EP1000
CSS

(more) HTML

- Before we can get to CSS, we need to cover a little more about HTML
  Review

  - HTML is for CONTENT
  - CSS is for LAYOUT

Often we “break-up” a page into sections (<div>, <section>)

- Easier management
- Can generate a template for consistent use
- Duplicate pages, changing only relevant sections.

The <head> Element

- the HTML <head> element is a container for
  - <title>
  - <link>
  - <style>
  - <script>
  - <meta>
- contains metadata (data about the HTML document)
- Metadata is not displayed.

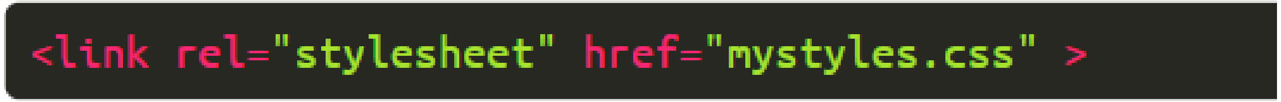
The <head> Element (cont'd)

<title>

- text which is displayed on the browser's bar or page's tab
- required in HTML documents
- purpose
  - defines a title in the browser's toolbar
  - provides a title for the page when added to favourites
  - displays a title for page in search engine results.

<link>

- defines the relationship between current document and an external resource.
- most often used to link external style sheets



```
<link rel="stylesheet" href="mystyles.css" >
```

The <head> Element (cont'd)

<style>

- used to define style information for a SINGLE HTML page.

<script>

- used to define client-side Javascripts

<meta>

- used to provide information about the page.
- used by browser to decide how to reload/render the page
- used by search engines to define keywords and content

Block and Inline

Block-level Elements

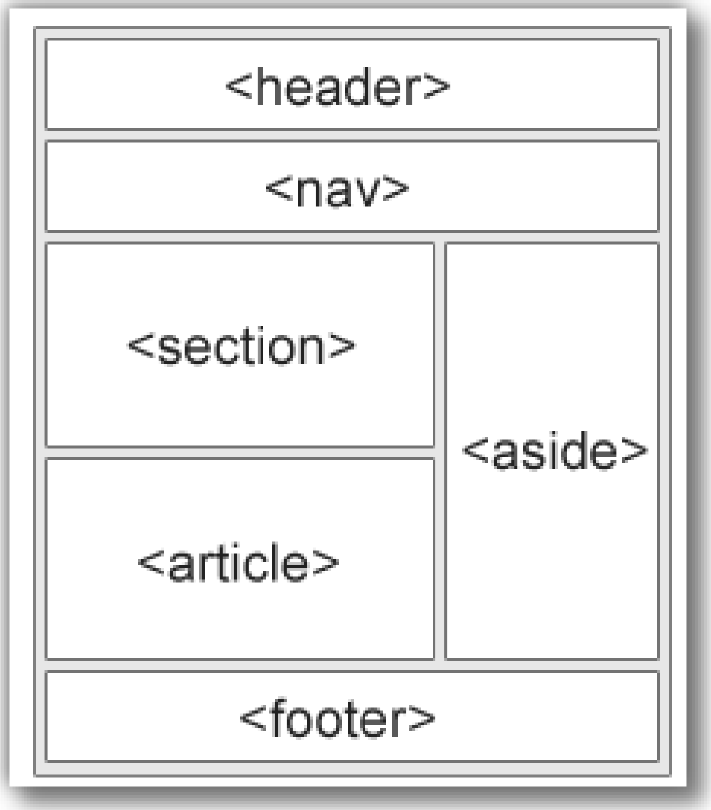
- Starts on a new line
- Takes up full width available
- Has top and bottom margin
- Example
    - <div> used as a container

Inline Elements

- Does not start on a new line
- Takes up as much space as necessary
- cannot contain a block-level element
- Example:
    - <span> used to mark up a block of text

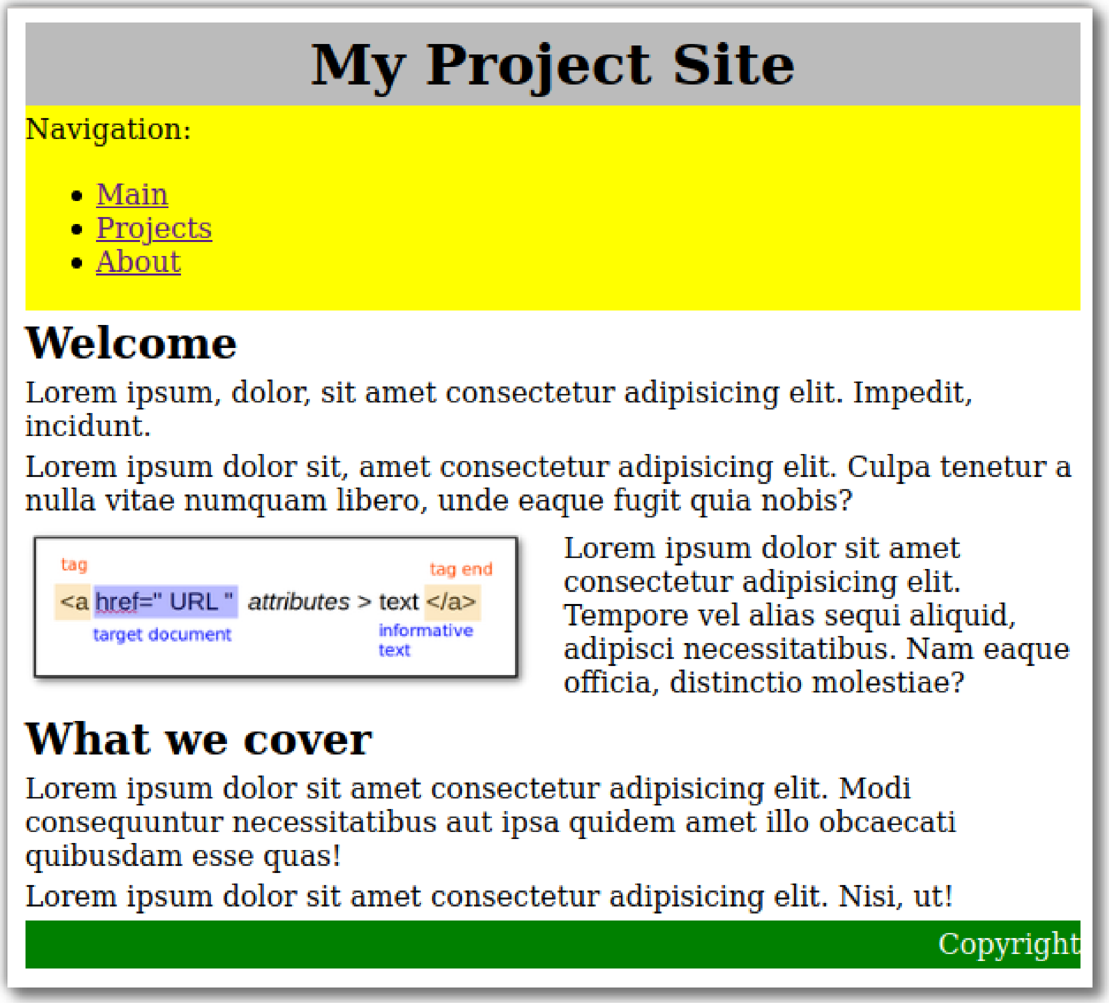
Layout Elements and Techniques

- uses a kind-of <div>
- (Semantic elements) that define the different parts of a web page
  - header
  - nav
  - section
  - article
  - aside
  - footer
- Clearly describes its meaning to both the browser and developer
- Layout techniques:
  - CSS framework
  - CSS float property
  - CSS flexbox
  - CSS grid



Typical Project Page

- Can you break down the blocks?
- Can you name them?
- How about the layout?



My Project Site

Navigation:

- Main
- Projects
- About

Welcome

Lorem ipsum, dolor, sit amet consectetur adipisicing elit. Impedit, incidunt.

Lorem ipsum dolor sit, amet consectetur adipisicing elit. Culpa tenetur a nulla vitae numquam libero, unde eaque fugit quia nobis?

tag

tag end

<a href=" URL " *attributes* > text </a>

target document

informative text

Lorem ipsum dolor sit amet consectetur adipisicing elit. Tempore vel alias sequi aliquid, adipisci necessitatibus. Nam eaque officia, distinctio molestiae?

