

Stredná odborná škola informačných technológií

Hlinícka 1, 831 52 Bratislava

ROČNÍKOVÁ PRÁCA

Školský rok: 2021/2022
Stredná odborná škola informačných technológií

Simon Živner

Hlinícka 1, 831 52 Bratislava

Študijný odbor: 2682 K mechanik počítačových sietí

Názov ročníkovej práce:

Tvorba školskej stránky

Bratislava 2022

Simon Živner

ČESTNÉ VYHLÁSENIE

Vyhlasujem, že som ročníkovú prácu vypracoval samostatne na základe vlastných teoretických a praktických poznatkov s použitím uvedených zdrojov informácií.

Bratislava, 30.3. 2022

.....

Simon Živner

Obsah

Úvod	5
1. Web Stránka	6
1.1 Druhy Web Stránky	6
1.2 História Web Stránok	6
1.3 Vytváranie Web stránky	7
2. HTML.....	7
2.1 Základe príkazy HTML.....	8
2.2 HTML Colors	12
3.2 Formátovanie textu v CSS.....	16
4. JavaScript	16
4.1 JavaScript Syntax	17
5. HTTP	19
6. PHP.....	19
6.1 Syntax PHP.....	20
6.2 Premenné	21
7. Praktická časť	23
Záver.....	29
Použitá literatúra.....	30

Úvod

Cieľom mojej ročníkovej práce je zostaviť súbor zaujímavých príkladov z tvorby web stránky, ako aj vysvetlenie načo tieto dané stránky slúžia .

V teoretickej časti som na základe uvedenej literatúry rozobral podstatu daných programovacích jazykov.

Príklady a pokusy sú zaradené v druhej hlavnej kapitole – Praktická časť .

Prácu som navyše doplnil množstvom obrázkov. Dovolím si povedať, že jedným zo základných stavebných kameňov šetrenia času je webová prezentácia, ktorú si môže užívateľ jednoducho a intuitívne pozmeniť podľa svojich potrieb a prostredníctvom nej zaistiť efektívne predanie informácií cieľovej skupine.

1.Web Stránka

Web stránka je dokument zvyčajne obsahujúci hypertext, obrázky, multimedialne a ďalšie prvky uložený obyčajne na webovom serveri prístupný prostredníctvom služby World Wide Web v sieti internet. Zobrazuje sa používateľovi pomocou webového prehliadača. Názov „webová stránka“ je metaforou papierových stránok spojených do knihy. Webové stránky sú obyčajne písané v značkovacom jazyku HTML alebo XHTML a prenos prebieha pomocou protokolu HTTP.

1.1 Druhy Web Stránky

Webové stránky sa delia na statické a dynamické weby.

Statický web - využíva HTML , CSS , nevytvára žiadnu interakciu, je to iba zobrazenie obsahu a častí inzertných priestorov

Dynamický web - sú stránky, ktoré sú generované zvlášť pre každého užívateľa na základe jeho požiadaviek alebo hodnôt premenných.

Interaktivita ("dynamickosť") stránky môže byť zabezpečená dvoma spôsobmi:

1) použitím skriptovacieho jazyka bežiaceho v prehliadači návštevníka na zmenu výzoru stránky po akciách spôsobených najmä myšou, klávesnicou alebo v určených časových intervaloch. Technológie používané na tieto zmeny majú spoločné označenie „Dynamické HTML“. Tieto metódy sa zvyčajne používajú na okamžitú zmenu výzoru stránky, napríklad na vytvorenie rolovaciego menu po nabehtutí myšou na položku menu.

2) použitím skriptovacieho jazyka bežiaceho na serveri. V tomto prípade je časté čítanie údajov z databáz, ukladanie užívateľom zadaných dát do databáz, vyhľadávanie údajov a podobne.

1.2 História Web Stránok

História pojmu World Wide Web má svoje korene v udalostiach, ako je vynájdenie telegrafu, vypustenie satelitu Sputnik a v niektorých ďalších historických udalostiach.

Prvá história stránok začala v marci 1989, keď Tim Berners-Lee a vedci z inštitútu CERN vo Švajčiarsku vytvorili dokument s názvom „Information Management : A proposal“. V tomto dokumente sa podľa niektorých médií a odbornej verejnosti vtedajšej doby nič nové neobjavilo. Až postupom času po dôkladnej analýze znamenal tento dokument zásadný obrat vo svete sprostredkovania informácií a informačných technológií.

1.3 Vytváranie Web stránky

Web stránky vytvárame hlavne pomocou programovacieho jazyku HTML.
Na dizajnovanie webu používame programovací jazyk CSS.
Na programovanie stránky používame JavaScript.

2. HTML

HTML – Hyper Text Markup Language (Hypertextový značkový jazyk)

Je to značkový jazyk na vytváranie webových stránok a iných informácií zobraziteľných vo webovom prehliadači.

HTML kladie dôraz skôr na prezentáciu informácií ako na význam slov. Pôvodne bol určený ako veľmi zjednodušená podmnožina jazyku SGML, ktorý sa používa v organizáciách s komplexnými publikačnými požiadavkami, ale neskôr sa stal samostatným štandardom.

