HTML

Tecnologie Internet

a.a. 2022/2023

Introduction

The *World Wide Web (Web)* is a network of information resources.

The Web relies on three mechanisms to make these resources readily available to the widest possible audience:

1. A uniform **naming scheme** for locating resources on the Web (e.g., URIs).

2. **Protocols**, for access to named resources over the Web (e.g., HTTP).

3. **Hypertext**, for easy navigation among resources (e.g., HTML).

Introduction

To publish information for global distribution, one needs a universally understood language, a kind of publishing mother tongue that all computers may potentially understand.
The publishing language used by the World Wide Web is **HTML** (HyperText Markup Language).

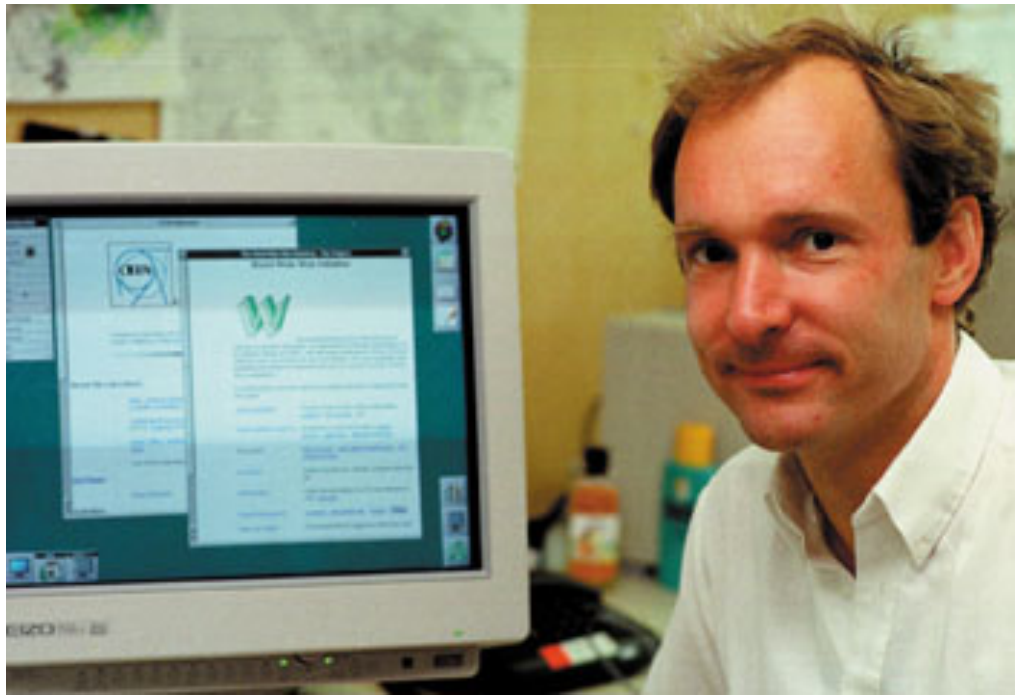
HTML gives authors the means to:

- Publish online documents with headings, text, tables, lists, photos, etc.
- Retrieve online information via hypertext links, at the click of a button.
- Design forms for conducting transactions with remote services, for use in searching for information, making reservations, ordering products, etc.
- Include spread-sheets, video clips, sound clips, and other applications directly in their documents.

History

HTML was originally developed by **Tim Berners-Lee** while at CERN, and popularized by the Mosaic browser developed at NCSA.
During the 1990s it has blossomed with the explosive growth of the Web.

During this time, HTML has been extended in a number of ways. The Web depends on Web page authors and vendors sharing the same conventions for HTML. This has motivated joint work on specifications for HTML.



History

HTML 2.0 (RFC 1866, November 1995) was developed under the aegis of the Internet Engineering Task Force (IETF) to codify common practice.

The efforts of the World Wide Web Consortium's HTML Working Group to codify common practice in 1996 resulted in **HTML 3.2** (W3C Recommendation, January 1997).

