

Univerzita Karlova
Pedagogická fakulta
Katedra informačních technologií a technické výchovy

DIPLOMOVÁ PRÁCE

Metodika výuky značkovacích jazyků
Markup languages educational methodology

Bc. Vojtěch Matějka

Vedoucí práce: PhDr. Josef Procházka, Ph.D.
Studijní program: N7504 Učitelství pro střední školy
Studijní obor: Učitelství VPP pro ZŠ a SŠ – technická a informační výchova

2017

Prohlašuji, že jsem diplomovou práci na téma Metodika výuky značkovacích jazyků vypracoval pod vedením vedoucího práce samostatně za použití v práci uvedených pramenů a literatury. Dále prohlašuji, že tato práce nebyla využita k získání jiného nebo stejného titulu.

Praha 21. 4. 2017

.....

podpis

Na tomto místě bych chtěl poděkovat vedoucímu mé diplomové práce PhDr. Josefu Procházkovi, Ph.D. za její odborné vedení, pomoc a trpělivost při zpracování této práce. Dále bych chtěl poděkovat ing. Jitce Zůnové a Jiřímu Hofmanovi za možnost vést výuku v jejich předmětu. Dále děkuji všem, kteří se svou ochotou zasloužili o to, aby práce v této podobě vznikla.

ABSTRAKT

Cílem této práce je vytvořit metodiku výuky značkovacích jazyků Markdown, HyperText Markup Language (HTML), eXtensible Markup Language (XML) a Scalable Vector Graphics (SVG) a tu ověřit na druhém stupni základní školy.

Součástí metodiky je seznámení se s historií a vývojem zmíněných značkovacích jazyků a teorie syntaxe značkovacích jazyků, která je nutná pro výuku podle této metodiky. Metodika využívá konkrétních příkladů, na kterých si žáci ověřují nově nabyté znalosti.

Ověření metodiky proběhlo formou výuky na dvou základních školách. Výuka proběhla v sedmé třídě v rámci povinně volitelného informatického předmětu a v deváté třídě během povinného předmětu. Očekávané rozdíly v pochopení výuky značkovacích jazyků ve prospěch devátého ročníku se nepotvrdily. Mnohem větší roli, než věk žáků, hrála jejich motivace. Ověření metodiky ukázalo, že je nutné důsledně dbát na formulaci zadání úloh a dalších pokynů pro žáky, aby nedocházelo k zaměňování zdrojového kódu dokumentu obsahujícího značky s výsledným formátovaným dokumentem.

Hlavním zjištěním je, že výuka značkovacích jazyků je pro žáky na druhém stupni pochopitelná, a že lze použít tuto metodiku pro výuku.

KLÍČOVÁ SLOVA

značkovací jazyk, značka, tag, Markdown, HyperText Markup Language, HTML, eXtensible Markup Language, XML, Scalable Vector Graphics, SVG, metodika výuky

ABSTRACT

The aim of this thesis is to develop an educational methodology for the following languages: Markdown, HyperText Markup Language (HTML), eXtensible Markup Language (XML) and Scalable Vector Graphics (SVG); and to verify the methodology at Czech elementary schools on the lower secondary level.

An integral part of the methodology is also an introduction to the history and development, as well as the theory of syntax of the mentioned markup languages, necessary for teaching according to this methodology. The methodology uses specific examples, on which the students can verify their newly acquired knowledge in practice.

The verification of the methodology took place in the form of teaching at two Czech elementary schools. The teaching took place in the seventh grade within the compulsory elective subject of informatics, in the ninth grade within a compulsory subject. The expected differences in understanding the markup languages education in favor of the ninth year have not been confirmed. The motivation of the students proved to play much greater role than their age. The verification of the methodology has shown that it is necessary to pay careful attention to the precise formulation of the task assignments and guidelines to the students, to avoid confusion and substitution of the source code of the document containing the tags and the final formatted document.

The key result is that the markup languages education is comprehensible for the lower secondary level students and the presented educational methodology can be applied.

KEYWORDS

markup language, markup, tag, Markdown, HyperText Markup Language, HTML, eXtensible Markup Language, XML, Scalable Vector Graphics, SVG, educational methodology

Obsah

1 Úvod.....	8
2 Značkovací jazyky.....	10
2.1 Historie a vývoj značkovacích jazyků.....	10
2.1.1 SGML.....	11
2.1.2 HTML.....	11
2.1.3 Od SGML k XML.....	12
2.1.4 Markdown.....	13
2.2 Syntaxe a značky.....	13
2.2.1 HyperText Markup Language (HTML).....	13
2.2.2 eXtensible Markup Language (XML).....	17
2.2.3 Markdown.....	19
2.2.4 Scalable Vector Graphics (SVG).....	22
2.3 Rozdělení značkovacích jazyků.....	24
2.3.1 Popisné jazyky.....	24
2.3.2 Výkonné jazyky.....	25
2.3.3 Prezentací jazyky.....	26
2.3.4 Shrnutí.....	27
3 Učebnice a skripta.....	28
3.1 Informatika pro základní školy: 1. díl Základy práce s PC.....	28
3.2 Informatika a výpočetní technika pro střední školy.....	28
3.3 Výpočetní technika – Hardware a Software.....	29
3.4 Internet a tvorba jednoduchých WWW stránek.....	29
3.5 Praktické činnosti pro 6.-9. ročník základních škol.....	31
3.6 Informatika pro základní školy: 3. díl.....	32
3.7 S počítačem nejen k maturitě: 2. díl.....	32
3.8 CSS a XHTML: tvorba dokonalých WWW stránek krok za krokem.....	34
3.9 Shrnutí.....	35
4 Metodika.....	36
4.1 Formátování textu podle značek.....	36
4.2 Formátovaný dokument.....	37
4.3 Nákupní seznam.....	39

4.4 Z Markdownu do HTML.....	41
4.5 Odkazy v HTML.....	43
4.6 Strukturování dokumentu.....	44
4.7 Pojmenování částí značkami.....	47
4.8 Seznam do XML.....	49
4.9 XML a atributy.....	51
4.10 XML do tabulky.....	51
4.11 Česká vlajka v SVG.....	52
4.12 Další úlohy.....	53
4.12.1 Značkování.....	53
4.12.2 Z popisu dokument.....	55
4.12.3 Datově orientovaný dokument.....	56
5 Ověření metodiky.....	58
5.1 Očekávané výsledky ověření.....	58
5.2 Průběh výuky.....	59
5.2.1 Úvodní dotazník.....	60
5.2.2 Úvod do značkování.....	63
5.2.3 Formátování podle značek.....	64
5.2.4 Jazyk Markdown.....	65
5.2.5 Příprava nákupního seznamu.....	66
5.2.6 Nákupní seznam v jazyce Markdown.....	67
5.2.7 Struktura dokumentu a XML.....	68
5.2.8 XML dokument do tabulky.....	69
5.2.9 Markdown do HTML.....	69
5.2.10 SVG česká vlajka.....	69
5.3 Závěrečný dotazník.....	70
5.4 Vyhodnocení ověření.....	71
6 Závěr.....	73
7 Seznam literatury.....	75
8 Seznam příloh.....	78

1 Úvod

Diplomová práce Metodika výuky značkovacích jazyků si klade za cíl vytvořit metodiku výuky značkovacích jazyků a tu ověřit na 2. stupni základní školy. Metodika vyžívá konkrétních příkladů, na kterých objasňuje jednotlivá pravidla užití značkovacích jazyků.

Žáci budou v rámci této metodiky seznámeni se základy značkovacích jazyků Markdown, HyperText Markup Language (HTML), Extensible Markup Language (XML) a Scalable Vector Graphics (SVG). Zvolené jazyky byly vybrány tak, aby pokrývaly širokou škálu možných využití značkovacích jazyků. Jazyk Markdown se využívá při formátování dokumentů v prostém textu (plain text), pomocí jazyka XML se dá popsat datová struktura, jazyk HTML je využíván při tvorbě webových stránek a jazyk SVG slouží k popisu grafiky. V případě jazyků HTML, XML a SVG se jedná o jazyky založené na stejném principu, což usnadňuje jejich výuku.

Jedním z důvodů volby tohoto tématu pro diplomovou práci byla skutečnost, že téma značkovacích jazyků je obtížně uchopitelné pro žáky a neoblíbené pro vyučující.

Značkovací jazyky lze zařadit do témat „Vytváření a publikování webových stránek (HTML, CSS, PHP apod.)“ a „Algoritmizace a základy programování – rozvoj algoritmického myšlení“, jak o nich hovoří publikace *Rozvoj informačně technologických kompetencí na základních školách* (Rambousek, 2013, s. 178 – 190). Ze 14 celkových témat informačních předmětů, jejich učitelé přisuzují nejmenší význam právě těmto položkám. Zároveň jejich ochota tato témata suplovat je velmi malá, jen práce s databázemi má ještě nižší míru ochoty (Rambousek, 2013, s. 246). To se dá přisuzovat nedostatečné informovanosti učitelů a nedostatku materiálům pro výuku.

Metodika uvádí pouze základy značkovacích jazyků, čímž umožňuje se s oblastí seznámit i žákům, kteří neprojevují o oblast informačních technologií takový zájem. Žákům, kteří se naopak o oblast informačních technologií zajímají, pomůžou získané informace rozšířit jejich znalosti a nabídnout jim směr, jakým se mohou dále samostatně vzdělávat.

O historii a vývoji značkovacích jazyků se pojednává v kapitole *2.1 Historie a vývoj značkovacích jazyků*. V této kapitole jsou popsány důvody vzniku značkovacích jazyků a jejich stručný vývoj. Historie je zaměřena na vznik jazyků HTML, XML a Markdown.

Tato kapitola by měla umožnit získat základní vhled do problematiky vzniku značkovacích jazyků. Teorie potřebná pro výuku značkovacích jazyků je uvedena v kapitole 2.2 *Syntaxe a značky*. Teorie je pojatá jako základ nutný pro výuku podle této metodiky. V kapitole 3 *Učebnice a skripta* je vybráno několik publikací, podle kterých lze vyučovat značkovací jazyky, většinou se jedná o značkovací jazyk pro tvorbu webových stránek HTML. V kapitole je komentován přístup publikací k výuce značkovacích jazyků.

Ověření metodiky proběhne výukou na Základní škole Gutova a Základní škole Donovalská v Praze.

2 Značkovací jazyky

„Značkovací jazyk (markup language) je formálně definovaný jazyk, který zavádí množinu speciálních symbolů. Tyto symboly se obvykle nazývají značky (markups).“ (Crhonek, 2003). Obecným principem značkování je vkládání těchto speciálních symbolů (značek) přímo do původního dokumentu. Značky vyznačují části dokumentu a obohacují dokument o další informace. Značky mohou podle způsobu použití určovat formát dané části dokumentu nebo strukturu dokumentu. Bližší rozdělení je uvedeno v kapitole *2.3 Rozdělení značkovacích jazyků*.

2.1 Historie a vývoj značkovacích jazyků

První použití značek v textu lze datovat do šedesátých let 20. století, kdy již docházelo k používání počítačů při zpracování a publikování textů. Při profesionálním tisku se do textu vložily řídicí sekvence a příkazy pro osvitovou jednotku (specific coding), které ovlivnily výsledný formát textu. Toto řešení s sebou neslo problém, kterým byla nutnost pro každou osvitovou jednotku různých výrobců dokument vždy znovu vytvořit a použít řídicí sekvence a příkazy daného výrobce.

Jako vhodné řešení se ukázalo použití obecných příkazů (generic coding) nezávislých na výrobci osvitové jednotky, kdy byl pro každou osvitovou jednotku napsán konvertor, který obecné příkazy převedl. Takovéto použití lze považovat za předchůdce značkování a značkovacích jazyků. Velkou zásluhu mělo sdružení GCA (Graphic Communications Association), které pracovalo na projektu generického kódování GenCode. Nakonec se projekt GenCode ukázal jako slepá ulička, nicméně velké množství poznatků z tohoto projektu bylo dále využito. Vývoj GenCode mimo jiné ukázal, že různé typy dokumentů potřebují různé značky.

„Asi prvním známým značkovacím jazykem byl GML (Generalized Markup Language).“ (Kosek, 2000, s. 13) Jazyk vytvořil Charles F. Goldfarb ve spolupráci s Edwardem Mosherem a Raymondem Lorie, když pracovali ve firmě IBM. IBM se tato myšlenka velmi líbila a vývoj jazyka osamostatnila pod názvem Integrated Text Processing. V roce 1971 ho Goldfarb pojmenoval GML¹, finální podoba byla zveřejněna v roce 1973.

¹ „Když se podíváme na jména autorů, můžeme začít pochybovat o tom, jaký je skutečný význam zkratky GML.“ (Kosek, 2000, s. 13)

Hlavní odlišností GML od GenCode byla skutečnost, že GenCode bylo pouze jednoduché značkování, zatímco GML byl „koncept formálně definovaného typu dokumentu (DTD) s explicitně stanovenou strukturou a tvarem elementů“. (Crhonek, 2003)

Protože se jazyk GML osvědčil, v roce 1978 na jeho základech začala organizace ANSI (American National Standards Institute – Americký národní standardizační institut) pracovat na vývoji jazyka, který by umožnil definici vlastních značkovacích jazyků. Díky tomu, že podobnou myšlenku řešila i organizace GCA, se subjekty spojily v rámci projektu ODA (Open Document Architecture). Výsledkem jejich snahy byl specifikační jazyk SGML (Standard Generalized Markup Language).

2.1.1 SGML

Jazyk SGML byl definován v normě „ISO 8879:1986 Information Processing – Text and office systems – SGML“ z roku 1986. Standard SGML definuje, jak sestavit množinu značek pro tvorbu strukturovaných dokumentů vytvořením DTD (Document Type Definition). DTD jsou pokyny pro parser (validátor), který čte dokument a kontroluje, zda odpovídá danému DTD. Pokud dokument odpovídá DTD, mluví se o validním dokumentu (valid). Dokument, který je dobře strukturovaný (well-formed), neporušuje žádnou syntaktickou zásadu SGML. Validní dokument je vždy i dobře strukturovaný.

Díky komplexnosti jazyka SGML je ale náročné a drahé provést jeho implementaci, čímž se výrazně zbrzdilo jeho rozšíření.

2.1.2 HTML

Nejrozšířenější aplikací SGML bylo po dlouhou dobu HTML (HyperText Markup Language). První návrh pochází od autorů Tim Berners-Lee a Robert Caillau, kteří pracovali pro CERN (Conseil Européen pour la Recherche Nucléaire – Evropská organizace pro jaderný výzkum). Jazyk HTML verze 1.0 používal velmi omezené množství značek (Bartels, 2011) a byl dokončen v červenci 1993 (Crhonek, 2003).

Další vývoj jazyka HTML byl rychlý, ale nepříliš koordinovaný. Začlenění všech novinek a vytvoření jazyka HTML 2.0 se ujalo konsorcium W3C (WWWC – World Wide Web Consortium).

„Konsorcium W3C je mezinárodní komunita, kde partnerské organizace, zaměstnanci W3C a veřejnost společně vyvíjejí webové standardy. Mísí konsorcia vedeného zakladatelem webu Tim Berners-Leeem a výkoným ředitelem Jeffrey Jaffem je vést web k jeho plnému potenciálu vyvíjením protokolů a doporučení, která zaručují dlouhotrvající růst webu.“ (About W3C, 2016; W3C Mission, 2016)

Jedním z problémů HTML bylo, že autoři webových stránek začali používat popisné značky (H1 je nadpis) jako prezentační (H1 je velké tučné písmo). To vedlo k nutnosti oddělit tyto dvě funkce, což vedlo k vytvoření kaskádových stylů CSS (Cascading Style Sheets), které oddělují prezentační funkci od popisného HTML.

Od začátku roku 2016 je k dispozici nejnovější verze HTML 5 se 111 značkami². Některé značky ze starších verzí jazyka HTML již podporovány nejsou.