Prvky HTML

- štruktúrové prvky : <h1>Stránka</h1> označuje, že slovo „Stránka“ je nadpisom prvej úrovne.
- prezentačné prvky : tučné popisujú výzor textu, bez ohľadu na jeho zmysel. Uvedený príklad zobrazí slovo „tučné“ tučným písmom.
- ovládacie prvky : vytvárajú tlačidlá, zaškrŕavacie políčka, zoznamy

```

<div class="container">
  <div class="row">
    <div class="col-md-6 col-lg-8"> <!-- _____ BEGIN NAVIGATION _____
      <nav id="nav" role="navigation">
        <ul>
          <li><a href="index.html">Home</a></li>
          <li><a href="home-events.html">Home Events</a></li>
          <li><a href="multi-col-menu.html">Multiple Column Men
          <li class="has-children"> <a href="#" class="current">
            <ul>
              <li><a href="tall-button-header.html">Tall But
              <li><a href="image-logo.html">Image Logo</a></li>
              <li class="active"><a href="tall-logo.html">Ta
            </ul>
          </li>
          <li class="has-children"> <a href="#">Carousels</a>
          <ul>
            <li><a href="variable-width-slider.html">Variab
            <li><a href="testimonial-slider.html">Testimoni

```

Obrázok č.1 – príklad HTML kódu



Obrázok č.2 - HTML logo

2.1 Základné príkazy HTML

Každý HTML súbor by mal obsahovať niekoľko hlavných tagov, hlavičky a telá. Tým sa zaručí, že všetci klienti pochopia o čo v danom dokumente ide.

Úprava textu: Zastaralý spôsob formátovania textu.

Tag	Popis
	Bold
	Definuje zdôraznený text
<i>	Kurzíva
<small>	Písanie menším písmom
<sub>	Dolný index
<sup>	Horný index
<ins>	Písanie vloženého textu
	Prečiarknutie textu
<mark>	Podčiarknutie/zvýraznenie textu

V tabuľke uvedenej hore sa nachádzajú tagy ktoré sa používali v “minulosti”.

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

<p><b>Tento text je bold</b></p>
<p><i>Tento text je kurzívou</i></p>
<p>Toto je<sub> dolný index</sub> a <sup>horný index</sup></p>

</body>
</html>
```

Obrázok č.3 - Príklad formátovania textu v HTML

Tento text je bold

Tento text je kurzívou

Toto je _{dolný index} a ^{horný index}

Obrázok č.4 - Príklad formátovania textu v HTML

V obrázku uvedenom vyššie môžeme vydiť príklad použitia tagov <i> alebo <sub> <sup>

HTML tagy ktoré používame na logické formátovanie

Tag	Popis
	Úsek v texte
	Zvýraznenie(tučné)
<cite>	Citácia
<code>	Výpis kódu
<acronym>	Skratka
<dnf>	Novo použitý termín

HTML tagy ktoré používame na prácu s obrázkami a tabuľkami

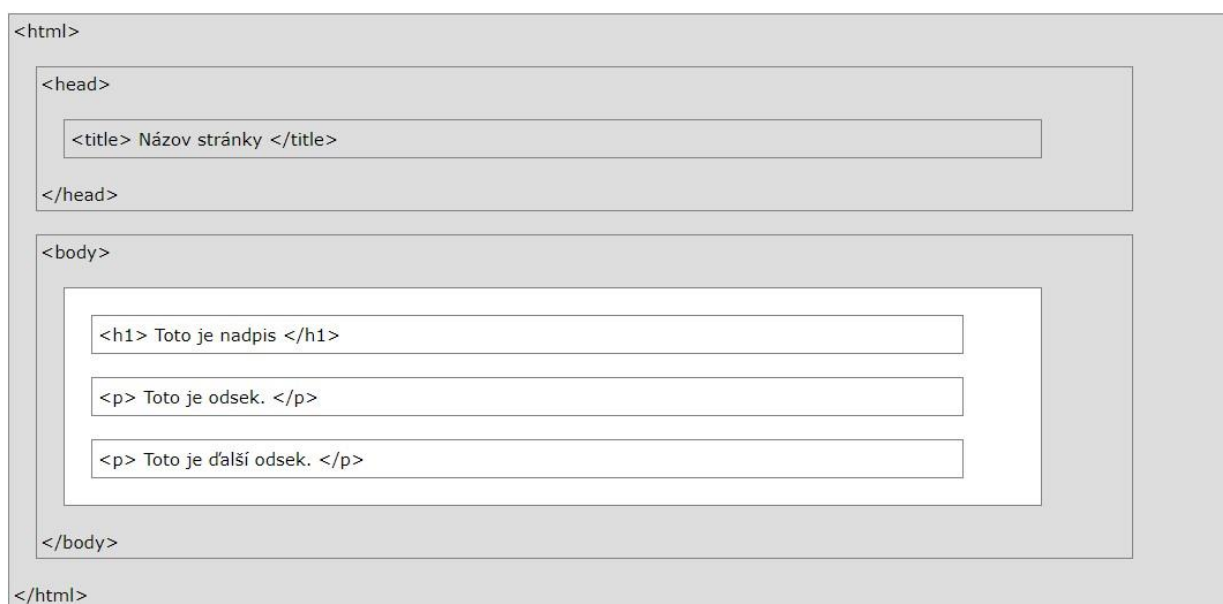
Tag	Popis
	Obrázok
<picture>	Rámček pre obrázok
<table>	Tabuľka
<tr>	Riadok tabuľky
<td>	Bunka tabuľky
<th>	Hlavička bunky tabuľky
<tbody>	Telo tabuľky
<thead>	Hlavička tabuľky

HTML tagy ktoré používame na prácu s multimédiami

Tag	Popis
<video>	Video súbor
<audio>	Audio súbor
<source>	Možný zdroj videa/audia
<track>	Titulky

Ďalšie dôležité príkazy ktoré používame v HTML:

- <!DOCTYPE html>** Vyhlásenie definuje, že tento dokument je HTML5 dokument
- <html>** Prvok je koreňový element stránky HTML
- <head>** Element obsahuje metainformácie o HTML stránky
- <title>** Element určuje názov stránky HTML (ktorý je zobrazený na titulnej lište prehliadača alebo na karte stránky)
- <body>** Prvok definuje telo dokumentu a slúži ako kontajner pre všetky viditeľné obsahy, ako sú nadpisy, odseky, obrázky, hypertextové odkazy, tabuľky, zoznamy, atď
- <h1>** Prvok definuje veľký nadpis
- <p>** Prvok definuje odsek



Obrázok č.5 – Vizualizácia štruktúry HTML stránky

2.2 HTML Colors

Farby v HTML sú určené preddefinovanými názvami farieb alebo hodnotami RGB, HEX, HSL, RGBA alebo HSLA.

HTML podporuje 140 štandardných názvov farieb.

Napríklad : “Brown” , “Blue” , “Tomato” , “Black” atď.

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

<h1 style="background-color:Tomato;">Tomato</h1>
<h1 style="background-color:Orange;">Orange</h1>

</body>
</html>
```

