**Web Page** - One document of information of a web site. Usually in HTML. May have CSS, JavaScript, Images and other...

**Web Site** - Collection of web pages having web resources. It has navigation between pages. Hosted on at least one server.

**World Wide Web** - Collection of all sites.

**Web App** - Next level. High interactivity. AJAX, Silverlight, Flash... Broken into tiers. Each tier has a role.

**WEB 1.0** - Ptoffessional autors who put content. Expensive. Users = consumers.

**WEB 2.0** - Users generate content. Anyone can publish. It has rich Internet applications. Social web. High content variaty. Users are autors too.

**WEB 3.0** - Computer generated information. Semantic web. AI. Personalization. Mobility.

**Semantic Web** - Can be read an categorized by computers. Search engines are faster. Hard to be implemented.

**AI**- Google Suggest, Google Translate

**Personalizations** - Sites show different content for different users.

**Trident-Based Web Browsers** - Internet Explorer, Netscape, Maxton

**Gecko-Based Web Browsers** - Firefox, Netscape, Sea Monkey

**Webkit-Based Web Browsers**- Chrome, Safari, Maxton

**Presto-Based Web Browsers** - Opera

**User Agent String** - Identify browser version, layout engine, OS

Windows NT 6.3 = Windows 8.1

Windows NT 6.2 = Windows 8

Windows NT 6.1 = Windows 7

Windows NT 6.0 = Windows Vista and Server 2008

Windows NT 5.2 = Windows 2003 Server ans XP x64

Windows NT 5.1 = Windows XP

Windows NT 5.0 = Windows 2000

Windows NT 4.0 = Windows NT

WOW64 = Win32 on Windows 64

x64 - 64 bit application on Windows 64

**Servers** - Database Server, File Server, Mail Server, Print Server, VPS Server

**Hardware Server** - Controlled by OS. Handles web requests. Redircted to APS.NET, PHP and other server-side programming languages.

Apache - 60.31% Of all servers

IIS - 19,34% (MS)

nginx - 7,65% (Igor Syslov)

GWS - 5,09% (Google)

lighttpd - 0,60%

**Client-Server Architecture:**

-server - machine(s) providing web apps or services to client. Cloud, PHP, ASP.NET

-client - software with UI to access services. Web Browsers, HTML5

**3-Tier Architecture:**

- Front End - client (UI)

- Middle Tier (Business Layer) - server software core system logic

- Backend - database, cloud

**ORM** - Object Relational Database. Virtually maps tables in database.

**SOA**- Service-oriented architecture - Create service - Register service - Secure service - Manage service - Virtualize service - Integrate service

**IDE Coding Tools** - Visual Studio, Web Develeper, Web Matrix, Eclipse, Aptana, Webstorm, Dream Weaver

**Lite Coding Tools**- Gedit, Notepad++

**Browser Tools**- Firebug, Webinspector, F12, Inspect, Console

**Drawing Tools** - Photoshop, Fireworks, Gimp, Paint.NET

**Source Control**- TFS, GIT, SVN

**DOM** - Document Object Model - defines logical structure of documents

**OSI Model** - Application Layer, Presentation Layer, Session Layer, Transport Layer, Network Layer, Datalink Layer, Physical Layer

**HTML**- Hypertext Markup Language

Структура на HTML документ

**<doctype>** - validator of the page (описва стандарта на HTML документа)

**<html>** - включва целия код (след doctype до края)/ можем да опишем езика на сайта <html lang=””>/

**<head>** - markup is not visibel for the userin the head, дава информация за браузарите, машините

В <head> може да има

<title> - заглавие на прозореца

<meta charseat=””> енкодинга (utf-8 или др.)

<meta name=””>

keywords – какво търси машината

descritpio – какво търси машината

…..др.

(Между **<head>** и **<body>** не трябва да пише нищо)

**<body>** - element contains all the visible to the user markup

Heading, text, hyperlinks, images, rtc…..