2.1.3 Od SGML k XML

Vzhledem ke své komplexnosti a složitosti bylo využití SGML omezené a využívali jej především velké společnosti. Velkým zastáncem využívání SGML bylo americké ministerstvo obrany, které od svých dodavatelů vyžadovalo dokumentaci právě ve formátu SGML. (Kosek, 2000) Proto se konsorcium W3C rozhodlo vytvořit jazyk, který by byl stejně dobrý jako SGML, ale byl by vhodnější pro masové použití stejně jako HTML. Při tvorbě XML (eXtended Markup Language) si konsorcium stanovilo 10 cílů³: (Bray, 1996)

1. XML by mělo být přímo použitelné přes Internet.
2. XML by mělo podporovat širokou škálu použití.
3. XML by mělo být kompatibilní se SGML.
4. Mělo by být jednoduché psát programy, které zpracovávají XML dokumenty.
5. Množství volitelných prvků XML by mělo být omezeno na absolutní minimum, ideálně žádné.
6. XML dokumenty by měly být lidmi čitelné a dostatečně jasné.

2 Seznam všech značek, které je možné použít v HTML5, nejsou zde zmíněny značky z jiných jmenných prostorů jako SVG a MathML. HTML Element Reference : HTML Tags Ordered Alphabetically. In: *W3Schools Online Web Tutorials* [online]. [cit. 2016-02-13]. Dostupné z: <http://www.w3schools.com/tags/>

3 Přeloženo z originálu, původní text je v příloze *A 10 návrhových cílů pro jazyk XML (orig. EN)*.

7. Návrh XML dokumentu by měl být vytvořen rychle.
8. Návrh XML dokumentu by měl být formální a stručný.
9. XML dokumenty by mělo být snadné vytvořit.
10. Stručnost (v XML značkování, pozn. aut. DP) má minimální význam.

Tyto cíle se objevily v dokumentu Pracovní návrh XML specifikace (Working Draft of an XML specification) uveřejněném 14. listopadu 1996. Definice vlastního jazyka XML 1.0 vyšla 10. února 1998 pod označením W3C Recommendation 10-February-1998.

Takto definované XML je podmnožinou SGML, ale na rozdíl od SGML je v něm mnoho parametrů předem určeno a nelze je měnit. Díky tomu je mnohem snazší a levnější vytvořit aplikace, které se značkovacím jazykem XML pracují.

2.1.4 Markdown

Markdown je poměrně mladý značkovací jazyk. Vytvořil ho John Gruber a první zmínka o vydání se datuje k 19. březnu 2004. Na začátku roku 2017 je aktuální verze 1.0.1 ze 17. prosince 2004. (Gruber, 2004)

Markdown je označení jak syntaxe značkovacího jazyka, tak nástroje pro programovací jazyk Perl k převodu textu do HTML. Cílem vytvoření značkovacího jazyka Markdown bylo navrhnout značkovací jazyk, jehož zdrojový kód bude pro lidi dobře čitelný a formátovaný. Dalším důležitým bodem byla jeho snadná konverze do HTML.

Jazyk Markdown používá například webový portál GitHub v souborech readme.md, které popisují jednotlivé hostované projekty.

2.2 Syntaxe a značky

V této kapitole bude zmíněno několik značkovacích jazyků, kterými se dále bude zabývat tato metodika. Uvedena bude syntaxe značkování. Také budou uvedeny základní značky daného jazyka, které budou využity v kapitole *4 Metodika*.

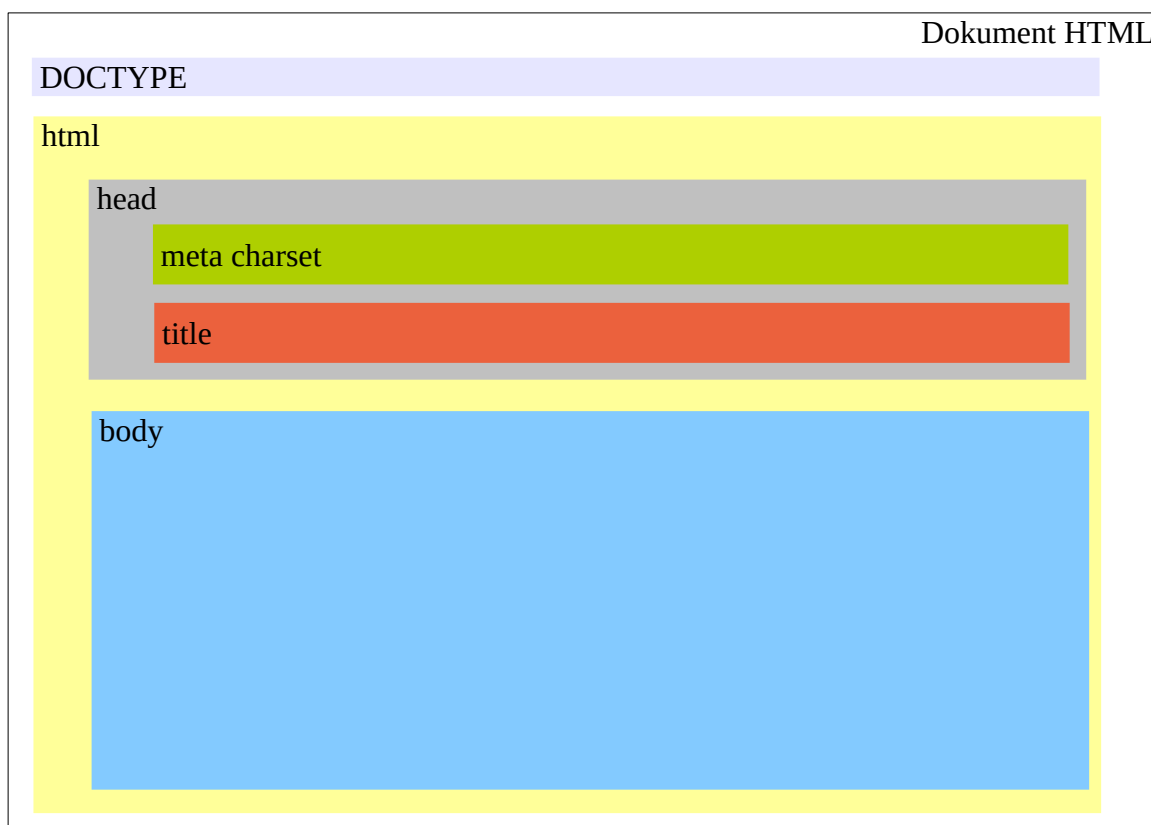
2.2.1 HyperText Markup Language (HTML)

Značkovací jazyk HTML se používá v dokumentech HTML pro popis struktury webové stránky. Využívá k tomu párové značky, kdy název značky je zapsán do lomených závorek < a > a ukončen stejným názvem, kterému předchází lomítko /. Nepárové značky mají

stejný zápis, ale vzhledem k jejich významu nemají žádný obsah a koncovou značku. Mezi takové značky patří například značka **img**, která vkládá obrázek. Na velikosti písmen, která jsou použita v názvech značek, nezáleží.

Syntaxe a povinná struktura dokumentu HTML

Základní struktura dokumentu HTML je stručně znázorněna na *Ilustrace 1*.



Ilustrace 1: Struktura dokumentu HTML

Dokument musí být uvozen značkou DOCTYPE, která určuje, o jaký dokument se jedná. Nejjednodušším způsobem je zapsat

```
<!DOCTYPE html>
```

Další značka **html** uvozuje celý dokument HTML a její ukončovací značka je až úplně na konci dokumentu. Přímými potomky této značky jsou vždy značky **head** a **body**. Značka **head** obsahuje informace o dokumentu, které se nezobrazují přímo ve vykreslovací oblasti prohlížeče, ale upravují jeho chování a nastavují například název stránky – pro ten se používá povinná značka **title**. Tento název je zobrazen v záhlaví okna prohlížeče nebo na oušku karty s vykreslenou stránkou v prohlížeči. Mezi další nejčastější značky ve

značce **head** patří nastavení kódování znakové stránky webového dokumentu, aby došlo ke správnému vykreslení znaků. Nejčastěji to bude kódování UTF-8, které se nastavuje značkou

```
<meta charset="UTF-8">
```

Značka **meta** je nepárová, proto nemusí obsahovat ukončovací značku. Značka **body** obsahuje veškerý obsah webové stránky.

Párové značky se nikdy nesmí křížit, a to ani v místech, kde by to z pohledu vykreslování webové stránky nebylo podstatné. Následující zápis je tedy chybný:

```
<strong><i>Tučné a kurzíva</strong></i>
```

Značky je nutné ukončovat v obráceném pořadí, než v jakém byly zapsány. Tedy tak, aby byly do sebe vnořeny a nekřížily se. Proto správný zápis vypadá následovně:

```
<strong><i>Tučné a kruzíva</i></strong>
```

Přehled základních značek

Nadpisy se označují značkou, která se skládá z písmene H následovaného číslicí. Číslice určuje úroveň nadpisu. Nadpis 1 tedy bude uveden mezi značky **<h1>** a **</h1>**.

Běžný text v **odstavcích** je označen značkou **<p>** z anglického paragraph. Bílé znaky jako mezery, tabulátory a nové řádky (entery) se berou vždy jako jeden znak, a to i v případě, že je jich více za sebou. Proto je možné ve zdrojovém kódu webové stránky HTML používat „grafickou“ úpravu pomocí mezer nebo tabulátorů, aniž by to mělo dopad na výsledné zobrazení. Pro zapsání **nového řádku** se používá nepárová značka **
**. Při tvorbě zdrojového dokumentu se doporučuje každou vnořenou značku odsadit o jeden tabulátor více a při ukončení tabulátor zase odebrat. Tím je zdrojový kód přehledný pro uživatele. Počítače žádnou takovou grafickou úpravu zdrojového kódu nepotřebují. Naopak se pro ně odstraňují veškeré nadbytečné mezery, tabulátory a nové řádky, aby se dosáhlo co nejmenšího zdrojového kódu. Na následující ukázce je zobrazen zdrojový kód formátovaný pro snadnou čitelnost uživateli, dále je ten samý zdrojový kód minimalizován pro použití ve výpočetní technice. Pak následuje obrázek, jak by vypadalo zobrazení takového zdrojového kódu stránky ve webovém prohlížeči. Zobrazení se vztahuje jak k formátovanému zdrojovému kódu, tak k jeho minimalizované verzi.

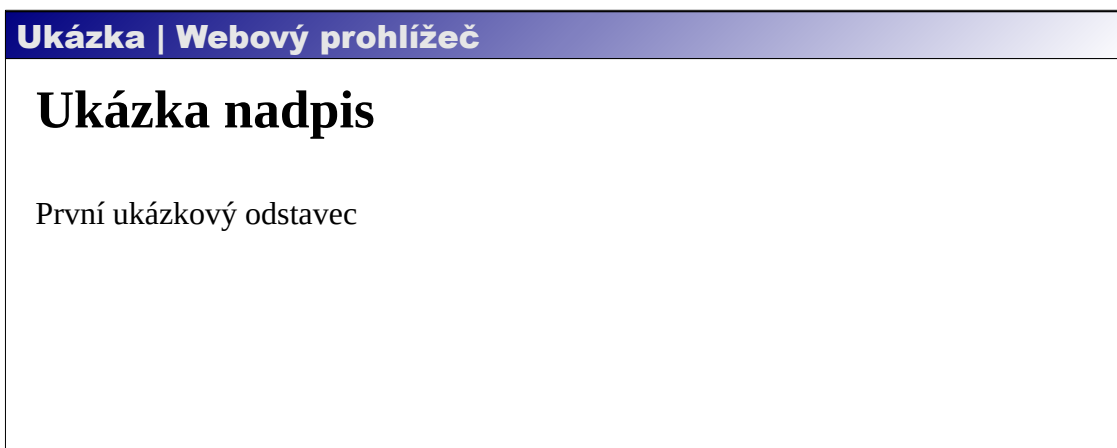
Ukázka formátovaného zdrojového kódu HTML dokumentu

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>
      Ukázka
    </title>
  </head>
  <body>
    <h1>Ukázka nadpis</h1>
    <p>
      První ukázkový odstavec
    </p>
  </body>
</html>
```

Ukázka minimalizované verze zdrojového kódu (zalomení řádku je zde jen z důvodu krátkého řádku v této práci, reálně se jedná o jeden dlouhý řádek):

```
<!DOCTYPE html><html><head><meta charset="utf-8"><title>Ukázka
</title></head><body><h1>Ukázka nadpis</h1><p>První ukázkový
odstavec</p></body></html>
```

Výše uvedené zdrojové kódy se ve webovém prohlížeči zobrazí následovně:



Pro formátování seznamů položek je možné použít dvě rozdílné značky. Pro **nečíslovaný seznam** (odrážky) se celý seznam vloží do značky **** (unordered list). Jednotlivé položky seznamu se poté uvádějí do značky **** (list item). V případě **číslovaného seznamu** se celý seznam vloží do značky **** (ordered list), položky se stejně jako v předchozím případě uvádějí do značek ****.

Značka kotva **<a>** z anglického anchor se používá hlavně pro **hypertextové odkazy**. Cíl odkazu se zapisuje jako atribut **href** značky **<a>**. **Atribut** se píše za mezeru za název značky uvnitř znaků **< a >**. Zápis se provádí uvedením jména atributu, znaku rovná se = a hodnoty do horních uvozovek ". Dalším využitím značky kotva je pojmenování místa v dokumentu, na které je poté možné odkazovat pomocí **fragmentu** v URL adrese. K tomu slouží atribut **name**. Na fragment stránky se odkazuje přidáním znaku kříž # na konec URL adresy a přidáním jména místa v dokumentu.

Příklad pojmenování místa v dokumentu:

```
<a name="poznámka">Poznámka 1:</a>
```

Odkaz vedoucí na fragment poznámka:

```
<a href="#poznámka">viz Poznámka 1</a>
```

Vkládání (linkování) **obrázků** se provádí nepárovou značkou **img**, kde atribut **src** nabývá hodnoty URL adresy obrázku.

2.2.2 eXtensible Markup Language (XML)

Značkovací jazyk XML slouží k popisu struktury dat a jejich ukládání a k jejich přenosu mezi různými druhy aplikací. Zdrojový kód zapsaný v jazyce XML by měl být snadno čitelný pro uživatele i výpočetní techniku. Díky možnosti tvorby vlastních názvů značek je jazyk velmi univerzální, ale oproti jazyku SGML je svázán pravidly tak, aby interpretace zdrojového kódu byla jednoduše implementovatelná. Hlavní cíle jazyka XML jsou zmíněny v kapitole *2.1.3 Od SGML k XML*.

Tvorba značek

Značky se uvádějí do lomených závorek **< a >** stejně jako v HTML. Část dokumentu je uvedena počáteční značkou a ukončena koncovou značkou, která má stejný název, a její název je uvozen lomítkem /. Je třeba dodržet stejnou velikost písmen v názvu značky.

Značky **<nazev>** a **<Nazev>** jsou dvě různé značky. Všechny značky jsou povinně párové, tedy každá značka musí být i ukončena. Pro značku, která nemá žádný obsah, lze použít zkrácený zápis lomítka za mezeru v uvozující značce. Značka **<nazev />** je značka s názvem „nazev“, která nemá žádný obsah a je hned i řádně ukončena. Stejně jako v HTML se značky jazyka XML nesmějí křížit, tedy při vnořování je nutné ukončovat jednotlivé značky v obráceném pořadí, než v jakém byly otevřeny.

Ukázka:

```
<nazev_kapitoly>Úvod</nazev_kapitoly>  
<podnazev />
```

Při tvorbě názvů značky je třeba dodržet tato pravidla:

- Název značky může obsahovat písmena ze všech abeced z Unicode kódování, číslice a znaky dvojtečka (:), podtržítka (_), mínus (-) a tečka (.).
- Název značky nesmí začínat číslicí, znaménkem mínus (-) nebo tečkou (.).
- Název značky nesmí začínat písmeny XML v libovolné kombinaci velikostí písmen (Xml, xMl, atp.).

Pravidla pro tvorbu názvů v originálním programátorském tvaru jsou uvedena v příloze C *Pravidla pro názvy značek XML*.

Dobré návyky při tvorbě názvů značek XML: (XML Elements, 2017)

- Znak mezera není v povolených znacích, proto značka nemůže v názvu obsahovat mezeru. Mezera může být nahrazena znakem podtržítka (_).
- Znak dvojtečka (:) má speciální význam, jeho použití se řídí dalšími pravidly. Tato pravidla nebudou v rámci této DP probírána, neboť se jedná o příliš odborné téma přesahující rámec této DP.
- Znaky tečka (.) a mínus (-) mohou být v názvu značky některými aplikacemi chybně interpretovány, proto se doporučuje tyto znaky v názvech nepoužívat.
- Ačkoliv definice jazyka XML umožňuje v názvech značek používat písmena i s diakritikou, některé aplikace nemusejí s nimi pracovat správně, proto autor této DP na základě vlastní zkušenosti doporučuje používat pouze základní znaky latinky bez diakritiky.

Každá značka může obsahovat atributy. Atributy se zapisují za název značky oddělený mezerou a to stále uvnitř znaků < a >. V případě více atributů se zapisují s mezerou mezi sebou. Zápis se provádí uvedením jména atributu, znaku rovná se = a hodnoty do horních uvozovek ". Na názvy atributů se vztahují stejná pravidla jako na tvorbu názvů značek.

Struktura XML dokumentu

Celý dokument XML musí být uzavřen v jedné rodičovské značce – stejně jako je tomu u HTML, kde tuto funkci plní značka **<html>**. Pro XML může být zvolen název značky více vystihující obsah dokumentu, naopak to nemůže být značka **<xml>**, jelikož název značky nesmí začínat „xml“.

Pro správné nastavení kódové stránky XML dokumentu by měla být na první řádku předcházející celému dokumentu definována verze XML a nastaveno kódování:

```
<?xml version="1.0" encoding="UTF-8"?>
```

Jedná se o speciální značku definující typ dokumentu, proto její zapsání neodpovídá pravidlům pro tvorbu názvů značek a jejich zápis. Značka není ukončena. Tato značka definuje XML jako verzi 1.0 a kódování nastavuje na UTF-8.

2.2.3 Markdown

Jak bylo zmíněno dříve v této DP, díky požadavku jednoduché převoditelnosti do HTML jazyk Markdown s HTML velmi úzce spolupracuje. V Markdownu se mohou používat i značky HTML. Jediným omezením je, že v blokových značkách HTML se již neprovádí konverze z jazyka Markdown. To neplatí pro řádkové značky HTML, jako například ****, kde se může i nadále používat syntaxe Markdown.

Základní syntaktická pravidla

Jazyk Markdown si klade za cíl vypadat čitelně a strukturovaně i ve své zdrojové podobě. Zároveň je přizpůsoben jednoduchému převodu zdrojového kódu do výsledné formátované podoby pomocí HTML. Z obou těchto požadavků vznikají požadavky na syntax jazyka.

V následujících odstavcích je pro představu uvedeno několik základních syntaktických pravidel pro psaní běžného textu a nejpoužívanější formátování jako jsou nadpisy, seznamy, oddělení pomocí horizontální čáry a tučné písmo či kurzíva. Další pravidla je možné najít přímo na stránce projektu Markdown na:

<https://daringfireball.net/projects/markdown/syntax>

Běžný text

Běžný text není nijak označen. Stejně jako v HTML, ani v jazyku Markdown nezáleží na množství mezer mezi slovy. Odstavce se oddělují prázdným řádem. Podle definice (Gruber, 2004) je zalomení na konci řádku ignorováno, ale mnoho Markdown editorů převádí každé zalomení řádku ve zdrojovém kódu na ruční zalomení řádku. Podle definice se ruční zalomení provádí přidáním dvou a více mezer na konec řádku.

Nadpisy

Pro formátování nadpisů je možné použít dva rozdílné způsoby zápisu. První zápis, graficky více oddělený ve zdrojovém kódu, umožňuje formátování pouze stylů Nadpis 1 (**h1**) a Nadpis 2 (**h2**). Nadpis 1, resp. Nadpis 2 se formátuje podtržením na dalším novém řádku pomocí rovná se =, resp. mínus -. Pro lepší a hezčí zdrojový text je vhodné podtrhnout nadpis v celé své délce. Před nadpisem musí být volný řádek, nebo musí být nadpis umístěn na začátku dokumentu.

Ukázka:

Toto je Nadpis 1

=====

Toto je Nadpis 2

Druhou možností je odsadit text příslušným počtem křížků (#). Například pro Nadpis 1 je to jeden křížek a pro Nadpis 6 se jedná o šest křížků.

Ukázka:

Toto je Nadpis 1

Toto je Nadpis 2

Toto je Nadpis 6

Alternativně může být pro lepší vzhled zdrojového kódu použito označení křížky i na konci nadpisu.

Toto je Nadpis 2

Seznamy

Markdown podporuje dva typy seznamů: řazené číslované seznamy a nečíslované seznamy. Pro vytvoření číslovaného seznamu je nutné, aby každá položka byla na vlastním řádku a začínala číslovkou následovanou tečkou a mezerou.

1. Otevřít
2. Upravit
3. Uložit

Nečíslovaný seznam také požaduje, aby každá položka byla na vlastním řádku. Řádek musí začínat hvězdičkou *, plusem +, nebo mínusem - a poté musí následovat mezera.

- * Modrá
- * Červená
- * Zelená

Horizontální čára

Oddělovat části dokumentu je možné pomocí horizontální čáry, která se dá vytvořit zapsáním tří a více stejných znaků hvězdička * nebo mínus - na samostatném řádku. Mezi znaky mohou být mezery.

Ukázka:

Tučné a kurzíva

Pro zvýraznění textu tučně nebo kurzívou se použije ohraničení daného textu hvězdičkou *, nebo podtržítkem _. Pro tučné písmo se uvádějí dva stejné znaky za sebou, pro kurzívu jen jeden.

Ukázka:

kurzíva označení hvězdičkou nebo _kurzíva označená podtržítkem_

tučné písmo nebo __tučné písmo__

2.2.4 Scalable Vector Graphics (SVG)

Scalable Vector Graphics, zkráceně SVG, je jazyk sloužící k popisu dvourozměrné vektorové grafiky v XML. SVG umožňuje zápis tří druhů vektorových grafických objektů. Umožňuje popsat tvary skládající se z přímých čar a křivek, obrázky a text.

Tato DP se zaměří na popsání příkazů pro kreslení kružnice, obdélníku a polygonu. Výhodou formátu SVG je jeho čitelnost ve zdrojové podobě, jelikož používá zápisu jazyka XML, a také možnost vložit SVG zdrojový kód přímo do HTML zdrojového kódu webové stránky. Všechny dnešní webové prohlížeče umí vektorovou grafiku SVG zobrazit.

Celá grafika je označena značkou **<svg>**. Mohou být doplněny údaje o velikosti kreslicího plátna atributy **height** a **width**. Při tvorbě grafiky pomocí jazyka SVG je třeba dodržovat všechna pravidla jazyka XML. Každá značka musí být ukončena, u SVG je velmi výhodné využívat zkrácený zápis ukončení pomocí lomítka na konci otevírací značky.

Příklad SVG dokumentu:

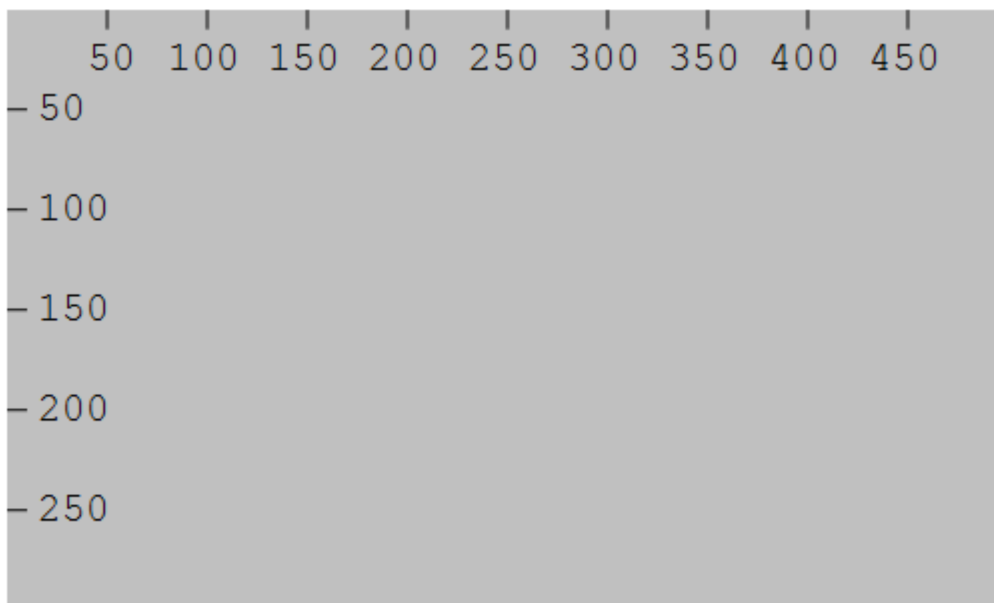
```
<svg height="100" width="200">
```

```
...
```

```
</svg>
```

Veškeré rozměry height, width, souřadnice X a Y, poloměry a jiné rozměry se udávají v pixelech bez přípony px. Vzhledem k tomu, že SVG je vektorová grafika, má velikost v pixelech vliv jen na její výchozí velikost vykreslení. Vektorovou grafiku lze zmenšovat či zvětšovat bez ztráty kvality (na rozdíl od bitmapové grafiky). Z tohoto důvodu lze na čísla rozměrů nahlížet i jako na poměry mezi velikostmi jednotlivých tvarů. Souřadnice jsou

vztaženy k levému hornímu rohu kreslicího plátna. Kladné hodnoty jsou vpravo a dolů. Ukázka kartézských souřadnic je uvedena na *Ilustrace 2* níže.



Ilustrace 2: Souřadnice SVG (Holica, 2007)

Každý grafický tvar, u kterého to má smysl, může mít barevnou výplň. Barva výplně se určuje atributem **fill**, který může nabývat hodnot anglicky označených barev (red, green, blue, yellow, pink, ...), hexadecimálního označení barvy #RRGGBB nebo zápisu barvy do funkce RGB(red, green, blue), případně s určením průhlednosti RGBA(red, green, blue, alfa).

Kružnice, kruh

Pro nakreslení kružnice nebo plného kruhu se používá značka **<circle>**. Ke značce je třeba doplnit atributy určující pozici středu kružnice a její poloměr. Pro nakreslení kruhu je nutné uvést atribut **fill** určující barvu kruhu.

Atributy:

- **cx, cy** Souřadnice X a Y středu kružnice
- **r** Poloměr kružnice
- **fill** Barva kruhu

```
<circle cx="100" cy="100" r="50" fill="yellow" />
```

Obdélník

Značkou pro nakreslení obdélníku je značka **<rect>**. Atributy se určí pozice levého horního rohu obdélníku, jeho šířka a výška.

Atributy:

- **x, y** Souřadnice levého horního rohu obdélníku
- **width** Šířka obdélníku
- **height** Výška obdélníku
- **fill** Barva obdélníku

```
<rect x="20" y="40" width="30" height="10" fill="yellow" />
```

Polygon

Polygon, což je libovolný n-úhelník jako například trojúhelník, čtyřúhelník, pětiúhelník, se kreslí pomocí značky **<polygon>**. Důležitým atributem je atribut **points**, který obsahuje seznam souřadnic vrcholů X,Y oddělených mezerou. V příkladě níže je vidět značka k nakreslení rovnoramenného trojúhelníku, který má vrcholy [0; 0], [150; 100] a [0; 200].

Atributy:

- **fill** Barva polygonu
- **points** Seznam vrcholů oddělených mezerou

```
<polygon fill="red" points="0,0 150,100 0,200" />
```

2.3 Rozdělení značkovacích jazyků

Značkovací jazyky můžeme rozdělit podle způsobu zápisu a dle výsledného použití do skupin popisných jazyků, procedurálních jazyků a prezentačních jazyků. Nejedná se o striktní definice, skupiny se mohou vzájemně prolínat a nejsou jednoznačně vyhraněny.

2.3.1 Popisné jazyky

Značkovací jazyky popisné, nebo také deskriptivní (z anglického descriptive), rozdělují dokument na jednotlivé informační části a popisují, co daná informace znamená. Příkladem může být XML nebo HTML. HTML podává informace, které části jsou například nadpis, odstavec, obrázek. V případě XML, které umožňuje použití vlastních značek, může určovat například, které informace jsou jméno, příjmení, ulice, telefonní číslo.

Příklad HTML:

<code><h1>Diplomová práce</h1></code>	Označení Nadpis 1 (h1)
<code><p>Toto je text odstavce doplněn výplňkovým textem. Dolor sit amet, consectetur adipiscing elit.</p></code>	Text odstavce (p)
<code></code>	Obrázek (img)

Příklad XML:

<code><osoba></code>	Začíná informace osoba
<code><jmeno>Jaroslav</jmeno></code>	Její jméno
<code><prijmeni>Novák</prijmeni></code>	Její příjmení
<code><ulice>Dlouhá</ulice></code>	Ulice, kde bydlí
<code><telefon>+420 123 456 789</telefon></code>	Její telefon
<code></osoba></code>	Konec informace o osobě

2.3.2 Výkonné jazyky

Výkonné značkovací jazyky, jinak také procedurální (z anglického procedural), obsahují programovací příkazy, kterými je možné řídit formát dokumentu. Například součástí jazyka TeX je i jednoduchý programovací jazyk BASIC. Díky příkazům umožňuje tato skupina jazyků velmi přesně popsat vizuální vzhled dokumentu. Někdy jsou tyto jazyky označovány zkratkou WYWIWYG (What You Want Is What You Get – Co chcete, to dostanete) v reakci na běžné textové editory označované WYSIWYG (What You See Is What You Get – Co vidíte, to dostanete). Dalšími procedurálními jazyky jsou například Troff nebo PostScript.

Příklad LaTeX (nadstavba TeX):

<code>\documentclass{article}</code>	Použití přednastaveného formátu
<code>\usepackage{graphicx}</code>	Použití balíčku s podporou grafiky
<code>\begin{document}</code>	Zde začíná „document“
<code>\title{Nadpis \LaTeX{}}</code>	Bude mít nadpis „Nadpis“ následované logem L ^a T _e X
<code>\author{David Dubský}</code>	Autor dokumentu bude „David Dubský“
<code>\maketitle</code>	Vytvoř úvodní stránku
<code>\begin{abstract}</code>	Zde začíná „abstract“

Tady je abstrakt. Class aptent taciti sociosqu ad litora.

`\end{abstract}` Zde končí „abstract“

`\section{Introduction}` Zde začíná sekce „Introduction“

Sekce Úvod a další výplňkový text. Duis risus. Excepteur sint occaecat cupidatat.

`\end{document}` Konec dokumentu

2.3.3 Prezentační jazyky

Úkolem prezentačních značkovacích jazyků, z anglického presentational, je řídit vzhled dokumentu pro zobrazení nebo vtištění, aniž by nějak specifikovaly strukturu dokumentu. Jako příklad lze uvést textové editory, z jejich zástupců program WordPad používající formát RTF (Rich Text Format).

Příklad RTF:

Zdrojový kód dokumentu:

```
{\rtf1\ansi\ansicpg1250\deff0\deflang1033{\fonttbl{\f0\fnil\fcharset0 Arial;}}\b\fs20 Zkou\'9aka \b0 textu\par}
```

Popsání zdrojového kódu:

{	Začátek dokumentu
\rtf1 \ansi \ansicpg1250 \deff0	Definice verze RTF, znakové stránky a jazyka
\deflang1033	
{	
\fonttbl	Definice tabulky písem
{	
\f0 \fnil \fcharset0 Arial;	Jedno použité písmo
}	
}	
\b	Přepnutí na tučné písmo
\fs20	Nastavení velikosti textu 20 půlbodů

Zkouš ⁹ ka	Text „Zkouška“ (jelikož RTF podporuje pouze 7-bitů, je znak š přepsán příkazem \ ⁹)
\b ⁰	Zrušení tučného písma
textu	Text „textu“
\par	Konec odstavce
}	Konec dokumentu

2.3.4 Shrnutí

Jak již bylo nastíněno v úvodu ke značkovacím jazykům, jejich rozdělení do přesných skupin není možné. U některých, například Tex, je zařazení více jednoznačné. U XML naopak záleží na konkrétní aplikaci značkovacího jazyka. Značkovacím jazykem XML může být skvěle popsána datová struktura, tabulka. Také může být popsán dokument se zaměřením na jeho budoucí vykreslení a zobrazení, kde se již prolíná popisný prvek a s prvkem prezentačním, jako v případě XHTML. Naopak při použití XML jako jazyka pro popis vektorové grafiky, jako v případě jazyka SVG, může být jazyk zmíněn spíše jako výkonný jazyk, kdy jednotlivé značky vykreslují grafické objekty.

3 Učebnice a skripta

V této kapitole je komentován přístup vybraných učebnic, skript nebo odborných knih k výuce značkovacích jazyků. Je analyzován přístup ke značkovacím jazykům, platnost přístupu a jeho aktuálnost a rozsah textu věnovaného těmto jazykům v rámci celé učebnice. Je vzato v potaz, že učebnice se starším datem vydání nemohou obsahovat nejaktuálnější informace.

3.1 Informatika pro základní školy: 1. díl Základy práce s PC

Svým určením je učebnice Informatika pro základní školy (Vaníček, 2004) zaměřena na základní ovládání a vzhled počítače a jeho softwaru. Učebnice zahrnuje i práci s webovými stránkami, ale pouze na nejzákladnější úrovni ovládání webového prohlížeče. Dále vysvětluje základní princip fungování webových stránek propojením pomocí hypertextových odkazů, kde na sebe jednotlivé stránky odkazují. Propojení pomocí odkazů učebnice ilustruje obrázkem se stránkami textových dokumentů. Modře obarvené a podtržené hypertextové odkazy a červené šipky znázorňují odkazy na ostatní dokumenty.

Učebnice vzhledem k popisu pouze nejzákladnějších principů fungování webových stránek nezmiňuje žádné aspekty značkovacího jazyka.

3.2 Informatika a výpočetní technika pro střední školy

Učebnice Informatika a výpočetní technika pro střední školy (Roubal, 2010) obsahuje v kapitole „5. Využívání služeb Internetu“ na straně 93 podkapitulu s názvem „5.1 WWW – World Wide Web“, ve které definuje některé pojmy z oblasti kódování webových stránek a jejích technologií. Například definuje hyperlink jako „odkaz na nějakou adresu (URL)“ (Roubal, 2010, s. 93). K pojmu HTML uvádí pouze, že prohlížeč obdrží HTML kód stránky a „podle pokynů v HTML souboru vše zobrazí“ (Roubal, 2010, s. 93). Jako zajímavost uvádí obrázek první webové stránky, která obsahuje pouze text s odkazy.

Učebnice popisuje kompletní pohled na výpočetní techniku a snaží se seznámit čtenáře s jejími aspekty, aniž by popisovala vše do velkých detailů. Proto je téma značkovacích jazyků v učebnici zmíněno jen okrajově.

3.3 Výpočetní technika – Hardware a Software

Výpočetní technika – Hardware a Software autora Klementa (2001) jsou skripta určená hlavně pro posluchače Pedagogické fakulty Univerzity Palackého v Olomouci. Skripta jsou podle autora koncipována tak, aby umožnila čtenáři získané znalosti aplikovat při konfiguraci počítačů a počítačových sítí.

První zmínka o značkovacích jazycích se vyskytuje v „Kapitole třetí – Software“, v podkapitole „3.4.3 Vyšší programovací jazyky“ na straně 95, kde je jazyk SGML uveden jako příklad dalších programovacích jazyků. Kapitola o softwaru je spíše seznamem pojmů, názvů a zkratk, které stručně vysvětluje. Obsah jednotlivých kapitol je řazen chronologicky.

„Kapitola čtvrtá – Síť a internet“ se věnuje webovým stránkám na straně 121 v podkapitole „4.5 WWW“. Princip webu popisuje jako „nový typ klientského programu, který běží na počítači u uživatele. ... jedná se o program, který dekoduje poměrně úsporná data a jednoduché instrukce zaslané serverem, sestaví z nich stránku jako ze stavebních cihliček a tu zobrazí ve svém hlavním okně.“ (Klement, 2001, s. 121) Popis webu je poněkud nepřesný. Pojmenování „poměrně úsporná data“ (Klement, 2001, s. 121) je nešťastné, jelikož z důvodu povahy zápisu HTML kódu nelze o úspornosti příliš mluvit. Vzhledem k tomu, že dále se autor vyjadřuje k další komprimaci (= úprava dat, jejímž cílem je zmenšit objem dat beze ztráty informace, pozn. aut. diplomové práce), kdy „přes Internet cestují pouze nezbytně nutné stavební prvky, často ještě v komprimovaném tvaru“, nelze výše uvedené tvrzení zaměnit za komprimaci síťového přenosu.

Dále skripta zmiňují „prostou genialitu webu“, propojení pomocí odkazů, nazývaných hyperlinky, které je doplněno schématem propojených dokumentů. Autor také zmiňuje, že vývoj webu nikdo neurčuje, díky čemuž je vývoj webu fantasticky rychlý.

Díky rychlému tempu vývoje v oblasti informačních technologií je mnoho informací v těchto skriptech, vydaných v roce 2001, již neaktuální. Tato skripta neposkytují žádné podrobnější informace ohledně značkovacích jazyků.

3.4 Internet a tvorba jednoduchých WWW stránek

Internet a tvorba jednoduchých WWW stránek (Vrbík, 2001) jsou skripta určená pro studenty učitelství a kladou si za cíl usnadnit studentům jejich první kroky v seznamování

s Internetem. Vzhledem k datu publikování skript v roce 2001 jsou informace pro dnešní dobu již zastaralé. Polovina skript se věnuje Internetu obecně, druhá polovina se pak věnuje tvorbě jednoduchých WWW stránek. K vytvoření stránek autor používá Microsoft Word verze 2000, přičemž občas doplňuje rozdíly pro uživatele starší verze 97.

Autor popisuje webovou stránku na straně 28 v „Části I – Internet“ v kapitole „7 – Shrnutí nejdůležitějších pojmů“ jako „soubor vytvořený pomocí jazyka HTML. Pomocí HTML lze zakódovat např. nadpisy a věty a přiřadit jim formátování.“

Dále v první kapitole „Úvod k tvorbě WWW stránek ve Wordu“ v „Části dvě – Tvorba jednoduchých WWW stránek“ vysvětluje, jak musí být stránka „naprogramována“ v jazyce HTML. Že se používají tzv. značky (tagy) a soubor lze přečíst v libovolném textovém editoru. Výslovně uvádí, že „jazyk HTML je neobyčejně přizpůsobivý různým operačním systémům.“

Vzhledem k tomu, že je k tvorbě webových stránek použit editor Word, většina kapitol jsou návody na obsluhu tohoto editoru v režimu tvorby WWW stránek a není zmiňován zdrojový kód stránek.

První zmínky o tvaru značkování v HTML se nachází v kapitole „11.2 Relativní versus absolutní hypertextové odkazy“ na straně 55, kde autor vysvětluje rozdíly mezi absolutním a relativním adresováním a navrhuje uživateli, jak změnit zdrojový kód souboru a upravit adresy na relativní. V této kapitole uvádí příklady značek **a** a **img** a vysvětluje úpravu **href**, resp. **src** atributu.

V kapitole „19 Odstraňování některých problémů“ na straně 71 autor popisuje způsob rozdílu generování webových stránek mezi programy Word 97 a Word 2000, kdy „zatímco v Office 97 byl výsledný generovaný kód složen z "čistého" HTML kódu, v Office 2000 se generuje daleko obsáhlejší kód XML.“ Dále se také zmiňuje o nejjobecnějším značkovacím jazyku SGML, pomocí kterého je definován značkovací jazyk HTML. Také upozorňuje na náročnou implementaci jazyka SGML a jeho „mdlou podporu“ pro rozdílné znakové sady.

V předposlední kapitole „20 Základy HTML“ stručně popisuje, co je jazyk HTML, a uvádí příklady jednoduchých HTML stránek s vysvětlením jednotlivých značek HTML, které jsou v příkladech použity.

Vzhledem k zaměření skript na tvorbu webových stránek pomocí textového editoru Microsoft Word nelze očekávat, že se skripta budou věnovat pečlivému popisu značkovacího jazyka. Ale i tak základní principy a značky skripta uvádějí a vysvětlují.

3.5 Praktické činnosti pro 6.-9. ročník základních škol

Učebnice Práce s počítačem (2) (Rambousek, 2003) je součástí řady učebnic Praktické činnosti pro 6. – 9. ročník základních škol. Tento díl se věnuje především počítačovým sítím, obsluze počítače a používání programů. Kapitola „8.7 Tvorba WWW stránek“ se na straně 140 věnuje struktuře a tvorbě webových stránek. Učebnice je koncipována jako plynulý text dělený do odstavců s občasným vloženým obrazovým doprovodem.

Kapitola „8.7 Tvorba WWW stránek“ začíná formátem webových stránek, jež jsou psány pomocí „zvláštního značkovacího jazyka HTML, který umožňuje úsporný a všeobecně srozumitelný textový popis www stránky.“ Je vysvětlena zkratka HTML, dále princip vytváření hypertextové (nelineární) struktury www stránek pomocí odkazů. Kapitola vysvětluje způsoby vytvoření a editace zdrojových kódů stránek HTML ve WYSIWYG editorech, kde uvádí i příklady takových editorů. Uvádí i přímé editace zdrojových kódů, kde opět zmiňuje příklady konkrétních editorů.

Dále následuje popis struktury zdrojového kódu webové stránky pomocí značek jazyka HTML. Jsou rozlišeny párové a nepárové značky. Kapitola popisuje práci se základními značkami stavby HTML dokumentu a dále s nejpoužívanějšími značkami, jejichž použití ilustruje na příkladu. K některým značkám uvádí i použití parametrů, které upravují jejich význam. Autoři končí tuto kapitolu zmínkou o kaskádových stylech (CSS) a existujících možnostech vytvářet dynamické webové stránky pomocí skriptů. Uvádí, že skripty je možné vytvářet buď na straně prohlížeče jako JavaScript, nebo na straně serveru, kde zmiňují Active Server Pages, Java Server Pages a Personal Home Page (PHP). Tuto kapitolu uzavírá zmínka o tvorbě animovaných stránek pomocí programu Macromedia Flash.

Uvedení do praxe práce se značkami podle této učebnice by vyžadovalo ještě získání dalších informací. Učebnice popisuje základy tvorby webových stránek a uvádí základní značky.

3.6 Informatika pro základní školy: 3. díl

Řada učebnic Informatika pro základní školy se skládá ze tří dílů, v posledním třetím dílu (Kovářová, 2004) se na straně 49 nachází kapitola „Tvorba webových stránek“. Celá řada učebnic je zpracována barevně, s barevným pruhem po straně s obdélníky „Pamatujte“.

V úvodu kapitoly „Tvorba webových stránek“ učebnice vysvětluje princip fungování webových stránek s nutností oddělení textu od obrázků. V podkapitole „Jazyk HTML“ učebnice čtenáři obecnými slovy vysvětluje programování v HTML a ukazuje příklad hlavičky HTML dokumentu vytvořeného v editoru Microsoft FrontPage 5.0 bez dalšího popisu. Zájemce o naučení se psaní přímo v kódu HTML učebnice odkazuje na internet, kde si mají vyhledat jiné publikace. Další text učebnice se věnuje výhradně práci s WYSIWIG editorem Microsoft FrontPage. Velká část kapitoly je zaměřena na vkládání obrázků a formátování jejich zobrazení. Učebnice uvádí ne úplně vhodný příklad zdrojového kódu webové stránky. Kód neobsahuje žádné tělo, ale zbytečné množství hlaviček, což může být matoucí a zbytečně to odvádí pozornost od základní stavební struktury HTML dokumentu.

Učebnice se věnuje především tvorbě webových stránek v editoru Microsoft FrontPage, o zdrojovém kódu se zmiňuje jen obecně.

3.7 S počítačem nejen k maturitě: 2. díl

„Druhý díl (S počítačem nejen k maturitě, pozn. autora DP) je zaměřen na speciálnější témata. Stěžejní kapitoly jsou ... tvorba internetových stránek různými metodami...“ (Navrátil, 2004, str. 4). Popisu HTML stránek je věnováno 40 stránek z celkových 175 stránek druhého dílu učebnice. Hlavními kapitolami pojednávajícími o HTML jsou kapitoly „HTML – Principy a vytváření“, „Tvorba stránek pomocí FrontPage“ a „Tvorba stránek pomocí HTML tagů“.

V kapitole „HTML – principy a vytváření“, začínající na stránce 50, učebnice vysvětluje, že HTML, neboli „HyperText Markup Language, což by se dalo přeložit jako "nadtextový značkovací jazyk", používá na rozdíl od jiných programovacích jazyků „namísto příkazů takzvané tagy neboli značky.“ (Navrátil, 2004, s. 50) Také hned v úvodu kapitoly je zmíněno, že spíše než o „programování“ se v případě webových stránek mluví o „tvorbě“. Dále učebnice vyzdvihuje to, že zdrojový kód webové stránky je textový, tudíž je možné

webovou stránku editovat i v běžných textových editorech a také je „neobyčejně "přizpůsobivá" různým počítačům, operačním systémům a prohlížečům“.

Z hlediska značkovacích jazyků je zajímavá malá podkapitola „Tagy“ v kapitole „HTML – principy a vytváření“ na straně 51. V této podkapitole je stručně vysvětleno, že tag je vlastně značka, podle které se prohlížeč řídí. Učebnice uvádí příklady jako „teď zarovnej text doprava“, „nyní zvětši písmo“. Také zmiňuje výhodu, že pokud je ve stránce chybný tag, prohlížeč ho ignoruje a nedojde k tomu, že by celý dokument nebyl načten. V závěru kapitoly je zmíněno, že „tagy se v HTML dokumentu uzavírají do znaků < >, kde < značí začátek tagu a > konec tagu.“

Další kapitola, začínající na straně 52 a nazvaná „Tvorba stránek pomocí FrontPage“, popisuje obsluhu programu FrontPage včetně úpravy uživatelského rozhraní. Kapitola zahrnuje témata jako formátování textu, odrážky a číslování, práce s obrázky, tabulky a vytváření hypertextových odkazů.

Na straně 68 začíná kapitola „Tvorba stránek pomocí HTML tagů“. Na začátku rozděluje tagy do dvou skupin, a to na skupinu párových a samostatných tagů. Jako příklad párového tagu uvádí tag ****, **** včetně ukázky a popisu fungování. Jako příklad samostatného tagu uvádí **<P>**, opět s ukázkou a popisem fungování. Pro editaci na operačním systému Windows učebnice doporučuje používat program Poznámkový blok. Učebnice dále doporučuje mít souběžně otevřený webový prohlížeč s načtenou stránkou, aby bylo vidět, jak v dané chvíli webová stránka vypadá. Kapitola obsahuje také popis vytváření formulářů.

Po podkapitole „Kostra dokumentu“, která popisuje strukturu HTML dokumentu a vysvětluje funkčnosti jednotlivých částí dokumentu, následuje seznam tagů, jejich popis a vysvětlení. V části seznam tagů také dojde k vysvětlení používání atributů tagů, jsou uváděny také příklady nejpoužívanějších atributů u některých tagů.

Po tomto seznamu tagů přichází kapitola „Dynamické WWW stránky“ (na straně 82), jež pojednává o tvorbě formulářů. Tím kapitoly o HTML jazyku končí.

Učebnice názorně a přehledně pojednává o tématu HTML jazyka včetně uvedení jak praktického řešení pomocí programu FrontPage, tak také popisu stavby HTML dokumentů

pomocí HTML jazyka. U všech pojmů jsou uvedeny příklady a názorné ukázky. Přístup k tématu je opět především popisný.

3.8 CSS a XHTML: tvorba dokonalých WWW stránek krok za krokem

Kniha CSS a XHTML (Procházka, 2011) by měla čtenáře seznámit se základními principy internetu. Obsahuje tedy nejen informace o vývoji stránek, ale čtenář dostane kompletní studijní materiál, který jej provede tvorbou webu, umístěním stránek na internet a laděním v různých internetových prohlížečích.

Na začátku knihy je seznam důležitých pojmů, především zkratk, s obsáhlejším popisem, co daný pojem znamená. U pojmu HTML je zmíněno především kódování pomocí značek, které uzavírají části dokumentu podle významu, a dále princip, že názvy značek se uvádějí mezi úhlové závorky. Dalším pojmem je XHTML, což je zkratka anglického „eXtensible HyperText Markup Language“. Tento jazyk měl původně nahradit HTML. U pojmu „HTML vs. XHTML – rozdíly“ jsou uvedeny základní rozdíly mezi HTML a XHTML, například povinné ukončování tagů, tagy malými písmeny, vhodnost uvádět XML deklarace a hodnoty atributů v závorkách.

V dalších kapitolách se prolíná popis různých technologií důležitých pro provoz internetových stránek a práci s nimi. Kniha také uvádí nejpoužívanější programy, které mohou s danou technologií čtenáři pomoci. Pro editaci webových stránek kniha doporučuje program PSPad, kde vyzdvihuje především jeho bezplatnost, dobrou práci s textem, podporu více kódování, šablony a nástroje pro práci s kódem HTML jako komprese či validace. Kniha dále uvádí všechny ukázky a návody právě pro program PSPad.

Na straně 33 začíná 2. kapitola „Základy jazyka XHTML“. Hned v první podkapitole „Zdrojový kód HTML stránek ve značkách“ je čtenář uveden do problematiky zápisu zdrojového kódu pomocí klasického ASCII textu v souborech s příponou HTM nebo HTML. Vyzdvihuje také výhodu tolerance chyb v prohlížeči, který je v případě opravitelné chyby schopen stránku zobrazit a nezobrazí jen chybové hlášení. Kniha mezi další výhody uvádí to, že zdrojový kód již není třeba dále kompilovat a v tomto formátu se kód předá prohlížeči.

Další podkapitola začíná vysvětlením principu značek a vysvětlením rozdílu mezi párovými a nepárovými značkami. Uváděné příklady jsou obvykle uvedeny v kódování

HTML a následuje popis převodu do XHTML. Kapitoly na rozdíl od ostatních učebnic a skript nejsou řazeny a děleny podle značek, ale podle jejich použití. Proto se v této knize setkáme s kapitolami jako „Práce s odstavci“, kde jsou uvedeny a vysvětleny značky **<P>**, **<DIV>**, **<CENTER>**, nebo „Fonty a velikost písma“, kde je představena značka **** včetně atributů **SIZE**, **FACE** a **COLOR**.

Ve třetí kapitole „Naše první WWW stránka“ kniha zmiňuje formátování pomocí CSS. V kapitole čtvrté „Kompletní průvodce XHTML tagy a formuláři“ jsou již všechny příklady jen v XHTML jazyce. Kniha končí představením jazyka (X)HTML na straně 105 a popisem formulářů.

Kniha pojednává srozumitelně a hodně do šíře o tématech souvisejících s tvorbou webových stránek. Zároveň zachází do hloubky jen tak, kolik je třeba, a nepřetěžuje čtenáře informacemi, které nezužítkuje při tvorbě stránek. Z pohledu značkovacích jazyků začíná s HTML a dále rozšiřuje příklady pro XHTML. Rozdělení kapitol do logických celků podle použití značek působí velmi přehledně a zároveň i čtivě. Celková orientace knihy je zaměřena více na tvoření webových stránek než na naučení jazyka (X)HTML.