Obrázok č.6 - Príklad určovania farieb podľa názvu v HTML

Hodnota RGB predstavuje červené , modré a zelené svetelné zdroje.

Farbu je možné určiť ako hodnotu RGB pomocou tohoto vzorca:

rgb (*červená* , *zelená* , *modrá*)

Každý parameter (červený, zelený a modrý) definuje intenzitu farby s hodnotou od 0 do 255.

To znamená, že existuje $256 \times 256 \times 256 = 16\,777\,216$ možných farieb!

Napríklad rgb (255, 0, 0) sa zobrazuje ako červená, pretože červená je nastavená na najvyššiu hodnotu (255) a ďalšie dva (zelená a modrá) sú nastavené na 0.

Iný príklad, rgb (0, 255, 0) sa zobrazuje ako zelená, pretože zelená je nastavená na najvyššiu hodnotu (255) a ďalšie dva (červený a modrý) sú nastavené na 0.

Ak chceme zobrazit' čiernu farbu, nastavíme všetky farebné parametre na 0, napríklad takto: rgb (0, 0, 0).

Ak chceme zobrazit' bielu farbu, nastavíme všetky farebné parametre na 255, napríklad takto: rgb (255, 255, 255).

```

<!DOCTYPE html>
<html>
<body>

<h1 style="background-color:rgb(255, 0, 0);">rgb(255, 0, 0)</h1>
<h1 style="background-color:rgb(0, 0, 255);">rgb(0, 0, 255)</h1>
<h1 style="background-color:rgb(60, 179, 113);">rgb(60, 179, 113)</h1>
<h1 style="background-color:rgb(238, 130, 238);">rgb(238, 130, 238)</h1>
<h1 style="background-color:rgb(255, 165, 0);">rgb(255, 165, 0)</h1>
<h1 style="background-color:rgb(106, 90, 205);">rgb(106, 90, 205)</h1>

</body>
</html>

```

Obrázok č.7 - Príklad nastavenia farby pomocou RGB

HEX farba je zadaná pomocou: #RRGGBB, kde hexadecimálne celé čísla RR (červená), GG (zelená) a BB (modrá) určujú komponenty farby.

Farbu je možné určiť pomocou hexadecimálnej hodnoty v tvare: **#rrggbb**

Kde rr (červená), gg (zelená) a bb (modrá) sú hexadecimálne hodnoty medzi 00 a ff (rovnaké ako desatinné miesta 0-255).

Napríklad # ff0000 je zobrazený ako červená, pretože červená je nastavená na najvyššiu hodnotu (ff) a ďalšie dva (zelená a modrá) sú nastavené na 00.

Ďalším príkladom je # 00ff00, ktorý sa zobrazuje ako zelená, pretože zelená je nastavená na najvyššiu hodnotu (ff) a ďalšie dva (červená a modrá) sú nastavené na 00.

Ak chceme zobraziť čiernu farbu, nastavíme všetky farebné parametre na hodnotu 00, napríklad takto: # 000000.

Ak chceme zobraziť bielu farbu, nastavíme všetky farebné parametre na hodnotu ff, napríklad takto: #ffffff

```

<!DOCTYPE html>
<html>
<body>

<h1 style="background-color:#ff0000;">#ff0000</h1>
<h1 style="background-color:#0000ff;">#0000ff</h1>
<h1 style="background-color:#3cb371;">#3cb371</h1>
<h1 style="background-color:#ee82ee;">#ee82ee</h1>
<h1 style="background-color:#ffa500;">#ffa500</h1>
<h1 style="background-color:#6a5acd;">#6a5acd</h1>

</body>
</html>

```

Obrázok č.8 -
Príklad
nastavenia
farby
pomocou
HEX

Farbu je možné určiť pomocou odtieňa, sýtosti a svetlosti (HSL) vo forme:

