Applied informatics 2

Pavla Langrová
Radek Němec
This material has been created within the project „Inovation of subjects of study in Fakulty of Education at Univerzity of Hradec Králové“, reg. no. CZ.1.07/2.2.00/28.0036.
Basic information regarding the subject

Lecturer: Radek Němec, Mgr.

Contact: Department of Informatics

Hours per week - direct tutoring: 2 hours per week

Hours per week – self-study: 12 hours

Placement of the subject: summer semester

Prerequisites: Applied informatics 1

Is prerequisite for: subject is not a prerequisite for any other subject

Rules regarding the communication with the tutor: students communicate with the tutor in person within guided consultations and electronically in MOODLE (Modular Object-Oriented Dynamic Learning Environment)

Annotation

The subject Applied informatics 2 follows the subject Applied informatics 1 and assumes its very good knowledge. Its focus is on development of work with information and communication technologies in tutoring and in preparation as well as online support of teaching at schools.

Goals of the Subject

The aim of the subject is to familiarize students with markup and scripting languages for the web, WYSIWYG editors and code editors, blogging and editorial systems, graphic formats and graphic editors as well as with general principles of creating websites. A successful graduate of the subject has to be able to create a website by means of one of the above mentioned ways. The website has to have at least 10 pages of related content with an easy navigation.

Synopsis of the Subject

1) 2 h: Principles of creating a website
2) 2 h: Creation of a HTML document via the WYSIWYG editor NVU
3) 2 h: Direct editing of HTML or XHTML via the editor PSPad
4) 2 h: Cascade Style Sheets (CSS) and the editor TopStyle Lite
5) 2 h: Creation of forms and their processing by scripts - JavaScript
6) 2 h: Displaying of a website on a web server, meta information, catalogues
SP) 2 h: Individual work

ad 1) Principles of creating a website

ad 2) Creation of HTML document via the WYSIWYG editor NVU

ad 3) Direct editing HTML or XHTML via the editor PSPad

ad 4) Cascade Style Sheets (CSS) and the editor TopStyle Lite

ad 5) Creation of forms and their processing by scripts - JavaScript

ad 6) Displaying of the website on a web server, meta information, catalogues

ad SP) Individual work on final projects

Requirements needed in order to complete the subject

The exam has two parts, written (a test of the basic knowledge about the creation of the web) and oral. The oral exam is a public defence of the website students created. The topic has to be related to the study field of its author or to their long-term provable interest. The extent and the content of the work are further described in the paragraph below. A public defence of students’ works will take place in the last study week of the given semester in the presence of students enrolled in a particular subject and possibly other people interested.

Basic (no need to discuss it with the tutor)

Web presentation – enclosed group of hypertext documents with a simple structure (either linear or hierarchic one) created in any WYSIWYG editor (for example NVU, MS FrontPage), or via direct editing HTML (XHTML). Total extent 7 to 10 pages of smaller extent. Pages have to be related, they have to have a uniform navigation (control) and a uniform colour scheme. Part of the website has to contain graphical objects (photos, drawings, logos etc.). The website should be related to one studied subject or provable (i.e. it is necessary to support it with evidence) long-term interest of the student. The website has to be valid when examined by the validator of the consortium W3 … http://validator.w3.org/ and has to be published on a web server.

The following four pages have to be included within the whole extent of the website:

● index.html … Main page of the presentation with a bold headline – a title and if possible also with a simple logo or motto.
● contact.html … A page with the author’s email address in the form of a picture in the GIF or PNG format as a protection from a spam.
● gallery.html … A gallery of several photos, when you click on the size reduced picture you will get the picture in the full size.
● links.html … A page with few external links to other websites which deal with similar
topics as student’s final task. These are the minimal requirements you can of course be more initiative.

**Alternative (no need to discuss it with the tutor)**
Alternatively, you can also create a computer presentation (in the OpenOffice.org Impress or MS PowerPoint program) which you then export in the PDF format. You can then create a valid website for your PDF presentation, index.html, contact.html and links.html (as above) and publish the final result on a web server.

**Alternative (you need to discuss it with the tutor)**
After a consultation with your tutor you can also submit your work of a similar extent in a different software. For example it can be a geometric construction in the GEONExT program prepared by a student of the physics on the topic geometric optics. Constructions will represent projection with lenses and mirrors, astronomical telescope and a microscope.
The meaning of the icons in the text

Goals
There is a list of goals at the beginning of each chapter.

