
Tutorial 13

Outline

1. Assignment 2 check-in

- a. Questions, concerns, more examples

2. Assignment 2.1 overview

3. Review: CSS Grid Layout system

- a. grid-template areas
- b. grid-template rows
- c. grid-template columns

4. Putting together all the parts of a website

- a. Create and link different pages to create a website
 - i. Directory structure

5. Individual activities

- a. Create and link different pages to create a website

Assignment 2 check-in

Questions, concerns, more examples...

New example @ A2L > MMEDIA2A06 > Assignments > Assignment 2 > sample.png

Assignment 2.1 Overview

Find the guidelines @ A2L > MMEDIA2A06 > Assignments > Assignment 2.1

Website layouts

A layout “is the placement or arrangement of visual elements — how they are organized on a page.” (Duckett, 2011, p. 387).

Layout examples:

<https://soundsoftheuniverse.com/>

<https://www.sfmoma.org/>

<https://www.poulos.co/>



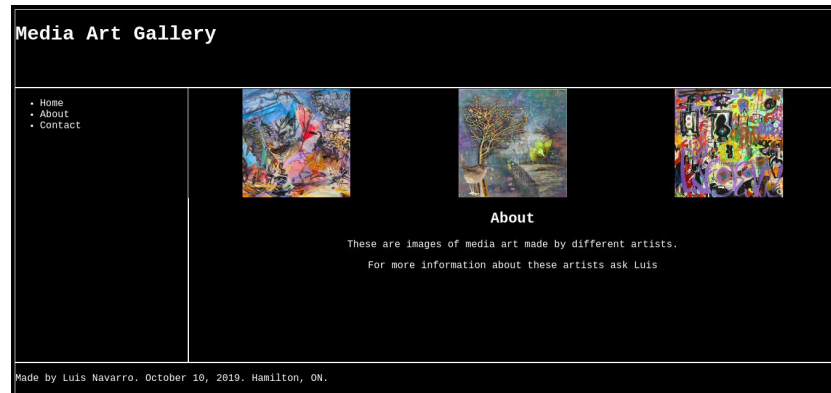
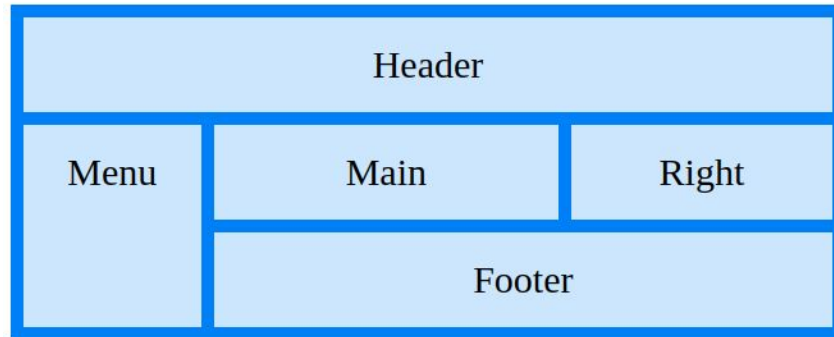
CSS layout: HTML `<div></div>` elements

1. To create **CSS layouts** you have to imagine every HTML **element** is inside a **box (also known as `<div>` element)**.

Ex.

```
<div class="header">  
  <h1> Media Art Gallery </h1>  
</div>
```

This grid layout contains six columns and three rows:



CSS layout properties

2. Use the CSS layout properties to arrange the div boxes according to your design or composition. Like the following CSS code:

```
.item1 {  
    grid-area: header;  
}  
.item2 {  
    grid-area: menu;  
}
```

The `grid-area` assigns a name to a grid item. Named grid items can then be referenced to by the `grid-template-areas` property of the grid container.

```
grid-container {  
    display: grid;  
    grid-template-rows: 10vh 80vh 10vh  
    grid-template-areas:  
        'header header '  
        'menu content'  
        'menu footer';  
}
```

The `display` property specifies the display behavior (the type of rendering box) of an element.

The `grid-template-areas` property specifies areas within the grid layout.

The `grid-template-rows` property specifies the number (and the heights) of the rows in a grid layout.

The `grid-template-columns` property specifies the number (and the widths) of columns in a grid layout.

Putting together all the parts of a website

1. **Create** different pages of a website by either:

- a. Making a new webpage with the sections you need.

... Or

- b. Duplicating a webpage you have already created. And then:
 - i. Erasing the sections you don't need and keep the sections you do need.
 - ii. Creating new sections.