HTML 4.0 (W3C Recommendation, December 1997) extends HTML with mechanisms for style sheets, scripting, frames, embedding objects, improved support for right to left and mixed direction text, richer tables, and enhancements to forms, offering improved accessibility for people with disabilities.

HTML 4

HTML 4.01 (W3C Recommendation, December 1999) is a revision of HTML 4.0 that corrects errors and makes some changes since the previous revision.

Main principles:

- separate structure and presentation
- consider universal accessibility of the web
- help user agents with incremental rendering

HTML 5 (W3C Recommendation, October 2014)

HTML 5.2 (W3C Recommendation, December 2017)

HTML living standard https://html.spec.whatwg.org/

HTML document representation

Character set

To promote interoperability, each HTML document must specify its *document character set.* A document character set consists of:

- *A Repertoire*: A set of abstract characters, such as the Latin letter "A", the Cyrillic letter "I", the Chinese character meaning "water", etc.
- *Code positions*: A set of integer references to characters in the repertoire.

HTML uses the **Universal Character Set (UCS)**, defined in ISO-10646. This standard defines a repertoire of thousands of characters used by communities all over the world.
The character set defined in ISO-10646 is character-by-character equivalent to Unicode.

HTML document representation

Character encoding

The document character set does not suffice to allow user agents to correctly interpret HTML documents as they are typically **encoded as a sequence of bytes** in a file or during a network transmission. User agents must also know the specific *character encoding* that was used to transform the document character stream into a byte stream.

Commonly used character encodings on the Web include ISO-8859-1 (also referred to as "Latin-1"; usable for most Western European languages), ISO-8859-5 (which supports Cyrillic), SHIFT_JIS (a Japanese encoding), EUC-JP (another Japanese encoding), and UTF-8 (an encoding of ISO 10646 using a different number of bytes for different characters). Names for character encodings are case-insensitive.

HTML document representation

Character encoding

How does a user agent know which character encoding has been used? The server should provide this information. The most straightforward way for a server to inform the user agent about the character encoding of the document is to use the **`charset`** parameter of the `Content-Type` header field of the HTTP protocol.

For example, the following HTTP header announces that the character encoding is EUC-JP:

```
Content-Type: text/html; charset=EUC-JP
```

Basic HTML data types

Lengths

- pixels
- percentage of the available horizontal or vertical space
- a sequence of lengths separated by “,”

URIs

- protocol
- domain name
- local name
- bookmark

Example:

```
http://www.example.com/mypage.html#bookmark_name
```

Basic HTML data types

Dates

The format is:

YYYY-MM-DDThh:mm:ssTZD

where:

YYYY = four-digit year
MM = two-digit month (01=January, etc.)
DD = two-digit day of month (01 through 31)
hh = two digits of hour (00 through 23) (am/pm NOT allowed)
mm = two digits of minute (00 through 59)
ss = two digits of second (00 through 59)
TZD = time zone designator (+1,+2,.. or -1,-2,.. vs UTC)

The global structure of an HTML document

An HTML 4 document is composed of three parts:
1 - a line containing HTML version information,
2 - a declarative header section (delimited by the **HEAD** element),
3 - a body, which contains the document's actual content. The body may be implemented by the **BODY** element or the **FRAMESET** element.

First HTML example: https://www.w3schools.com/html/tryit.asp?filename=tryhtml_basic

```
<!DOCTYPE HTML>
<HTML>
   <HEAD>
      <TITLE>My first HTML document</TITLE>
   </HEAD>
   <BODY>
      <P>Hello world!</P>
   </BODY>
</HTML>
```

Every HTML document **must** have a **TITLE** element in the **HEAD**.