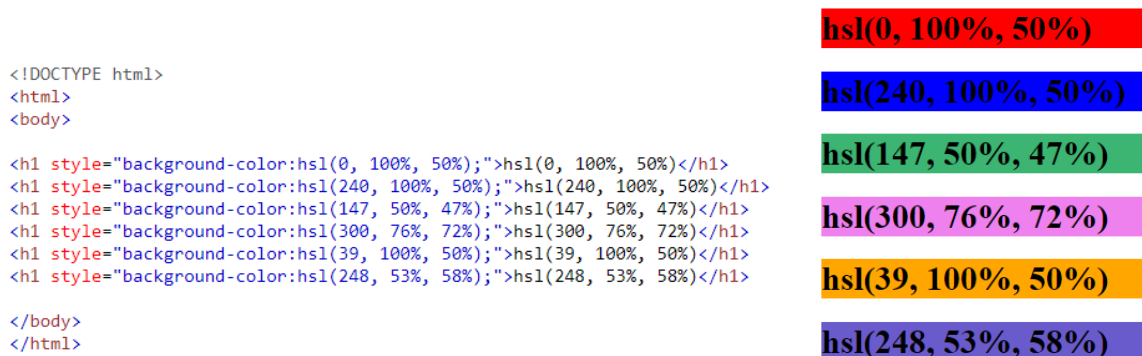
hsl (odtieň , sýtosť , ľahkosť)

Odtieň je stupeň na farebnom koliesku od 0 do 360.

0 je červená, 120 je zelená a 240 je modrá.

Sýtosť je percentuálna hodnota, 0% znamená odtieň šedej a 100% je plná farba.

Svetlosť je tiež percentuálnou hodnotou, 0% je čierna a 100% je biela.



Obrázok č.9 a č.10 – Príklad nastavenia farby pomocou HSL

3. CSS

CSS – Cascading Style Sheets (Kaskádové štýly)

Je to všeobecné rozšírenie HTML alebo XHTML. Označuje sa ako jednoduchý mechanizmus na vizuálne formátovanie internetových dokumentov.

Pomocou CSS sa vytvárajú štruktúrované dokumenty, teda oddeluje sa obsah dokumentu (HTML) od jeho vzhľadu (CSS). Získa sa tým prehľadný a jednoduchý kód.

CSS je možné presunúť do externých súborov, zmenší sa tým dátová veľkosť a dá sa jedným súborom zmeniť celý štýl stránky. Cieľom každého webdesignera je vytvoriť stránku tak, aby vyzerala čo najviac podobne v najpoužívanejších prehliadačoch.

CSS nezaručuje rovnaké vykresľovanie vo všetkých prehliadačoch.

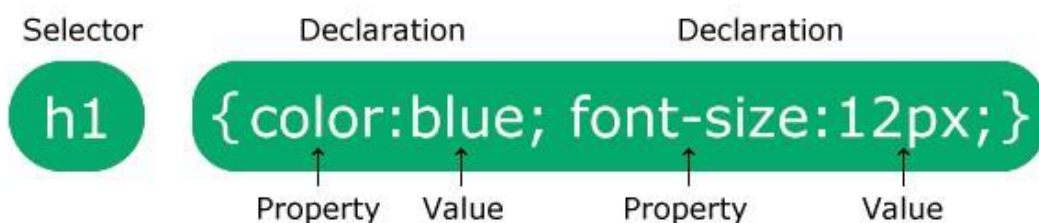
Vzhľadom k rôznym interpretáciám CSS s rôznymi prehliadačmi je nemožné aby stránka vyzerala na pixel rovnako. Najväčší problém v interpretácii CSS má internet explorer.



Obrázok č.11 - CSS logo

3.1 Základné príkazy CSS

Pravidlo CSS pozostáva zo selektora a bloku vyhlásenia.



Obrázok č.12 – Príklad CSS selektora

Selektor ukazuje na prvok HTML, ktorý chceme upraviť.

Deklaračný blok obsahuje jednu alebo viac deklarácií oddelených bodkočiarkami. Každá deklarácia obsahuje názov vlastnosti CSS a hodnotu oddelenú dvojbodkou. Viaceré vyhlásenia CSS sú oddelené bodkočiarkami a bloky vyhlásení sú obklopené zloženými zátvorkami.

p - je selektor v CSS (odkazuje na prvok HTML, ktorý chcete upraviť: <p>)

color - je vlastnosť a **red** je hodnotou vlastnosti **text-**

align - je vlastnosť a **center** je hodnotou vlastnosti

3.2 Formátovanie textu v CSS

CSS má veľa vlastností na formátovanie textu.

Napríklad:

color - slúži na nastavenie farby textu.

Farbu určuje: názov farby - napríklad „ **green** “ hodnota HEX - napríklad „ # 0000ff “

hodnota RGB - napríklad „ rgb (0,255,0) “

text-align - slúži na nastavenie vodorovného zarovnania textu

Text môže byť zarovnaný doľava alebo doprava, na stred alebo zarovnano.

napr. **h1** {

text-align: **center**;

}

Keď je **text-align** nastavený na „zarovnanie“, každý riadok sa natiahne tak, aby mal každý riadok rovnakú šírku a ľavý a pravý okraj boli rovné.

4. JavaScript

JavaScript, je skriptovací programovací jazyk, ktorý je používaný najmä pri tvorbe webových stránok. Pôvodne ho vyvíjal Brendan Eich zo spoločnosti Netscape Communications pod názvom Mocha, neskôr pod menom LiveScript. Pred uvedením na verejnosť bol premenovaný na „JavaScript“, najmä pre vtedajšiu popularitu jazyka Java. Aj na základe jeho názvu je rozšírený názor, že syntax Javascriptu sa podobá Jave, v skutočnosti bol jeho tvorca najviac inšpirovaný jazykom Self.