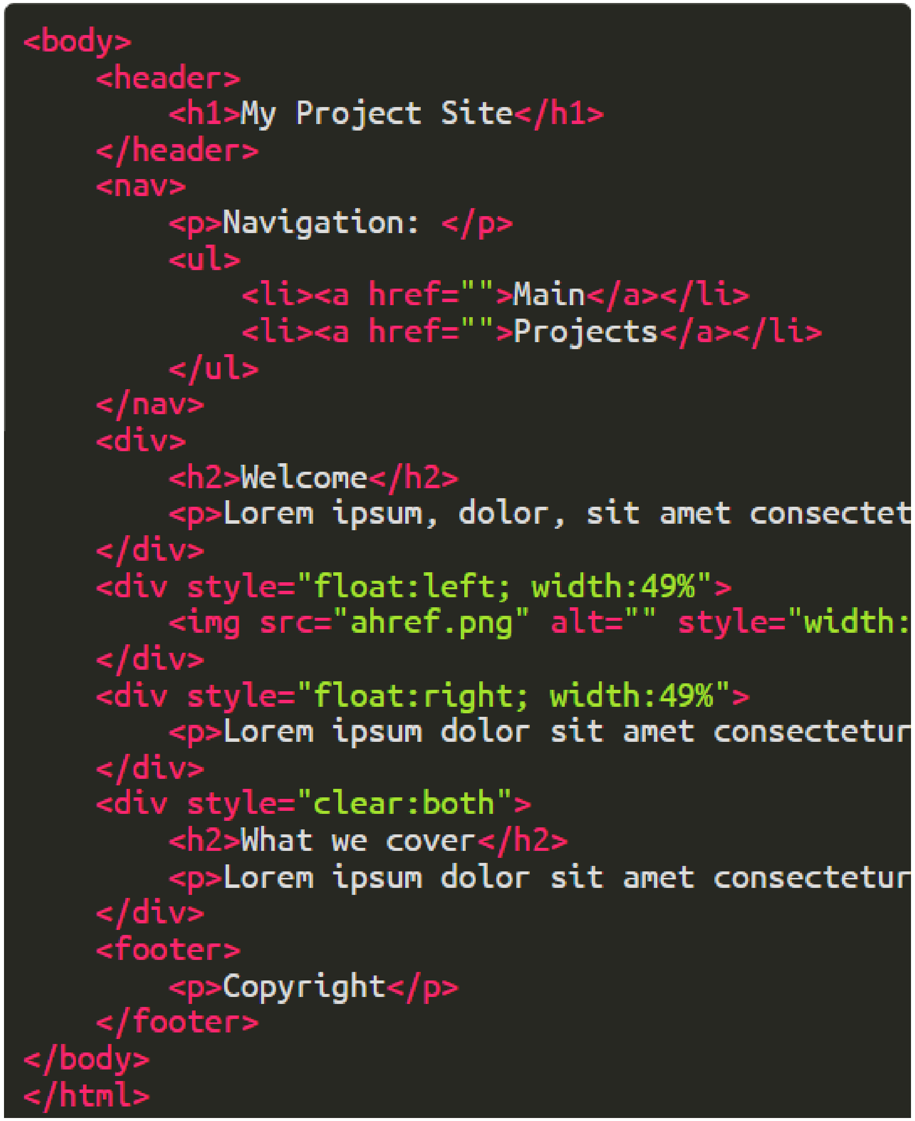
What we cover

Lorem ipsum dolor sit amet consectetur adipisicing elit. Modi consequuntur necessitatibus aut ipsa quidem amet illo obcaecati quibusdam esse quas!

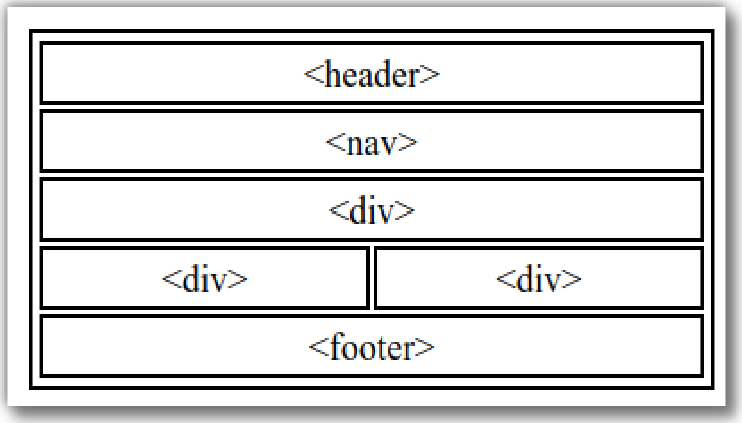
Lorem ipsum dolor sit amet consectetur adipisicing elit. Nisi, ut!

Copyright

Typical Page Breakdown



```
<body>
    <header>
        <h1>My Project Site</h1>
    </header>
    <nav>
        <p>Navigation: </p>
        <ul>
            <li><a href="">Main</a></li>
            <li><a href="">Projects</a></li>
        </ul>
    </nav>
    <div>
        <h2>Welcome</h2>
        <p>Lorem ipsum, dolor, sit amet consectet
    </div>
    <div style="float:left; width:49%">
        <img src="ahref.png" alt="" style="width:
    </div>
    <div style="float:right; width:49%">
        <p>Lorem ipsum dolor sit amet consectetur
    </div>
    <div style="clear:both">
        <h2>What we cover</h2>
        <p>Lorem ipsum dolor sit amet consectetur
    </div>
    <footer>
        <p>Copyright</p>
    </footer>
</body>
</html>
```



<header>

<nav>

<div>

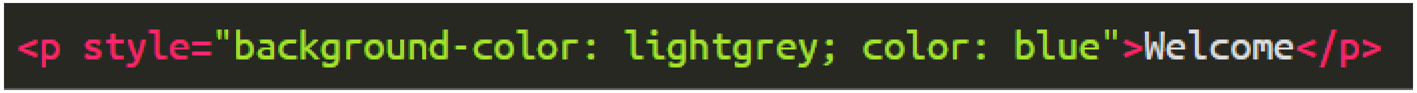
<div> <div>

<footer>

Cascading Style Sheets (CSS)

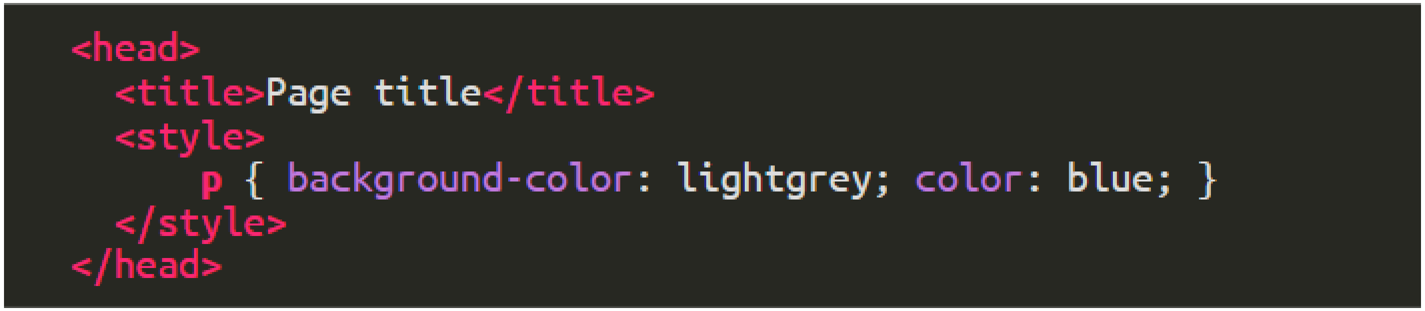
- Describes how HTML elements are to be displayed.
- Controls LAYOUT

INLINE



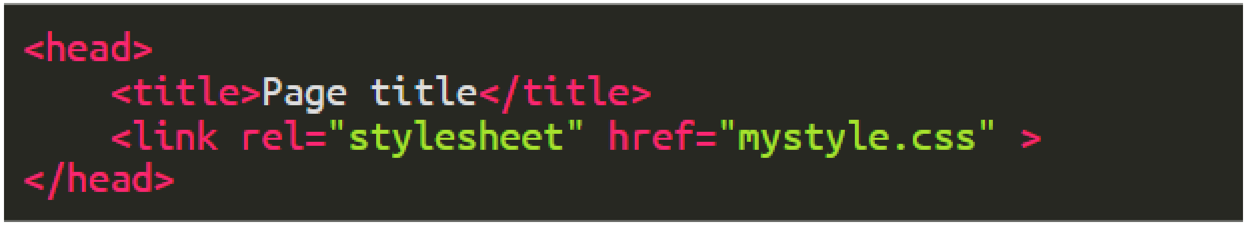
```
<p style="background-color: lightgrey; color: blue">Welcome</p>
```