The global structure of an HTML document

Other tags for the header section:

– **BASE**: the base URI, to be used with relative URIs

```
<BASE href="http://www.w3schools.com/images/" target="_blank">
```

– **LINK**: links to external documents (e.g., style sheets)

```
<LINK href="special.css" rel="stylesheet" type="text/css">
```

– **SCRIPT**: references to scripts

```
<SCRIPT type="text/vbscript" src="http://someplace.com/progs/vbcalc">
```

– **STYLE**: style definition

```
<STYLE>
H1 {color:red;}
P {color:blue;}
</STYLE>
```

– **META**: meta-information about the document

```
<META name="Author" content="John Doe">
```

The global structure of an HTML document

Regarding the body, the **DIV** and **SPAN** elements, in conjunction with the **id** and **class** attributes, offer a generic mechanism for adding structure to documents. These elements define content to be inline (SPAN) or block-level (DIV) but impose no other presentational idioms on the content. Thus, authors may use these elements in conjunction with style sheets, the **lang** attribute, etc., to tailor HTML to their own needs and tastes.

```
<DIV id="client-boyera" class="client">
<P><SPAN class="client-title">Client information:</SPAN></P>
<TABLE class="client-data">
<TR><TH>Last name:<TD>Boyera</TR>
<TR><TH>First name:<TD>Stephane</TR>
<TR><TH>Tel:<TD>(212) 555-1212</TR>
<TR><TH>Email:<TD>sb@foo.org</TR>
</TABLE>
</DIV>
```

The global structure of an HTML document

A heading element briefly describes the topic of the section it introduces. There are six levels of headings in HTML with **H1** as the most important and **H6** as the least. Visual browsers usually render more important headings in larger fonts than less important ones.

The following example shows how to use the **DIV** element to associate a heading with the document section that follows it.

```
<DIV class="section" id="forest-elephants" >
<H1>Forest elephants</H1>
<P>In this section, we discuss the lesser known forest elephants.
...this section continues...
<DIV class="subsection" id="forest-habitat" >
<H2>Habitat</H2>
<P>Forest elephants do not live in trees but among them.
...this subsection continues...
</DIV>
</DIV>
```

Text

Lines and paragraphs

A *line break* is defined to be a carriage return (
), a line feed (
), or a carriage return/line feed pair.

The **BR** element forcibly breaks (ends) the current line of text.

```
This text contains<BR>a line break.
```

The HTML markup for *defining* a paragraph is straightforward: the **P** element defines a paragraph.

```
<P>This is a paragraph</P>
<P>This is another paragraph</P>
```

Text

Preformatted text

The **PRE** element tells visual user agents that the enclosed text is "preformatted".

```
<PRE>
       Higher still and higher
         From the earth thou springest
       Like a cloud of fire;
         The blue deep thou wingest,
And singing still dost soar, and soaring ever singest.
</PRE>
```

Text

Phrase elements

EM: Indicates emphasis.
STRONG: Indicates stronger emphasis.
CITE: Contains a citation or a reference to other sources.
DFN: Indicates that this is the defining instance of the enclosed term.
CODE: Designates a fragment of computer code.
SAMP: Designates sample output from programs, scripts, etc.
KBD: Indicates text to be entered by the user.
VAR: Indicates an instance of a variable or program argument.
ABBR: Indicates an abbreviated form (e.g., WWW, HTTP, URI, Mass., etc.).
ACRONYM: Indicates an acronym (e.g., WAC, radar, etc.).

Text

Phrase elements

```
As <CITE>Harry S. Truman</CITE> said,
<Q lang="en-us">The buck stops here.</Q>
More information can be found in <CITE>[ISO-0000]</CITE>.
Please refer to the following reference number in future
correspondence: <STRONG>1-234-55</STRONG>
```

The presentation of phrase elements depends on the user agent. Generally, visual user agents present EM text in italics and STRONG text in bold font. Speech synthesizer user agents may change the synthesis parameters, such as volume, pitch and rate accordingly.

```
<ABBR title="World Wide Web">WWW</ABBR>
<ABBR lang="fr" title="Soci&eacute;t&eacute; Nationale des Chemins de Fer">
   SNCF
</ABBR>
<ABBR title="Abbreviation">abbr.</ABBR>
```

Text

Quotations

BLOCKQUOTE is for long quotations (block-level content) and **Q** is intended for short quotations (inline content) that don't require paragraph breaks.

```
<BLOCKQUOTE cite="http://www.mycom.com/tolkien/twotowers.html">
<P>They went in single file, running like hounds on a strong scent,
and an eager light was in their eyes. Nearly due west the broad
swath of the marching Orcs tramped its ugly slot; the sweet grass
of Rohan had been bruised and blackened as they passed.</P>
</BLOCKQUOTE>
```

Visual user agents generally render BLOCKQUOTE as an indented block. Visual user agents must ensure that the content of the Q element is rendered with delimiting quotation marks. Authors should not put quotation marks at the beginning and end of the content of a Q element.