Neexistuje žiadny vzťah medzi jazykmi Java a JavaScript. Ich najväčšia podobnosť je v syntaxi založenej na spoločnom predkovi - jazyku C. Microsoft vyvinul jazyk s podobnou syntaxou pod názvom JScript, ktorý ale nedodržiava špecifikáciu ECMAScript. Rozdiely v implementácii majú za následok čiastočnú neprenosnosť kódu medzi webovými

prehliadačmi ako je napríklad implementácia JScript (Microsoft Internet Explorer) a JavaScript (Mozilla Firefox).

V súčasnosti je dostupné množstvo implementácií virtuálnych strojov jazyka JavaScript, i s otvoreným zdrojovým kódom, pričom niektoré z týchto implementácií je možné používať ako knižnice do iných programov a mať tak podporu tohto jazyka v prakticky ľubovoľnej aplikácii. Spôsoby interpretácie sa rôznia medzi implementáciami.

Niektoré využívajú jednoduché spracovanie kódu do abstraktného syntaxného stromu, ktorý následne prechádzajú a interpretujú interpreterom priamo zdrojový kód. Oveľa efektívnejšie sú implementácie, ktoré kód spracujú do podoby bajtkódu - vlastnej reprezentácie jednotlivých inštrukcií programového kódu. Tento bajtkód je interpretovaný následne virtuálnym strojom, ktorý môže byť registrový alebo zásobníkový, čo tiež ovplyvňuje jeho výkon.



Obrázok č.13 - JavaScript logo

4.1 JavaScript Syntax

JavaScript Syntax je sada pravidiel, pomocou ktorých sa zostavujú programy JavaScript

`var x, y, z;` - vyhlási premenné

`x = 5; y = 6;` - priradí hodnoty

`z = x + y;` - vypočíta hodnoty

Syntax jazyka JavaScript definuje dva typy hodnôt :

- a) Pevné hodnoty (fixed values)
- b) Premenné hodnoty (variable values)

Fixné hodnoty sa nazývajú Literals

Hodnoty premenných sa nazývajú Variables

4.1.1 Literals

Dve najdôležitejšie pravidlá syntaxe pre pevné hodnoty sú:

- čísla sa píše s desatinnými miestami alebo bez nich
- reťazce sú text napísaný dvojitémi alebo jednoduchými úvodzovkami

4.1.2 Variables

V programovacom jazyku sa na ukladanie dátových hodnôt používajú premenné .

JavaScript používa **var** kľúčové slovo na deklaráciu premenných.

Znamienko rovnosti sa používa na priradenie hodnôt do premenných.

Napr. **var** x;

x = 6;

Na výpočet hodnôt využíva aritmetické operátory (+ - / *)

Napr. (9 + 4) / 5

Na priradovanie hodnôt sa používa =

Napr . **var** x, y;

x =

5; y

= 6;

5. HTTP

HTTP – Hypertext transfer protocol (hypertextový prenosový protokol) je protokol na prenos html dokumentov medzi servermi a klientmi služby WWW.

Pôsobí ako primárna metóda prepravy informácií na world wide webe. Pôvodný účel bol poskytovať prostriedky na publikáciu a získavanie HTML stránok.

HTTP je protokol definujúci požiadavky a odpovede medzi klientmi a servermi.



Obrázok č.14 - HTTP protokol

6. PHP

PHP začalo ako malý open source projekt, ktorý sa vyvíjal, keď stále viac ľudí zisťovalo, aký užitočný je. Rasmus Lerdorf spustil prvú verziu PHP už v roku 1994.

PHP je skratka pre „Hypertext Preprocessor“.

PHP je server side scripting language ktorý je vložený do HTML. Používa sa na správu dynamického obsahu, databáz, sledovanie relácií, dokonca aj na vytváranie celých stránok elektronického obchodu.

Je integrovaný s množstvom populárnych databáz, vrátane MySQL, PostgreSQL, Oracle, Sybase, Informix a Microsoft SQL Server.

PHP je vo svojom vykonávaní príjemne rýchle, najmä keď je kompilované ako modul Apache na strane Unixu. Po spustení server MySQL vykoná aj veľmi zložité dotazy s obrovskými súbormi výsledkov v čase nastavenia rekordov. PHP podporuje veľké množstvo hlavných protokolov ako POP3, IMAP a LDAP.

PHP4 pridalo podporu pre Java a distribuované objektové architektúry (COM a CORBA), vďaka čomu je po prvýkrát možnosť n-tierového vývoja.

Jazyk PHP sa snaží byť čo najviac zhovievavý.

Syntax PHP je podobná C.

6.1 Syntax PHP

Syntaktický modul PHP potrebuje spôsob, ako odlíšiť kód PHP od ostatných prvkov na stránke.

Mechanizmus ako to urobiť, je známy ako „escaping to PHP“.

Existujú štyri spôsoby, ako to urobiť :

1. PHP tagy - univerzálny a najviac efektívny spôsob

`<?php ?>`

2. SGML - style tagy

`<??>`

3. ASP – style tagy

- značky v štýle ASP napodobňujú značky používané službou Active Server Pages na vymedzenie blokov kódu

`<%%>`

4. HTML script tagy

`<script language = "PHP">...</script>`

6.2 Premenné

Hlavným spôsobom ukladania informácií uprostred programu PHP je použitie premennej.

