

# INFORMATIKA 9

## Obsah

---

Řád učebny, bezpečnost práce, učebnice, sešity, plán práce, přihlášení do sítě, plocha účtu,.....	2
Opakování 7. a 8. třídy .....	2
EXCEL .....	3
Druhy buněk .....	3
Grafická úprava buňky.....	5
VZORCE – ZADÁVÁNÍ .....	5
Úvod do sítí LAN .....	6
Teorie počítačové grafiky .....	7
Pokročilejší úpravy fotografií.....	8
Koláže .....	9
Animace .....	9
Tvorba www stránek .....	9
Webové stránky mohou obsahovat: .....	10
Uložení souboru .....	10
Tvorba www pomocí HTML jazyka .....	10
Tagy .....	11
Desatero zásad pro dobrý web: .....	11
Kostra stránky.....	12
Nadpisy .....	13
Tučnost, kurzíva.....	13
Font písma .....	14
Barva písma .....	14
Otevření odkazu v novém okně .....	15
Obrázky na webu .....	15
Neuspořádaný seznam .....	16
Uspořádaný seznam .....	17
Tabulka .....	17
Bublina na přejetí myší.....	18
Anketa, počítadlo, diskuse, návštěvní kniha .....	18
Validátor W3C .....	18

## **Řád učebny, bezpečnost práce, učebnice, sešity, plán práce, přihlášení do sítě, plocha účtu,...**

---

Desky na vytvořené práce

### **Opakování 7. a 8. třídy**

---

Založíme si fiktivní firmu (např.: obchod s květinami, kurýrní a doručovatelská služba, detektivní kancelář, cukrárnu a kavárnu, prodejnu automobilů, cestovní kancelář,...)

Nejdříve vymyslete její jméno a náplň činnosti (Word)

Navrhněte její logo (Malování)

Navrhněte vizitku pro jejího ředitele (Word nebo Callisto) – velikost 5 x 9 cm

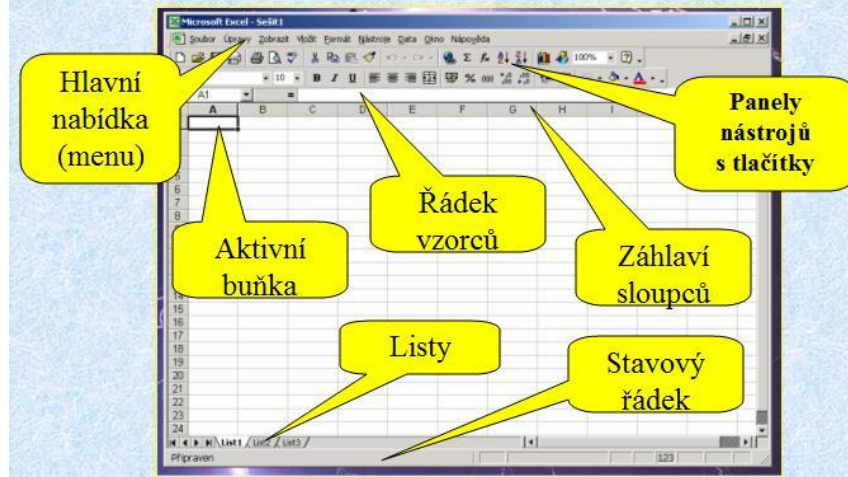
Vytvořte inzerát do novin (Malování, Word nebo Callisto) – velikost 12 x 18 cm

Připravte 2 - 4 minutovou prezentaci (automaticky přehrávanou) do regionální televize, ve které svoji firmu co nejpoutavěji představíte