within HTML file <head>



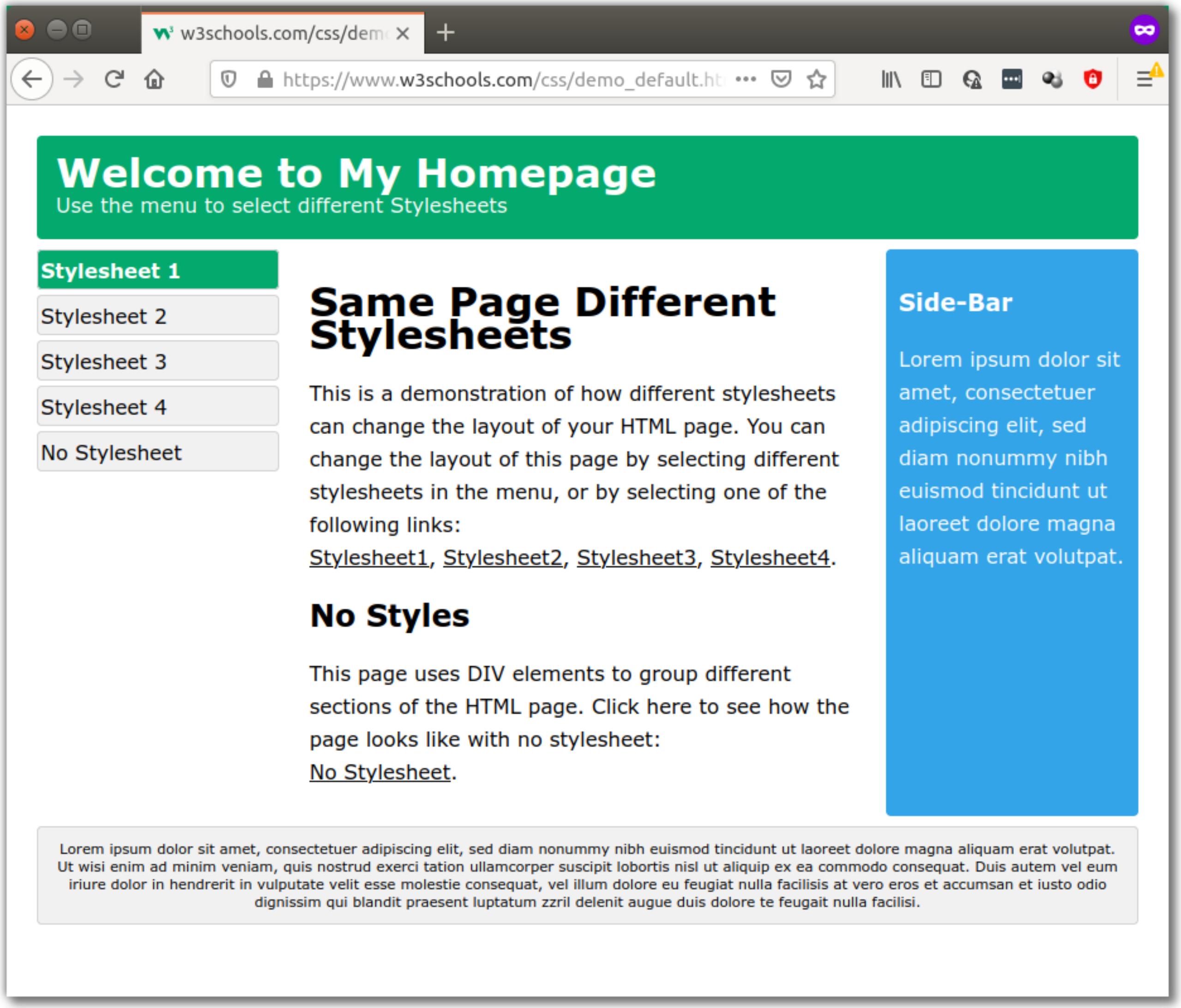
```
<head>
  <title>Page title</title>
  <style>
      p { background-color: lightgrey; color: blue; }
  </style>
</head>
```

In an external CSS file (recommended)



```
<head>
    <title>Page title</title>
    <link rel="stylesheet" href="mystyle.css" >
</head>
```

One HTML Multiple Styles



Welcome to My Homepage

Use the menu to select different Stylesheets

Same Page Different Stylesheets

This is a demonstration of how different stylesheets can change the layout of your HTML page. You can change the layout of this page by selecting different stylesheets in the menu, or by selecting one of the following links: Stylesheet1, Stylesheet2, Stylesheet3, Stylesheet4.

No Styles

This page uses DIV elements to group different sections of the HTML page. Click here to see how the page looks like with no stylesheet: No Stylesheet.

Side-Bar

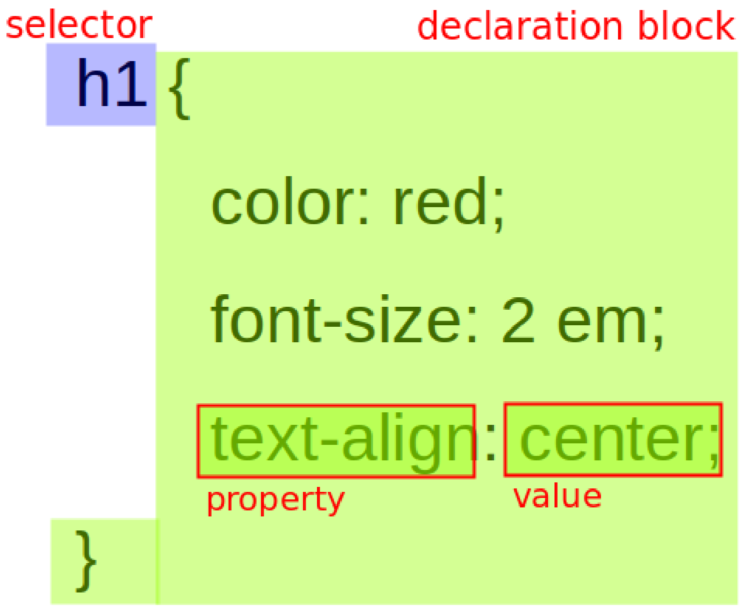
Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blandit praesent luptatum zzril delenit augue duis dolore te feugait nulla facilisi.

Example:
W3Schools One HTML Page - Multiple Styles

CSS Rule Syntax

- A CSS Rule consists of
  - a selector which identifies the HTML element
  - a declaration block which holds the formatting



```
h1 {
  color: red;
  font-size: 2 em;
  text-align: center;
}
```

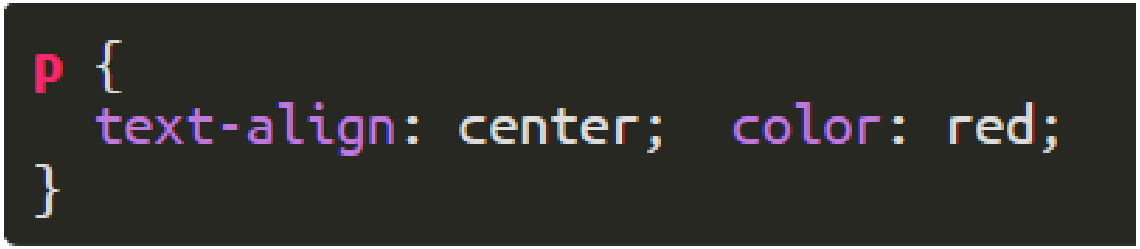
- Selector points to the HTML element to style
- Declaration defines how to format it using
- a property
- a separator (colon (:))
- value for that property
- each declaration is separated by a semi-colon (;)
- you can have many declarations in a block
- you can have many CSS rules in a file.
- All selected elements on the page will be formatted

CSS Selectors

Selectors are used to "find" the HTML element you want to style

element Selector

- Selects the HTML element based on the element name.
- Affects all elements on the page

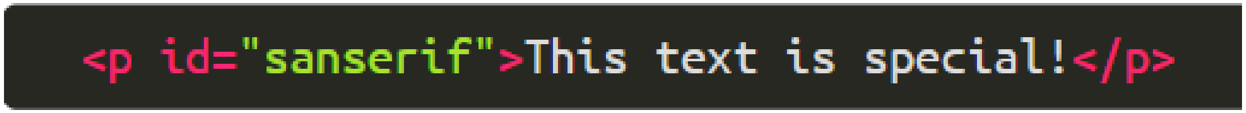


```
p {
  text-align: center;  color: red;
}
```

CSS Selectors - id

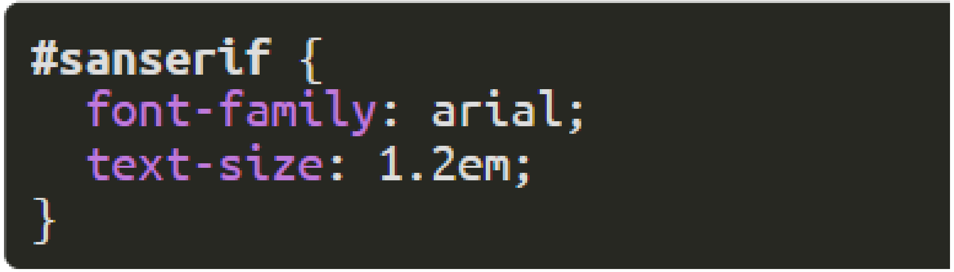
id Selector

- Uses the id attribute of an HTML element
- Unique within a page, should be used only once
- To select the specific id, write a hash (#) followed by the id



```
<p id="sanserif">This text is special!</p>
```