3.9 Shrnutí

Celkem bylo posouzeno 8 publikací. Kromě dvou publikací se jedná o publikace vydané před rokem 2004, čemuž odpovídá aktuálnost uvedených informací. Je proto pochopitelné, že vzhledem k rychlému vývoji v IT oblasti některé publikace pojednávají o již starém způsobu tvorby webových stránek.

Přímo o značkovacích jazycích zmíněné publikace příliš neinformují. Jen v publikacích zaměřených na tvorbu webových stránek je možné získat informace o značkovacím jazyku HTML.

Jelikož v Rámcovém vzdělávacím programu se vyskytují spíše požadavky na práci se značkovacím jazykem HTML než se soubory v XML, i zaměření učebnic reflektuje tyto potřeby a věnuje se převážně HTML. Dalším důvodem je skutečnost, že v době tvorby výše uvedených publikací byla problematika XML nová a její rozvoj procházel rychlým vývojem.

4 Metodika

Metodika je pojata jako systém problémových úloh, které na sebe navzájem navazují. Úlohy pracují s nákupním seznamem, který je upravován a formátován různými značkovacími jazyky.

Metodika začíná jednoduchou úlohou, kde se žáci poprvé setkají se značkami v textu. Dále jsou uvedeny jazyky Markdown, HTML, XML a na závěr je uvedena jedna úloha k jazyku SVG. Teorie značkovacích jazyků je uvedena v kapitole 2.2 *Syntaxe a značky* začínající na straně 13 a v metodice již není znovu rozebírána.

4.1 Formátování textu podle značek

V této úloze dostanou žáci vytištěný text, který obsahuje značky formátování. Úkolem žáků je text podle značek upravit. Žáci budou potřebovat tužku nebo propisku a pastelku.

Cíl

Cíle této úlohy jsou seznámit žáky s možností značkování a vyzkoušet si vliv použití značek v textu na výsledný vzhled dokumentu.

Tato úloha rozvíjí kompetence žáka v oblasti orientace v textu, kde jsou doplněny další znaky a symboly označující formát textu. Žáci by měli být schopni se orientovat v textu a být schopni rozlišit mezi obsahem textu a značkami upravujícími formátování textu.

Zadání

Zadání vhodné k oboustrannému tisku na stranu A4 je v souboru *formatovani.pdf*. Součástí souboru je i text pro další úlohu 4.2 *Formátovaný dokument*, s kterým se následně bude také pracovat, a proto je vhodné jej vytisknout současně se zadáním této úlohy. Text se po vytištění rozdělí na dvě strany A5 a každý žák obdrží jednu oboustrannou kopii formátu A5.

Připravte si tužku nebo propisku a pastelku. V následujícím textu označte části textu podle následujících pravidel:

- Text, který se nachází mezi #, vybarvi pastelkou
- Text, který se nachází mezi jednou hvězdičkou, zakroužkuj
- Text, který se nachází mezi dvěma hvězdičkami, podtrhni

Česká republika

Česko, úředním názvem *Česká republika*, je stát ve ****střední Evropě****. Jako formálně svrchovaný národní stát vznikla tehdejší ****Česká socialistická republika**** 1. ledna 1969 v rámci federalizace Československa. Od 6. března 1990 nese tento stát název Česká republika. Dne 1. ledna 1993 se v souvislosti se zánikem *Československa* stal subjektem mezinárodního práva a od téhož data je účinná ****ústava****, podle jejíž preambule navazuje novodobý český stát na tradice státnosti Československa a bývalých zemí *Koruny české*. ****Hlavou státu**** je *prezident* republiky, vrcholným a jediným zákonodárným orgánem je dvoukomorový Parlament České republiky.

Vyhodnocení

Posuzuje se, jak žáci vybarví text podle zadání. Pokud si žáci správně přečetli zadání, neměly by se objevit zásadní problémy s porozuměním.

4.2 Formátovaný dokument

Ze zdrojového kódu napsaného v jazyce Markdown by žáci měli vytvořit formátovaný dokument podle své intuice. Znalost jazyka Markdown není žádoucí. Žáci by měli využít textového editoru, který umožňuje základní formátování nadpisů, tučného písma a kurzívy, číselovaného a nečíselovaného seznamu. Doporučenými editory jsou Microsoft Word nebo OpenOffice / LibreOffice Writer.

Pro tuto úlohu je třeba, aby žák měl kompetence k práci se styly, s formátováním textu a vytvářením číslovaného a nečíselovaného seznamu. Dále by měl mít kompetenci k práci ve zvoleném textovém editoru.

Žákům je možné poskytnout zdrojový text v elektronické podobě, aby se ušetřil čas s přepisováním textu do počítače. Zároveň se tím zamezí problému, kdy by žáci nechali v textu formátovací značky jazyka Markdown.

Cíl

Úloha rozšiřuje porozumění textu obsahujícímu značky určující formát dokumentu. Cílem úlohy je přimět žáky k intuitivnímu naformátování textu na základě zdrojového kódu dokumentu obsahujícímu značky v jazyce Markdown. Cílem není výuka jazyka Markdown.

Zadání

Text (Šokové stavy, 2017) vhodný k oboustrannému tisku na stranu A4 je v souboru formatovani.pdf. Součástí souboru je i zadání předchozí úlohy a je vhodné tisknout je oboustranně společně. Text se po vytištění rozdělí na dvě strany A5 a každý žák obdrží jednu oboustrannou kopii formátu A5.

Vytvořte z následujícího textu dokument s formátováním ve vámi zvoleném textovém editoru. Využijte stylů Nadpis 1 a Nadpis 2, tučného písma a kurzívy, číslovaného a nečíslovaného seznamu. Do zdrojového textu nic dalšího nedoplňujte.

První pomoc

=====

První pomoc jsou úkony a opatření, která při náhlém ohrožení nebo postižení zdraví člověka omezují rozsah a důsledky tohoto ohrožení či postižení. Poskytnout první pomoc je v rámci svých schopností a sil povinen každý občan České republiky starší 18 let, pokud tím neohrozí svoje zdraví či život.

Postup resuscitace

1. Nehrozí nebezpečí?
2. Bez známek života? (Nereaguje, nedýchá)
3. Přivolejte pomoc, telefon 155 nebo 112
4. Resuscitujte – Mačkejte propnutýma rukama střed hrudní kosti 100 – 120 x za minutu do hloubky 5 – 6 cm.
5. Pokračujte do příjezdu záchranky

Postup v případě šoku

Pravidlo 5T

- **T**eplo: Zajistit dostatečný tepelný komfort, ale ne přehřátí (v létě do stínu)

- ****T**icho**: Zajistit klidné prostředí, ale s postiženým komunikovat
- ****T**ransport**: Zajistit odsun k lékaři
- ****T**išení bolesti**: V žádném případě nepodávat léky, spíše uklidňovat, vše vysvětlit
- ****T**ekutiny**: V žádném případě nedávat napít, ale pouze otírat a zvlhčovat rty, obličej, maximálně několik kávových lžiček vody, nebo čaje.

Zdrojový text pro žáky, který neobsahuje formátování a značky značkovacího jazyka Markdown je v příloze *B Zdrojový text První pomoc*.

Vyhodnocení

Očekává se, že žáci použijí formátování pomocí stylů Nadpis. Někteří žáci nemusí rozlišit mezi prvním nadpisem, který by měl mít formát Nadpis 1, a druhým a třetím nadpisem, které by měly mít formát Nadpis 2. Dále je třeba zkontrolovat, zda číslování seznamů nezapsali ručně jako čísla, ale použili formát „číslovaný seznam“.

Žákům by na tomto příkladu měla být vysvětlena základní syntaktická pravidla pro jazyk Markdown. Není nutné, aby se žáci učili nazpaměť, stačí, aby si je poznamenali a měli je k dispozici. Je možné využít pomůcku z přílohy *I Markdown pomůcka*. Žáci by na základě nově získaných informací měli uvést svůj naformátovaný dokument do souladu se zdrojovým kódem v Markdownu.

4.3 Nákupní seznam

Cílem této úlohy je vypracování nákupního seznamu, se kterým se dále bude pracovat, a seznámit žáky se syntaxí značkovacího jazyka Markdown.

Žáci vytvoří nákupní seznam s několika položkami. Žáci budou pro tvorbu seznamu používat textový editor nepodporující formátování, například aplikaci Notepad (Poznámkový blok), která je součástí základní instalace operačního systému Microsoft Windows.

Lze očekávat, že žáci vytvoří seznam oddělený čárkami, nebo seznam s každou položkou nákupního seznamu na novém řádku. V některých případech bude možné se setkat i s určitými pseudoodrážkami na začátku každého řádku s položkou. Jako odrážky se mohou vyskytovat *, - nebo dokonce čísla.

Nyní by žáci měli své nákupní seznamy upravit tak, aby každá položka měla svůj řádek. Pro lepší přehlednost se žáci mohou pokusit začátek řádku označit vždy nějakou vhodnou odrážkou (symbolem, znakem), doplnit seznam nadpisem a rozdělit jej do dvou nadepsaných kategorií (například kategorie potraviny, drogerie apod.). Výběr je ponechán žákům. Pokud by na svém seznamu nemohli najít více kategorií, musí seznam doplnit o další vhodné položky.

Takto vytvořený seznam bude výchozí pro práci se značkovacím jazykem Markdown. Žáci si uloží kopii tohoto seznamu pro práci v podkapitole 4.7 *Pojmenování částí značkami*. Pro práci s jazykem Markdown může být použita aplikace MdCharm, kterou je ale třeba instalovat na všech žákovských stanicích. Alternativou může být online dostupný Markdown editor, například Dillinger na URL adrese <http://dillinger.io>. V tomto online nástroji mohou být psány dokumenty v jazyce Markdown a tlačítkem s ikonkou oko je možné hned zobrazit výsledný formátovaný dokument. V této DP se dále předpokládá použití online verze Dillinger editoru.

Žáci vloží své vytvořené seznamy do Markdown editoru a zobrazí si jeho výsledný dokument. Žáci by se měli snažit, aby nadpis vypadal jako Nadpis 1, název kategorie položek nákupního seznamu jako Nadpis 2 a nákupní seznam jako seznam s odrážkami, nebo číslovaný seznam.

Aby žáci dosáhli takového formátování, musí být seznámeni se základními syntaktickými pravidly, jak zapsat formátováním Nadpis 1, Nadpis 2 a jak vytvořit seznam.

Žáci by měli podle těchto pravidel upravit svůj zápis a vložit jej do editoru.

Výsledný nákupní seznam po provedení úprav by měl být správně naformátovaný a přehledný. Zároveň i zdrojový kód by měl být přehledný, čitelný a srozumitelný běžným uživatelům bez nutnosti převodu ze zdrojového kódu nebo znalosti jazyka Markdown.

Ukázka nákupního seznamu

Nákupní seznam

=====

Elektro

- * Žárovka LED 5W
- * Počítačová myš
- * Klávesnice

Drogerie

- * Zubní kartáček
- * Toaletní papír
- * Mýdlo

4.4 Z Markdownu do HTML

Tato úloha ukazuje převod zdrojového kódu napsaného v Markdown značkovacím jazyce do jazyka HTML. Příklad vychází z již vytvořeného nákupního seznamu z kapitoly 4.3 *Nákupní seznam*. Žáci si svůj nákupní seznam napsaný v jazyce Markdown nechají vyexportovat do HTML. Online nástroj Dillinger toto umožňuje volbou z menu „Export as“ a dále výběrem položky „HTML“. Exportovat dokument jako „Styled HTML“ není pro výukové účely doporučováno, jelikož tím se zdrojový kód HTML komplikuje o definice CSS stylů, což by pro žáky bylo matoucí.

Cíl

Úloha si klade za cíl ukázat žákům možnost převodu jazyka Markdown do jazyka HTML. Dále je cílem této úlohy seznámit žáky se zdrojovým kódem jazyka HTML a vést žáky k porovnání zdrojového kódu a hledání spojitostí mezi formátovaným vzhledem a jeho zápisem pomocí značek.

Ukázkový nákupní seznam v HTML

<code><!DOCTYPE html></code>	Typ obsahu dokumentu
<code><html></code>	Začátek HTML
<code><head></code>	Začátek sekce HEAD
<code><meta charset="utf-8"></code>	Označení kódové stránky dokumentu
<code><title>Untitled Document.md</title></code>	Název stránky
<code><style></style></code>	Vlastní styly (prázdné, nepoužívá se)
<code></head></code>	Konec sekce HEAD
<code><body id="preview"></code>	Začátek sekce BODY
<code><h1></code>	Nadpis 1
<code></code>	Pojmenování nadpisu (není vidět)
Nákupní seznam	Nadpis nákupního seznamu
<code></h1></code>	Konec Nadpis 1
<code><h2></code>	Nadpis 2
<code></code>	
Elektro	Nadpis kategorie
<code></h2></code>	
<code></code>	Nečíslovaný seznam (odrážky)
<code>Žárovka LED 5W</code>	Položka seznamu
<code>Počítačová myš</code>	Položka seznamu
<code>Klávesnice</code>	Položka seznamu
<code></code>	
<code><h2></code>	
<code></code>	
Drogerie	Nadpis další kategorie
<code></h2></code>	
<code></code>	Nečíslovaný seznam (odrážky)
<code>Zubní kartáček</code>	Položka seznamu
<code>Toaletní papír</code>	Položka seznamu
<code>Mýdlo</code>	Položka seznamu
<code></code>	
<code></body></code>	Konec sekce BODY
<code></html></code>	Konec dokumentu HTML

Žáci by se měli po vlastním srovnání zdrojového kódu v jazyce Markdown a exportovaného zdrojového kódu v HTML pokusit zformulovat pravidla pro zápis v jazyce HTML.

Mohlo by jít například o tato pravidla: celý dokument je ohraničen značkou **html**; většina názvů značek je psána dvakrát; podruhé se začíná zpětným lomítkem. Dále např. pravidlo pro zápis položky seznamu pomocí značky **li** či o označení nadpisů značkami **h1** a **h2**.

Žáci si mohou ověřit své myšlenky přímou editací zdrojového kódu v HTML a následným zobrazením v internetovém prohlížeči. Zde ovšem může docházet ke zkreslení, jelikož internetový prohlížeč může ignorovat některé syntaktické chyby a snažit se je sám opravit. Tato metoda proto není zcela vhodná pro naučení se pravidel, ale spíše jen k ověření myšlenky.

Poté, co žáci vyjádří své návrhy ohledně syntaxe jazyka HTML na základě převedeného nákupního seznamu, by měli být seznámeni se syntaxí jazyka HTML. Důležité pojmy značkovacího jazyka HTML jsou párové a nepárové značky (tagy). Základním uvedením do značkovacího jazyka HTML je struktura dokumentu HTML.

Po vysvětlení základní struktury dokumentu HTML by se žáci měli seznámit se značkami Nadpis 1 (**h1**), Nadpis 2 (**h2**), nečíslovaný seznam (**ul**), číslovaný seznam (**ol**), položky seznamu (**li**) a kotva (**a**). Kromě číslovaného seznamu jsou všechny značky použity v ukázkovém nákupním seznamu.

4.5 Odkazy v HTML

Tato úloha vede žáky k tomu, aby svůj nákupní seznam upravili s použitím hypertextových odkazů.

Cíl

Cílem úlohy je seznámit žáky se značkou **a** pro hypertextový odkaz a použití atributů **src** a **title**. Dále budou žáci získávat další zkušenosti s prací s párovými značkami jazyka HTML.

Zadání

Žáci na základě znalosti použití značky kotva (**a**) s atributem `href` upraví nákupní seznam a položky propojí hypertextovým odkazem na e-shopy, které položky prodávají. Atribut `title` u hypertextového odkazu by měl obsahovat název odkazované stránky.

Důraz by měl být kladen především na správné použití zápisu značky. Párová značka kotva (**a**) musí být vždy ukončena. Žáci si také musí zkontrolovat, zda nedošlo ke křížení se značkou **li**. To by znamenalo, že značka kotva začínala později než začíná položka seznamu, ale byla ukončena později než je ukončena položka seznamu. Tímto by pak přesahovala mimo položku seznamu, což není dovoleno. Značka **a** by měla být uvedena uvnitř značky **li** z logiky věci, kdy nejdříve je položka seznamu a pak teprve samotný obsah položky seznamu.

4.6 Strukturování dokumentu

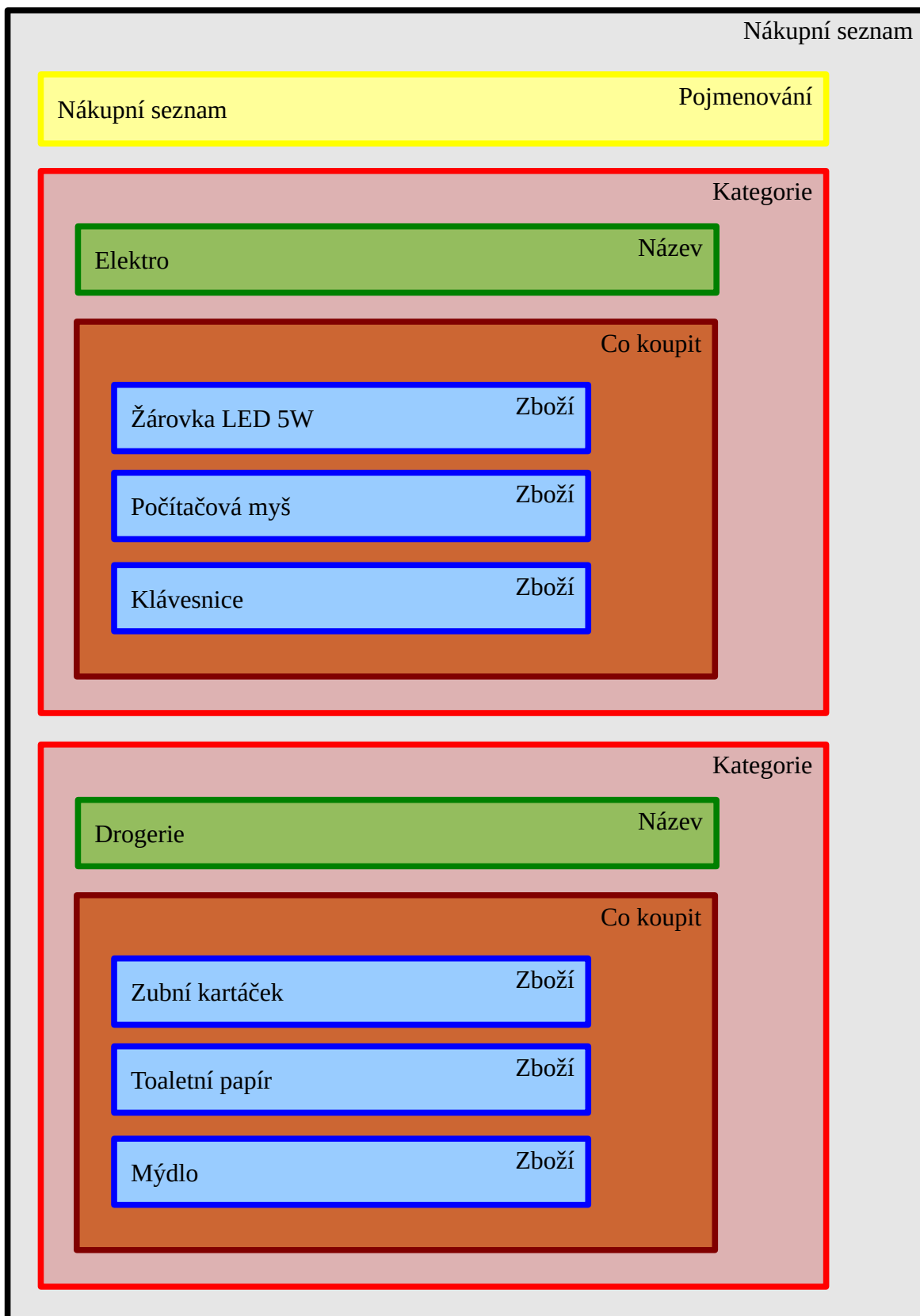
Vzhledem k tomu, že strukturování dokumentu a popis struktury dat již vyžaduje určitý způsob analytického myšlení a práci s abstraktními pojmy, je možné zařadit tuto úlohu jako úlohu, která se vypracuje společně a ukáže žákům příklad, jak lze strukturovat dokument.

Cíl

Úloha vypracovaná společně by měla žákům ukázat možnosti a požadavky popisu struktury dat. Hlavní náplní je hledání nadřazených zobecněných pojmů pro dané oblasti dokumentu. Úloha vychází z ukázkového seznamu, který obsahuje nadpis, dvě kategorie, dva nadpisy kategorie a celkově šest položek seznamu.

Postup

Žáci dostanou text „Nákupní seznam“ vytištěný a společně si zaznamenávají do kroužku oblasti dat a popisují jejich význam. Výsledek může být stejný, jako je znázorněno na straně 45, *Ilustrace 3*. Začíná se od největší oblasti. Ta zahrnuje úplně vše, co je na papíře vytištěno. Jelikož se jedná o přípravu nákupního seznamu k převodu do XML, je třeba striktně dodržovat, že se jednotlivé oblasti nesmějí křížit, ale musí být vždy vloženy do větší oblasti celým svým objemem. Popisy jednotlivých oblastí se můžou lišit a je na žácích, jaké popisy si zvolí. Pravidla tvorby značek XML není třeba dodržovat, tomu se bude věnovat úloha *4.8 Seznam do XML*. Níže je uveden seznam možných pojmenování



Ilustrace 3: Struktura dat nákupního seznamu

jednotlivých oblastí. Nejedná se o konečný výčet jediných správných pojmenování daných oblastí nákupního seznamu, ale o příklady.

- Nákupní seznam Seznam, Nákup
- Pojmenování Titulek, Název, Nadpis
- Kategorie Typ, Oddělení, Druh
- Název Nadpis, Podnadpis, Označení, Jméno
- Co koupit Seznam položek, Položky, Seznam zboží, Produkty
- Zboží Položka, Věc, Produkt

Je třeba dbát na to, aby pojmenování oblastí nebylo příliš konkrétní. V případě seznamu je dobrým pomocníkem možnost přenést bez úprav dané označení oblastí i na druhou kategorii. Například pojmenování „Kategorie elektro“ není možné aplikovat na druhou kategorii a bylo by nutné pojmenování kategorie změnit.

Smyslem označování oblastí a tím strukturování dokumentu je rozdělit dokument na co nejmenší části a pojmenovat jejich význam. Dále je úkolem dávat jednotlivé části do větších logických celků a tyto celky opět pojmenovat. Jak je zmíněno dále, XML vyžaduje, aby celý obsah byl v jedné značce, proto úloha začíná označením celého nákupního seznamu.

Zdrojový text

Zdrojový text vhodný k tisku a následnému označování žáky je v příloze v souboru seznam.pdf. Tisk se provede 2 stránky na 1 stranu A4 a stránka se následně rozpůlí pro dva žáky.

Nákupní seznam

Elektro

Žárovka LED 5W

Počítačová myš

Klávesnice

Drogerie

Zubní kartáček

Toaletní papír

Mýdlo

4.7 Pojmenování částí značkami

Pro nejjednodušší práci žáci použijí vytvořený seznam bez formátování, který si uložili v kapitole 4.3 *Nákupní seznam*. Žáci označí jednotlivé části seznamu podle jejich významu a logických celků. Inspirovat se mohou zápisem jazyka HTML, pokud jazyk znají, ale používat by měli své vlastní názvy značek. Oblasti budou označeny jménem, které bude uvedeno mezi symboly < a >. Označená oblast bude ukončena stejným jménem, kterému bude předcházet znak lomeno /, a celý název bude uveden mezi symboly < a >. Žáci by se měli soustředit na označení dokumentu, nadpisů, seznamů a jednotlivých položek.

Pokud žáci vypracovali označení seznamu z úlohy 4.6 *Strukturování dokumentu*, mohou využít tento zápis a aplikovat ho na svůj osobní nákupní seznam.

Cíl

Cílem úlohy je přenést označení logických celků dokumentu přímo do dokumentu v textové podobě.

Příklad výsledného dokumentu:

```
<nákupní seznam>
  <pojmenování>
    Nákupní seznam
  </pojmenování>
  <kategorie>
    <název>
      Elektro
    </název>
    <co koupit>
      <zboží>
        Žárovka LED 5W
      </zboží>
      <zboží>
        Počítačová myš
      </zboží>
      <zboží>
```

```

                                Klávesnice
                                </zboží>
                                </co koupit>
                                </kategorie>
                                <kategorie>
                                <název>
                                Drogerie
                                </název>
                                <co koupit>
                                <zboží>
                                Zubní kartáček
                                </zboží>
                                <zboží>
                                Toaletní papír
                                </zboží>
                                <zboží>
                                Mýdlo
                                </zboží>
                                <co koupit>
                                <kategorie>
                                </nákupní seznam>

```

Vyhodnocení

Je velmi pravděpodobné, že žáci své seznamy nebudou dělit na takové množství částí jako ukázkový seznam – například nadpis kategorie bude nejspíš na stejné úrovni jako část „co koupit“. Proto je nutné, aby žáci své seznamy popsali co nejpodrobněji, snažili se přitom zachovat logické celky u sebe a tyto celky byly správně zanořeny.

Při určování správného logického celku může pomoci převod na cesty. Například část dokumentu „Žárovka LED 5W“ je v cestě „nákupní seznam“ > „kategorie“ > „co koupit“ > „zboží“. Zjistit název této kategorie je možné v cestě „nákupní seznam“ > „kategorie“ > „název“.

Nelze rozhodnout, které pojmenování částí je správné a které špatné. Naproti tomu lze dokument rozdělit do celků, které nedávají smysl. Protože seznam obsahuje dvě kategorie, měl by se jejich popis shodovat. Stálé opakování stejných částí není chybou.

Rozhodně se doporučuje odsadit jednotlivé bloky podle jejich vnoření pomocí tabulátorů, jak je to uvedeno u příkladů. Přispívá to celkové čitelnosti a pomáhá při práci s ukončovacími značkami.

4.8 Seznam do XML

V této úloze se žáci seznámí s pravidly pro tvorbu XML dokumentů a vytváření názvů značek XML. Žáci by po seznámení se základními syntaktickými pravidly značkovacího jazyka XML měli upravit své nákupní seznamy tak, aby daná pravidla splňovaly.

Cíl

Úloha si klade za cíl seznámit žáky se základními pravidly pro tvorbu značek ve značkovacím jazyce XML. Pravidla by poté měli být žáci schopni samostatně aplikovat na již vytvořené vlastní seznamy a upravit již použité značky. Tím dojde k procvičení pravidel. Závěrem je možné seznámit žáky s online validátorem XML.

Postup

Seznamte žáky se základními pravidly XML dokumentů. Pro seznámení žáků s pravidly XML může být využito přílohy *K XML pomůcka*.

Upravený nákupní seznam podle pravidel XML:

```
<nakupni_seznam>
  <pojmenovani>
    Nákupní seznam
  </pojmenovani>
  <kategorie>
    <nazev>
      Elektro
    </nazev>
    <co_koupit>
      <zbozi>
```

```

                Žárovka LED 5W
            </zbozi>
            <zbozi>
                Počítačová myš
            </zbozi>
            <zbozi>
                Klávesnice
            </zbozi>
        </co_koupit>
    </kategorie>
    <kategorie>
        <nazev>
            Drogerie
        </nazev>
        <co_koupit>
            <zbozi>
                Zubní kartáček
            </zbozi>
            <zbozi>
                Toaletní papír
            </zbozi>
            <zbozi>
                Mýdlo
            </zbozi>
        </co_koupit>
    </kategorie>
</nakupni_seznam>

```

Kontrolu upraveného nákupního seznamu, který odpovídá pravidlům a doporučením pro tvorbu názvů značek pro XML, mohou žáci provést na URL adrese <https://www.xmlvalidation.com>. XMLvalidation.com je online validační nástroj, který

kontroluje syntaxi dokumentu ve značkovacím jazyce XML. Nástroj je dostupný pouze v anglickém jazyce.

4.9 XML a atributy

Dalším krokem úpravy nákupního seznamu je přidání cen k jednotlivým položkám nákupního seznamu a pomocí atributu nastavení měny.

Cíl

Žáci se seznámí se zápisem atributů v XML a získané znalosti aplikují při doplnění nákupního seznamu o ceny zboží a měnu ceny.

Zadání

Doplňte do svého seznamu ke každému zboží jeho cenu, pomocí atributu určete měnu.

Ukázka části seznamu

...

