

## Session 2

### HTML Images:

**Images** can improve the design and the appearance of a web page.

The HTML `<img>` tag is used to embed an image in a web page.

**Images** are not technically inserted into a web page; images are linked to web pages.

The `<img>` tag creates a holding space for the referenced image.

The `<img>` tag is empty, it contains attributes only, and does not have a closing tag.

The `<img>` tag has two required attributes:

- **src** - Specifies the path to the image
- **alt** - Specifies an alternate text for the image

The images can be in internet or in our computer. If images in internet we will copy the image link address, if images in our computer we will use this code:

### Example of images:

```







```

## ***Forms***

***An HTML form*** is used to collect user input the user input is most often sent to a server for processing.

### ***Example:***

First name:

Last name:

### ***The <form> Element:***

The HTML <form> element is used to create an HTML form for user input:

<form>

.

*form elements*

.

</form>

The <form> element is a container for different types of input elements, such as: text fields, checkboxes, radio buttons, submit buttons, etc.

**<form>** element can contain one or more of the following form elements:

- **<input>**
- **<label>**
- **<select>**
- **<textarea>**
- **<button>**
- **<fieldset>**
- **<legend>**
- **<datalist>**
- **<output>**
- **<option>**
- **<optgroup>**

**Example:**

```
<label for="fname">First name:</label>  
<input type="text" id="fname" name="fname">
```

## HTML Input Types

Here are the different input types you can use in HTML:

- **<input type="button">**
- **<input type="checkbox">**
- **<input type="color">**
- **<input type="date">**
- **<input type="datetime-local">**
- **<input type="email">**
- **<input type="file">**
- **<input type="hidden">**
- **<input type="image">**
- **<input type="month">**
- **<input type="number">**
- **<input type="password">**
- **<input type="radio">**

- `<input type="range">`
- `<input type="reset">`
- `<input type="search">`
- `<input type="submit">`
- `<input type="tel">`
- `<input type="text">`
- `<input type="time">`
- `<input type="url">`
- `<input type="week">`

*The default value of the type attribute is "text".*

### **Example**

```
<form>
  <label for="username">Username:</label> <br>
  <input type="text" id="username" name="username"> <br>
  <label for="pwd">Password: </label> <br>
  <input type="password" id="pwd" name="pwd">
</form>
```

This is how the HTML code above will be displayed in a browser:

Username:

Password:

## ***HTML Entities***

Reserved characters in HTML must be replaced with character entities

If you use the less than (<) or greater than (>) signs in your text, the browser might mix them with tags. Character entities are used to display reserved characters in HTML.

*Advantage of using an entity name:* An entity name is easy to remember.

*Disadvantage of using an entity name:* Browsers may not support all entity names, but the support for entity numbers is good.

***A character entity looks like this:***

*&entity\_name;*

OR

*&#entity\_number;*

To display a less than sign (<) we must write: **&lt;** or **&#60;**

## ***Non-breaking Space***

A commonly used entity in HTML is the non-breaking space: **&nbsp;**;

A non-breaking space is a space that will not break into a new line.

### ***Examples:***

- § 10
- 10 km/h
- 10 PM

Another common use of the non-breaking space is to prevent browsers from truncating spaces in HTML pages.

If you write 10 spaces in your text, the browser will remove 9 of them. To add real spaces to your text, you can use the **&nbsp;** character entity.

## ***HTML semantic tags***

Semantic elements = elements with a meaning.

### ***What are Semantic Elements?***

A semantic element clearly describes its meaning to both the browser and the developer.

Examples of **non-semantic** elements: <div> and <span> - Tells nothing about its content.

Examples of **semantic** elements: <form>, <table>, and <article> - Clearly defines its content.

## ***Semantic Elements in HTML***

Many web sites contain HTML code like: `<div id="nav">` `<div class="header">` `<div id="footer">` to indicate navigation, header, and footer.

In HTML there are some semantic elements that can be used to define different parts of a web page

<code>&lt;article&gt;</code>	Defines independent, self-contained content
<code>&lt;aside&gt;</code>	Defines content aside from the page content
<code>&lt;details&gt;</code>	Defines additional details that the user can view or hide
<code>&lt;figcaption&gt;</code>	Defines a caption for a <code>&lt;figure&gt;</code> element
<code>&lt;figure&gt;</code>	Specifies self-contained content, like illustrations, diagrams, photos, code listings, etc.
<code>&lt;footer&gt;</code>	Defines a footer for a document or section
<code>&lt;header&gt;</code>	Specifies a header for a document or section
<code>&lt;main&gt;</code>	Specifies the main content of a document
<code>&lt;mark&gt;</code>	Defines marked/highlighted text
<code>&lt;nav&gt;</code>	Defines navigation links
<code>&lt;section&gt;</code>	Defines a section in a document
<code>&lt;summary&gt;</code>	Defines a visible heading for a <code>&lt;details&gt;</code> element
<code>&lt;time&gt;</code>	Defines a date/time

## ***Explanation of some important semantic tags***

### **1- HTML <header> Tag**

#### *Definition and Usage:*

The **<header>** element represents a container for introductory content or a set of navigational links.