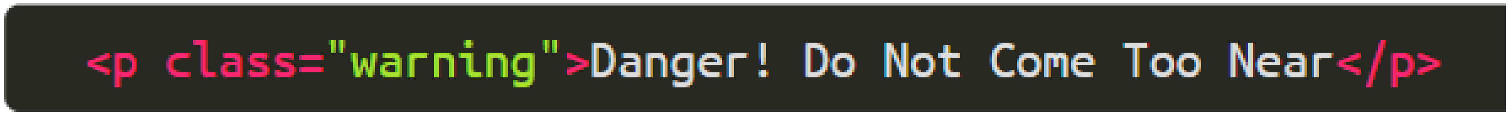
with



```
#sanserif {
  font-family: arial;
  text-size: 1.2em;
}
```

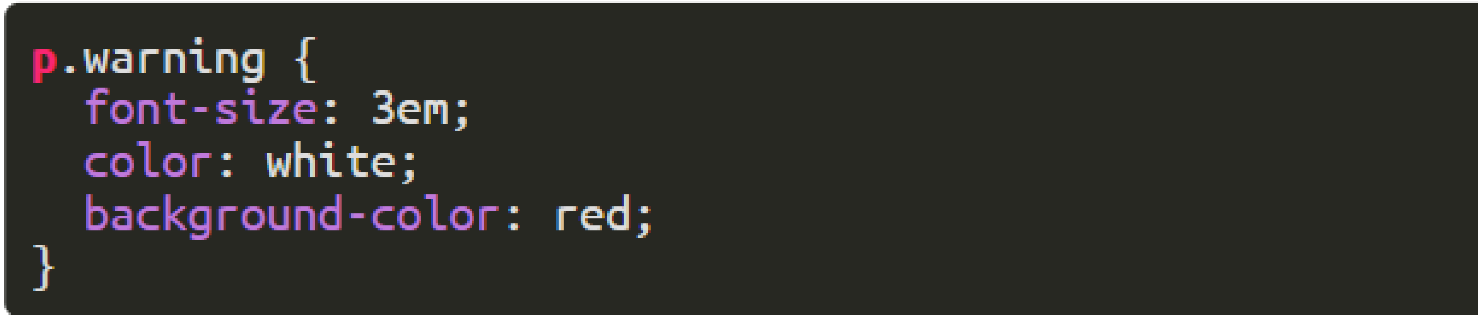
CSS Selectors - class

- Selects HTML elements with a specific class attribute
- There may be more than one occurrence of this class
- To select elements with a specific class, write a period (.), followed by the class name.



```
<p class="warning">Danger! Do Not Come Too Near</p>
```

with

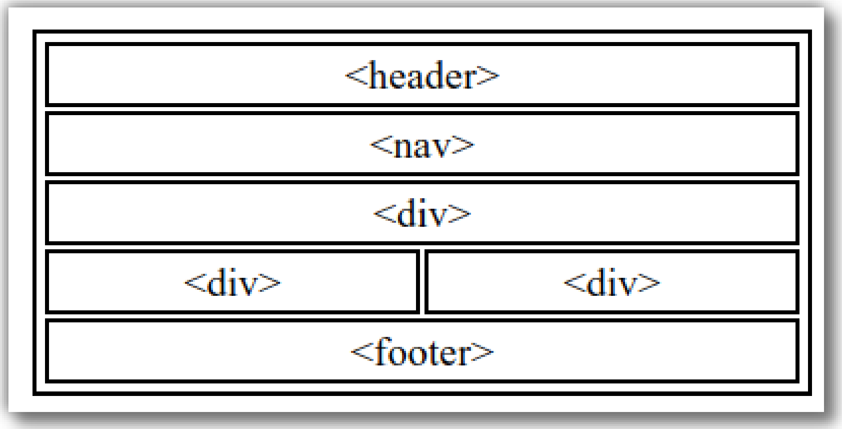


```
p.warning {
  font-size: 3em;
  color: white;
  background-color: red;
}
```

CSS Class vs ID: Which One To Use

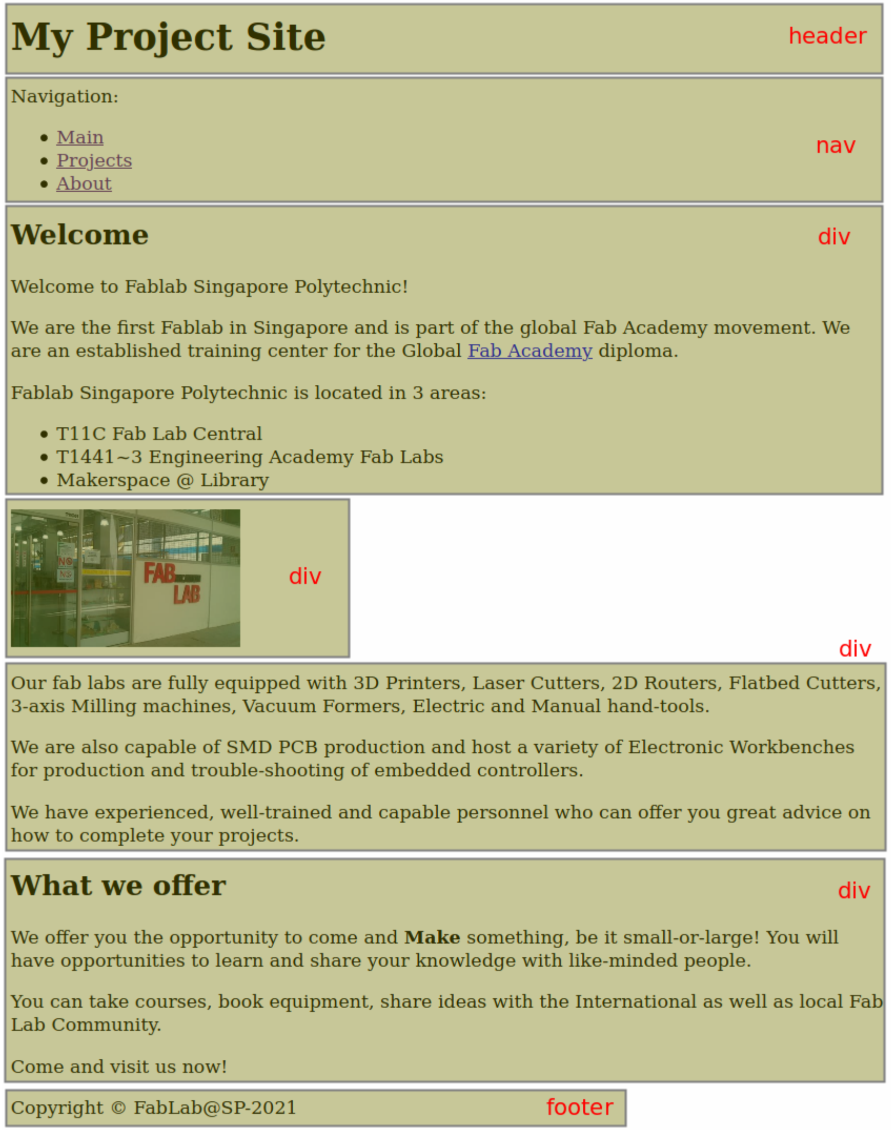
Exercise

- We start learning CSS with a page which has some content.
- The structure of the page is shown on the right.
- We will look into formatting the page.



Try building the HTML with CSS.

HTML Content Blocks



My Project Site

Navigation:

- Main
- Projects
- About

Welcome

Welcome to Fablab Singapore Polytechnic!

We are the first Fablab in Singapore and is part of the global Fab Academy movement. We are an established training center for the Global Fab Academy diploma.

Fablab Singapore Polytechnic is located in 3 areas:

- T11C Fab Lab Central
- T1441~3 Engineering Academy Fab Labs
- Makerspace @ Library

Our fab labs are fully equipped with 3D Printers, Laser Cutters, 2D Routers, Flatbed Cutters, 3-axis Milling machines, Vacuum Formers, Electric and Manual hand-tools.

We are also capable of SMD PCB production and host a variety of Electronic Workbenches for production and trouble-shooting of embedded controllers.

We have experienced, well-trained and capable personnel who can offer you great advice on how to complete your projects.

