

# Adventures in HTML, CSS, and Creativity

Instructor: Adelina Ajdini



# Recap of the last class

- ❖ Adding in text formatting (underline, italic, bold etc).
- ❖ Adding in strong text
- ❖ Adding links
- ❖ Adding an image
- ❖ Editing the height and the width of an image
- ❖ Hyperlinks in text
- ❖ Styling with CSS classes

# Session 4: More HTML customization

Comments, media and HTML lists

- ❖ Comments in HTML
- ❖ Comments in CSS
- ❖ Styling link text
- ❖ Adding the alt property to an image tag
- ❖ Adding a clickable link to an image
- ❖ Media (videos and files)
- ❖ HTML lists

# Comments into our code

Comments → form of documentation, explaining the purpose, functionality, or intent of a particular section of code.

Valuable → for developers who might read or work on the code in the future, including yourself.

In HTML, comments are enclosed between `<!--` and `-->`.

In CSS, comments are enclosed between `/*` and `*/`.

# Comments into our code

## HTML sample

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>My Web Page</title>
    <!-- This is an HTML comment in the head section --> ←
    <link rel="stylesheet" href="styles.css">
</head>
<body>
    <h1>Hello, World!</h1>
    <!-- This is an HTML comment in the body section --> ↘
    <p>This is a paragraph.</p>
</body>
</html>
```

# Comments into our code

CSS sample

```
h1 {  
    color: blue; /* This is a comment for the color property */  
}
```



# Styling hyperlink text

```
a {  
    text-decoration: none; /* Remove underlines by default */  
    color: #0077cc; /* Set the default color */  
    font-weight: bold; /* Make the text bold */  
}
```

# Adding the alt property to an image tag

The alt attribute in an HTML **img** tag is used to provide alternative text for the image. This text is displayed **if the image cannot be loaded**, and it also serves as a description for users who rely on screen readers. Here's an example of how you can add the alt attribute to an image tag:

```
<!-- Image with alt attribute -->  

```

```
<!-- Another example with a more functional description -->  

```

# Adding the alt property to an image tag

You can make an image clickable by wrapping it with an anchor () tag.

Example:

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

# Adding the alt property to an image tag

1. **The `<a>` tag** → used to create a hyperlink.
2. **The `href` attribute within the `<a>` tag** → specifies the destination URL when the image is clicked.
3. **The `<img>` tag inside the `<a>` tag** → represents the clickable image.
4. **The `alt` attribute** → provides alternative text for the image.

# Making images clickable

You can make an image clickable by wrapping it with an anchor () tag.

Example:

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

# Media (videos and files)

To include videos and files in HTML:

1. With **<video>** element for embedding videos and
2. the **<a>** (anchor) element for linking to files.

Here are examples for both scenarios >>

# VIDEOS in HTML

```
<video width="640" height="360" controls>  
    <source src="example-video.mp4" type="video/mp4">  
</video>
```

- ❖ The **<video>** element is used to embed a video.
- ❖ The **width** and **height** attributes define the dimensions of the video player.
- ❖ The **controls** attribute adds playback controls (play, pause, etc.)
- ❖ The **<source>** element provides the video file (example-video.mp4)
- ❖ **type** attribute specifies the MIME type.

# FILES in HTML

<!-- Link to a PDF file -->

```
<a href="example-document.pdf" download> Download PDF</a>
```

<!-- Link to a ZIP file -->

```
<a href="example-archive.zip" download> Download ZIP</a>
```

- ❖ The **<a>** (anchor) element is used to create links.
- ❖ The **href** attribute specifies the path to the file.
- ❖ The **download** attribute, when present, suggests that the browser should download the linked file rather than navigating to it. (If we do not add this, the browser will open the file once we click on it)

# HTML Lists

In HTML, there are **three** main types of lists:

1. unordered lists **<ul>**
2. ordered lists **<ol>**
3. definition lists **<dl>**

# Unordered lists <ul>

An unordered list is a list of items where the order of the items doesn't matter. The items are typically marked with bullet points.

```
<ul>  
  <li>Item 1</li>  
  <li>Item 2</li>  
  <li>Item 3</li>  
</ul>
```

# Ordered Lists <ol>

An ordered list is a list of items where the order of the items is important. The items are numbered.

```
<ol>
  <li>First item</li>
  <li>Second item</li>
  <li>Third item</li>
</ol>
```

# Definition Lists <dl>

- ❖ A definition list consists of terms and their definitions. It uses `<dt>` for terms and `<dd>` for definitions.