**tag** - smallest peice in HTML

**attribute** - property of a tag

**ellement**- combination of opening, closing tag and attributes

**<b>** - bold (не е добре да се използва, а де се замества с CSS)

**<i>** - italic

**<sup>** - superscript

**<sub>**- subscript

**<strong>** - stronger text (използва се за указване на важен текст (може да се забави и т.н.))**(указва смисъл)**

**<em>** - emphasized text (да се използва вместо <i>)

**<blockquote>** - quoted text

**<del>** - deleted text

**<pre> tag** - preformatted text (изглежда като код)

**Inline elements** - <span> <a> <em> <img>

**<br />** - new line

**<a href="..."> </a>** - hyperlink (а – anchor) (линка може да бъде релативен, абсолютен или текущ)

**<img src="..." alt="..." />** - image (когато сочим до файл от папка е добре да посочим релативния път до нея спрямо html файла: imgsrc=”images/SEB-Ninja.png”)

**alt**  - използва се за hover-ване с мишката или от търсещите машини (или за незрящи)

**<span>** - inline element. Used with styles (работи на един ред span span)

**Block elements** - <div> <p> <h1> <ul> <dl> <ol>

**<h1> - <h6>** - headings

**<p>** - paragraph

**<div>** - rectangular block element. Used with styles (работи на отделни редове div div)

**<ol>**- ordered list(1, A, a, I, i)

**<ul>**- unordered list (circle, disc, square)

**<dl>** - definition list

**<dt>**- data title

**<dd>** - data description

**Section elements/tags**

**<div> -** отделям част от прозореца за самостоятелно оформяне

**<span>** - отделям част от текст за допълнителна обработка

**HTML 5 tags / Semantic structural tags:** (вместо div в div в div в div)

**<nav>** - navigation. Used with links, lists. == (<div id=“ nav”>)

**<header>**- header of a section, site == (<div id=“header”>)

**<footer>** - footer of a section, site == (<div id=“footer”>)

**<section>** - section of a website == (<div id=“section”>)

**<article>** - part of a section == (<div id=“article”>)

**<aside>** - content not directly with the site. Ads for example == (<div id=“ aside”>)

**<hgroup>**- header of a section

**<figure>**- self-contained flow content

**<figcaption>** - optional caption for figure

**<video> <audio>** - media

**<track>** - subtitles for video

**<mark>** - highlighted text

**<progress>** - progress bar

**<meter>** - measurement

**<time>**- date or time

**<bdi>** - isolated text

**<wbr>** - line-break opportunity

**<canvas>** - dynamic content

**<datalist>** - set of predefined options (can be hided or shown)

**<keygen>** - key generator

**<output>**- output of a calculation

**<meta>**- meta tag

**Literals**

**&** - to insert symbol. For example &amp shows ampersand

**&nbsp** – literal for single space

**&copy** – literal for copy symbol “©”

**&bull** – literal for bull point “●”

**<!-- ... -->** - comment

**<div title="...">** - gives tooltip on the div

**<code>** - gives you code

**<cite>**- citedd text

**<big>** - bigger font

**<small>** - smaller font

**Tables –** само за репрезентиране на таблици, не за loyaut

**<table></table>-** begin / end of table

**<caption>** with <table> - description of the table

**<th>** - header cells (bold and centered, have important meaning)

**<tr>**- table row (таблиците винаги започват с <tr>)

**<td>** - table data (can be used for footer)

**Table rows split into semantic sections** (указва информация на машините/google)

**<table>**

**<colgroup>** and **<col>** - define columns (used to set the vision of some specific tables)

<colgroup>

<col style=”width:100px”/><col /> (прави се в CSS по принцип)

</colgroup>

**<thead>** denotes table header and contains **<th>** elements, instead **<td>** elements

<tr><th>Column 1</th><th>Column 2</th></tr>

**</thead>**

**<tfoot>** denotes table footer but comes BEFORE the **<tbody>**

<tr><th>Footer 1</th><th> Footer 2</th></tr>

**</tfoot>**

**<tbody>** denotes collection of table rows that contain the very data.

<tr><th>Cell 1</th><th> Cell 2</th></tr>

**</tbody>**

**</table>**

|  |  |  |
| --- | --- | --- |
| **cellspacing**- space between cells  (margin) |  |  |
| **cellpadding** - margins of cells |  |

|  |  |
| --- | --- |
| **colspan** - merges columns |  |
| **rowspan** - merges rows |  |
| (не могат да бъдат комбинирани) |  |

**tabborder, tabindex** - for TAB key

**<hr>** - horizontal line

**Forms –** primary method for gathering information data from site visitors

Can contain:

**Text fields** for the user to type

**Buttons** for interactions like “**Register**”, “**Login**”, “**Search**”

**Menus**, **Sliders**, etc….

|  |  |
| --- | --- |
| **<form name="...">** - the name of the form | The name attribute specifies the name of an <input> element.  The name attribute is used to reference elements in a JavaScript, or to reference form data after a form is submitted.  Note: Only form elements with a name attribute will have their values passed when submitting a form. |
| **<form method="...">** - how to send data (post) | The method attribute specifies how to send form-data (the form-data is sent to the page specified in the action attribute).  The form-data can be sent as URL variables (with method="get") or as HTTP post transaction (with method="post"). |
| **<form action="...">** - where to send data (to urls) | The action attribute specifies where to send the form-data when a form is submitted. |
| **<form value="...">** - | The value attribute specifies the value of an <input> element. The value attribute is used differently for different input types:   * For "button", "reset", and "submit" - it defines the text on the button * For "text", "password", and "hidden" - it defines the initial (default) value of the input field * For "checkbox", "radio", "image" - it defines the value associated with the input (this is also the value that is sent on submit)   (In HTML5, the action attribute is no longer required.) |

|  |  |
| --- | --- |
| **<input type="text" />** - textbox |  |
| **<input type="reset" />** - reset button for the form |  |
| **<textarea>** - multiline textbox  + атрибут placeholder (изписва текст, който изчесва, когато тръгнем да въвеждаме) |  |
| **<input type="password" />** - \*-textbox |  |
| **<input type="submit" />** - submit button |  |
| **<input type="image" />**- gets coordinates from image |  |
| **<input type="button" />** - with javasctipt |  |
| **<input type="checkbox" />**- checkbox    value – информацията, която се изпраща на сървъра ако чекнем  (fruit==apple) |  |
| **<input type="radio" name="..." /> - radio buttons group by "name"** |  |
| **<select><option value="..." selected="selected"></option></selected>** - dropdown menu |  |
| **<select multiple="multiple"></select>** - multiple choice menu |  |
| **<input type="hidden" />**- hidden data not seen by user  Потребителя не попълва и не вижда, но се изпращат до сървъра |  |
| **<label for="example">**- are used to associate an explanatory text to a form field using the fields ID  тук липсва name  **name** и **value** се изпращат на сървъра |  |
| **<fieldset>** - are used to enclose a group of related form fields:  **<legend** **>** - caption for the fieldset | C:\Users\M.Boykov\Desktop\2016-04-14_155459.jpg  Разделя тематично различните форми за попълване: Лични данни, допълнителни въпроси,…др. |

**Sliders and Spinboxes**

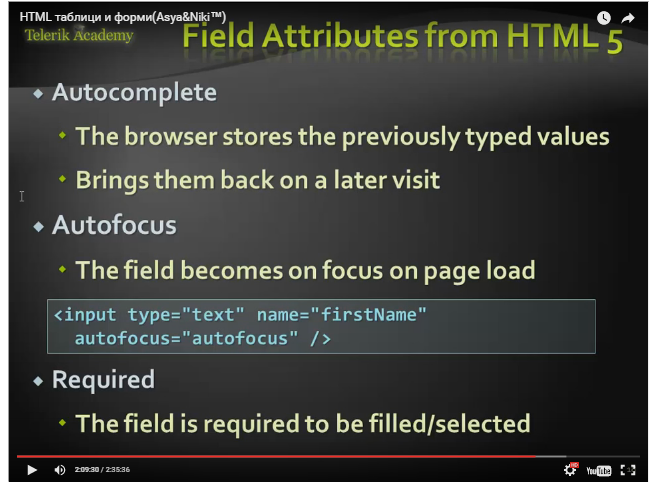
|  |  |  |
| --- | --- | --- |
| **<input type="range">**- spinbox (does not work in Firefox)  **<input type="number">** - slider (does not work in Firefox) |  |  |

**<input type="email" />** - email pattern

**<input type="url" />**- URL pattern

**<input type="tel" />** - brings telephone keypad

**<iframe>** - shows one web site in another



**HTML is used for content, not layout. Presents information in a meaningful manner. Defines headers paragraphs, textboxes. Does not define colors, size, positions.**

**Semantic HTML is easier to read by developers. Easier to render by browsers. Easier for search engines.**

**<details> + <summary>** - accordion like widget

**To show new content on older browsers - HTML5Shiv and Modernizr.**

**<dfn>** - definition

**<kbd>** - user input

**<menu>** - list of commands

**<object>**- external resource

**<optgroup>** - list of option elements with a common label

**<param>** - parameters for plugins. Invoked by object element.

**<s>**- no longer acurate element

**<samp>**- sample output

**<source>** - to show alternative media

**For CSS**

**External CSS file** - <link rel=stylesheet type="text/css" href = "..."></link>

**id** – добре е да имаме една единствена променлива с подобен клас

**class** – можем да имаме няколко различни типа данни и променливи с подобен клас

**name** – стилизиране на форми

**style** – не е препоръчително да се използва вътре в тага на елемента като атрибут (отделен CSS файл или да се изнесе горе в head)