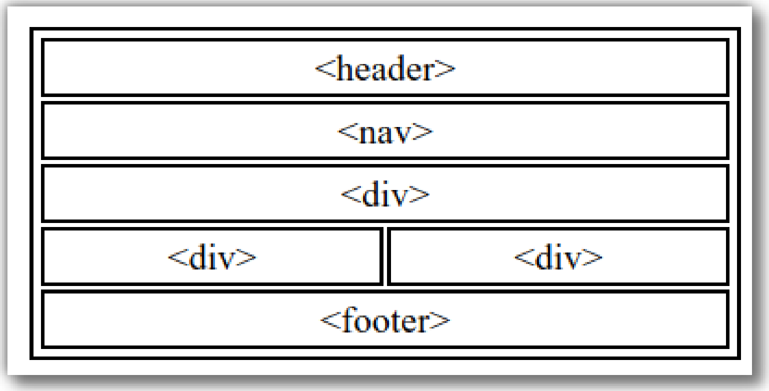
What we offer

We offer you the opportunity to come and **Make** something, be it small-or-large! You will have opportunities to learn and share your knowledge with like-minded people.

You can take courses, book equipment, share ideas with the International as well as local Fab Lab Community.

Come and visit us now!

Copyright © FabLab@SP-2021



<header>

<nav>

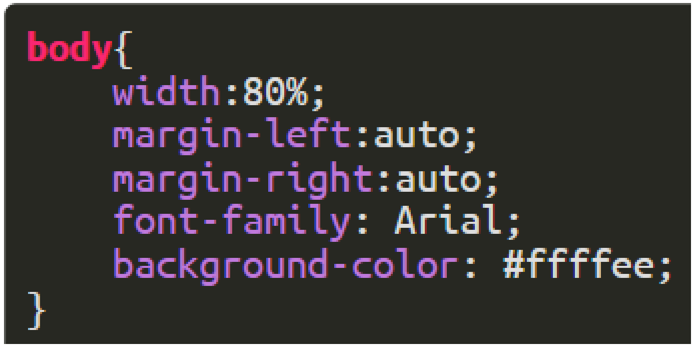
<div>

<div> <div>

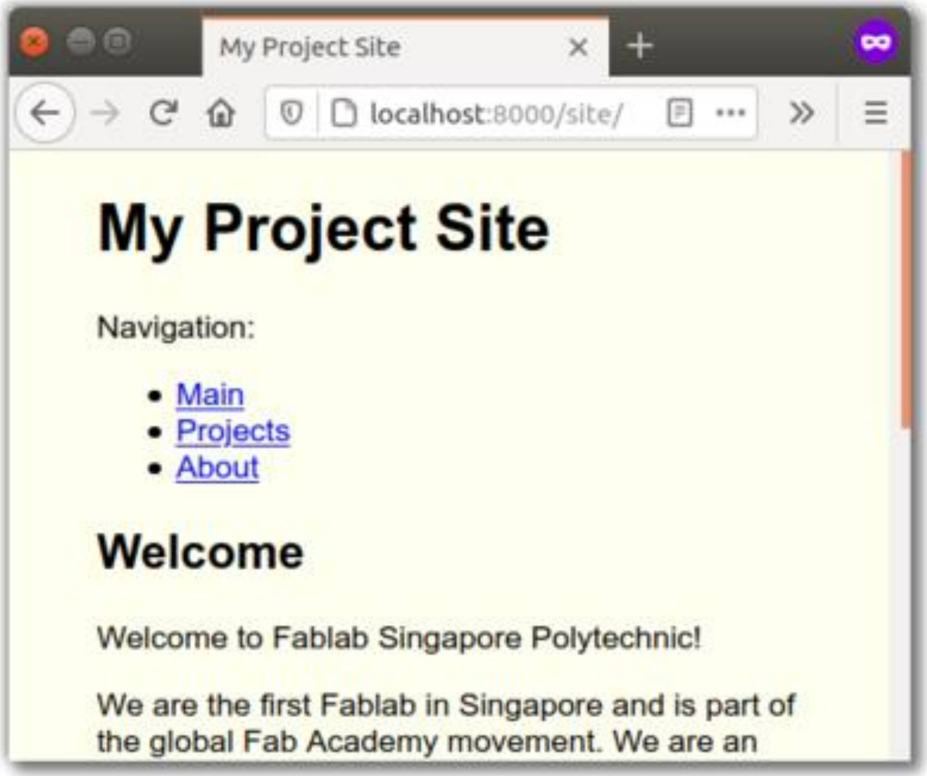
<footer>

Centering the page

- Before we start let's centre the entire page
    - have the content cover only 80% of the window
    - centre the contents
    - change all fonts to Arial
    - change the background color



```
body{
    width:80%;
    margin-left:auto;
    margin-right:auto;
    font-family: Arial;
    background-color: #ffffee;
}
```



My Project Site

Navigation:

- Main
- Projects
- About

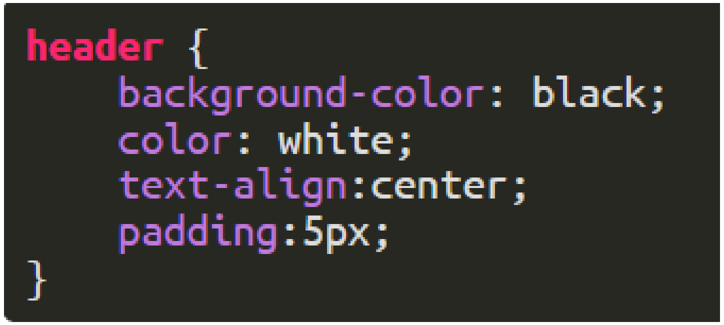
Welcome

Welcome to Fablab Singapore Polytechnic!

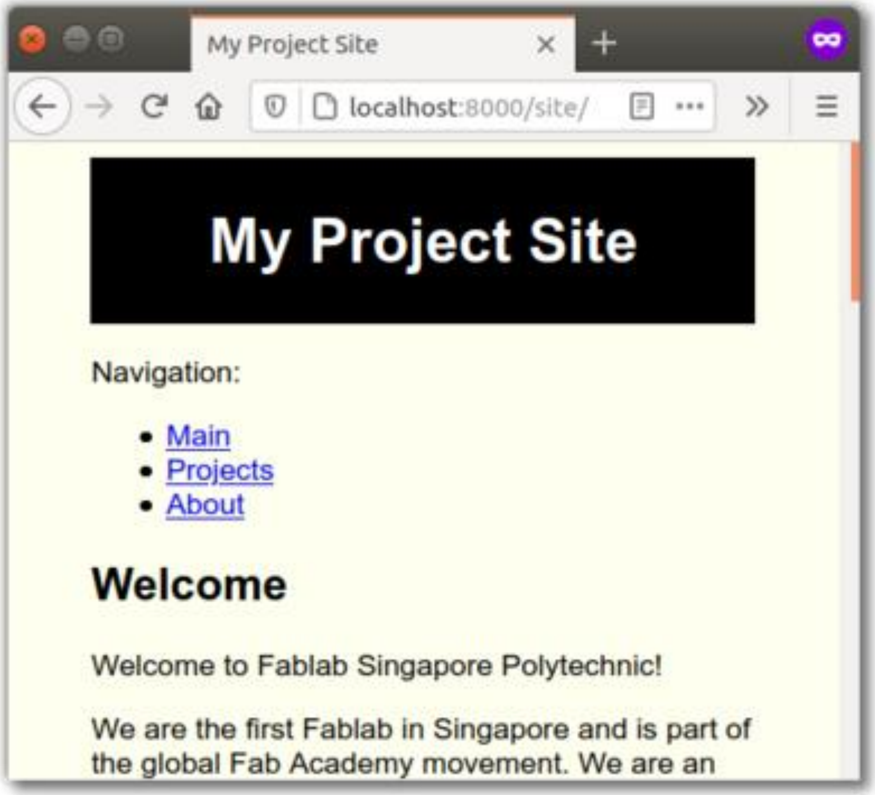
We are the first Fablab in Singapore and is part of the global Fab Academy movement. We are an

Changing the Header title

- Let's make the header a bit more prominent
    - inverse the colours
    - centre the heading text
    - add some spacing



```
header {
    background-color: black;
    color: white;
    text-align:center;
    padding:5px;
}
```



My Project Site

Navigation:

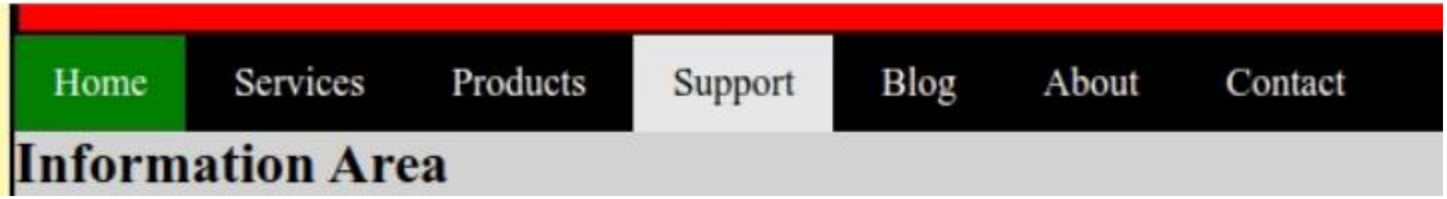
- Main
- Projects
- About

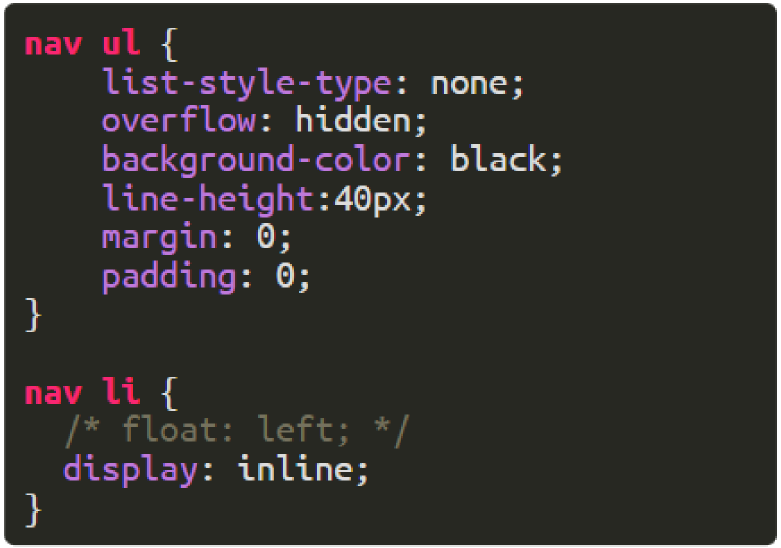
Welcome

Welcome to Fablab Singapore Polytechnic!

We are the first Fablab in Singapore and is part of the global Fab Academy movement. We are an

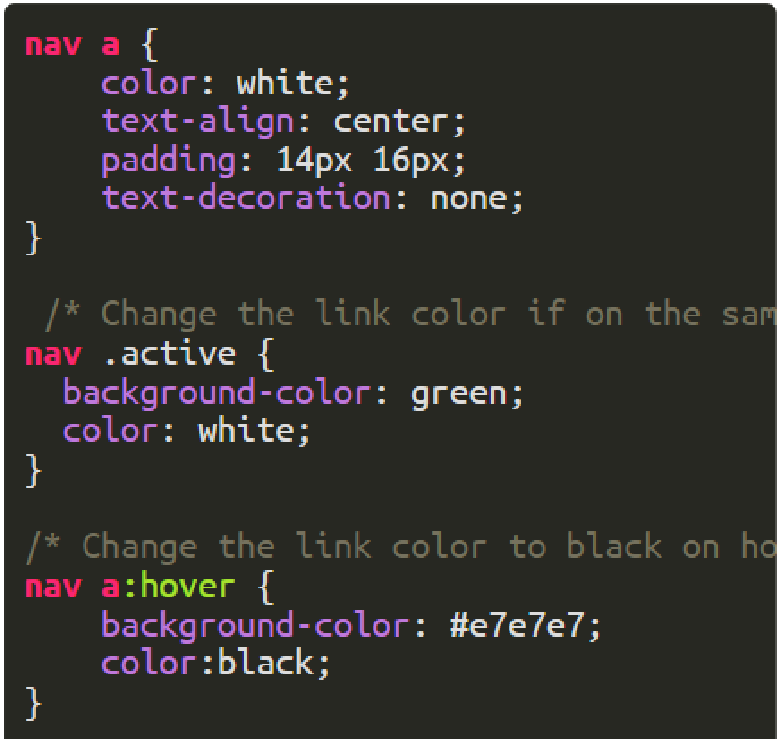
Navigation Bar





```
nav ul {
    list-style-type: none;
    overflow: hidden;
    background-color: black;
    line-height:40px;
    margin: 0;
    padding: 0;
}

nav li {
  /* float: left; */
  display: inline;
}
```

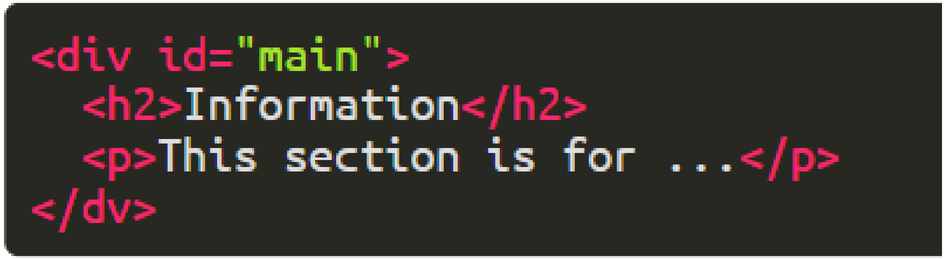
Examples from [W3 Schools](), [horizontal nav bar]()



```
nav a {
    color: white;
    text-align: center;
    padding: 14px 16px;
    text-decoration: none;
}

 /* Change the link color if on the sam
nav .active {
  background-color: green;
  color: white;
}

/* Change the link color to black on ho
nav a:hover {
    background-color: #e7e7e7;
    color:black;
}
```

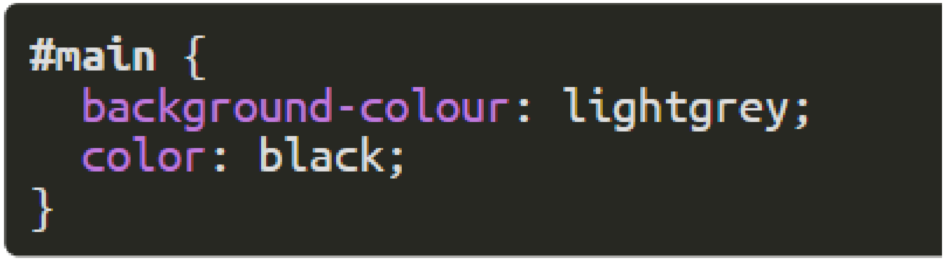
<div> sections

- use "id" or class" to designate them
- create CSS rules for each id, class



```
<div id="main">
  <h2>Information</h2>
  <p>This section is for ...</p>
</dv>
```

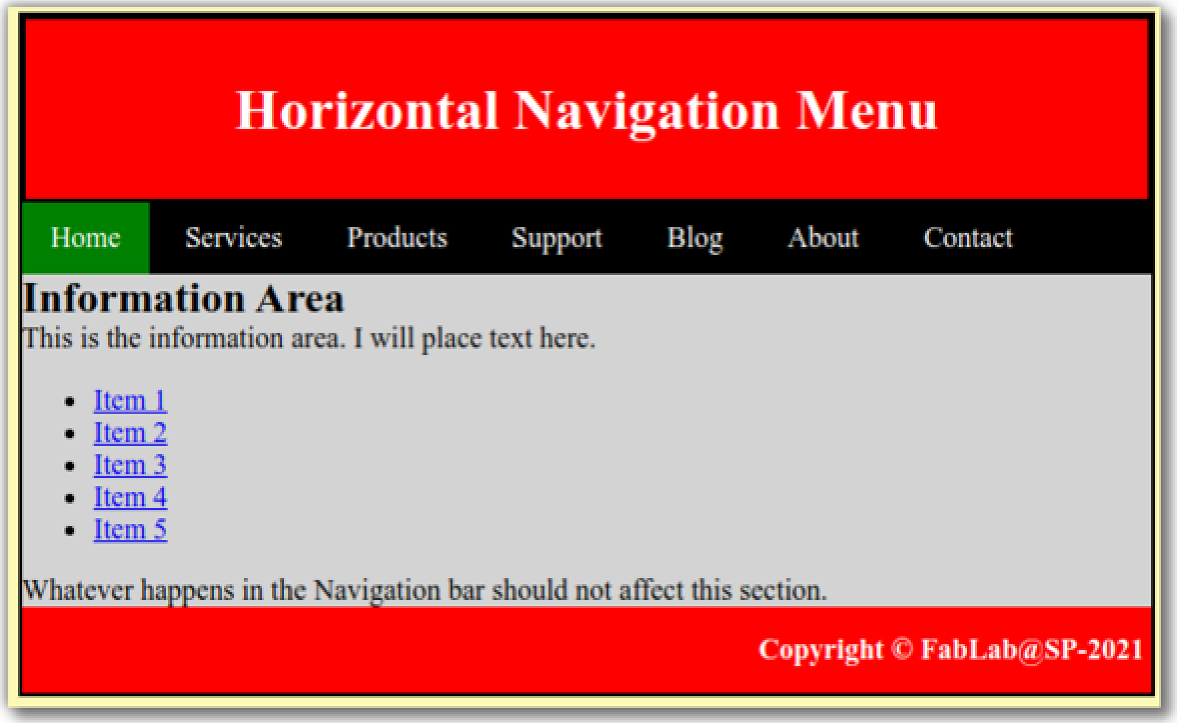
CSS Rules



```
#main {
  background-colour: lightgrey;
  color: black;
}
```

Tips

- It's easier to work with blocks, work on each block-at-a-time.
- Use colours, gradient, borders to designate areas
- Get/Use ideas from other templates
- Use Firefox/Chrome Developer Tools to troubleshoot
  - Firefox F12 key [14 Firefox Developer Tools]
  - Chrome F12 key [Tutorial for Beginners]



Horizontal Navigation Menu

Information Area

This is the information area. I will place text here.