A **<header>** element typically contains:

- one or more heading elements (<h1> - <h6>)
- logo or icon
- authorship information

#### **Note:**

You can have several **<header>** elements in one HTML document. However, **<header>** cannot be placed within a <footer>, <address> or another **<header>** element.

```
<header>
  <h1>A heading here</h1>
  <p>Posted by John Doe</p>
  <p>Some additional information here</p>
</header>
```

### **2- HTML <nav> Tag**

#### *Definition and Usage:*

The **<nav>** tag defines a set of navigation links.

Notice that NOT all links of a document should be inside a **<nav>** element.

The **<nav>** element is intended only for major block of navigation links.



```
<nav>
  <a href="/html/">HTML</a>
  <a href="/css/">CSS</a>
  <a href="/js/">JavaScript</a>
  <a href="/python/">Python</a>
</nav>
```

### 3- HTML `<article>` Tag

#### *Definition and Usage:*

The `<article>` tag specifies independent, self-contained content.

An article should make sense on its own and it should be possible to distribute it independently from the rest of the site.

Potential sources for the `<article>` element:

- Forum post
- Blog post
- News story

#### **Note:**

The `<article>` element does not render as anything special in a browser. However, you can use CSS to style the `<article>` element (see example below).

```
<article>
  <h2>Google Chrome</h2>
  <p>Google Chrome is a web browser developed by Google, released in 2008. </p>
</article>

<article>
  <h2>Mozilla Firefox</h2>
  <p>Mozilla Firefox is an open-source web browser developed by Mozilla. </p>
</article>
```

## 4- HTML <aside>Tag

### *Definition and Usage:*

The `<aside>` tag defines some content aside from the content it is placed in.

The aside content should be indirectly related to the surrounding content.

**Tip:** The `<aside>` content is often placed as a sidebar in a document.

### **Note:**

The `<aside>` element does not render as anything special in a browser. However, you can use CSS to style the `<aside>` element (see example below).

```
<aside>
  <h4>Epcot Center</h4>
  <p>Epcot is a theme park at Walt Disney World Resort featuring exciting
  attractions, international pavilions, award-winning fireworks and seasonal
  special events. </p>
</aside>
```

## 5- HTML <footer>Tag

### *Definition and Usage:*

The `<footer>` tag defines a footer for a document or section.

A `<footer>` element typically contains:

- authorship information
- copyright information
- contact information
- sitemap
- back to top links
- related documents

You can have several `<footer>` elements in one document.

**Tip:** Contact information inside a `<footer>` element should go inside an `<address>` tag.

```
<footer>
  <p>Author: Hege Refsnes</p>
  <p><a href="mailto:hege@example.com">hege@example.com</a></p>
</footer>
```

## 6- HTML `<main>` Tag

### *Definition and Usage:*

The `<main>` tag specifies the main content of a document.

The content inside the `<main>` element should be unique to the document. It should not contain any content that is repeated across documents such as sidebars, navigation links, copyright information, site logos, and search forms.

### **Note:**

There must not be more than one `<main>` element in a document.

The `<main>` element must NOT be a descendant of an `<article>`, `<aside>`, `<footer>`, `<header>`, or `<nav>` element.

```
<main>
  <h1>Most Popular Browsers</h1>
  <p>Chrome, Firefox, and Edge are the most used browsers today.</p>

  <article>
    <h2>Google Chrome</h2>
    <p>Google Chrome is a web browser developed by Google, released in 2008</p>
  </article>

  <article>
    <h2>Mozilla Firefox</h2>
    <p>Mozilla Firefox is an open-source web browser developed by Mozilla.</p>
  </article>
</main>
```

## 7- HTML <figure>Tag

The <figure> tag specifies self-contained content, like illustrations, diagrams, photos, code listings, etc.

While the content of the <figure> element is related to the main flow, its position is independent of the main flow, and if removed it should not affect the flow of the document.

**Tip:** The <figcaption> element is used to add a caption for the <figure> element.

```
<figure>
  
  <figcaption>Fig.1 - Trulli, Puglia, Italy.</figcaption>
</figure>
```

## 8- HTML <section>Tag

*Definition and Usage:*

The <section> element defines a section in a document.