```
<dl>
```

```
  <dt>Term 1</dt>
```

```
  <dd>Definition 1</dd>
```

```
  <dt>Term 2</dt>
```

```
  <dd>Definition 2</dd>
```

```
</dl>
```

# Nested Lists

- ❖ Lists can also be nested inside one another, combining both ordered and unordered lists.



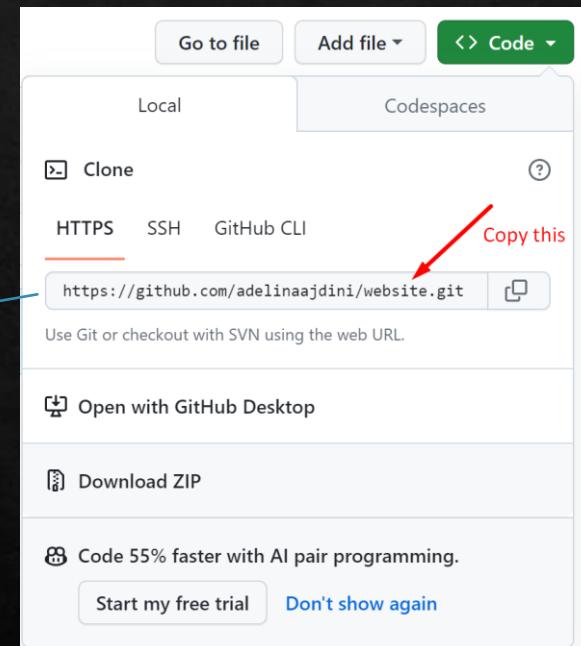
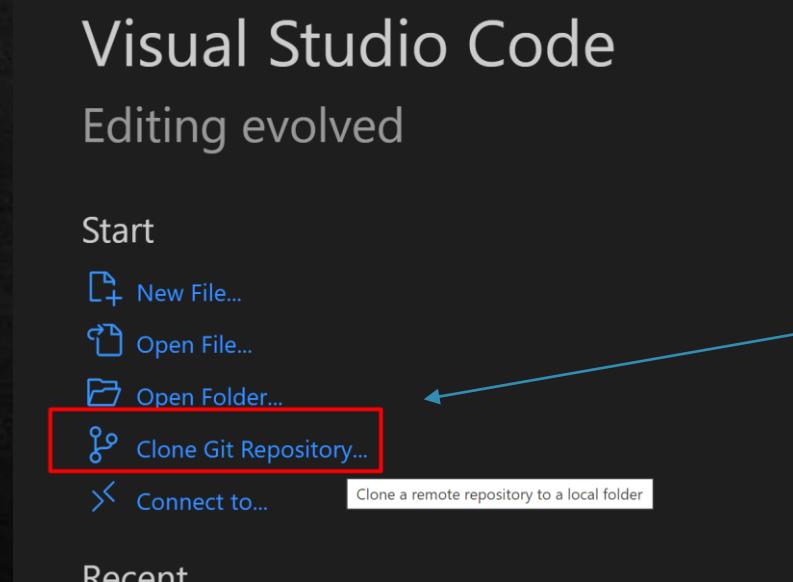
```
<ul>
  <li>Item 1</li>
  <li>Item 2
    <ul>
      <li>Subitem 2.1</li>
      <li>Subitem 2.2</li>
    </ul>
  </li>
  <li>Item 3</li>
</ul>
```

Additionally, lists are often used for navigation menus, content organization, and presenting information in a structured manner on web pages.

# Clone Project from Git

1. Create a new GIT folder
2. Upload Website – folder (with the “website.html” and “style.css”)
3. Save it and make sure we can view it into the browser
4. After this, open Visual Studio Code
5. Clone the project
6. Save

# Clone Project from Git



Thank you for your  
attention



QUESTIONS?

# Homework 4

Copy the “Website” folder THAT we created on class with the files inside.

Clone and rename the folder “**Homework3**”

**Inside the HTML file “index.html” add 3 lists:**

1. Ordered: With your favourite movies
2. Unordered: Your favourite books
3. Definition List: Explaining by definition:

HTML:

CSS:

(Hint: Use those 2 titles for the **<dt> </dt>** tags)

```
<dl>
  <dt>HTML</dt>
  <dd>Definition 1</dd>

  <dt>CSS</dt>
  <dd>Definition 2</dd>

</dl>
```