## EXCEL

---

**Užívá se hlavně v ekonomických oblastech pro tvorbu tabulek a grafů.**



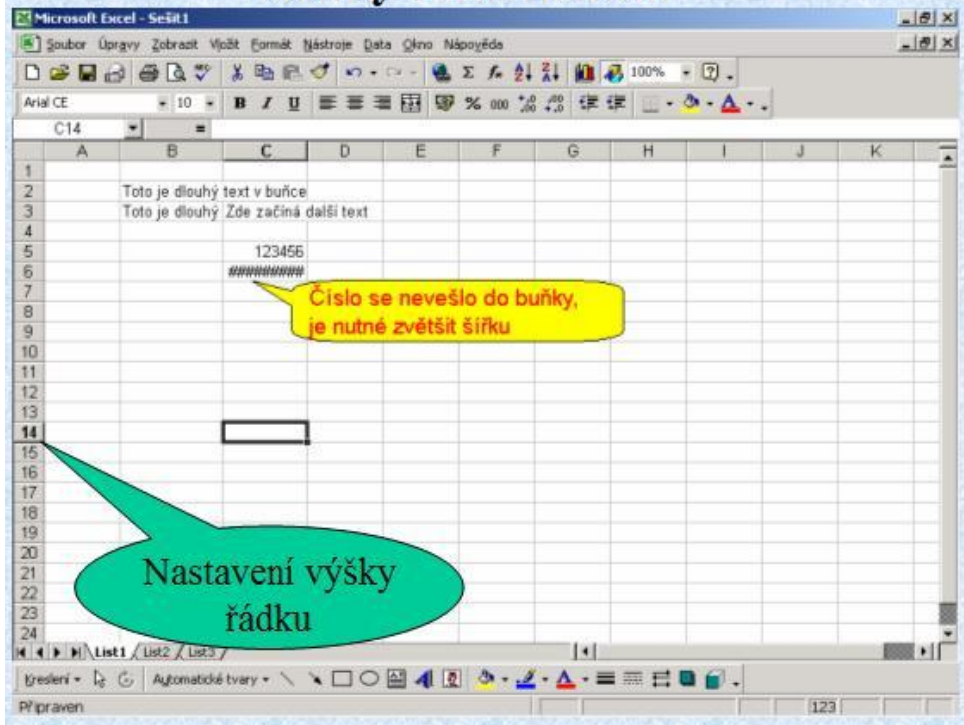
**Buňka** - označení písmenem ve svislém směru a číslem ve vodorovném směru, např.: **C4**. Počet buněk v Excelu – několik milionů.

### Druhy buněk

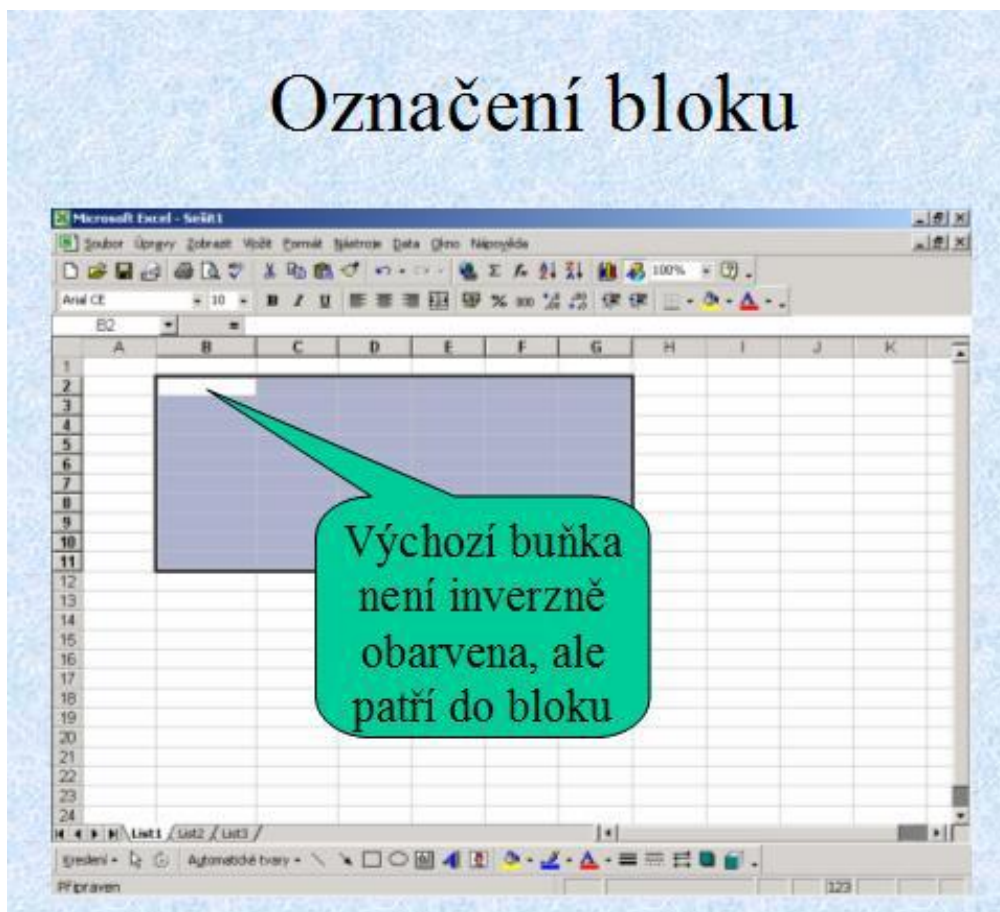
---

- **řetězec** (text) – může obsahovat cokoliv (číslo, text, znaky), např.: **250 tun**
- **číslo** - obsahuje číslo, ne text! Např.: **121,50**
- **vzorec** – vždy začíná znakem =, např.: **=C1+C2**
- **datum** – v předem stanovené formě zápisu, např.: **21. 05. 2004**