Terms to remember
The list of important terms and main points which should not be omitted when studying a certain topic.

Note
There are less important or specifying information in the note.

Control questions
These questions test to which extent has the student understood the text and the problems and remembered essential and important information.

Summary
The summary of the topic.

Bibliography
The bibliography used in the study material for supplementing and broadening the knowledge.
Content

1 The history of the internet.................................................................................................................. 8
  1.1 The beginnings of the internet....................................................................................................... 8
  1.2 The Czech internet ......................................................................................................................... 8
  1.3 Peering........................................................................................................................................... 8

2 Model ISO/OSI, TCP/IP, basic protocols............................................................................................ 9
  2.1 Reference model ISO/OSI................................................................................................................ 9
  2.2 Architecture TCP/IP...................................................................................................................... 10

3 Domain Name System, DNS................................................................................................................ 11
  3.1 Basics of the operation.................................................................................................................. 11
  3.2 DNS servers (name servers).......................................................................................................... 11
  3.3 Root servers................................................................................................................................... 12

4 The basics for creating a website.......................................................................................................... 13
  4.1 What is HTML?............................................................................................................................. 13
  4.2 What is needed in order to create a website?................................................................................. 13
  4.3 HTML editors - programs ............................................................................................................ 14
  4.4 General HTML syntax ............................................................................................................... 14
  4.5 Uploading a website on the internet............................................................................................. 15
  4.6 Few tips that will make your work easier ..................................................................................... 15
    4.6.1 Htm or html suffix?.................................................................................................................. 15
    4.6.2 Names of the files ................................................................................................................... 15
    4.6.3 Directories (or files)............................................................................................................... 15
    4.6.4 Relative and absolute link (addresses).................................................................................. 16
    4.6.5 Starting file ........................................................................................................................... 16

5 Creation of the HTML document....................................................................................................... 17
  5.1 Creation of the HTML document.................................................................................................. 17
  5.2 Line wrap and paragraph.............................................................................................................. 19
  5.3 Insertion of pictures...................................................................................................................... 19
    5.3.1 Graphics in the HTML documents ....................................................................................... 19
    5.3.2 Hypertext graphics.................................................................................................................. 21
    5.3.3 Positioning of pictures ........................................................................................................... 21
  5.4 Insertion of tables......................................................................................................................... 22
    5.4.1 Use of the tables..................................................................................................................... 23
  5.5 Lists................................................................................................................................................ 23

6 Cascade styles CSS............................................................................................................................ 24
  6.1 When to use CSS.......................................................................................................................... 24
  6.2 The outline of the possibilities of CSS......................................................................................... 24
  6.3 Three uses of CSS......................................................................................................................... 24

7 PHP – dynamic WWW pages.............................................................................................................. 26

8 Common editorial systems on the basis of the PHP language.......................................................... 28
  8.1 Basic functions of the CMS.......................................................................................................... 28
  8.2 WordPress...................................................................................................................................... 28
  8.3 Joomla!............................................................................................................................................ 29
  8.4 Drupal............................................................................................................................................ 29

9 Individual work on final projects...................................................................................................... 30
1 The history of the internet

Goals
You will get information about the history/founding of the world and the Czech internet.

Terms to remember
- Time and place of the founding of the internet
- The beginnings of the Czech internet
- Peering, NIX

1.1 The beginnings of the internet
The history of the internet is tied to the creation of computers (after 1945) and computer networks which enabled their mutual communication. The first functional network was built in the USA by the DARPA agency. This network was decentralized so it could function even after a power failure in some of its parts.

1.2 The Czech internet
The history of the Czech internet goes back to 1990, to a former Czechoslovakia. Among the first networks in Czechoslovakia was the network EUnet which interconnected particularly the Unix stations. Afterwards there was so called EARN (European Academic and Research Network) network which was already run on non-switched circuits. The first node of this network was a central computer in Oblastní Výpočetní Centrum (OVC) ČVUT in Prague connected to Austrian national intersection in Linz. The operation of this network was provided by CESNET (Czech Educational and Scientific NETwork).

1.3 Peering
There were many primary providers on the market who offered a connection of their own line abroad. In case that two local users connected to different providers wanted to communicate, their communication had to go through the line of one provider abroad and through the line of the other provider then back. That is why the need for so called peering – direct connection of networks – was discussed from the very beginning.
Goals
The basics of communication on the internet.

