

Web Engineering & Development (SWE 363)

2.2 More HTML (Hypertext Markup Language)

Created by: Dr. Omar Hammad

Lectured by: Dr. Khadijah Al Safwan

Announcements

- We allow teams of 4 but with higher expectations
- Inform us about your Github account: <https://forms.office.com/r/HuvttUTp0A>
- Graders assigned and will be looking into your code

GitHub Classroom's Repository Push Issue:

Set Your Git Username and Email to Match GitHub

```
git config --add --global user.name "Your GitHub Name"  
git config --add --global user.email "your_email@example.com"
```

Why This Matters?

GitHub links commits to your account using the email in your Git config. If the email doesn't match any email in your GitHub account, your commits won't be linked to your profile

In today's Lecture:

- Links
- Containers
- Forms
- Form Widgets

Reference:

- Zybook: 2.6 to 2.7

Links

Connect different pages together

a simple Link

```
<a href="https://google.com">Link</a>
```

A link with an image

```
<a href="https://google.com">  
  
</a>
```

Board activity:

How can we link to a page located in the same directory?

Containers

grouping elements together

Common Container tags

- `<header>` : Header
- `<main>` : Main Content
- `<section>` : Section
- `<article>` : Article
- `<footer>` : Footer
- `<nav>` : Navigation
- `<div>` : Division
- `` : Span

Messy Code Example

```
<html>
  <head>
    <title>My Page</title>
  </head>
  <body>
    <h1>Welcome</h1>
    <p>This is a paragraph with no spacing or proper indentation making it very hard to read and maintain</p>
    <div>
      <ul>
        <li>Item 1</li>
        <li>Item 2</li>
        <li>Item 3</li>
      </ul>
    </div>
    <a href="link.html">Click here</a>
    <script>
      function doSomething(){var x=1;var y=2;var result=x+y;return result;}
    </script>
  </body>
</html>
```

Board Challenge:

How would you structure this page using containers?

Clean Code Example

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>My Page</title>
</head>
<body>
  <header>
    <h1>Welcome</h1>
  </header>
  <main>
    <p>This is a paragraph with proper spacing and indentation, making it easy to read and maintain.</p>
    <section class="content">
      <ul>
        <li>Item 1</li>
        <li>Item 2</li>
        <li>Item 3</li>
      </ul>
    </section>
    <a href="link.html">Click here</a>
    
  </main>
</body>
</html>
```

Block vs Inline Elements

- **Block**: Takes up the full width of the container: `<div>`, `<section>`, `<article>`, `<header>`, `<footer>`, `<p>`, `<h1>`, `<h2>`, `<h3>`, `<h4>`, `<h5>`, `<h6>`
- **Inline**: Takes up only the width of the content: ``, `<a>`, ``, ``, ``

Forms

Make website interactive

A simple form

```
<form action="process_form.php" method="post">  
  <label for="name">Name:</label>  
  <input type="text" id="name" name="name">  
  <input type="submit" value="Submit">  
</form>
```

action: The URL where the form data will be sent

method: The HTTP method to use (POST or GET)

input: The input field where the user will enter the data

submit: The button that will submit the form

Get vs Post

- Get: Sends the data in the URL
- Post: Sends the data in the body of the request

Data look like:

Client Request

```
GET /process_form.php?name=John
```

```
POST /process_form.php
```

Data (key-value pairs)

```
name=John
```

Common Form Widgets (Elements)

- `<input>` : Input (text, password, email, number, etc.)
- `<select>` : Select (dropdown menu)
- `<textarea>` : Textarea (large text field)
- `<button>` : Button (submit, reset, etc.)
- `<checkbox>` : Checkbox (multiple choice)
- `<radio>` : Radio (single choice)
- `<submit>` : Submit (submit the form)
- `<reset>` : Reset (reset the form)

30m

Let's build a social media app

Starter code at: github.com/Web-Engineering-KFUPM/2.2_social_media_app

Next Class

- Basics of CSS