

HTML: introduction

On the program

1. HTML?
2. A tag language
3. Syntax
4. Validation
5. Semantic

On the program

1. HTML?
2. A tag language
3. Syntax
4. Validation
5. Semantic

HTML

What does **HTML** mean?

HTML

HyperText Markup Language

HTML

HyperText Markup Language *a tag language for hypertext*

HTML

HyperText Markup Language *a tag language for hypertext*

a language

for describing web documents, it can be used to structure the content of these documents,

HTML

HyperText Markup Language *a tag language for hypertext*

a language

for describing web documents, it can be used to structure the content of these documents,

With tags

the structure of documents is organized using tags,

HTML

HyperText Markup Language *a tag language for hypertext*

a language

for describing web documents, it can be used to structure the content of these documents,

With tags

the structure of documents is organized using tags,

for hypertext

documents contain [hyperlinks](#) providing direct access to another part of the document

Example

HTML5

Current major version of HTML where there is an addition of tags compared with previous versions

XHTML

- An HTML version conform to XML language
- XML: eXtensible Meta Language
- imposes rules for writing

XHTML5

- XHTML5 $\equiv$ HTML5
- The same tags
- More rigorous syntax

On the program

1. HTML?
2. A tag language
3. Syntax
4. Validation
5. Semantic

What is a language?

A Language

Vocabulary + Syntax + Semantic

A Language

Vocabulary + Syntax + Semantic

Syntax = Grammar

Semantic=Meaning

A Language

Vocabulary + Syntax + Semantic

Syntax = Grammar

Semantic=Meaning

Vocabulary enables you to construct texts that respect the **Syntax** and make **sense**

Vocabulary

Predefined **elements**, identified by **tags**

Vocabulary

Predefined **elements**, identified by **tags**

- **Opening** <element>

Vocabulary

Predefined **elements**, identified by **tags**

- **Opening/Closing** `<element>/</element>`

Vocabulary

Predefined **elements**, identified by **tags**

- **Opening/Closing** `<element>/</element>`
`<h1> <p> <strong> ...`

Vocabulary

Predefined **elements**, identified by **tags**

- **Opening/Closing** `<element>/</element>`
 `<h1> <p> <strong> ... / </h1> </p> </strong> ...`

Vocabulary

Predefined **elements**, identified by **tags**

- **Opening/Closing** `<element>/</element>`
`<h1> <p> <strong> ... / </h1> </p> </strong> ...` **Self**

Closing tags: `<element/>`

Vocabulary

Predefined **elements**, identified by **tags**

- **Opening/Closing** `<element>/</element>`
`<h1> <p> <strong> ... / </h1> </p> </strong> ... Self`
- Closing** tags: `<element/>`
`<br/> <img/>`

Vocabulary

Predefined **elements**, identified by **tags**

- **Opening/Closing** `<element>/</element>`
`<h1> <p> <strong> ... / </h1> </p> </strong> ... Self`
- **Closing** tags: `<element/>`
`<br/> <img/>`
- Case-insensitive HTML: `<eLEmENt> = <eLeMeNT>`.

Vocabulary

Predefined **elements**, identified by **tags**

- **Opening/Closing** `<element>/</element>`
`<h1> <p> <strong> ... / </h1> </p> </strong> ...`
- **Self-Closing** tags: `<element/>`
`<br/> <img/>`
- Case-insensitive HTML: `<eLEmENt> = <eLeMeNT>`.
- but XHTML $\Rightarrow$ **lowercase**: `<element>`

We name **a content of an element** the code of its opening and closing tags

<p> content of p element </p>

We name **a content of an element** the code of its opening and closing tags

<p> content of p element </p>

NB: self-closing tags have empty content.

On the program

1. HTML?
2. A tag language
3. Syntax
4. Validation
5. Semantic

On the program

1. HTML?
2. A tag language
3. **Syntax**
4. Validation
5. Semantic

Syntax

The syntax defines

- The structure of the document
- the rules of writing

Minimum structure

Minimum structure

```
<DOCTYPE html>
```

← DOCTYPE declaration

```
<html xmlns="http://www.w3.org/1999/xhtml">
```

```
<!-- document header -->
```

```
<head>
```

```
<title>Minimal HTML 5 document</title>
```

```
<meta charset="UTF-8"/>
```

```
</head>
```

```
<!-- document body -->
```

```
<body>
```

```
<!-- we place the page content here -->
```

```
...
```

```
</body>
```

```
</html>
```

Minimum structure

```
<DOCTYPE html
```

← DOCTYPE declaration

```
<html xmlns="http://www.w3.org/1999/xhtml">
```

← a root element

```
<!-- document header -->
```

```
<head>
```

```
<title>Minimal HTML 5 document</title>
```

```
<meta charset="UTF-8"/>
```

```
</head>
```

```
<!-- document body -->
```

```
<body>
```

```
<!-- we place the page content here -->
```

```
...
```

```
</body>
```

```
</html>
```