Terms
- Model ISO/OSI, 7 layers
- TCP/IP, 4 layers
- Protocols
- Application protocols (services)

2.1 Reference model ISO/OSI
The reference model ISO/OSI was developed by the ISO organization as a part of the effort to standardize computer networks called OSI and was accepted as the international norm in 1984. The reference model ISO/OSI is used as a clear example of a solution of communication in computer networks thanks to a multi-layered model where each layer is independent and easily replaced.

Each of the seven layers is responsible for clearly defined functions necessary for a communication. Each layer uses services of its neighbouring lower layer. Then it provides services to the neighbouring upper layer.
Architecture TCP/IP

A communication between the same layers of two different systems is operated by a communication protocol while using the connection created by the neighbouring lower layer. Architecture enables the exchange of the protocols of one layer without any impact on other layers. As an example can be the possibility of communication via various physical mediums – an Ethernet, an optic fibre or a serial line.

Architecture TCP/IP is divided into four layers (unlike the reference model OSI with seven layers):
- application layer
- transport layer
- internet layer
- network interface
### Domain Name System, DNS

#### Goals
DNS, the translation of domain names and IP addresses in the network

#### Terms
- Domain name
- Servers DNS

### 3.1 Basics of the operation
The use of names is much more convenient than the use of numbered IP addresses. There was a need to implement this change at the very beginning of the internet.

The space of the domain names comprise of a tree with one root. Every node of this tree contains information about a part of the name (domain) which is assigned to it and links on its subordinate domains. The root of the tree is so called the root domain which is defined as a dot. Below the root domain there are so called Top-Level Domains (TLD) in a hierarchical manner. They can either be thematic (com for commerce, edu for educational institutions etc.) or national (cz for the Czech republic, sk for Slovakia, jo for Jordan etc.).

The advantage of this organization lies in the option to divide the zone and to give its administration to someone else. Newly created zone will then become authoritative for the assigned name space.

### 3.2 DNS servers (name servers)
DNS server can play one of the three parts towards the domain (to be more specific towards the zone but these terms are in most cases interchangeable):

- **Primary server** is the one where all the data are created. If you need to change something in the domain you need to edit data on its primary server. Each domain has one primary server.
- **Secondary server** is the automatic copy of the primary server. This server continuously updates the data and serves as a backup in case the primary server fails and it is also used for data load distribution. Each domain has to have at least one secondary server.
- **Supporting (caching only) servers** serve as a compensatory memory for lowering the data load of the whole system. It stores the answers and provides them when repeating inquiries unless their “time to live“ (TTL) is up.

The answer coming directly either from the primary or secondary server is authoritative therefore it is considered correct. On the contrary, the answer from the compensatory memory is not authoritative. A client can ask for the authoritative answer but in most cases any answer is good enough.
3.3 Root servers

Root name servers represent a basic part of the technical infrastructure of the internet. Reliability, correctness and safety of the internet operations are dependent on the root name servers. These servers provide a root zone file to the other DNS servers. They are a part of the world wide distributed database which is used for the translation of unique domain names to other identifiers.
4. The basics for creating a website

Goals

- Gain basic knowledge about HTML
- You will know what you need in order to create a website
- Formats of the graphics
- Basic rules and principles for creating a website
- Placement of a website on the internet

Terms to remember

- HTML
- Tags
- Absolute and relative URL address

4.1 What is HTML?

The acronym HTML means Hypertext Markup Language. It is a language used for creating documents which defines the visual aspect of the text (the size of headlines, used fonts, margins etc.). HTML language was specifically developed (and is continuously being developed) for the purpose of publishing documents on the World Wide Web (WWW).

4.2 What is needed in order to create a website?

What is needed:

- A computer with some text editor, for example the Notepad or the PSPad editor (http://www.pspad.com/).
- WWW browser (Internet Explorer, Google Chrome, Mozilla Firefox, ...)

What is not needed:

- Functional internet connection.
- Expensive or complicated programs.
- The knowledge of programming.

The internet page is a file with a suffix htm or html. It is an ordinary text file which contains signs of the HTML language (tags).

There are two important steps you need to do in order to create a website:

1) You need to write the website, i.e. create the htm file (or more interconnected files) in the text editor
2) Upload these files on the internet

There are two ways how to open the website:
Either you can look at it in the browser in which case it is not possible to edit it.
Or you can open it as a text (for example in the Notepad) which is called the source, code or the source code and which is possible to edit.