## Dlouhý text v buňce



## Označení bloku



## Oprava obsahu buňky

---

1. Pomocí klávesy F2
2. Dvojklik v buňce
3. V řádku vzorců

Vymazání obsahu buňky - pomocí klávesy Del

## Grafická úprava buňky

---



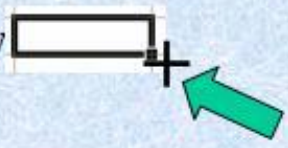
Každá buňka může mít jinou velikost písma, ohraničení, zarovnání, ...

**Volba pomocí tlačítek na panelu nástrojů nebo ve FORMÁTu – BUŇKY**

1. Číslo – druh (měna, datum, vlastní, ..), des. místa, symbol, ...
2. Zarovnání
3. Písmo
4. Ohraničení
5. Vzorky
6. Zámek

## VZORCE – ZADÁVÁNÍ

---

- a) Přesně zapíšeme vzorcem, např.:  $=(C2 +C3)*C4$
- b) Použijeme průvodce (u složitějších funkcí),  např.: statistické – průměr, logické – když, ....
- c) Suma  - tlačítko na panelu nástrojů
- d) Kopírováním – přes schránku nebo tahem myši v pravém spodním rohu buňky 

## Úvod do sítí LAN

---

Local Area Network (též LAN, lokální síť, místní síť)

označuje počítačovou síť, která pokrývá malé území (např. domácnosti, malé firmy).

přenosové rychlosti jsou vysoké, řádově Gb/s.

nejrozšířenější technologií v dnešních LAN sítích je Wi-Fi

Slouží ke snadnému sdílení prostředků (diskového prostoru, tiskáren, připojení k Internetu)

Sítě LAN označují všechny malé sítě, které si mnohdy vytváří sami uživatelé na své vlastní náklady. Jedná se o sítě uvnitř místností, budov nebo malých areálů; ve firmách i v domácnostech. Dále je charakterizuje levná vysoká přenosová rychlost (až desítky [Gbps](#)) a skutečnost, že si je na vlastní náklady pořizují sami majitelé propojených počítačů.

Síť se skládá z aktivních a pasivních prvků. Aktivní prvky se aktivně podílejí na komunikaci. Patří mezi ně například switch, router, síťová karta apod. Pasivní prvky jsou součástí, které se na komunikaci podílejí pouze pasivně (tj. nevyžadují napájení) – propojovací kabel, konektory, hub.

Opačným protipólem k sítím LAN jsou sítě WAN, jejichž přenosovou kapacitu si uživatelé pronajímají od specializovaných firem a jejichž přenosová kapacita je v poměru k LAN drahá.



## Teorie počítačové grafiky

---

Existují dva základní typy: [vektorová](#) a [rastrová grafika](#).

### Vektorová grafika

- ukládá přesná geometrická data, například souřadnice bodů, propojení mezi body (úsečky a křivky) a vyplnění tvarů
- vektorový obrázek je složen ze základních geometrických útvarů jako jsou body, přímky, křivky a mnohoúhelníky

Výhody vektorové grafiky:

- Je možné libovolné zmenšování nebo zvětšování obrázku bez ztráty kvality
- Je možné pracovat s každým objektem v obrázku odděleně.
- Výsledná paměťová náročnost obrázku je obvykle mnohem menší než u rastrové grafiky.

Nevýhody vektorové grafiky:

- Oproti rastrové grafice zpravidla složitější pořízení obrázku. V rastrové grafice lze obrázek snadno pořídít pomocí fotoaparátu nebo skeneru.
- Překročí-li složitost grafického objektu určitou mez, začne být vektorová grafika náročnější na operační paměť a procesor než grafika bitmapová.

### Použití

Vektorová grafika se používá zejména pro počítačovou sazbu, tvorbu ilustrací, diagramů a počítačových animací. Pro práci s vektorovou grafikou se používají vektorové editory (např. Adobe Illustrator, CorelDraw, Zoner Callisto, atd).

Formáty

- .ai - Adobe Illustrator
- .cdr - Corel Draw
- .zmf - Zoner Callisto

### Rastrová neboli bitmapová grafika

- základem rastrové grafiky je pravidelná síť pixelů
- každý pixel nese specifické informace, například o jasu, barvě, průhlednosti bodu...