Minimum structure

```
<DOCTYPE html
```

← DOCTYPE declaration

```
<html xmlns="http://www.w3.org/1999/xhtml">
```

← a root element

```
<!-- document header -->
```

```
<head>
```

```
<title>Minimal HTML 5 document</title>
```

```
<meta charset="UTF-8"/>
```

```
</head>
```

← The header of the document

```
<!-- document body -->
```

```
<body>
```

```
<!-- we place the page content here -->
```

```
...
```

```
</body>
```

```
</html>
```

Minimum structure

```
<DOCTYPE html
```

← DOCTYPE declaration

```
<html xmlns="http://www.w3.org/1999/xhtml">
```

← a root element

```
<!-- document header -->
```

```
<head>
```

```
<title>Minimal HTML 5 document</title>
```

```
<meta charset="UTF-8"/>
```

```
</head>
```

← The header of the document

← a title

← declaration of the character
encoding used

```
<!-- document body -->
```

```
<body>
```

```
<!-- we place the page content here -->
```

```
...
```

```
</body>
```

```
</html>
```

Minimum structure

```
<DOCTYPE html
```

← DOCTYPE declaration

```
<html xmlns="http://www.w3.org/1999/xhtml">
```

← a root element

```
<!-- document header -->
```

```
<head>
```

← The header of the document

```
<title>Minimal HTML 5 document</title>
```

← a title

```
<meta charset="UTF-8"/>
```

← declaration of the character
encoding used

```
</head>
```

```
<!-- document body -->
```

```
<body>
```

← the body of the document

```
<!-- we place the page content here -->
```

```
...
```

```
</body>
```

```
</html>
```

Writing rules

A valid HTML document must be parenthetical

Writing rules

A valid HTML document must be parenthetical
Every opening `<element>` tag must have an associated closing `</element>` tag.

Writing rules

A valid HTML document must be parenthetical

1. Every opening `<element>` tag must have an associated closing `</element>` tag.
2. The elements must not overlap:

first open, last close

ex: `<p>` start `<code>....?</p>` more elements...
`</code>`

! illegal !

Writing rules

A valid HTML document must be parenthetical

1. Every opening `<element>` tag must have an associated closing `</element>` tag.
2. The elements must not overlap:
first open, last closed

ex: `<p>` start `<code>` more elements..... `</code>``<p>`

Writing rules

Embedded elements the content of an element can be made up of other embedded elements

Writing rules

Embedded elements the content of an element can be made up of other embedded elements

```
<p>  
  start of the content of p  
  <code>  
    content 1  
    <strong>  
      Embedded content2  
    </strong> end  
    Embedded content1  
  </code> content of p  
</p>
```

Writing rules

**Embedd elements must obey to rules,
not all embedded elements are possible!**

Writing rules

**Embedd elements must obey to rules,
not all embedded elements are possible!**

examples:

- an element `<p>` cannot contain an element `<h1>`.
- an element `<ul>` necessarily contains at least one element `<li>`
- an element `<li>` is necessarily nested within an element `<ul>`
etc.

On the program

1. HTML?
2. A tag language
3. Syntax
- 4. Validation**
5. Semantic

Validation

An (X)HTML document is **valid** if it complies with all these rules.

Example

Validation

Validation tools can be used to check that a document is syntactically correct.

`http://validator.w3.org/nu (xhtml extension)`

On the program

1. HTML?
2. A tag language
3. Syntax
4. Validation
5. **Semantic**

Semantic

Each element (tag) is associated with a semantic that defines its use. It lets you know when and why to use an element.

Semantic

- Each element (tag) is associated with a semantic that defines its use.
- It lets you know when and why to use an element.
- The<p> element is used to represent a paragraph

[MDN](#)

Semantic

- Each element (tag) is associated with a semantic that defines its use.
- It lets you know when and why to use an element.
- The<p> element is used to represent a paragraph
- the <time> element is used to identify a time or date
-

[MDN](#)

[MDN](#)

Semantic

- Each element (tag) is associated with a semantic that defines its use.
- It lets you know when and why to use an element.
- The<p> element is used to represent a paragraph [MDN](#)
- the <time> element is used to identify a time or date [MDN](#)
- The <td> element represent a box in a table [MDN](#)
- the element is used to give importance to a text [MDN](#)

[Review the list of tags](#)

Semantic

- Each element (tag) is associated with a semantic that defines its use.
- It lets you know when and why to use an element.
- The<p> element is used to represent a paragraph [MDN](#)
- the <time> element is used to identify a time or date [MDN](#)
- The <td> element represent a box in a table [MDN](#)
- the element is used to give importance to a text [MDN](#)

[Review the list of tags](#)

In addition to the tags, there is the text content to give the overall semantics (meaning) of the document.