4.3 HTML editors - programs

HTML editors are programs used to create websites. The WYSIWYG editors in which you write the text and you do not need to worry about the html code are especially useful for the beginners. Editor itself fills in the html code.

Commonly used WYSIWYG editors are for example Dreamweaver, FrontPage and NVU. WYSIWYG editors however do have their drawbacks. The main one being the debatable quality of the created code.

Apart from the WYSIWYG editors there are also structural editors. In these editors you write the HTML code directly but the program helps (gives clues, complements etc.). Structural editors are for example HomeSite, PSPad, EasyPad, UltraEdit, jEdit.

An example of the HTML code in the structural editor:

```
1 <html>
2 <head>
3  <title>Ma první stránka</title>
4  </head>
5 </body>
6  Moje první html stránka.
7  A nějaké další texty.
8 </body>
9 </html>
```

Figure 4.3 - PSPad

4.4 General HTML syntax

HTML language has a few principles which are worth mentioning:
- It is not case sensitive <body> is exactly the same as <BODY>.
- However, in the addresses and names of the files it is case sensitive and there cannot be any space or the Czech language.
- Tags that the browser does not know as if did not exist.
- There cannot be a space at the beginning of a tag, for example < br> is wrong.
Two or more consecutive spaces have the same meaning as one space.
The end of the line in the source code is regarded as a space.
It is always good to put the values of the attributes into inverted commas. For example the link
\<a href=www.seznam.cz\>seznam\</a\> works but it is better to write
\<a href="www.seznam.cz"\>seznam\</a\>.
Special characters such as © are inserted into the source code as a sequence of the characters &something; for example &copy:, non-breakable space is &nbsp;
A comment is inserted in the source code in between the characters "<!-- a -->". This is a comment that will not show. -->

4.5 Uploading a website on the internet
If you already have the website ready on the disc (html files) then you need to upload it on the internet. You can either be a host for free or you can order a paid hosting (usually with the second level domain).

Free
- Use services of public servers, free hosting.
  The first one was Geocities. Nowadays there are lots of similar servers (e.g.:
- Use servers from your employers or school.

Paid hosting
If you want to have a second level domain you always have to choose this option.

4.6 Few tips that will make your work easier

4.6.1 Htm or html suffix?
It is recommended to use the suffix html. Html suffix is the norm. Try to make your own convention so you do not get confused.

4.6.2 Names of the files
When naming HTML files follow these principles:
- Do not use Czech language in the names of the files and directories. Although it might work somewhere, it will not work everywhere. Avoid naming a website kocicka.html. It is always better to use kocicka.html.
- Do not use spaces. If there is a link to a file with a space it has to be replaced in the address with the string "%20". For example the file hezke odkazy.html is being linked as hezke%20odkazy.html. However, more suitable is to rename the file and replace the space with a character -, for example hezke-odkazy.html.
- You cannot use the following characters in the name of the file / \ * : ? # < >.
- It is recommended to write names of the files in lower case. The address of the file is case-sensitive therefore the case of the letters matters.
- Use names which indicate what the file contains. More understandable than sdsfo24x.html is kapitola1.html.

4.6.3 Directories (or files)
It is convenient to save the files on the web into directories. All the files in the server
should be in one directory and its subentries. This makes the structure of the web well-arranged.

Recommendations:
- Same principles apply for the folder name as for the name of the files: without spaces, diacritic, etc., preferably in lower case. The name of the directory is a part of the URL address.
- Do not create structures that are too nested.
- You can leave all the files in one directory, it is not a mistake.
- Pictures are usually in a special directory, for example "pictures".

4.6.4 Relative and absolute link (addresses)
When you enter an absolute address in the browser the web page will show. It is so called URL. A relative address directly contains the name of the file placed in the same directory (folder) or the name of the directory and then the name of the file. Therefore the relative address depends on the place it is used from.

This is for example the absolute address http://www.nekde.cz/soubory.htm. From the main page www.nekde.cz to the file leads a relative link written down as <a href="soubory.htm">. If a browser finds a relative link it automatically adds a path to the current page in front of it (which makes it an absolute link).