- Item 1
- Item 2
- Item 3
- Item 4
- Item 5

Whatever happens in the Navigation bar should not affect this section.

Copyright © FabLab@SP-2021

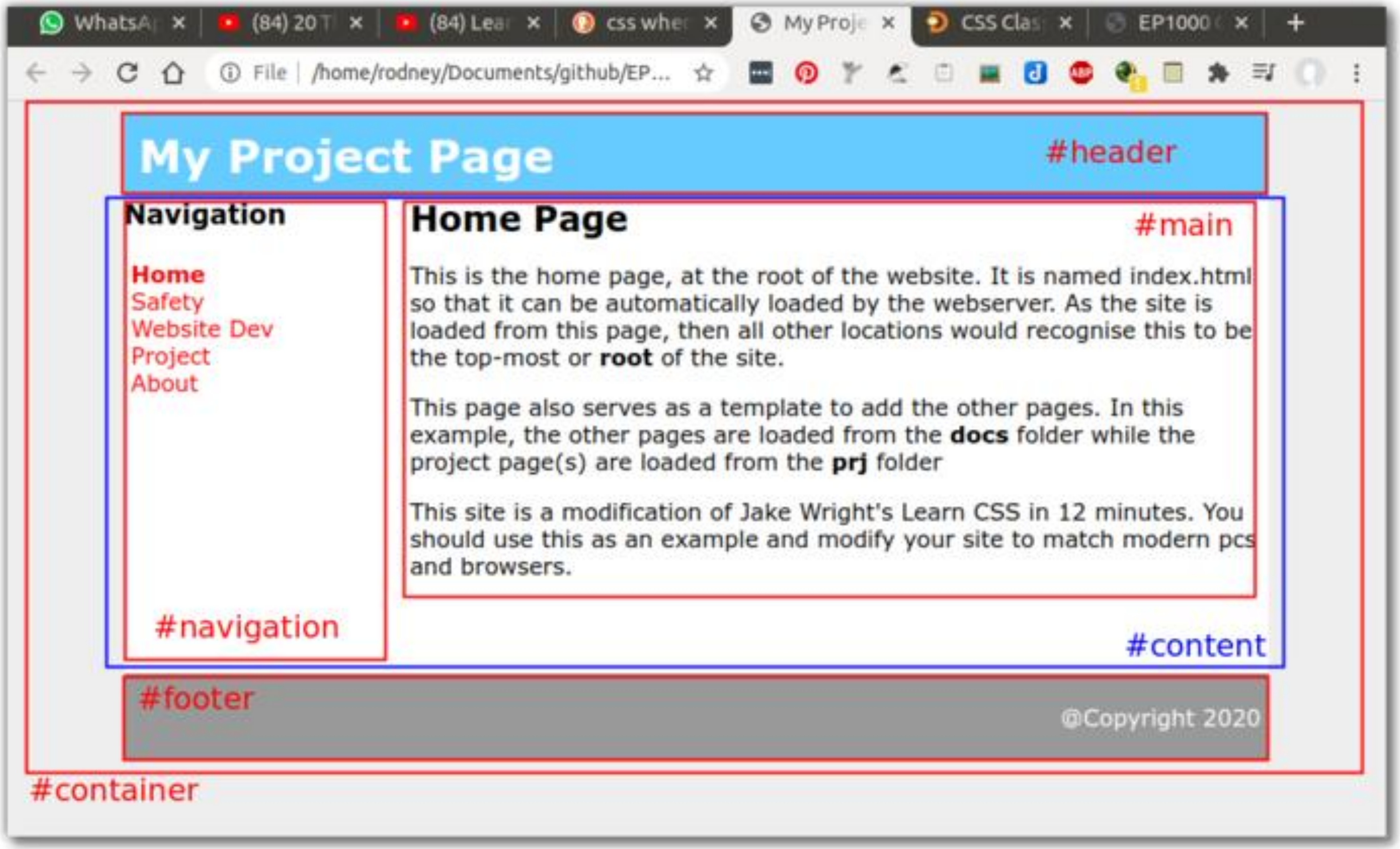
Tutorial

- Jake Wright – Learn CSS in 12 minutes
  - shows the creation of a basic site content
  - shows how an effective layout can be planned
  - shows creation of a simple Project Site

- Task:
  1. watch JW's video in its entirety
  2. use paper and pencil, follow development of structure/layout
  3. create the content of the site
  4. create the layout of the site
  5. Think:
     - how can you add your About page to the site
     - how can you add Project pages to the site

JW Layout

Jake Wright uses id (e.g. #container, #navigation) to define his rules for his page.



My Project Page

Navigation

Home
Safety
Website Dev
Project
About

Home Page

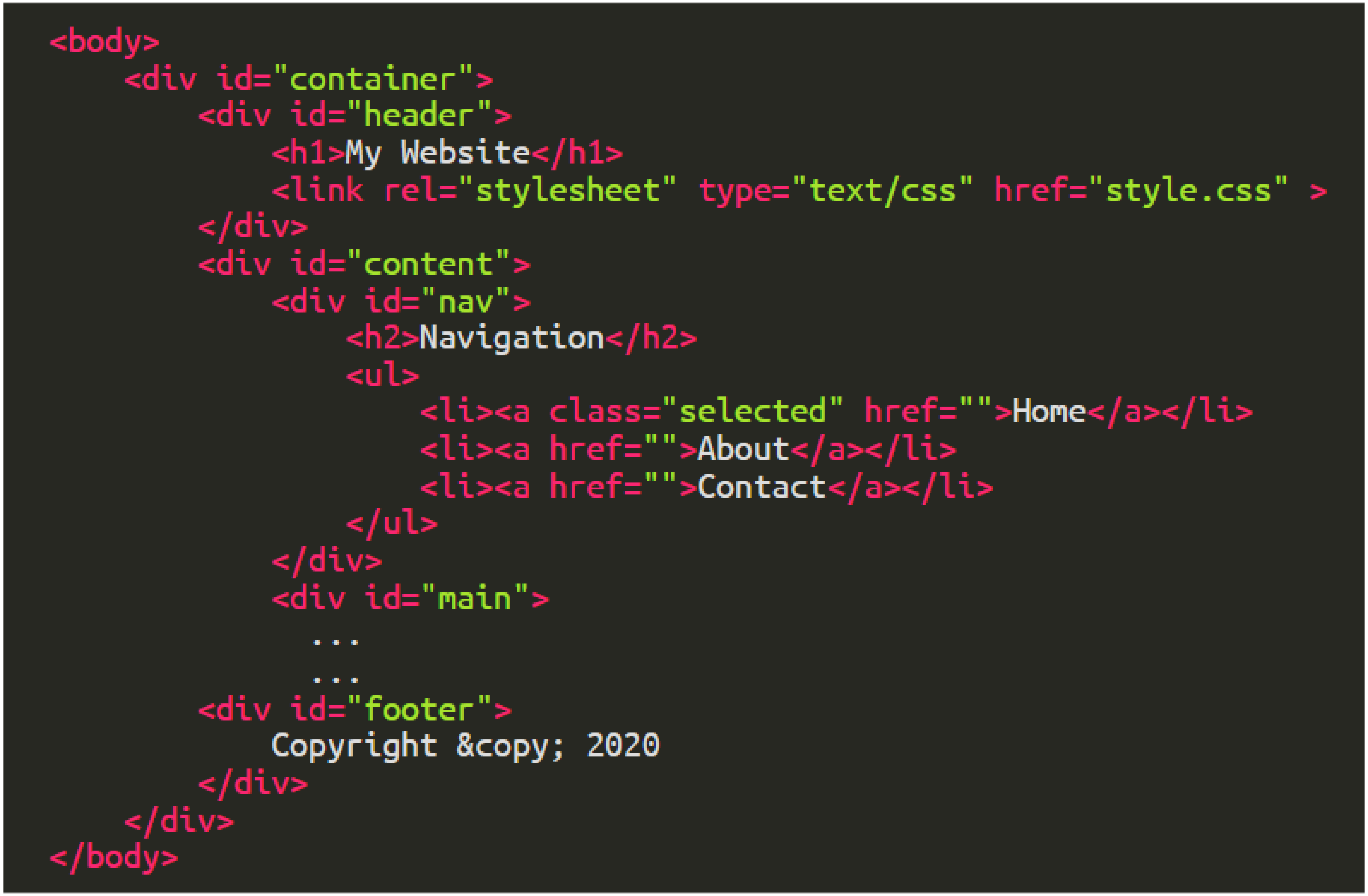
This is the home page, at the root of the website. It is named index.html so that it can be automatically loaded by the webserver. As the site is loaded from this page, then all other locations would recognise this to be the top-most or **root** of the site.