- obrázek v rastrové grafice má omezené rozlišení, které se udává počtem řádek a sloupců
- dnes se často kombinuje rastrová a vektorová grafika v souborových formátech jako **PDF**

### Výhody bitmapové grafiky

- pořízení obrázku je velmi snadné například pomocí fotografie nebo pomocí skeneru

### Nevýhody bitmapové grafiky

- velké nároky na úložný prostor (při vysokém rozlišení a barevné hloubce velikost obrázku dosahuje i jednotek megabytů, v profesionální grafice se běžně operuje i s podklady o desítkách megabytů)
- změna velikosti (zvětšování nebo zmenšování) vede ke zhoršení obrazové kvality obrázku (na výsledném obrázku je patrný rastr)

### Formáty

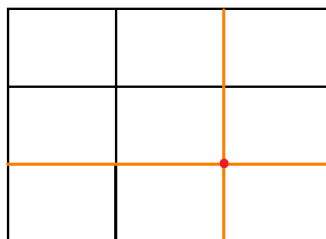
- BMP, GIF, JPG, PCX, PNG, TIFF

## Pokročilejší úpravy fotografií

---

– v programu Zoner Photo Studio a PhotoFiltre

- změna rozměrů a velikosti
- oříznutí
- pootočení
- zrcadlení, překlopení
- zlatý řez
- změna jasu a kontrastu
- barevný nádech – sepie, stará fotka, černobílá fotografie
- retuše (odstranění vad) – klonovací razítko
- rámečky a ohraničení
- odstranění červených očí
- olejomalba, reliéf
- bělení zubů





## Koláže

---



## Animace

---

- je způsob vytváření zdánlivě se pohybujících věcí.
- slovo pochází z cizího slova znamenajícího oživení
- animace se využívá mj. v animovaném filmu.

Princip animace je zaznamenání sekvence snímků, které jsou každý o sobě statický a drobně se od sebe liší. Při rychlém zobrazování těchto snímků za sebou vzniká díky setrvačnosti lidského oka dojem pohybu. Snímky se však musí přehrávat takovou rychlostí, kterou už oko nepostřehne.

Když zobrazíme těchto šest snímků vhodnou rychlostí za sebou, vznikne dojem pohybu.

Animace je i základem pro film, který využívá stejného efektu. Lidské oko má určitou setrvačnost, tzn. obraz zůstává po určitou dobu zaznamenán na sítnici. Díky tomu se dá vytvořit iluze spojitého pohybu pomocí rychle se střídajících obrázků. Film obvykle využívá 24 snímků za jednu sekundu. Je to frekvence, při které již lidské oko vidí spojitý obraz. Televize v normě PAL používá frekvenci 25 snímků za sekundu, v normě NTSC 29,97 snímků za sekundu.

## Tvorba www stránek

---

Co se dozvíte?

**jak webová stránka funguje**

**co je potřeba k jejímu vytvoření**

**jak stránky zviditelnit - publikovat**

Internet – **spousty PC navzájem propojených: servery a klientské PC**

**Na serverech jsou uloženy informace a klientské PC si je pomocí programu - prohlížeče (browseru) mohou prohlížet**

**Každá stránka na internetu musí mít svou jedinečnou adresu**

**Webmaster – autor web. prezentace**

Možnosti jak vytvořit web

na nějaké zdarma nabízené šabloně – bez nutnosti znát cokoli o tvorbě www, např. na e-stránkách, na blogu atd.

v redakčním systému – bez nutnosti znát cokoli o tvorbě www, vkládáte pouze neformátovaný text a obrázky, systém sám je upravuje pro použití na stránce pomocí editorů

WYSIWYG editory (vizuální) – přímo vytváříte webovou stránku, např.: ve Wordu, v MS FrontPage atd. (jednoduché, ale vznikají nepřesné zdrojové kódy)

Pomocí programovacích jazyků (např. HTML) – přímo se píše zdrojový kód, např. v Notepadu, PSPadu atd.

## Webové stránky mohou obsahovat:

---

- Texty
- Odkazy
- Formuláře
- Obrázky
- Tabulky
- Videá atd.

## Uložení souboru

---

**Formáty** (přípony) stránek – htm nebo html (lepší je jednu zvolit a držet se jí)

Název souboru nesmí obsahovat českou diakritiku, mezery a některé další znaky (tečka, dvojtečka, atd.)