- Všetky premenné v PHP sú označené znakom dolára (\$).
- Hodnota premennej je hodnota jej posledného priradenia.
- Premenným sa priraduje operátor =, pričom premenná je na ľavej strane a výraz, ktorý sa má vyhodnotiť, je na pravej strane.

PHP má celkovo osem dátových typov, ktoré používame na zostavenie našich premenných:

- **Integers** – sú celé čísla bez desatinnej čiarky, napríklad 4195

```
$int_var = 12345;  
$another_int = -12345 + 12345;
```

- **Doubles** – sú čísla s pohyblivou desatinnou čiarkou, napríklad 3,14159 alebo 49,1

```
<?php  
    $many = 2.2888800;  
    $many_2 = 2.2111200;  
    $few = $many + $many_2;  
  
    print("$many + $many_2 = $few <br>");  
?>
```

- **Booleans** – majú iba dve možné hodnoty buď true alebo false

```
if (TRUE)  
    print("This will always print<br>");  
  
else  
    print("This will never print<br>");
```

- **NULL** – je špeciálny typ, ktorý má iba jednu hodnotu: NULL

```
$my_var = NULL;
```

- **Strings** – sú sekvencie znakov, ako napríklad 'PHP podporuje reťazcové operácie

```
$string_1 = "This is a string in double quotes";  
$string_2 = 'This is a somewhat longer, singly quoted string';  
$string_39 = "This string has thirty-nine characters";  
$string_0 = ""; // a string with zero characters
```

- **Arrays** – sú pomenované a indexované kolekcie iných hodnôt
- **Objects** – sú inštanciami tried definovaných programátorom, ktoré môžu zbalit' aj iné druhy hodnôt a funkcií, ktoré sú špecifické pre triedu
- **Resources** – sú špeciálne premenné, ktoré obsahujú odkazy na zdroje mimo PHP (ako napríklad pripojenia k databáze)

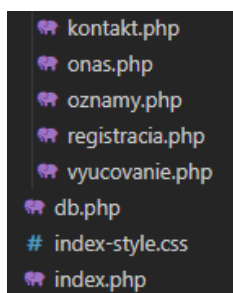


Obrázok č.15 PHP logo

7. Praktická časť

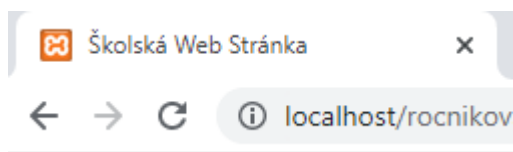
V praktickej časti mojej ročníkovej práce budem spracovávať vytváranie samostatnej školskej web stránky pomocou PHP.

Na začiatok si vytvorím súbory kam si budem ukladať samostatný kód ku každej podstránke.



Značka `<title>` definuje názov dokumentu. Názov musí byť iba textový a zobrazuje sa v záhlaví prehliadača alebo na karte stránky.

```
<title>Školská Web Stránka</title>
</head>
<body>
<div class="navigation">
<ul>
<li class="list">
<b></b>
<b></b>
<a href="index.php?stranka=onas">
<span class="title">0 nás</span>
</a>
</li>
<li class="list">
<b></b>
<b></b>
<a href="index.php?stranka=oznamy">
<span class="title">Oznamy</span>
</a>
</li>
<li class="list">
<b></b>
<b></b>
<a href="index.php?stranka=kontakt">
<span class="title">Kontakt</span>
</a>
</li>
<li class="list">
<b></b>
<b></b>
<a href="index.php?stranka=vyucovanie">
<span class="title">Vyučovanie</span>
</a>
</li>
<li class="list">
<b></b>
<b></b>
<a href="index.php?stranka=registracia">
<span class="title">Prihlásenie</span>
</a>
</li>
</ul>
</div>
```



Pomocou atribútu href si prepojím podstránku s hlavnou stránkou.

```
<a href="index.php?stranka=onas">  
    <span class="title">O nás</span>  
</a>
```

Po prepájaní všetkých podstránok mi vznikne nav bar.

O NÁS

OZNAMY

KONTAKT

VYUČOVANIE

PRIHLÁSENIE

Následne po vytvorení nav baru sa môžem presunúť na samostatnú podstránku.

```
podstranky > onas.php  
1  <!DOCTYPE html>  
2  <html lang="en">  
3  <head>  
4      <meta charset="UTF-8">  
5      <meta http-equiv="X-UA-Compatible" content="IE=edge">  
6      <meta name="viewport" content="width=device-width, initial-scale=1.0">  
7      <link rel="stylesheet" href="index-style.css">  
8      <title>Document</title>  
9  </head>  
10 <body>  
11     <div class = "main">  
12         <?php  
13             echo "Ak sa doteraz sústreďoval vývoj v oblasti IT na stolné počítače a zvyšov  
14             echo "<br>";  
15             echo "Škola prechádza neustálymi zmenami s cieľom dosahovať najvyššiu kvalitu  
16         ?>  
17     </div>  
18  
19 </body>  
20 </html>  
21  
22
```

V tejto konkrétnej podstránke onas.php som si do časti body vypísal pomocou príkazu echo skutočnosti o vybranej škole.