4.6.5 Starting file
Each server has a set name of the starting file. The most commonly used one is index.html. This is how you name your main page file.
5 Creation of the HTML document

Goals
- An overview of the basics for creating the HTML document
- Insertion and work with pictures
- Creation of tables

Terms to remember
- Tag (command)
- Attribute
- URL address

5.1 Creation of the HTML document
The HTML document is in principle created by two basic components – by displayed text and by commands (tags). Displayed text is the text which will be displayed. Tags are the operating elements of the document and determine what will the text and other media components (pictures, animation, sounds etc.) do.

Some tags are obligatory and create a basic frame for every HTML document. The whole document has to be enclosed between tags `<HTML>` and `</HTML>`. Apart from this, every document also needs to have its header which is marked with tags `<HEAD>` and `</HEAD>`. It is convenient to enter the title of the document in the heading part of the document via tag `<TITLE>` and `</TITLE>`. The content of the document is then entered in the area limited by tags `<BODY>` and `</BODY>`.

The format of tags is usually the following:
```
<keyword>Any text</keyword>
```

Tag is marked by an opening tag and is closed by the same tag which however contains an extra forward slash – its pair closing character.

This is what the basic structure of the HTML document looks like:
<html> type of the document
The tag is placed both at the beginning and at the end of the HTML code and is used to determine the type of the document.

<head> header
The tag <head> contains all the information about the document such as <title>, <base> or <meta>. It does not contain any text that is a part of the document (compare: tag <body>). Tag <head> follows the tag <html> and precedes <body>.

<title> title of the document
This tag contains a complete one-lined title of the document (which you can see in the browser window at the top) and it should be placed between the <head>.

<meta> metadata
The function of this tag is to provide information regarding the document which is not defined by other elements. Servers can use this kind of information for identification and cataloguing. The majority of the search engines use the tag <title>, description and keywords.

Name= description, keywords
Will tell the browser if the attribute Content contains a description of the document or a list of keywords.

Content= text or a list separated by commas
According to the value of the attribute Name the attribute Content offers a description of the document or a list of keywords, separated by commas.

<body>
A tag <body> contains all the information which is a part of the document. <body> should be placed right after the closing tag </head>. The closing tag </body> should be
placed at the end of the whole document before the tag </html>.

5.2 Line wrap and paragraph
<br> line wrap
This tag cause the wrap of the text, i.e. the following text will appear on the next line.
Example:

hello<br>good morning

Result:
hello
good morning

<p> paragraph
The tag for a paragraph will skip one line before the following text appears. The end of the paragraph can be, but does not have to be, closed by the closing tag </p>.

Example:

...and the prince kissed the princess.<p>And they lived happily ever after.

Result:
...and the prince kissed the princess.

And they lived happily ever after.

5.3 Insertion of pictures

5.3.1 Graphics in the HTML documents
The HTML language enables to insert pictures directly into the document via the tag <IMG>.

Pictures are not a part of the source code of the web page but they are saved somewhere on the server. Mostly in the following graphic formats GIF, JPEG, PNG. The name of the file with a picture is inserted into the document via SRC attribute.

Example can be the following

<IMG SRC="pictures/grafika.gif">

Non pair tag which inserts a graphic object into the document:

<src> URL of the graphic object which should be inserted into the text
**lowsrc** URL of the graphic draft (browsing program will transfer the draft of the graphic, complete the whole document and only then will transfer the full versions of the graphic determined by attribute *src*)

**Alt** alternative text which is displayed in case the browsing program cannot (or must not) display the graphics

**align** The placement of the graphic object against the surroundings

- **top** the top edge of the graphics will be lined up with the top edge of the line
- **middle** the middle of the graphics will be lined up with baseline
- **bottom** the bottom edge of the graphics will be lined up with the bottom edge of the line
- **left** the horizontal placement on the left edge; text wraps the graphics form the right
- **right** the horizontal placement on the right edge; text wraps the graphics from the left
- **texttop** the top edge of the graphics will be lined up with the top edge of the text on the line
- **absmiddle** the middle of the graphics will be lined up with the middle of the line
- **baseline** the bottom edge of the graphics will be lined up with the baseline
- **absbottom** the bottom edge of the graphics will be lined up with the bottom edge of the line

**width** the required width of the graphics (the browser should increase/decrease the graphics in a way so that its width when displayed is equal with the number of pixels)

**height** the required height of the graphics

**vspace** defines how much space will be left out around the graphics in the vertical direction

**hspace** defines how much space will be left out around the graphics in the horizontal direction

**units** unit (implicitly pixels)

**border** the width of the border created around the graphics presents a link;

- **border=0** there will be no border displayed

**ismap** denotes that when you click on the mouse in the area of the graphics the browser should pass the coordinate of this area to the CGI anchor script