Všechny soubory pro jeden web ukládat do stejné složky, vytvořit si podsložky, jako např. obrázky (images), privátní – styl nebo vzor a může se skrýt, atd. – pro přehlednost

Úvodní soubor pojmenovat index.htm nebo default.htm

## Tvorba www pomocí HTML jazyka

---

### Co budeme potřebovat

Programy:

Poznámkový blok (Notepad) neboť webová stránka má textový formát (vše je zapsané texty)

Prohlížeč, ve kterém si zkontrolujete, zda stránka odpovídá vašim představám

**Zdrojový kód** (recept, podle kterého prohlížeč vaří)

popisuje jak má stránka vypadat

je možné ho zobrazit v prohlížeči v menu Zobrazit – Zdrojový kód

píšeme ho značkovacím HTML jazykem, zápis obsahuje tzv. značky – tagy

## Tagy

---

párové, např.: `<b>Tento text bude zobrazen tučným písmem</b>`

nepárové ``

jdou do sebe libovolně vnořovat

často mívají různé atributy např. `<p align="right"> odstavec zarovnaný do prava</b>`

**HTML (Hypertext Markup Language)** je základním jazykem pro vytváření webových stránek a jeho zvládnutí je nutné i pro používání složitějších skriptovacích jazyků, jako je PHP či ASP. Spousta věcí se dělá pomocí CSS stylů namísto HTML. Pomocí CSS je to funkčnější, systematictější a asi i jednodušší než čistým HTML. U CSS se hodnoty nepíše do uvozovek.

Oficiální verzi jazyka HTML utváří konsorcium [W3C](#), neustále se vyvíjí

HTML je pružný – pokud prohlížeč narazí na chybný příkaz, přeskočí ho

### Volba prohlížeče

existuje jich spousta: MS Internet Explorer (90%), Opera, Mozilla, Safari, atd.

každý je zobrazuje trochu jinak (každý umí něco jiného lépe)

všechny jsou zdarma

Pomocníci na internetu – stránky věnované tvorbě www, např.:

<http://www.jakpsatweb.cz/>

## Desatero zásad pro dobrý web:

---

1. přizpůsobit svoji práci několika skutečnostem o čtenářích: nečtou dlouhé texty, nerolují, nad odkazem se zamyslí, zda stojí zato na něj kliknout
2. nemít odkazy na prázdné stránky (žádné Připravujeme, Ve výstavbě atd.)
3. aktuálnost – neustálá obměna podle nových skutečností
4. musí se dát jednoduše celé změnit, grafiku používat jen v omezené míře, ne jako nosnou část stránky, aby se rychle načetli
5. vyplatí se mít jasnou strukturu odkazů – na úvodní mít odkazy na všechny ostatní a na ostatních mít možnost návratu do úvodní
6. všechny informace max. po 2 – 3 odkazech
7. psát spisovně a respektovat typografická pravidla
8. používat nadpisy a styly
9. u dlouhých textů používat záložky
10. u odkazů nepsat *klikněte zde*, ale rovnou na slovo vyjadřující cíl udělat odkaz

## Kostra stránky

---

Nejdůležitějším a dá se říct, že jediným opravdu nezbytným meta tagem je meta tag pro kódování češtiny. Pokud ho nepoužijete, text stránek, resp. české znaky se budou zobrazovat jako všelijaké paznaky, čtverečky a podobně.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="cs-cz" lang="cs-cz">
```

### HEAD – hlava stránky

```
<head>
<meta http-equiv="Content-Type" content="text/html; charset=windows-1250" />
<title>Titulek stránky</title>
</head>
```

```
<body>
<p>Odstavec textu.</p>
</body>
```

```
</html>
```

Tak takto by měla vypadat kostra stránky. Soubor uložíme s příponou **htm** nebo **html**

### BODY – tělo stránky

#### Barvy stránky a odkazů

```
<body text="black" link="blue" alink="green" vlink="red">
```

(barva textu, odkazu a použitého odkazu)

```
<body bgcolor="#FFFF00" background="green">
```

(barva pozadí...)

```
<body style="background-color: green; color: yellow;">
```

#### přes CSS - zápis stylu

Tag <style> patří mezi tagy <head> a </head>:

```
<style type="text/css">
```

```
body {color: black; background-color: white; background-image: url("pozadi.gif");} /* text černý, barva pozadí bílá, obrázek na pozadí */
```

```
a:link {color: blue;} /* nenavštívený odkaz */
```

```
a:visited {color: navy;} /* navštívený odkaz */
```

```
a:hover {color: red;} /* odkaz, přes který se jede myší*/
```

```
</style>
```

**white** - bílá, **red** – červená, **yellow** – žlutá, **orange** – oranžová, **pink** – růžová, **brown** – hnědá, **grey** – šedá, **blue (#FFFF00)** – modrá

Nastavení okrajů

