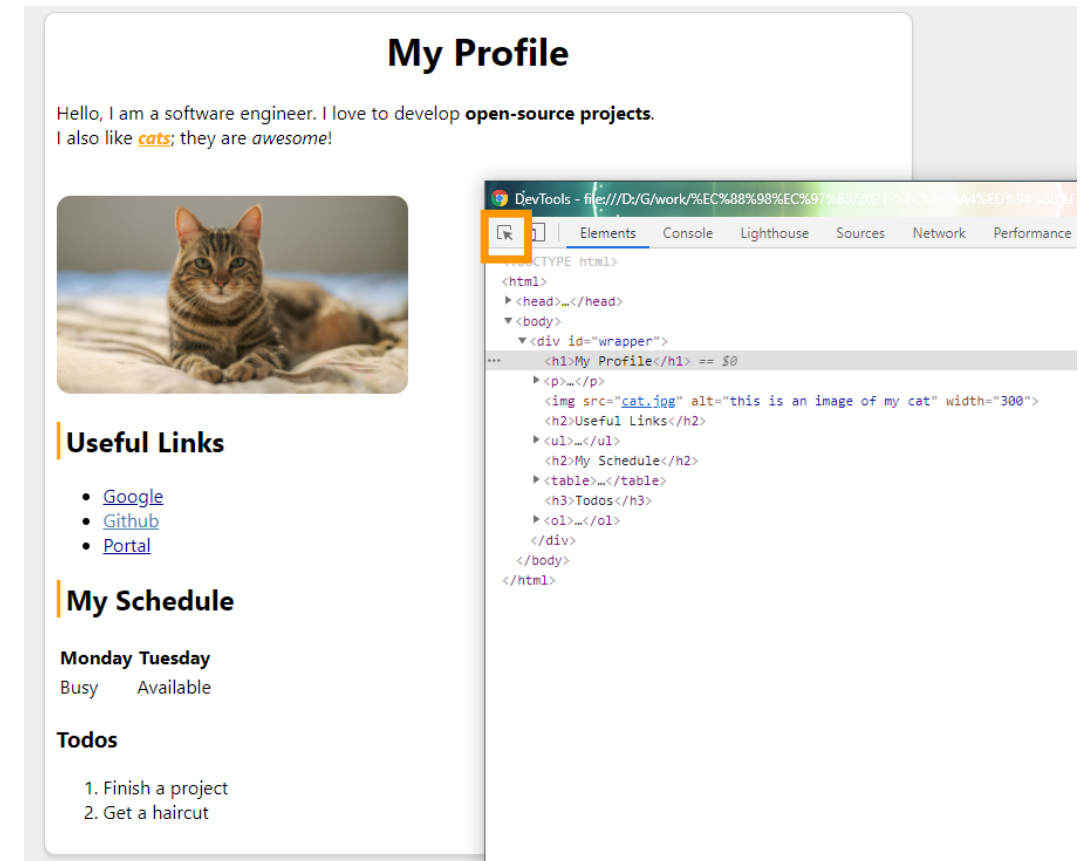
Debugging CSS

- Sometimes, your CSS rule does not apply!
- Menu -> Tools -> Developer Tool
- If you are using Chrome on Windows, press <F12>.
- In the “Elements” tab, you can see the HTML code that your browser is showing currently.