Text

Subscripts and superscripts

Many scripts (e.g., French) require superscripts or subscripts for proper rendering. The **SUB** and **SUP** elements should be used to markup text in these cases.

```
H<sub>2</sub>O
<br>
E = mc<sup>2</sup>
<br>
<SPAN lang="fr">M<sup>lle</sup> Dupont</SPAN>
```

Lists

HTML offers authors several mechanisms for specifying lists of information. All lists must contain one or more list elements. Lists may contain:

- Unordered information.
- Ordered information.
- Definitions.

The previous list, for example, is an **unordered list**, created with the **UL** element:

```
<UL>
<LI>Unordered information.
<LI>Ordered information.
<LI>Definitions.
</UL>
```

Lists

An **ordered list**, created using the **OL** element, should contain information where order should be emphasized, as in a recipe:

1. Mix dry ingredients thoroughly.
2. Pour in wet ingredients.
3. Mix for 10 minutes.
4. Bake for one hour at 300 degrees.

```
<OL>
<LI>Unordered information.
<LI>Ordered information.
<LI>Definitions.
</OL>
```

Lists

Definition lists, created using the **DL** element, generally consist of a series of term/definition pairs (although definition lists may have other applications).

```
<DL>
<DT><STRONG>Lower cost</STRONG>
<DD>The new version of this product costs significantly less than the
previous one!
<DT><STRONG>Easier to use</STRONG>
<DD>We've changed the product so that it's much easier to use!
<DT><STRONG>Safe for kids</STRONG>
<DD>You can leave your kids alone in a room with this product and
they won't get hurt (not a guarantee).
</DL>
```

Tables

Tables are defined with the **TABLE** tag.
Tables are divided into **table rows** with the **TR** tag.
Table rows are divided into **table data** with the **TD** tag.
A table row can also be divided into **table headings** with the **TH** tag.

```
<TABLE>
  <TR>
    <TD>Jill</TD>
    <TD>Smith</TD>
    <TD>50</TD>
  </TR>
  <TR>
    <TD>Eve</TD>
    <TD>Jackson</TD>
    <TD>94</TD>
  </TR>
</TABLE>
```

Tables

To make a cell span more than one row, use the **rowspan** attribute:

```
<TABLE>
  <TR>
    <TH>Name:</TH>
    <TD>Bill Gates</TD>
  </TR>
  <TR>
    <TH rowspan="2">Telephone:</TH>
    <TD>555 77 854</TD>
  </TR>
  <TR>
    <TD>555 77 855</TD>
  </TR>
</TABLE>
```

Tables

To make a cell span more than one column, use the **colspan** attribute:

```
<TABLE>
  <TR>
    <TH>Name</TH>
    <TH colspan="2">Telephone</TH>
  </TR>
  <TR>
    <TD>Bill Gates</TD>
    <TD>555 77 854</TD>
    <TD>555 77 855</TD>
  </TR>
</TABLE>
```

Tables

To add a caption to a table, use the **CAPTION** tag:

```
<TABLE>
  <CAPTION>Monthly savings</CAPTION>
  <TR>
    <TH>Month</TH>
    <TH>Savings</TH>
  </TR>
  <TR>
    <TD>January</TD>
    <TD>$100</TD>
  </TR>
  <TR>
    <TD>February</TD>
    <TD>$50</TD>
  </TR>
</TABLE>
```

Links

In HTML, links are defined with the **A** tag:

```
<A href="http://www.unipr.it/">University of Parma</A>
```

A local link (link to the same web site) is specified with a relative URL (without http://www....).