Ako nasledujúci krok som si vytvoril podstránku, kde budú informácie o každej triede ako sú napríklad. rozvrh, žiaci a iné.

```
<body>
  <div class = "main">
    <div class = "civic-box">
      <a href="index.php?stranka=4.A">
        <div class = "4.A">
          <span class="title"><b>4.A</b></span>
        </div>
      </a>
      <a href="index.php?stranka=4.B">
        <div class = "4.B">
          <span class="title"><b>4.B</b></span>
        </div>
      </a>
    </div>
  </div>
</body>
```

Ďalším následným krokom je vytvorenie podstránky, kde sa bude prihlasovať do žiackých, rodičovských alebo učiteľských účtov.

```
<body>
<div class = "main">
  <div class = "prihlasenie-place">
    <div class = "prihlasenie">
      <?php

        session_start();
        require_once('db.php');
        if($_POST){
          if($_POST['rok'] != date('Y'))
            echo '<p>Chybne vyplneny antispam.</p>';
          else if($_POST['heslo'] == '')
            echo "<p>Nevyplnili ste heslo.</p>";
          else if($_POST['heslo'] != $_POST['oheslo'])
            echo "<p>Hesla sa nezhoduju.</p>";
          else if($_POST['email'] == '')
            echo "<p>Nevyplnili ste email.</p>";
          else if($_POST['meno'] != ''){
            $meno = $_POST['meno'];
            $email = $_POST['email'];
            $sql = "SELECT name_user FROM uzivatelia WHERE name_user = '$meno' OR email_user = '$email' LIMIT 1;";
            $result = $conn->query($sql);

            if($result->num_rows > 0)
              echo "<p>Uzivatel s tymto menom alebo emailom uz existuje</p>";
            else{
              $heslo = password_hash($_POST['heslo'], PASSWORD_DEFAULT);
              $meno = $_POST['meno'];
              $email = $_POST['email'];
              $sql = "INSERT INTO uzivatelia (name_user, user_password, email_user) VALUES ('$meno', '$heslo', '$email')";
              $conn->query($sql);
              $posledny = $con->insert_id;
              $_SESSION['uzivatel_id'] = $con->insert_id;
              $_SESSION['uzivatel_meno'] = $_POST['meno'];
              $_SESSION['uzivatel_email'] = $_POST['email'];
              $_SESSION['uzivatel_admin'] = 0;
              //header('Location: admin/administracia.php');
              header('Location: admin/admin_view.php?stranka=administracia');
              exit();
            }
          }
        }
      </div>
    </div>
  </div>
</body>
```

Pri vytvorení prihlasovania som nastavil pomocou kódu uvedenom nižšie

```
if($_POST){  
    if($_POST['rok'] != date('Y'))  
        echo '<p>Chybne vyplneny antispam.</p>';  
}
```

antispam, ktorý zabraňuje vytvoreniu účtu, ak sa rok nezhoduje s aktuálnym rokom.

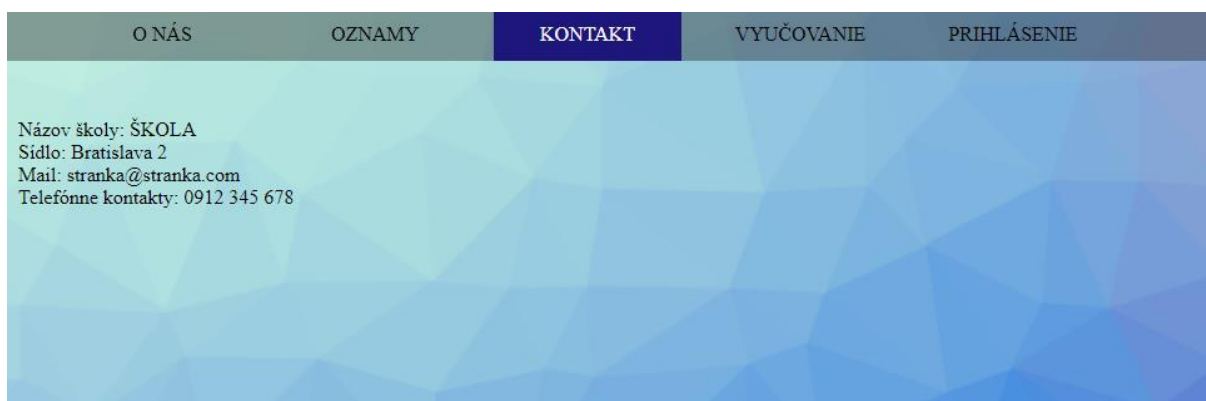
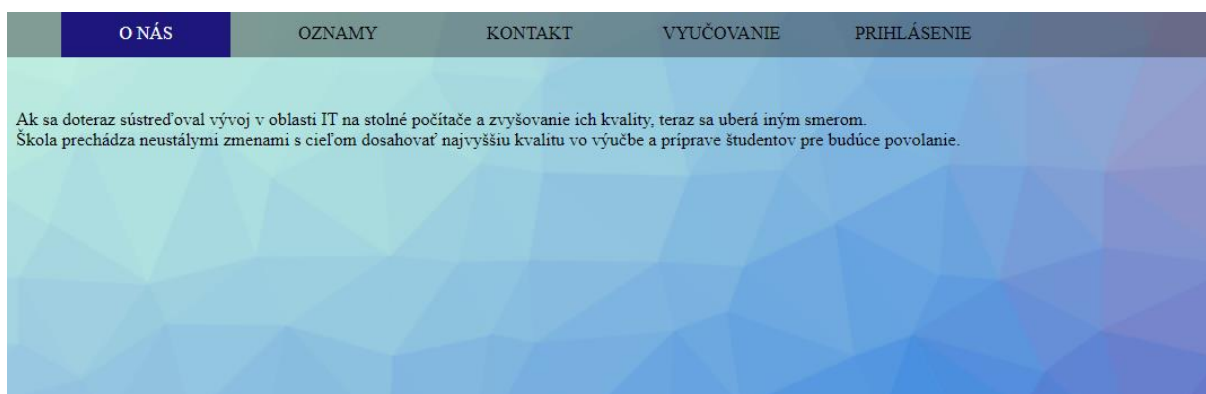
Následne je potrebné zadať heslo duplicitne ako overenie správnosti.

```
else if($_POST['heslo'] == '')  
    echo "<p>Nevyplnili ste heslo.</p>";  
else if($_POST['heslo'] != $_POST['oheslo'])  
    echo "<p>Hesla sa nezhoduju.</p>";
```