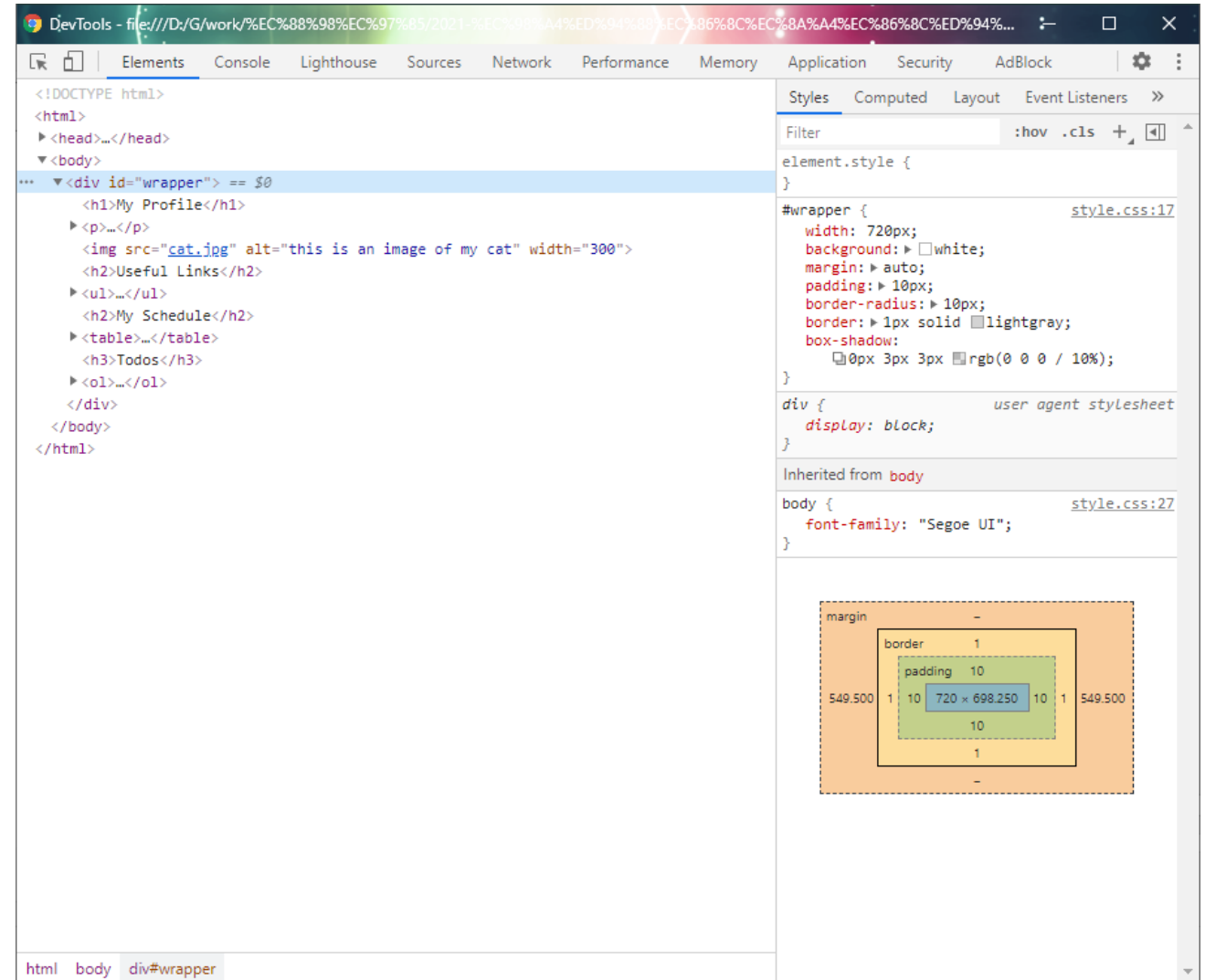
Debugging CSS

- Use the inspector tool to debug.
- Click on the inspector tool (top left).
- Click on the tag you want to inspect.
- You can choose a tag in the HTML view as well.



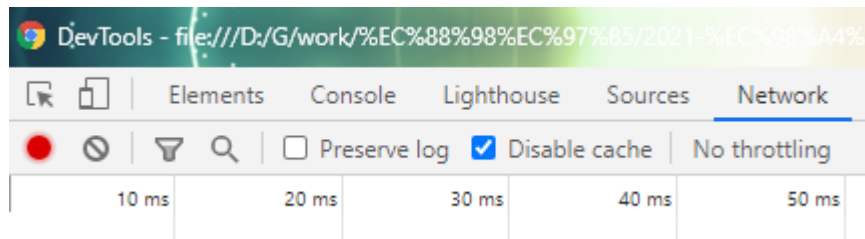
Debugging CSS

- Styles applied to the element
 - You can change values or apply new styles.
- Box model
 - margin and padding
- Hierarchy
 - html > body > div#wrapper



Tips

- Sometimes, your page does not update even though you changed the source code.
- This can be a cache issue.
- Open DevTool -> Go to Network -> Check “Disable cache”
 - This option works only when DevTool is open. So keep it open.



Resources

- Source code: <https://github.com/e-/profile>
- <https://e-.github.io/profile/>
- HTML Tag Reference: <https://www.w3schools.com/TAGS/default.ASP>
- CSS Reference: <https://www.w3schools.com/cssref/>
- HTML cheat sheet: <https://htmlcheatsheet.com/>
- CSS cheat sheet: <https://htmlcheatsheet.com/css/>

Summary: HTML & CSS

- **HTML** for the structure of a web page
- **CSS** for the style
- **JavaScript** for the interaction

- An HTML document consists of tags (hierarchy!).
 - `<html>`, `<head>`, `<body>`, `<p>`, ``, ``, ``, `<table>`, ...

- CSS rule = selector + (property: value)
 - Id selector: `#id`, class selector: `.class`, tag name selector: `tag_name`
 - Properties: color, margin, padding, background, font-size, ...
 - Selector specificity