Putting together all the parts of a website

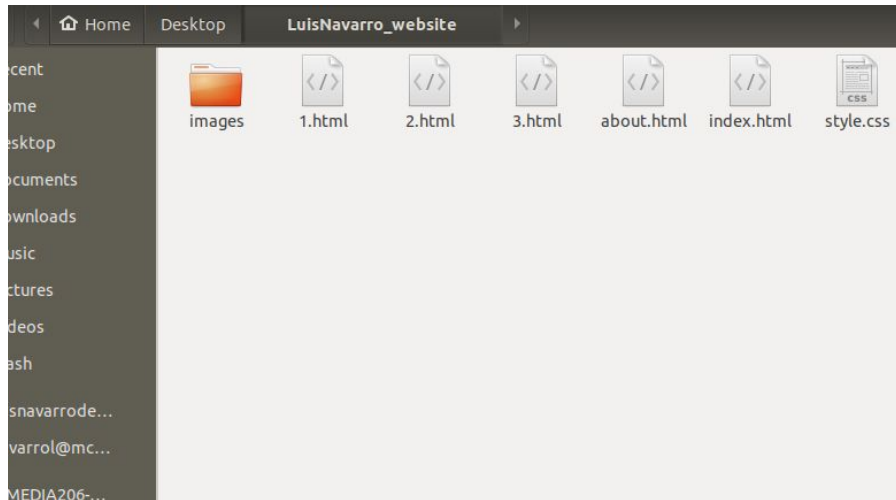
2. **Link** the different pages of a website:

- a. Transform the **menu items** of your website into **links** with the `<a>` element. Like the following example:

```
<a href="biography.html"> biography <a>
```

Dealing with your website files

- “When you are working on a website locally on your computer, you should **keep all the related files in a single folder.**”
- “**This folder can live** anywhere you like, but you should put it somewhere where you can easily find it, **maybe on your Desktop, in your Home folder, or at the root of your hard drive.**”
- “The most common things we'll have on any website project we create are an **index HTML file** and **folders to contain images, style files, and script files.**”



(Source

https://developer.mozilla.org/en-US/docs/Learn/Getting_started_with_the_web/Dealing_with_files)

Activity: Create contact page and link different webpages

General Instructions:

- 1) Follow the steps in the following slides to create a contact page.**
- 2) Follow the steps in the following slides to link the home, biographies and contact pages.**
- 3) Submit your final webpage to A2L.**

Part 1: Create a contact page of a website:

- Objectives:
 - Familiarize with the processes of creating additional pages for a website

1. **Download** the "ContactPage-activity" folder from A2L -> tutorial 12.
2. **Create** a copy of the "index.html" file inside of the "ContactPage-activity" folder. To do this do the following:
 - + Right-click the index.html files, and select "Copy" from the popup menu.
 - + Right-click inside the "ContactPage-activity" folder, and select "Paste items" from the popup menu..
3. **Rename** the copy of the "index.html" file with the name "contact.html". To do this do the following:
 - + Right-click on the copy of the "index.html" file and choose the "Rename" option from the popup menu.
 - + Type the new name "contact.html" and hit Enter

Part 2: Create a contact page of a website:

- Objectives:
 - Familiarize with the processes of creating additional pages for a website

1. **Setup** TextEdit's preferences. Do the following:
 - + In the TextEdit app on your Mac choose TextEdit > Preferences.
 - + Select "Plain text"
 - + On the same screen, uncheck any box related to spelling correction.
 - + Now, click "Open and Save", and select "Display HTML files as HTML code instead of formatted text."
2. **Open** the "contact.html" file in TextEdit (at this point TextEdit should only display the code, if it isn't, go back to step #1).
3. **Delete** the code you won't need for the contact page and keep the code you would need.
 - + For example, keep the all the code referring to the header, the menu, and the footer, and erase the code referring to the "home-content".

Part 3: Create a contact page of a website:

- Objective:
 - Familiarize with the process of creating additional pages for a website

1. **Make** a new 'div' box with a class named "contact". Follow the next example:

```
+ <div class = "contact"> </div>
```

2. **Create** a text with your name and email address inside the div box. Follow the next example:

```
+ <div class = "contact">
```

```
<p> David Smith , smith@mcmaster.ca </p>
```

```
< /div>
```

3. **Save** your changes in TextEdit.
4. **Visualize** your changes by opening the "contact.html" in the browser.

Part 4: link the home, biographies and contact pages.

- Objective:
 - Familiarize with linking multiple webpages

1. **Locate** the menu section inside the "contact.html" file in TextEdit.
2. **Create** a link for the menu items to their respective HTML documents. Follow the example below:

```
<ul >
```

```
<a href= "index.html"> <li> Home </li> </a>
```

```
<a href= "biographies.html"> <li> Bio </li> </a>
```

```
<a href= "contact.html"> <li> Contact </li> </a>
```

```
</ul >
```

3. **Visualize** your changes by opening the "contact.html" in the browser.

Part 5: link the home, biographies and contact pages.

- Objective:
 - - Familiarize with linking multiple webpages

1. **Open** the "index.html" file and the "biography.htm" file in TextEdit and **create** links for the menu items of these pages, similarly to how you did in part 4.
2. **Save** your changes in TextEdit and visualize them in the browser.
3. Submit your work to A2L

Additional resources

Link different webpages:

<https://openclassrooms.com/en/courses/2479876-build-your-website-with-html5-and-css3/2490386-creating-links>