According to W3C's HTML documentation: "A section is a thematic grouping of content, typically with a heading."

Examples of where a <section> element can be used:

- Chapters
- Introduction
- News items
- Contact information

### **Note:**

A web page could normally be split into sections for introduction, content, and contact information.

## Example:

```
<section>
  <h1>WWF</h1>
  <p>The World Wide Fund for Nature (WWF) is an international organization
working on issues regarding the conservation, research and restoration of the
environment, formerly named the World Wildlife Fund. WWF was founded in 1961.</p>
</section>

<section>
  <h1>WWF's Panda symbol</h1>
  <p>The Panda has become the symbol of WWF. The well-known panda logo of WWF
originated from a panda named Chi Chi that was transferred from the Beijing Zoo
to the London Zoo in the same year of the establishment of WWF.</p>
</section>
```

## ***Iframes***

An HTML iframe is used to display a web page within a web page.

*Iframe Syntax:*

An inline frame is used to embed another document within the current HTML document.

*Syntax:*

```
<iframe src="url" title="description"></iframe>
```

*Iframe - Set Height and Width*

Use the height and width attributes to specify the size of the iframe

***Example***

```
<iframe src="demo_iframe.htm" height="200" width="300" title="Iframe  
Example"></iframe>
```

*Iframe - Remove the Border*

By default, an iframe has a border around it.

To remove the border, add the style attribute and use the CSS border property:

***Example***

```
<iframe src="demo_iframe.htm" style="border:none;" title="Iframe  
Example"></iframe>
```

- *The HTML <iframe> tag specifies an inline frame*
- *The src attribute defines the URL of the page to embed*
- *Always include a title attribute (for screen readers)*
- *The height and width attributes specifies the size of the iframe*
- *Use border:none; to remove the border around the iframe.*

## ***HTML 5***

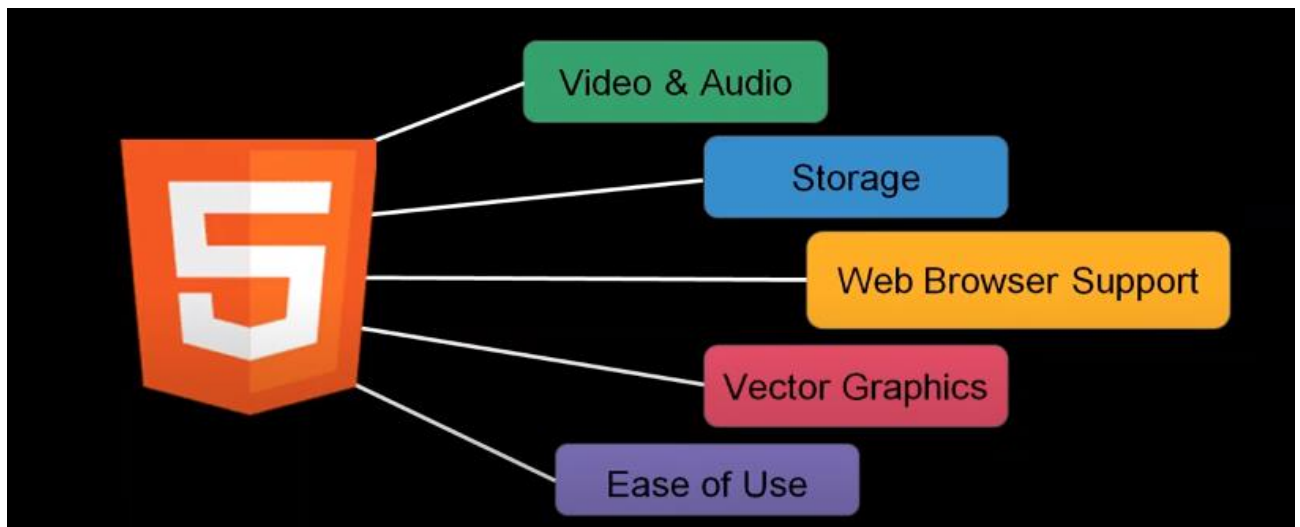
### ***Definition of HTML5:***

*HTML5*: is the fifth and last major HTML version. It is Hypertext Markup Language revision 5