```
<body margin-left="50pt" margin-right="0pt">
```

nebo

```
< style type="text/css">
```

```
body{margin-left:0pt; margin-top:0pt (vzdálenost od horního okraje); margin-  
right:0pt;}  
</ style>
```

**style** - atribut tagu <body>, který určuje vzhled celého dokumentu. V tomto případě nastavuje barvu pozadí stránky (background-color) na zelenou a barvu textu (color) na žlutou. V zásadě se již jedná o použití kaskádových stylů = CSS. Kaskádové styly jsou základní prostředek na úpravu vzhledu stránek. Atributem style se dá měnit vzhled.

## Nadpisy

---

Existuje 6 úrovní nadpisů, označují se tagy *h1*, *h2*, *h3*, *h4*, *h5* a *h6*, kde 1 je úroveň nejvyšší. Úrovně se liší velikostí písma, všechny nadpisy jsou implicitně zarovnány vlevo.

```
<h1>Nadpis 1. úrovně</h1>
```

```
<h2>Nadpis 2. úrovně</h2>
```

```
<h3>Nadpis 3. úrovně</h3>
```

```
<h4>Nadpis 4. úrovně</h4>
```

```
<h5>Nadpis 5. úrovně</h5>
```

```
<h6>Nadpis 6. úrovně</h6>
```

### Zarovnání odstavce

Normálně se odstavce (i jiné tagy) zarovnávají doleva. Zarovnání jednoho odstavce doprava:

```
<p style="text-align: right">Odstavec zarovnaný doprava.</p>
```

Zarovnání na střed a do bloku:

```
<p style="text-align: center">Odstavec zarovnaný na střed.</p>
```

```
<p style="text-align: justify">Odstavec zarovnaný do bloku.</p>
```

Lepší je ale nastavit všechny odstavce najednou globálním stylem:

```
<style type="text/css">
```

```
p {text-align: center;} /* všechny odstavce budou zarovnaný na střed */  
</style>
```

## Tučnost, kurzíva

---

Normální písmo, <b> tučné písmo </b>, <i> normální kurzíva </i>.

```
<strong>také tučné písmo (zvýraznění)</strong>
```

```
<em>také kurzíva (zvýraznění)</em>
```

### Odstavce a vynucené zalomení řádku

```
<p>Odstavec zalamuje řádky podle šířky prostoru, který má (nejčastěji okna). Na konci odstavce se zalomí řádek.</p>
```

Je-li potřeba uvnitř odstavce zalomit řádek, použije se tag <br>, v normálních textech k tomu ale není důvod:

```
<p>Text odstavce plyne<br> a najednou se zalomí řádek. </p>
```

## Font písma

---

```
<font size="2" color="black" face="Arial">
aaaa
</font>
```

## Barva písma

---

```
<span style="color: red;">Červené písmo</span>
nebo třeba pro odstavec
```

```
<p style="color: blue;">Odstavec modrým písmem</b>
```

Dá se to napsat do libovolného tagu, ale nemusí to fungovat u odkazů (tag <a>), u nich se to musí udělat globálním stylem <style>, což je ale stejně lepší i v jiných případech:

```
<style type="text/css">
```

```
body {color: maroon; background-color: white;}/* hnědé písmo pro celý
dokument, barva pozadí bílá */
```

```
p {color: green;} /* zelené písmo pro všechny odstavce <p> */
```

```
.zlute {color:yellow;} /* pro všechny tagy s class="zlute" */
```

```
a:link {color: blue;} /* nenavštívený odkaz modrý*/
```

```
a:visited {color: navy;}/* navštívený odkaz tmavě modrý */
```

```
a:hover {color: red} /* odkaz, přes který se jede myší červený */
</style>
```

Odkazy

```
<a href="adresa">text odkazu </a>
```

text (nebo obrázek) mezi těmito dvěma tagy se zobrazí jako text odkazu ("a" jako angl. "anchor" = odkaz). Po kliknutí na něj se prohlížeč přepne na adresu, která je zadaná atributem HREF.

**href**

atribut tagu "a" se rovná cestě k odkazovanému souboru. Používá se zde buď relativní, nebo absolutní adresa.

*absolutní adresa*

používá se ve formě <http://www.atakdale.cz>. V příkladu je takto použita adresa Seznamu.

*relativní cesta*

pokud se chci odkázat na jiný svůj soubor, je zbytečné vypisovat tam to <http://> a celou cestu. Nejlepší je, když je odkazovaný soubor ve stejném adresáři jako ten aktuální. Pak se na něj dá ukázat relativně. Stačí zapsat jenom **jméno souboru** a prohlížeč to pochopí. (V příkladu je takto odkázaný druhý soubor prikklad2.htm.) Jestliže je ten cílový soubor v jiném blízkém adresáři, stačí napsat `href="adresář/soubor.htm"`. Pro nadřazený adresář se používají konvenční dvě tečky.

Záložka někde v dokumentu:

```
<a name="jmeno_zalozky"></a>
```

a takhle se na ni po kliknutí odroluje:

```
<a href="#jmeno_zalozky">odrolovat</a>
```

## Otevření odkazu v novém okně

---

```
<a href="http://www.webtvorba.cz" target="_blank">WEBtvorba.cz</a>
```

Pozn.: Atribut *target* není povolen v XHTML 1.0 Strict, v Transitional ano.

### Titulek odkazu

Každý odkaz by měl mít svůj titulek. Tím je text, který uživateli přiblíží, kam vlastně odkaz směřuje. Titulek se zobrazí při najetí kursoru myši na odkaz. Titulky odkazů jsou důležité také např. pro čtečí zařízení, umožňují lepší orientaci na webu.

```
<a href="http://www.webtvorba.cz" title="Úvodní stránka tohoto webu">WEBtvorba.cz</a>
```

Poznámka v HTML:

```
<!-- toto je poznámka v HTML -->
```

```
<span>, </span>
```

dvojice tagů vymežující nějak odlišný text.

```
<span style="color: red"> </span>
```

V tomto případě má obalený kus textu červenou barvu písma. Jde o první příklad tagu, který má tak zvaný *atribut*. Atributem je "style" a má hodnotu "color: red" (angl. barva: červená). Pomocí tohoto atributu style= se dá detailně nadefinovat vzhled textu, obaleného libovolným tagem.

## Obrázky na webu

---

Pro vložení obrázku slouží nepárový element **img**. Povinnými atributy jsou *src*, jehož hodnotou je název obrázku, případně i s cestou a *alt* obsahující alternativní text, který se zobrazí, pokud je obrázek prohlížeči nedostupný, buď proto, že se na zadané adrese nenachází nebo proto, že prohlížeč obrázky nezobrazuje. Alternativní text by měl tedy nějakým způsobem přiblížit co je na obrázku těm, kteří jej nevidí.

```

```

Obrázky na webu bychom měli na webu používat opatrně. Kromě výběru vhodných obrázků je třeba dbát na jejich datovou velikost, zdaleka ne každý si může dovolit prohlížet stránky, jejichž velikost se dá počítat v megabytech. Obrázky, které nemají jinou funkci než „zlepšení designu“ stránky by měli být vkládány pouze pomocí CSS stylu a to jako obrázek na pozadí.

### Velikost obrázku

Velikost obrázku není povinné zadávat, přesto je velkou chybou toto opomenout. Pokud totiž velikost obrázku nezadáte, prohlížeč si na jeho zobrazení nevyhraní dostatečný prostor a jelikož okolní text se samozřejmě načte dřív než obrázek (protože je co se týče velikosti dat menší), způsobí to jakési „poskakování“ stránky ve chvíli, kdy se začínají načítat obrázky, které potřebují více prostoru než mají a okolní text jim vlastně musí uhybat.

Stejně tak je vhodné zadávat obrázku jeho skutečnou velikost, nezmenšovat ani nezvětšovat pomocí prohlížeče, ale pomocí grafického editoru. Pokud totiž obrázek o velikosti např. 100×100 pixelů zmenšíme v prohlížeči na 50×50, docílíme tak sice požadované velikosti, ale uživatel bude zbytečně stahovat více dat, protože bude stahovat samozřejmě obrázek v původní velikosti.



Pro určení velikosti obrázku se používají atributy *width* (šířka) a *height* (výška) nebo stejnojmenné vlastnosti v CSS.

Vložení čáry

```
<hr size="10" noshade (bez stínování) with="50%" align="center">
```

**Obrázek**

Obrázek ze stejného adresáře, ve kterém je html stránka:

```

```

nebo obrazek.jpg nebo obrazek.png. Zadané rozměry by měly být skutečné rozměry obrázku (jednotka se v HTML neuvádí, ale rozměry jsou v pixelech = obrazkových bodech).

Obrázek z nadřazeného adresáře (to jsou ty dvě tečky):

```

```

Obrázek z webu (absolutní adresa, začíná http://):

```

```

Obrázek fungující jako odkaz (kliká):

```
<a href="cilova_stranka.html"></a>
```

Obrázek fungující jako odkaz, ale bez modrého rámečku:

```
<a href="cilova_stranka.html"></a>
```

**Div = oddíl**

Tag div slouží nejčastěji na obalení více blokových prvků. Například když budu chtít mít tři odstavce červeným písmem a odsazené zleva o 30px:

```
<div style="color: red; margin-left: 30px;">
```

```
<p>odstavec</p>
```

```
<p>odstavec</p>
```

```
<p>odstavec</p>
```

```
</div>
```

Ale dá se to dělat samozřejmě i jinak, stejně tak <div> má širší použití, většinou na rozvržení designu stránky.

## Neuspořádaný seznam

---

Neuspořádaný neboli nečíslovaný seznam se značí tagem **ul** (unordered list). Položka seznamu je **li** (list item). Položka může obsahovat i více odstavců. Před každou položkou se standardně vytváří odrážka.