The **target** attribute specifies where to open the linked document. This example will open the linked document in a new browser window or in a new tab:

```
<A href="http://www.w3schools.com/" target="_blank">Visit W3Schools!</A>
```

_blank: Opens the linked document in a new window or tab
_self: Opens the linked document in the same frame as it was clicked (this is default)
_parent: Opens the linked document in the parent frame
_top: Opens the linked document in the full body of the window
framename: Opens the linked document in a named frame

Links

HTML **bookmarks** are used to allow readers to jump to specific parts of a Web page.
Bookmarks are practical if your website has long pages.

You must first create the bookmark:

```
<H2 id="tips">Useful Tips Section</H2>
```

and then add a link to it.

```
<A href="#tips">Visit the Useful Tips Section</A>
```

When the link is clicked, the page will scroll to the location with the bookmark.

Images and objects

The **IMG** element embeds an image in the current document at the location of the element's definition. The **IMG** element has no content; it is usually replaced inline by the image designated by the **src** attribute, the exception being for left or right-aligned images that are "floated" out of line.

```
<P>Parma, University Campus, Centro S. Elisabetta:
<IMG src="http://www.qis.unipr.it/img/photos/SElisabetta2.png"></IMG>
</P>
```

Images and objects

The **OBJECT** element defines an embedded object within an HTML document.

```
<object width="400" height="400" data="helloworld.swf"></object>
```

The term "object" is used to describe the things that people want to place in HTML documents; other commonly used terms for these things are: resources, applets, plug-ins, media handlers, etc.

Forms

An HTML form is a section of a document containing normal content, markup, special elements called **controls** (checkboxes, radio buttons, menus, etc.), and labels on those controls. Users generally "complete" a form by modifying its controls (entering text, selecting menu items, etc.), before submitting the form to an agent for processing (e.g., to a Web server, to a mail server, etc.)

```
<FORM action="http://somesite.com/prog/adduser" method="post">
   <P>
   <LABEL for="firstname">First name: </LABEL>
            <INPUT type="text" id="firstname"><BR>
   <LABEL for="lastname">Last name: </LABEL>
            <INPUT type="text" id="lastname"><BR>
   <LABEL for="email">email: </LABEL>
            <INPUT type="text" id="email"><BR>
   <INPUT type="radio" name="sex" value="Male"> Male<BR>
   <INPUT type="radio" name="sex" value="Female"> Female<BR>
   <INPUT type="submit" value="Send"> <INPUT type="reset">
   </P>
</FORM>
```

Forms

The browser collect user data by means of forms (tag **FORM**). Data are sent to a server, over HTTP, located to the URI specified by the **action** attribute. Which HTTP method (either GET or POST) will be used to submit data is specified by the **method** attribute.

The INPUT tag allows for specifying controls:
- text
- password
- checkbox
- radio
- submit
- image
- reset
- button
- hidden
- file

Frames

`iframe` is used to embed another document within one:

```
<!DOCTYPE html>
<html>
<body>

<h2>HTML Iframes</h2>
<p>You can use the height and width attributes to specify the size of the
iframe:</p>

<iframe src="demo_iframe.htm" height="200" width="300"></iframe>

</body>
</html>
```

Other interesting elements and APIs

Elements:

- **semantic** elements like <HEADER>, <FOOTER>, <ARTICLE>, and <SECTION>.
- form **control attributes** like number, date, time, calendar, and range.
- **graphic** elements: <SVG> and <CANVAS>.
- **multimedia** elements: <AUDIO> and <VIDEO>.

APIs:

- HTML Geolocation
- HTML Drag and Drop
- HTML Local Storage
- HTML Application Cache
- HTML Web Workers
- HTML Server-Sent Events (SSE)

References

https://html.spec.whatwg.org/

http://www.w3schools.com/html/

http://www.w3schools.com/html/tryit.asp?filename=tryhtml_basic