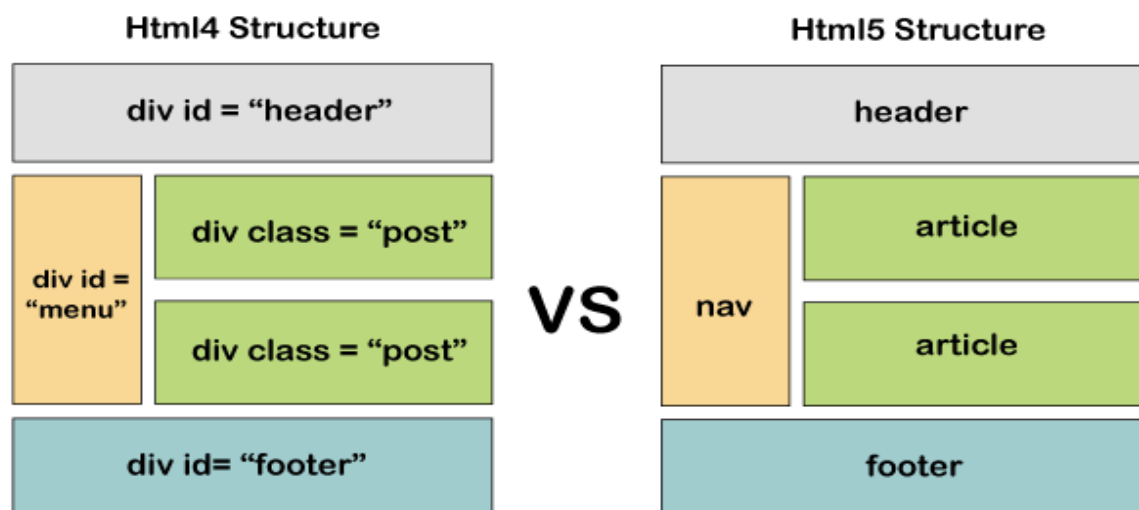
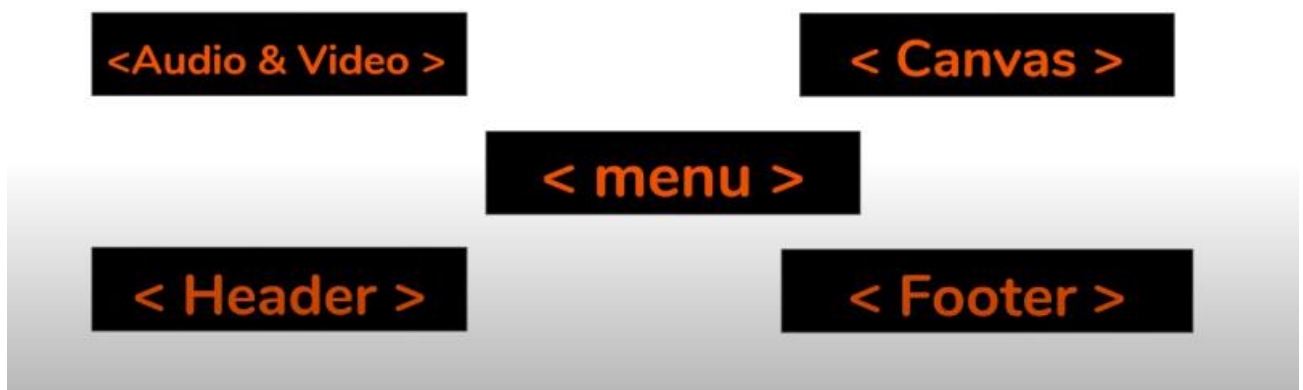
*HTML5*: is markup language for the structure and presentation of World Wide Web contents. HTML5 supports the traditional HTML and XHTML-style syntax and other new features in its markup, New APIs, XHTML and error handling.

### ***New features of HTML5:***

- New parsing rules that are not based on SGML but are oriented towards flexible parsing and compatibility.
- Support of use of inline Scalar Vector Graphics (SVG) and Mathematical Markup Language (MathML) in text/html.
- New available elements include article, aside, audio, bdi, canvas, command, datalist, details, embed, figcaption, figure, footer, header, hgroup, keygen, mark, meter, nav, output, progress, rp, rt, ruby, section, source, summary, time, video and wbr.
- New available types of form controls include dates and times, email, url, search, number, range, tel and color.
- New available attributes of charset on meta and async on script.
- Global attributes that can be applied for every element that include id, tabindex, hidden, data-\* or custom data attributes.
- Most of HTML5 tags is semantic tags that have clear meaning and this make the search engines such as google find website faster.



Some of the important tags that were added in HTML5:





# Travel reservation form

**\* denotes mandatory**

Full name\*:

Email address\*:

Select Tour Package\* :

Arrival date\*:

Number of persons\*:

What would you want to avail?\*

Boarding ☐

Fooding ☐

Sight seeing ☐

Discout Coupon code:

Terms and conditions\*

☒ I agree ☐ I disagree