**usemap** URL with the definition of sensitive areas in the picture (MAP element)
5.3.2 Hypertext graphics
Similarly as a text, within the container of the tag <A> (hypertext links) there can be placed also pictures. In that case it is so called "hypertext graphics". This way you can very easily implement graphic menus.

```
<BODY>
<A HREF="#content">
<IMG SRC="pictures/obsah.gif"></A>
<A HREF="#img">
<IMG SRC="pictures/img.gif"></A>
</BODY>
```

Final result:

![Obsah Příkaz IMG](pictures/obsah.gif)

Figure 5.4.2.1 - resulting processing

**Note:** You can achieve a better effect by positioning this menu to the center and disabling the border around the hypertext graphics (BORDER=0)

```
<BODY>
<CENTER>
<A HREF="#content">
<IMG SRC="pictures/obsah.gif" BORDER=0></A>
<A HREF="#img">
<IMG SRC="pictures/img.gif" BORDER=0></A>
</CENTER>
</BODY>
```

Final result:

![Obsah Příkaz IMG](pictures/obsah.gif)

Figure 5.4.2.2 - resulting processing

5.3.3 Positioning of pictures
It is possible to position pictures within the line of the text. In this case you can manipulate with the picture just like with any other character of the text. The height of the picture for example defines the height of the whole line. In the standard case it is not possible to have more lines of the text next to one picture. When you want to position picture within the line of the text use the attribute ALIGN.
5.4 Insertion of tables

Tables are a powerful tool in the HTML language. By using tables you can not only print a very well-arranged columns of various figures but also create a professionally looking web page thanks to different table „tricks“.

The definition of the table is defined by the tag <TABLE> and </TABLE>. The attribute BORDER of this tag makes the table visible. Same as with the forms, in this case it is also necessary to use other tags in the container of the table which enable to specify the final appearance of the whole table. The tag <TR> defines the row of the table which contains the tag (usually several) <TD> representing the columns of the data. The tag <TH> (table header) defines the heading of individual columns of the data. It is possible to add a caption to the table via character <CAPTION> which will be lined up with the centre of the table.

Example of using a table:

```html
<TABLE BORDER>
  <CAPTION>Specification of the names of the HTML teaching files</CAPTION>
  <TR>
    <TH><PRE></PRE></TH> <TH>Alena</TH> <TH>Lada</TH> <TH>Karel</TH>
  </TR>
  <TR><TD>Windows</TD> <TD>html_wa.zip</TD></TR>
  <TR><TD>html_wl2.zip</TD> <TD>html_wk.zip</TD></TR>
  <TR><TD>Unix</TD> <TD>html_ua.tgz</TD></TR>
  <TR><TD>html_ul2.tgz</TD> <TD>html_uk.tgz</TD></TR>
</TABLE>
```
Result:

<table>
<thead>
<tr>
<th>Specification of the names of the HTML teaching files</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
<tr>
<td>Windows</td>
</tr>
<tr>
<td>Unix</td>
</tr>
</tbody>
</table>

5.4.1 Use of the tables

Table tags are not currently used for aligning the numbers but for the complete adjustment of the HTML page. You can for example place pictures into various columns of the table which will be evenly positioned regardless of the width of the browser window.

Thanks to using tables you can also make the forms noticeably neater and better arranged. Check boxes can be placed in one table cell while their caption is in the next column.

5.5 Lists

HTML language contains several types of the lists. The first one is unnumbered list <UL>. The individual list items are preceded by opening tag <LI> (list item). Items are then automatically indented and there is a bullet point before them. The tag <LH> is used to define the heading (title) of the list.
6 Cascade styles CSS

Goals
An overview about the basics of formatting documents via CSS.

Terms
- CSS
- Cascade style and its way of notation.

CSS is a collection of methods for formatting websites. This acronym means Cascading Style Sheets. We use the term cascade styles because you can stratify style definitions.