```
<ul>
```

```
<li>žirafa</li>
```

```
<li>slon</li>
```

```
<li>velbloud</li>
```

```
</ul>
```

## Uspořádaný seznam

---

Uspořádaný neboli číslovaný seznam se značí tagem **ol** (ordered list). Položka seznamu je opět **li**. Před položku se automaticky vypisuje její pořadové číslo.

```
<ol>
  <li>žirafa</li>
  <li>slon</li>
  <li>velbloud</li>
</ol>
```

## Tabulka

---

se vytváří pomocí elementu **table**. Pro řádek tabulky slouží tag **tr** (table row), pro buňky tabulky tagy **th** (table head; pro buňky v záhlaví tabulky) a **td** (table data). Buňky se vkládají uvnitř řádku tabulky, kolik buněk, tolik bude mít tabulka sloupců. Buňky záhlaví jsou standardně formátovány tučným písmem, zarovnány na střed.

```
<table >
  <tr>
    <th>1</th>
    <th>2</th>
  </tr>
  <tr>
    <td>aaa</td>
    <td>bbb</td>
  </tr>
  <tr>
    <td>ccc</td>
    <td>ddd</td>
  </tr>
</table>
```

Nejjednodušší tabulka 2x2:

```
<table cellpadding="0">
<tr>
  <td>První buňka prvního řádku</td>
  <td>Druhá buňka prvního řádku</td>
</tr>
<tr>
  <td>První buňka druhého řádku</td>
  <td>Druhá buňka druhého řádku</td>
</tr>
</table>
```

Zapnutí rámečku (border) a vnitřních okrajů buněk (cellpadding):

```
<table border="1" cellpadding="6" cellspacing="0">...
```

Jednoduchý rámeček (nebude dvojitý):

```
<table border="1" style="border-collapse: collapse;">
<tr>první buňka...<td></td><td></td></tr>
<tr><td></td><td></td></tr>
</table>
```

Buňka přes dva řádky (v druhém řádku se zapíše o jednu buňku méně):

```
<td rowspan="2">buňka přes dva řádky</td>
```

Buňka přes tři sloupce:

```
<td colspan="3">Buňka přes tři sloupce</td>.
```

Nadpis tabulky

Pro nadpis tabulky se používá element **caption**. Ten by měl být uveden uvnitř tagu **table** ještě před prvním řádkem.

```
<table>
  <caption>Nadpis tabulky</caption>
  <tr>
    <th>1</th>
    <th>2</th>
  </tr>
  <tr>
    <td>aaa</td>
    <td>bbb</td>
  </tr>
  <tr>
    <td>ccc</td>
    <td>ddd</td>
  </tr>
</table>
```

## Bublina na přejetí myší

---

Do libovolného tagu stačí napsat atribut `title="text bubliny"`, například:

```
<span title="text, který se objeví ve žluté bublině">text, přes který když se přejeđe myší, se to objeví</span>
```

## Anketa, počítadlo, diskuse, návštěvní kniha

---

To není sranda naprogramovat, musí to běžet na serveru. Zaregistrujte na [www.blueboard.cz](http://www.blueboard.cz) a vložte si do stránky vygenerovaný kód.

Nejlepší počítadlo se statistikami zdarma je na [www.toplist.cz](http://www.toplist.cz).

## Validátor W3C

---

Různé validátory jsou v HTML editorech. Já budu psát přímo o validátoru W3C, je asi nejpoužívanější a jak název napovídá, bude i nejlepší :-)

Online validátor W3C nalezneme na adrese [validator.w3.org](http://validator.w3.org). Práce s ním je jednoduchá. Na úvodní stránce je třeba sdělit validátoru, který soubor se bude validovat. Zvolit můžete buď **Validate by URL**, kde se do políčka **Address** zadá URL adresa stránky. Nebo můžete zvolit **Validate by File Upload** a pomocí tlačítka **browse** projít svůj disk a vybrat (a potvrdit tlačítkem **check**) soubor k validaci.