This page also serves as a template to add the other pages. In this example, the other pages are loaded from the **docs** folder while the project page(s) are loaded from the **prj** folder

This site is a modification of Jake Wright's Learn CSS in 12 minutes. You should use this as an example and modify your site to match modern pcs and browsers.

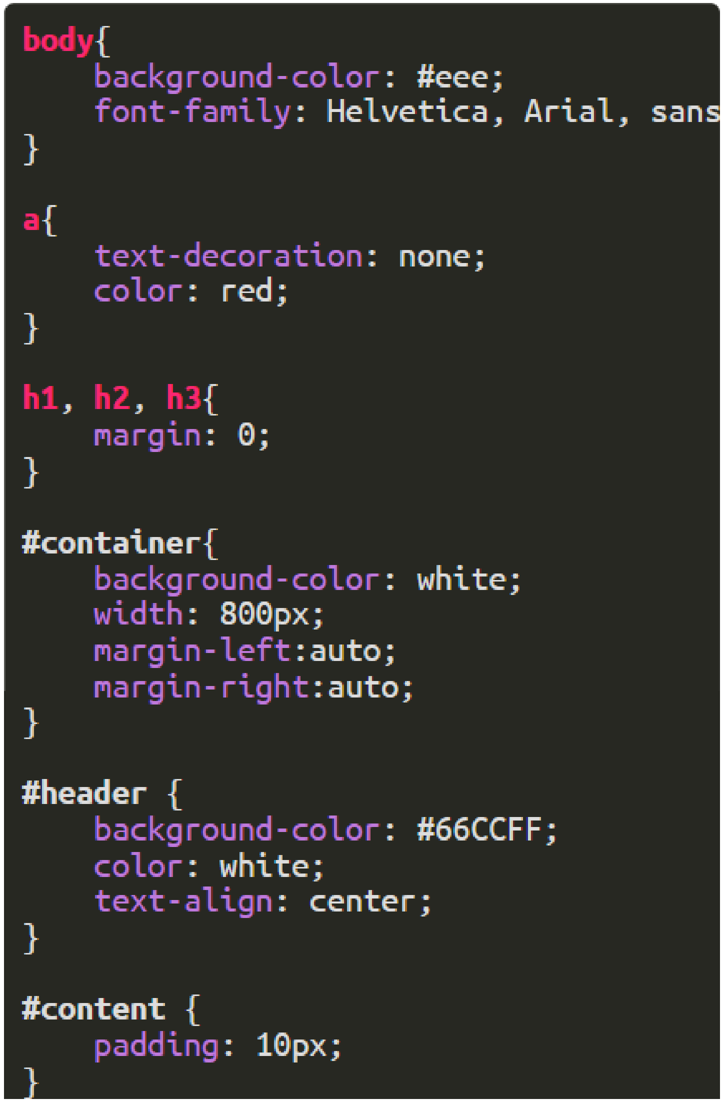
@Copyright 2020

JW Content

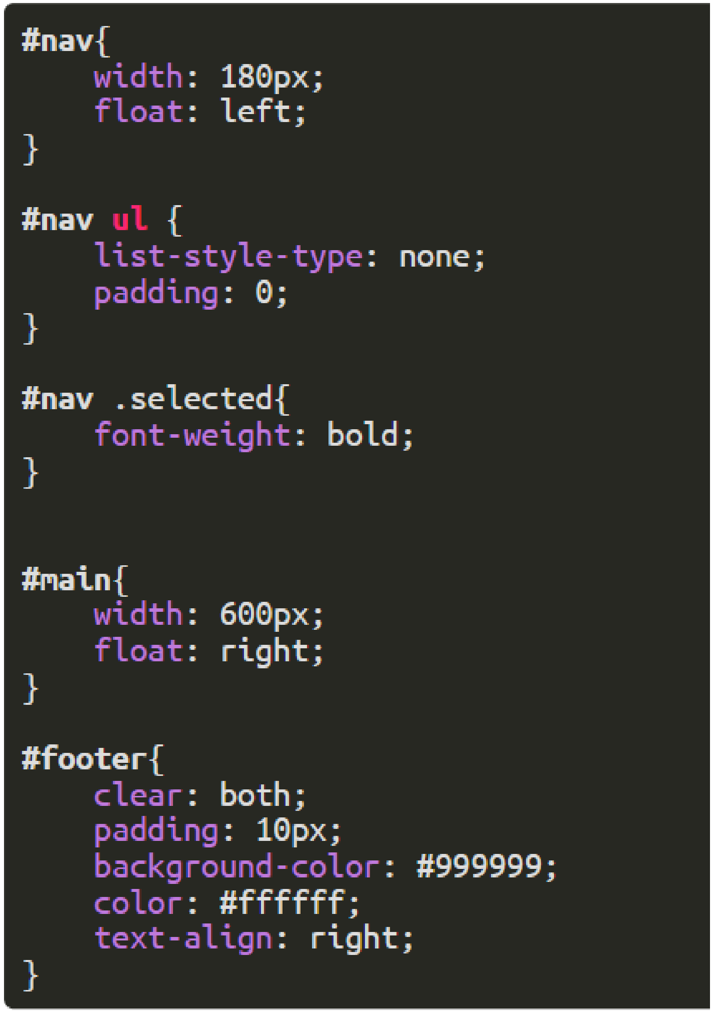


```
<body>
    <div id="container">
        <div id="header">
            <h1>My Website</h1>
            <link rel="stylesheet" type="text/css" href="style.css" >
        </div>
        <div id="content">
            <div id="nav">
                <h2>Navigation</h2>
                <ul>
                    <li><a class="selected" href="">Home</a></li>
                    <li><a href="">About</a></li>
                    <li><a href="">Contact</a></li>
                </ul>
            </div>
            <div id="main">
              ...
              ...
        <div id="footer">
            Copyright &copy; 2020
        </div>
    </div>
</body>
```

JW Layout



```
body{
    background-color: #eee;
    font-family: Helvetica, Arial, sans
}

a{
    text-decoration: none;
    color: red;
}

h1, h2, h3{
    margin: 0;
}

#container{
    background-color: white;
    width: 800px;
    margin-left:auto;
    margin-right:auto;
}

#header {
    background-color: #66CCFF;
    color: white;
    text-align: center;
}

#content {
    padding: 10px;
}
```



```
#nav{
    width: 180px;
    float: left;
}

#nav ul {
    list-style-type: none;
    padding: 0;
}

#nav .selected{
    font-weight: bold;
}


#main{
    width: 600px;
    float: right;
}

#footer{
    clear: both;
    padding: 10px;
    background-color: #999999;
    color: #ffffff;
    text-align: right;
}
```

Marked Assignment (CA1)

Task

1. Create a Project Documentation Site using HTML and CSS
2. Your site must have
    1. a Main page, explaining what this site is all about (Also known as the Index page)
    2. an About page, introducing yourself. You must include some biodata, a selfie and means of contact. You should also write one or two paragraphs about yourself.
    3. a Project(s) page (eventually you will need to have more project pages to showcase all your exercises and assignemnt)
    4. a Final Project page.
    5. navigation to move from one page to the other
3. Project page(s)
    1. links to projects/assignments that you have created
    2. link to your final project
    3. an initial page listing the tools that you have used/are using to create your website and the reasons to use them.
    4. include the table we discussed in class.

Marked Assignment (CA1) cont'd

Must-haves

1. You may use the JW Site (HTML & CSS) as a base
   - Modify to suit modern browsers of min. width 1024 pixels.
   - Modify the colours and font to your liking
   - Have a project page showing what you have changed.
2. Your site needs to be self-contained within a folder, but have a structure for
   - Organisation of images (i.e Images Folder, Video Folders etc.)
   - Projects (this must be a separate section)
   - On one of the project pages, show whether this is a viable method for documenting your project
     - Why, why-not
     - A simple how-to to add pages/projects

- Deadline

3. 4th week of Semester (more to come)

Questions ?

Ask now or forever be silent…

EP1000

CSS

End