6.1 When to use CSS
- when you need to place some formatting element on the website which cannot be done in a standard way,
- when you often write texts which are meant for the internet and you do not want to waste time with formatting,
- when you administrate (or are planning to administrate) a bigger website with many web pages which could look similarly

6.2 The outline of the possibilities of CSS
The main meaning of the CSS lies in the fact that they function automatically while the visual aspect of the whole website is simply defined by one file, for example:
- Set any or a specific type size, interleaving and caps
- Indent the first line of the paragraph, enlarge the line height
- Delete or enlarge the empty space left after the paragraph
- Automatically format the headings (for instance make them all green)
- Highlight the links when passed over by the mouse
- Automatically do the graphical bullet points
- Make some parts of the text invisible, transparent or do not display them at all
- Redefine the graphical meaning of the commonly used tags (for example everything that is in italics, make also bold)
- Set the background of anything, a page, a table or even of a paragraph; the background does not have to recur and it could have a specific position
- Place any object (for example a part of a text) anywhere on the page

6.3 Three uses of CSS
A style can be defined in three different ways:
- Directly in the source code by the formatted element via attribute style="...". This is what I call a direct style.
- Via "stylesheet" in the page header. Stylesheet is a list of styles. There you find
information what should be formatted and in what way, for example that headings should be green. In the web page you write the stylesheet in between the tags `<style>` and `</style>`.

- By using the external stylesheet - which is the file *.css, to which a web page refers to via tag `<link>`. The stylesheet is placed in the file. The main advantage is that you can link one file to many web pages so all of them then look similarly.

It is enough to be able to operate one of the ways. External css file is the one most commonly used.
Goals
Basic information about the PHP programming language

Terms
- Server script
- Dynamic web
- PHP

PHP is a server script language used when creating dynamic web pages. PHP is short for Hypertext Pre-processor.

Each web page (file) which contains PHP scripts has to have a suffix .php. This suffix does not affect in any way other scripts which are already on the web page. PHP can be added anywhere in the code. However, you do have to separate it from the code. To separate the PHP from the HTML code use the following directives `<?` and `?>`. Most commonly used are the directives `<?php` and `?>`.

Advantages of PHP:
- very effective and elegant processing of the forms, data and email
- the ability to access both files and cookies
- cooperation with database (+SQL)
- very wide possibilities of use

Disadvantages of PHP:
- a relatively complicated installation on the server
- complexity

A simple example of use:
Na této stránce již bylo 1962 lidí.

ZDROJ:

```php
<?php
if(file_exists("pocet_priklad.txt"))
{
    print "";
}
else
{
    $file=fopen("pocet_priklad.txt","a");
fputs($file,"0");
fclose($file);
}
$soubor="pocet.txt";
$file=fopen("$soubor","r+");
$pocet=fgets($file,100);
$pocet++;
fseek($file,0);
fputs($file,$pocet);
fclose($file);
echo "$pocet";
?>
```

Figure 7 - PHP
8 Common editorial systems on the basis of the PHP language

Goal
An overview of selected editorial systems.

Terms
- CMS
- WordPress
- Joomla!
- Drupal

CMS (content management system) is a software providing the management of the documents, most often of the web content.

CMS can sometimes be called editorial or publishing system.

8.1 Basic functions of the CMS
Among the basic functions of the CMS (usually divided into administrator and user one) belong:
- a creation, a modification and publishing of documents (articles), usually via web interface, often by utilizing a simple online WYSIWYG editor or a simple system of text formatting (no need to know HTML),
- a management of the access to documents, usually with the administration of the users and access rights, often with the workflow or groupware functions,
- a management of either published or general discussions or comments
- a management of the files,
- a management of the pictures or galleries,
- a calendar,
- the statistics of the traffic.

The market with the CMS programs is wide. There are a lot of programs as a free software as well as commercial ones. Most commonly used are WordPress, Joomla! and Drupal.

8.2 WordPress
WordPress has its roots as a blogging platform. Currently, there are a lot of non-blogging websites which include everything from simple few-pages catalogue websites to full social networks.
8.3 Joomla!
Editorial system Joomla! was used almost for everything besides ordinary websites. From systems for administration of properties, booking systems to complete company directories.

8.4 Drupal
Drupal is a very popular editorial system. It contains many functions for creating internal and external websites and many tools for organizing the content.
Create a web presentation in the extent of five mutually linked HTML pages (with pictures) on one of the following topics:

- my hometown
- a simple catalogue with products and its description ("minishop")
- CV including photographs, hobbies, favourite books etc.
- a favourite music band, a singer
- a favourite movie
- a hotel with the offer of the accommodation
- a restaurant including the menu
- a pet
- my holiday
- a sports team
- a small recipe book
- a computer game - review

Please discuss other topics with your lecturer.
Bibliography