Na záver treba zadať meno a email. V prípade ak sa meno alebo email zhoduje s databázou, systém nepovolí vytvoriť daný účet a oznámi duplicitu už existujúceho účtu. Ak sa účet v databáze nenachádza, vytvorí ho a následne uloží do pamäte databázy zvolené meno, heslo a email.

```
else if($_POST['email'] == '')  
    echo "<p>Nevyplnili ste email.</p>";  
else if($_POST['meno'] != ''){  
    $meno = $_POST['meno'];  
    $email = $_POST['email'];  
    $sql = "SELECT name_user FROM uzivatelia WHERE name_user = '$meno' OR email_user = '$email' LIMIT 1;";  
    $result = $conn->query($sql);  
  
    if($result->num_rows > 0)  
        echo '<p>Uzivatel s tymto menom alebo emailom uz existuje</p>';  
    else{  
        $heslo = password_hash($_POST['heslo'], PASSWORD_DEFAULT);  
        $meno = $_POST['meno'];  
        $email = $_POST['email'];  
        $sqli = "INSERT INTO uzivatelia (name_user, user_password, email_user) VALUES ('$meno', '$heslo', '$email')";  
        $conn->query($sqli);  
        $posledny = $con->insert_id;  
        $_SESSION['uzivatel_id'] = $con->insert_id;  
        $_SESSION['uzivatel_meno'] = $_POST['meno'];  
        $_SESSION['uzivatel_email'] = $_POST['email'];  
        $_SESSION['uzivatel_admin'] = 0;  
        //header('Location: admin/administracia.php');  
        header('Location: admin/admin_view.php?stranka=administracia');  
        exit();  
    }  
}
```

Finálny vzhľad stránky:



O NÁS OZNAMY KONTAKT VYUČOVANIE **PRIHLÁSENIE**

Meno

Heslo

Opakuj Heslo

Email

Aktuálny Rok

Registovať sa

Máte už účet? [Prihlásiť sa.](#)

DOMOV O NÁS OZNAMY KONTAKT **REGISTRÁCIA**

Meno

Heslo

Prihlásiť sa

Nemáte ešte účet? [Registovať sa.](#)

Záver

V mojej ročníkovej práci som vás chcel oboznámiť s tvorbou školskej web stránky . Venoval som sa programovacím jazykom pomocou ktorých dokážeme vytvoriť danú web stránku. Ako vieme , medzi základné formy doručenia informácie patrí osobné stretnutie, telefonický hovor, pošta, e-mailový kontakt, webová prezentácia, prípadne iné dostupné možnosti. Ale medzi najrýchlejšie (získanie času) a najlacnejšie (jednorazová investícia do kúpy softwaru) formy riešenia predania informácie patrí webová prezentácia. Zdroje som čerpal najmä z internetu, linky na konkrétne stránky a knihy sú uvedené v použitej literatúre. Práca na tejto téme ma veľmi zaujala, bavila a bola pre mňa veľkým prínosom.

Použitá literatúra.

https://sk.wikipedia.org/wiki/Web_site

https://sk.wikipedia.org/wiki/Dynamick%C3%A1_webov%C3%A1_str%C3%A1nka

<https://www.jakpsatweb.cz/html/>

<https://dizajn-web.webnode.sk/vznik-a-historia-web-stranok/>

https://sk.wikipedia.org/wiki/Web_page

https://sk.wikipedia.org/wiki/Hypertextov%C3%BD_zna%C4%8Dkov%C3%B4_jazyk

<https://pixabay.com/sk/photos/k%C3%B3d-html-digit%C3%A1lnyk%C3%B3dovanie-web-1076533/>

https://sk.wikipedia.org/wiki/Hypertextov%C3%BD_prenosov%C3%BD_protokol

[https://sk.wikipedia.org/wiki/PHP_\(skriptovac%C3%AD_jazyk\)](https://sk.wikipedia.org/wiki/PHP_(skriptovac%C3%AD_jazyk))

[https://sk.wikipedia.org/wiki/PHP_\(skriptovac%C3%AD_jazyk\)#/media/S%C3%BAbor:PHP-logo.svg](https://sk.wikipedia.org/wiki/PHP_(skriptovac%C3%AD_jazyk)#/media/S%C3%BAbor:PHP-logo.svg)

<https://www.w3schools.com/>

https://www.tutorialspoint.com/php/php_introduction.htm

Knižná literatúra:

HTML and CSS (autor: Jon Duckett)

[Nakladateľ: John Wiley & Sons Inc , Rok vydania: 2011]

PHP pro úplné začátečníky (autor: Jakub Mach)

[Dátum vydania 2002]

JavaScript – Programátorské techniky a webové technológie

(autor: Ondřej Žára)

[Vydavateľstvo Computer Press , Dátum vydania 2015]