```
<zbozi>
  <nazev>
    Mýdlo
  </nazev>
  <cena mena="CZK">
    7,90
  </cena>
</zbozi>
```

...

4.10 XML do tabulky

Jako zkoušku, zda se žáci orientují v datovém modelu XML, lze použít převod nákupního seznamu do tabulky. Pro začátek se bude vycházet ze zadané tabulky, kterou žáci vyplní. Žáci převedou svůj nákupní seznam do tabulky, čímž se seznámí i s datovým významem prvků nákupního seznamu.

Cíl

Cílem úlohy je pochopení datového modelu XML nákupního seznamu a jeho převod do tabulky.

Zadání

Doplňte tabulku na základě svého nákupního seznamu.

Tabulka k doplnění:

Kategorie	Zboží	Cena (měna)

Do prvního řádku by si žáci měli doplnit název svého nákupního seznamu.

Ukázková tabulka:

Nákupní seznam		
Kategorie	Zboží	Cena (měna)
Elektro	Žárovka LED 5W	89 (CZK)
Elektro	Počítačová myš	129 (CZK)
Elektro	Klávesnice	11 (EUR)
Drogerie	Zubní kartáček	150 (CZK)
Drogerie	Toaletní papír	6,50 (CZK)
Drogerie	Mýdlo	13,90 (CZK)

Pozn.: Tím, že měna ceny byla uložena jako atribut, pracuje se s ní v rámci sloupce Cena a nikoliv jako se samostatným sloupcem. Název kategorie se bude několikrát opakovat.

4.11 Česká vlajka v SVG

Žáci v rámci této úlohy vytvoří vektorový obrázek české vlajky v jazyce SVG, který využívá zápisu XML značkovacího jazyka. Žáci nejdříve nakreslí českou vlajku ve čtvercové síti. Následně by měly být žákům vysvětleny příkazy jazyka SVG a práce v souřadnicové síti, která je jinak orientována než síť využívaná v matematice.

Cíl

Žák by měl být schopen sám navrhnout jednoduchou grafiku a zapsat ji jazykem SVG.

Zadání

Na čtverečkovaný papír nakreslete českou vlajku. Všechny důležité body (rohy vlajky a vrchol modrého trojúhelníku) nakreslete tak, aby se nacházely na mřížce čtverečkovaného papíru.

Poznámka: Česká vlajka má formát 2:3, bílý a červený pruh dělí vlajku vodorovně na půl, červený pruh je vespod, modrý klín sahá zleva do středu vlajky.

Ukázka výsledku:

```
<svg>
  <rect x="0" y="0" width="300" height="100" fill="white" />
  <rect x="0" y="100" width="300" height="100" fill="red" />
  <polygon fill="blue" points="0,0 150,100 0,200" />
</svg>
```

4.12 Další úlohy

Dále v této kapitole následují doplňkové úlohy, které rozšiřují možnosti procvičení a výkladu značkovacích jazyků. Úlohy je možné vřadit do výuky dle uvážení vyučujícího. Úlohy jsou koncipovány jako návrhy a doporučení k dalšímu procvičování. Jejich konkrétní realizace se ponechá na vyučujícím.

4.12.1 Značkování

V této úloze by žák měl navrhnout jednoduchý způsob k zaznamenání formátovaného textu. Formátovaný text může být žákům rozdán vytištěný nebo promítnutý na plátno. Žáci by měli k zaznamenání textu se značkami využít program Poznámkový blok (notepad), který nepodporuje formátování textu a tím usnadní tvorbu textu.

Cíl

Cílem úlohy je, aby žáci sami navrhli značkový jazyk k označení formátu textu. Tato úloha by měla žákům ukázat, že je možné zachytit formát dokumentu i bez vizuálního zobrazení jen pouhým textem. Jak je psáno v kapitole *2.1 Historie a vývoj značkovacích jazyků* na straně 10, první využití značkování bylo například u tiskáren.

Úloha

Zkuste navrhnout způsob, jakým v textu označovat tučný text a kurzívu pouze s použitím písmen, čísel a znaků, které jsou na české klávesnici. Navržený způsob aplikujte na následující text. Přepište ho tak, aby bylo možné zrekonstruovat text zpět s plným formátováním.

A: Duis *biendum*, **lectus** ut *viverra* rhoncus.

B: Ala *oreet*, magna **id** *viverra tincidunt*, **sem** odio.

Poté si vyměňte vytvořené texty se svým sousedem a zkuste zrekonstruovat původní formátovaný text.

Zkonzultujte se svým spolužákem zkušenosti s kódováním a dekódováním výše uvedeného textu a vymyslete způsob, jak výše uvedené úpravy zaznamenat (můžete použít jeden z již vymyšlených nebo utvořit nový).

Popište detailně způsob kódování (zápisu) a dekódování (opětovného vytvoření formátovaného dokumentu).

Vyhodnocení

Pokud nikdo nepřišel s nápadem „přepínání“ formátů (příkazy pro pisatele), je vhodné žákům tento způsob předvést.

Ukázka 1

Popis: Tučné je uzavřeno v *, kurzíva _. (Tento způsob označování tučného písma * lze využít v Microsoft Wordu nebo OpenOffice/LibreOffice Writeru, kdy se automaticky použije tučné písmo)

Zápis A: *D*uis _biend_um, *lectus* ut _*viv*_erra rho*n*cus.

Zápis B: Alao_reet_, magna *id* viverra *_tincid_*unt, se*m* odio.

Ukázka 2

Popis: Tučné, resp. kurzíva je uvedeno slovem TUČNÉ:, resp. KURZÍVA: a uzavřeno v uvozovkách. Jedná se o zdlouhavější zápis, ale na druhou stranu je čitelnější pro lidi. Vzhledem k tomu, že slova tučné a kurzíva nejsou označena žádným speciálním znakem, může dojít k problémům, pokud se tato slova nacházejí ve výchozím textu.

Zápis A: TUČNÉ:“D“uis KURZÍVA:“biend“um, TUČNÉ:“lectus“ ut
TUČNÉ:“KURZÍVA:“viv““erra rhoTUČNÉ:“n“cus.

Zápis B: AlaoKURZÍVA:“reet“, magna TUČNÉ:“id“ viverra
KURZÍVA:“TUČNÉ:“tincid““unt, seTUČNÉ:“m“ odio.

Ukázka 3

Popis: Použití příkazů pro změnu formátu písma uzavřené ve složených závorkách.

Zápis A: {tučné}D{normální}uis {kurzíva}biend{normální}um,
{tučné}lectus{normální} ut {tučné a kurzíva}viv{normální}erra
rho{tučné}n{normální}cus.

Zápis B: {normální}Alao{kurzíva}reet{normální}, magna
{tučné}id{normální} viverra {tučné a kurzíva}tincid{normální}unt,
se{tučné}m{normální} odio.

4.12.2 Z popisu dokument

Tato úloha slouží k procvičení značkování a k porozumění významu značek. Žáci by měli vytvořit formátovaný dokument z textů, které obsahují formátovací značky. Značky jsou voleny tak, aby jim žáci rozuměli.

Cíl

Cílem této úlohy je ověření, zda žáci chápou principy značkovacích jazyků.

Úloha

Vytvořte formátovaný dokument z následujících zápisů:⁴

- <tučné>Kůň domácí</tučné> (<kurzíva>Equus caballus</kurzíva>) nebo pouze <tučné><kurzíva>kůň</kurzíva></tučné> je domestikované zvíře.
- Slovo {tučné 3 písmena}kůň označuje {kurzíva 6 písmen}obecně jak {tučné 5 písmen}samce, tak i {tučné 6 písmen}samici {tučné 4 písmena} {kurzíva 4 písmena}koně.
- /tučné(Předci) dnešního /kurzíva(koně) jsou zařazeni do /kurzíva(období) před /tučné(/kurzíva(60 miliony)) let.

4 Kůň. In: *Wikipedie* [online]. 28. 4. 2016 v 13:43 [cit. 2016-05-15]. Dostupné z: <https://cs.wikipedia.org/wiki/K%C5%AF%C5%88>

Výsledek

- **Kůň domácí** (*Equus caballus*) nebo pouze **kůň** je domestikované zvíře.
- Slovo **kůň** označuje *obecně* jak **samec**, tak i **samici koně**.
- **Předci** dnešního *koně* jsou zařazeni do *období* před **60 miliony** let.

4.12.3 Datově orientovaný dokument

V této úloze by žáci měli navrhnout způsob záznamu datově orientovaného dokumentu (tabulky) pouze pomocí znakového zápisu.

Cíl

Tato úloha si klade za cíl, aby žáci vytvořili způsob zápisu strukturovaných dat pouze prostým textem.

Úloha

Zkuste navrhnout způsob, jakým lze přepsat tabulku pouze s použitím písmen, čísel a znaků, které jsou na české klávesnici. Navržený způsob aplikujte na následující tabulku. Přepište ji tak, aby bylo možné podle vámi přepsaného textu znovu sestavit tabulku se stejným významem.

Název	Cena
Notebook WR52	25 999 Kč
Fén QW2	958 Kč

Nebojte se využít více slov k popisu tabulky. Zkuste se podívat nejen na vzhled tabulky, ale dejte důraz na informace a jejich provázání.

Ukázka 1

Využití dokumentové orientace dokumentu

Popis: Tabulka má sloupce A a B, řádky 1, 2, 3.

Zápis:

Buňka A1, tučný text: Název

Buňka B1, tučný text: Cena

Buňka A2: Notebook WR52

Buňka B2, zarovnání vpravo: 25 999 Kč

Buňka A3: Fén QW2

Buňka B3: 958 Kč

Podle tohoto zápisu lze sestavit přesně výše zmíněnou tabulku, ale tento zápis nevyužívá popisu, co které informace znamenají.

Ukázka 2

Využití datové orientace dokumentu. Tabulka nemusí mít vzhled přesně podle zadání, ale bude vyjadřovat stejné informace.

Zápis:

Produkt 1 – Název: Notebook WR52; Cena 25 999 Kč

Produkt 2 – Název: Fén QW2; Cena: 958 Kč

5 Ověření metodiky

Ověřování metodiky proběhlo výukou na druhém stupni Základní školy Gutova v Praze 10 v březnu a dubnu 2017. Výuka probíhala v sedmém ročníku ve volitelném předmětu Prezentace informací, který vedla paní učitelka ing. Jitka Zůnová. Druhé ověření probíhalo na druhém stupni Základní školy Donovalská v Praze 11. Výuka byla vedena v devátém ročníku v povinném předmětu Informatika, který vedl pan učitel Jiří Hofman. Tyto školy byly zvoleny, neboť zde jednak byla výuka umožněna, jednak tyto školy splňovaly požadavky na žáky různých věkových kategorií a na různě zaměřené předměty (povinný a povinně volitelný).

5.1 Očekávané výsledky ověření

Cílem ověřování této metodiky je zjistit, zda způsob výuky podle této metodiky je pro žáky srozumitelný a pochopitelný. Pro ověření jsou zvolena témata jazyk Markdown a jazyk XML. V případě sedmého ročníku je přidáno téma SVG. Jelikož žáci již byli seznámeni se značkovacím jazykem HTML v dřívějších hodinách a též tvořili webové stránky, jazyk HTML bude jen stručně zopakován.

Podle výzkumu publikace *Rozvoj informačně technologických kompetencí na základních školách* patří témata „vytváření a publikování webových stránek“ a „programování“ z hlediska žáků mezi nejtěžší témata v informatickém předmětu (Rambousek, 2013, s. 283). Také se jedná o témata, které žáci považují za nejzbytečnější (Rambousek, 2013, s. 281). Na základě těchto zjištění lze očekávat, že žáci budou mít s pochopením značkovacích jazyků problémy.

U úlohy 4.1 *Formátování textu podle značek* je možné očekávat, že nebudou problémy s jejím řešením. U následující úlohy 4.2 *Formátovaný dokument* bude záležet na tom, jak jsou žáci zkušení s prací v textovém editoru. Můžeme očekávat nutnost připomenout postup při formátování pomocí stylů. Výsledky budou záležet na zkušenostech a intuici žáků při formátování dokumentu.

S vytvářením nákupního seznamu v úloze 4.3 *Nákupní seznam* by žáci neměli mít těžkosti. Můžeme předpokládat, že žáci na druhém stupni již budou schopni sestavit nákupní seznam se dvěma kategoriemi dle svého výběru. Nepochopení může vzniknout při

vytváření nadpisů kategorií či nadpisů celého seznamu. Formátování nadpisů stylem jazyka Markdown, kdy se nadpis podtrhává, může být pro žáky méně pochopitelné než označování nadpisu křížky. Naopak s tvorbou číslovaných a nečíslovaných seznamů by žáci neměli mít problém. Jen je třeba dbát na správné umístění mezer.

Vzhledem k vysoké náročnosti abstraktního myšlení při strukturování dokumentu bude využito úlohy *4.6 Strukturování dokumentu* jako propedeutiky k strukturování pomocí jazyka XML. Protože úloha bude vypracovávána společně, bude obtížnější zjistit, zda žáci způsob strukturování dat dostatečně pochopili. V úloze *4.7 Pojmenování částí značkami* žáci aplikují nově získané znalosti o pravidlech tvoření značek XML. Vzhledem k jednoduchosti pravidel by se neměly vyskytovat větší nesrovnalosti, spíše půjde o nedostatky způsobené nepozorností při aplikování pravidel tvorby XML názvů značek.

U práce s jazykem SVG v úloze *4.11 Česká vlajka v SVG* může žákům způsobovat komplikace více faktorů. Může jít o problémy s vyjádřením souřadnic, a to zvláště v tom případě, kdy je orientace souřadnicové sítě jiná, než žáci znají z předmětu matematika. Další komplikací mohou být anglicky psané příkazy pro kreslení a především správné vyplnění atributů. Pokud žáci porozuměli zápisu značek XML, zápis značek jazyka SVG by neměl činit potíže.

Celkově lze očekávat, že téma Markdown bude snadnější k pochopení. Naopak téma XML značkovacího jazyka lze považovat za náročnější, kdy lze očekávat, že zvláště v sedmém ročníku bude toto téma pro žáky obzvláště obtížné.

5.2 Průběh výuky

Na začátku samotné výuky byl žákům zadán jednoduchý úvodní dotazník, který je přiložen v příloze *D Úvodní dotazník*. Dotazník zjišťuje postoj a znalosti žáků ohledně značkovacích jazyků Markdown, HTML, XML a SVG. Dotazník byl koncipován tak, aby žáci byli schopni odpovědět na jasné otázky a zároveň, aby časová náročnost jeho vypracování byla minimální. Dotazník nezkoumal předpoklady k řešení jednotlivých úloh, což se v případě výuky na ZŠ Donovalská ukázalo jako jeho nedostatek. Dotazník byl realizován pomocí nástroje Google Forms, ukázka vzhledu je v příloze *E Vzhled úvodního dotazníku*.

Výuka na ZŠ Gutova zahrnovala tři vyučovací hodiny v průběhu jednoho měsíce. Výuka byla zahájena dotazníkem, po kterém následovaly úlohy *4.1 Formátování textu podle značek*, *4.2 Formátovaný dokument* a *4.3 Nákupní seznam*. První hodina končila vložením vytvořeného seznamu do online editoru Dillinger. Na úplném závěru byli žáci požádáni o anonymní napsání volné zpětné vazby na papír.

Na začátku druhé hodiny, která následovala až po 14 dnech (z důvodu týdenního lyžařského kurzu žáků), byl stručně zopakován jazyk Markdown a následovala úloha *4.6 Strukturování dokumentu*. V té byla vysvětlena pravidla pro tvorbu XML názvů značek a následovala úloha *4.8 Seznam do XML*, ve které žáci aplikovali pojmenování na své seznamy. Hodina byla ukončena kontrolou jednotlivých seznamů. Kontrolu žáci provedli v online XML validátoru.

Poslední hodina byla věnována jazykům XML, Markdown, HTML a nově SVG. V prvním cvičení se žáci věnovali přepisu XML jídelního lístku do tabulky podle úlohy *4.10 XML do tabulky*. Při převodu jazyka Markdown do HTML byly zopakovány základní HTML značky a konec hodiny byl věnován jazyku SVG a úloze *4.11 Česká vlajka v SVG*. Na závěr hodiny dostali žáci za úkol vyplnit závěrečný dotazník. Jeho zadání je uvedeno v příloze *F Závěrečný dotazník*.

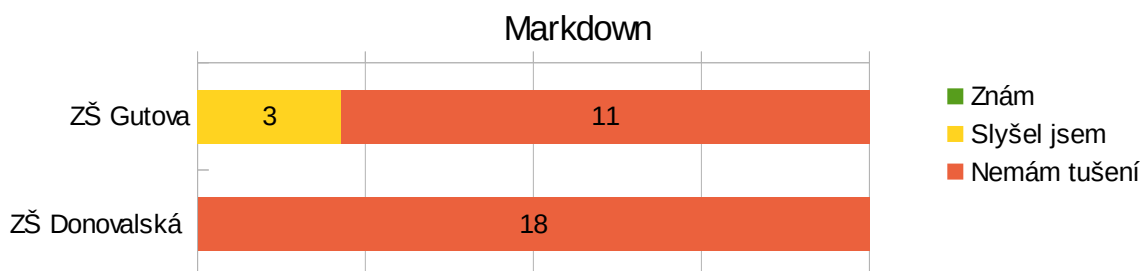
Na ZŠ Donovalská výuka probíhala ve dvou po sobě jdoucích hodinách. Struktura první hodiny byla podobná hodině na ZŠ Gutova. V druhé hodině se pokračovalo s jazykem Markdown. Druhá polovina hodiny se věnovala úloze *4.6 Strukturování dokumentu*. Téma bylo zakončeno vysvětlením pravidel tvorby značek v XML a opravou chyb v názvech značek z ukázky v úloze *4.7 Pojmenování částí značkami*, kde značky ještě nejsou upraveny pravidly v XML. Posledních 5 minut hodiny bylo věnováno vyplnění závěrečného dotazníku, viz příloha *F Závěrečný dotazník*.

5.2.1 Úvodní dotazník

Na ZŠ Gutova se úvodního dotazníku z přílohy *D Úvodní dotazník* zúčastnilo 14 žáků. Vzhledem ke skutečnosti, že výuka probíhala v rámci povinně volitelného předmětu, odpovědi žáků korespondují s vyšším zájmem žáků o počítače a technologie s nimi spojené. Žáci si sami zvolili tento předmět buď z důvodu zájmu o počítače, nebo naopak přílišnému nezájmu o paralelní tělovýchovný předmět. Na otázku „V budoucnu bych chtěl

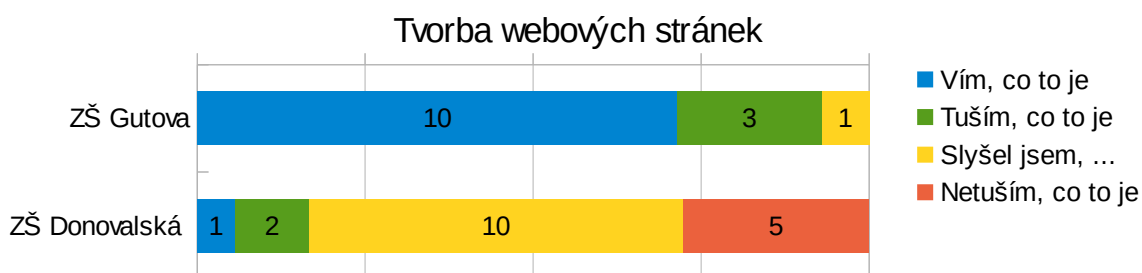
umět“ 6 žáků odpovědělo, že chce umět programovat, 5 žáků vytvářet webové stránky a 6 žáků odpovědělo, že chce umět řešit problémy s počítači.

Na ZŠ Donovalská se dotazníku zúčastnilo 18 žáků. Na této škole se naopak jednalo o povinný předmět, ve kterém lze očekávat menší zájem o tyto technologie. Tomu nakonec i odpovídaly výsledky ověřování. Na otázku „V budoucnu bych chtěl umět“ 2 žáci odpověděli, že chtějí umět programovat, 3 žáci řešit problémy s počítači, 3 žáci tvořit webové stránky a 10 žáků si nevybralo ani jednu z nabízených možností.



Ilustrace 4: Otázka Markdown. Zdroj: Úvodní dotazník

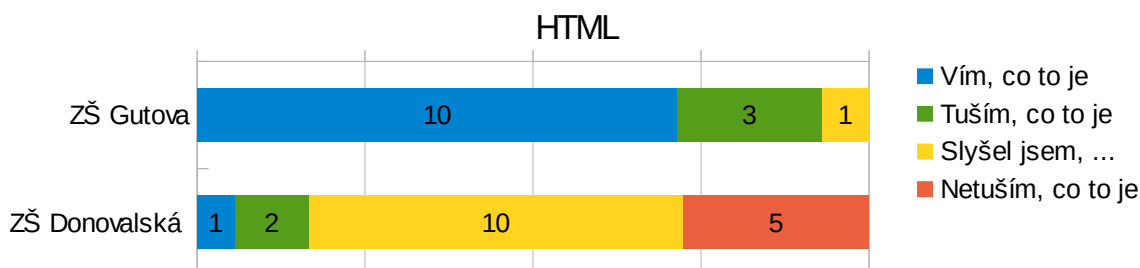
Na otázku o znalosti jazyka Markdown 3 žáci ZŠ Gutova odpověděli, že o jazyku již slyšeli, ostatní žáci ZŠ Gutova odpověděli, že nemají tušení. V případě ZŠ Donovalská všech 18 žáků uvedlo, že nemají tušení. Takové výsledky se dají přisuzovat tomu, že jazyk Markdown je poměrně mladý značkovací jazyk a běžní uživatelé počítačů nemají příliš možností se s ním setkat. Také je možné, že i když se s tímto jazykem setkají, nerozeznají, že jde o značkovací jazyk díky jeho velmi přehlednému a čitelnému zdrojovému kódu.



Ilustrace 5: Otázka Tvorba webových stránek. Zdroj: Úvodní dotazník

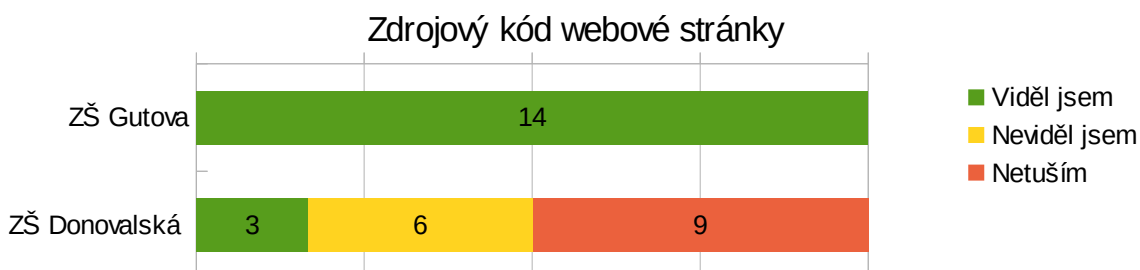
Vzhledem k tomu, že žáci volitelného předmětu na ZŠ Gutova již pracovali se zdrojovým kódem a tvořili vlastní webové stránky, na další tři otázky ohledně tvorby webových stránek, jazyka HTML a zdrojového kódu webových stránek, odpovídali převážně kladně.

9 žáků již tvořilo webové stránky, 3 ostatní tušili, co to je, a 1 žák odpověděl, že webové stránky dělá běžně. 1 žák odpověděl, že netuší. Co je HTML vědělo 10 žáků, 3 žáci tušili a jeden žák slyšel, ale víc o tom neví. Zdrojový kód webové stránky viděli všichni.



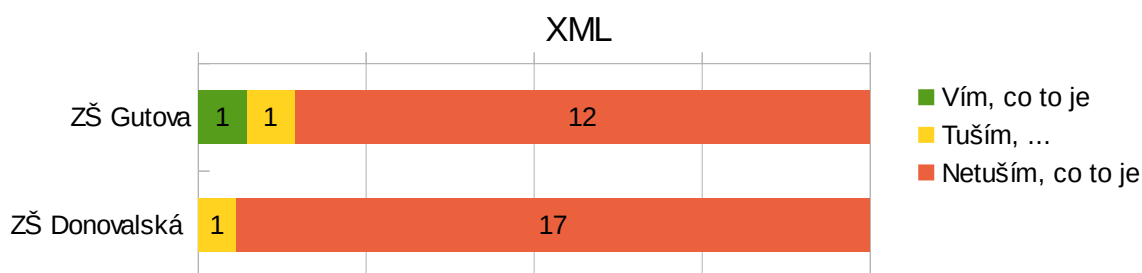
Ilustrace 6: Otázka HTML. Zdroj: Úvodní dotazník

Na ZŠ Donovalská na otázku „Tvorba webových stránek“ odpověděl 1 žák, že stránky tvoří běžně. 2 žáci odpověděli, že už stránky tvořili, 1 žák odpověděl, že netuší, co to je. Zbývajících 14 žáků uvedlo, že tuší, co to je. Na otázku „HTML“ odpověděl 1 žák, že ví, co to je, a 2 žáci tuší, co to je. Odpověď „Slyšel jsem, ale víc o tom nevím“ zvolilo 10 žáků. 5 žáků netuší, co to je. Zdrojový kód webové stránky viděli 3 žáci, 6 žáků odpovědělo, že nevidělo. 9 žáků vůbec netuší, co zdrojový kód webové stránky je.



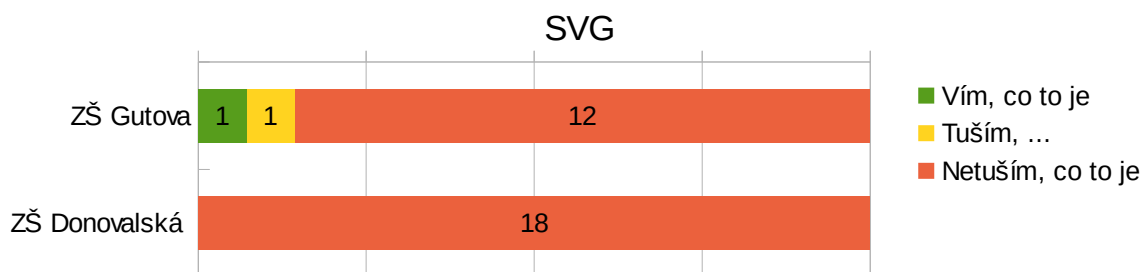
Ilustrace 7: Otázka Zdrojový kód webové stránky. Zdroj: Úvodní dotazník

Značkovací jazyk XML a struktura dat je již pokročilá látka, proto nelze u žáků očekávat jeho znalost, což se opět v dotazníku potvrdilo. Na ZŠ Gutova jeden žák věděl, co to je a jeden žák tušil, co by to mohlo být. Na ZŠ Donovalská jeden žák tušil, co by to mohlo být.



Ilustrace 8: Otázka XML. Zdroj: Úvodní dotazník

Na otázku „SVG“ na ZŠ Gutova jeden žák odpověděl, že ví, co to je a jeden, že tuší, co by to mohlo být. Na ZŠ Donovalská všichni netušili, co to je.



Ilustrace 9: Otázka SVG. Zdroj: Úvodní dotazník

Náročnost a veřejná známost jednotlivých témat by se dala seřadit následovně: Tvorba webových stránek, HTML a zdrojový kód, XML a nejméně známé SVG a Markdown. V celém dotazníku panovala konzistence v tom, že žáci, kteří byli obeznámeni s méně známými tématy, se v tématech známějších orientovali dobře. Například žák, který v dotazníku uvedl, že ví, o co se jedná v případě SVG, také odpověděl, že dělá běžně webové stránky, ví, co je to HTML a ví, co je to XML.

5.2.2 Úvod do značkování

Žákům byl rozdán vytištěný text z kapitoly 4.1 *Formátování textu podle značek*, který měli za úkol podle zadání vybarvit, zakroužkovat nebo podtrhnout. Text k vytištění je uveden v přílohách v souboru *formatovani.pdf*. Než žáci začali úkol vypracovávat, bylo jim vysvětleno, proč se v textu nacházejí speciální značky. K vysvětlení byl využit příklad s posíláním dokumentu přes SMS, kdy je třeba v textu zaznamenat požadavky na podtrhávání a další formátování. Jako další příklad byl uveden email, který nepodporuje

formátování. Toto cvičení přišlo žákům z obou škol jednoduché a lehké. Podle očekávání se nevyskytly žádné problémy.

5.2.3 Formátování podle značek

Dalším úkolem žáků byla úloha 4.2 *Formátovaný dokument*. Žáci dostali vytištěný text s formátovacími značkami jazyka Markdown a byli upozorněni, že tento text je formátován podle jiných pravidel než předchozí úloha. Text k vytištění je v přílohách v souboru *formatovani.pdf* na straně dvě. Jejich úkolem bylo převést vytištěný text do textového editoru podporujícího formátování a naformátovat ho podle svého citu tak, jak by si dané značky vyložili. Vzhledem k tomu, že žáci v dotazníku zvolili, že jazyk Markdown neznají, nikdo z žáků neznal význam značek v tištěném textu. Původní text bez značek Markdown dostali žáci v elektronické podobě, protože cílem cvičení nebylo procvičit přepisování textu. Text je v příloze *B Zdrojový text První pomoc*.

Na ZŠ Gutova se objevil problém, jak formátovat styl Nadpis 2, pokud není ve výběru stylů Microsoft Wordu. Vzhledem k tomu, že nový Microsoft Word 2013/2016 má ve výchozím nastavení skryt styl Nadpis 2, návod na formátování stylu Nadpis 2 byl žákům předveden na následující hodině.⁵ Zdrojový text byl upraven aktivováním stylu Nadpis 2, proto při otevření souboru z přílohy *zdroj-prvni_pomoc.v2.doc* je styl Nadpis 2 již v seznamu.

Na ZŠ Donovalská žáci používali místo textového editoru Microsoft Word volně dostupný editor LibreOffice Writer. Někteří žáci měli problémy s ukládáním dokumentů, používáním stylů a formátováním dokumentu, proto v rámci této hodiny byly předvedeny názorné komentované ukázky, jak zmíněné akce provádět. Byť by se mělo jednat o dovednosti, které by žáci devátého ročníku základní školy měli již ovládat, nebylo tomu tak. Chybou bylo, že úvodní dotazník kvůli snaze o stručnost, neobsahoval otázky zabývající se základními dovednostmi obsluhy počítače, a tak někteří žáci nesplnili předpoklady této úlohy.

5 Formátování stylem Nadpis 2 se poprvé musí provést složitějším postupem. Šipkou dolů „Více“ (nikoliv šikmou šipkou) se zobrazí další styly, kde je volba „Použít styly...“. V nově otevřeném dialogu je možné ručně napsat styl Nadpis 2 a potvrdit stiskem tlačítka „Použít“. Poté se již styl Nadpis 2 zobrazí v seznamu stylů a je možné ho používat

Protože někteří žáci začali značky ze zdrojového textu doplňovat do textu v textovém editoru, byli v průběhu práce upozorněni, že znaky rovná se, pomlčky, podtržítka a hvězdičky mohou mít v textu určitý význam a ve výsledném textu se nemusí objevit. Na ZŠ Donovalská byli žáci výslovně upozorněni, že do textu nemají nic doplňovat, i přesto se našlo několik žáků, kteří začali do textu přepisovat značky.

Úloha nebyla náročná na provedení. Nejobtížnější na úloze bylo pochopení zadání práce. Jediný rozdíl mezi sedmým a devátým ročníkem byl v dovednostech práce s textovým editorem. V pochopení úlohy rozdíly shledány nebyly.

Vzhledem k tomu, že úlohu bylo pro žáky obtížnější pochopit, je nutné při zadávání této úlohy pečlivě dbát na používání pojmů „zdrojový text“ a „výsledný formátovaný dokument“, aby bylo žákům zcela jasné, o čem se mluví a co mají dělat. Je vhodné žákům vysvětlit důvody a význam značkování. Na základě této zkušenosti je vhodné mít připraveny všechny mezivýsledky, aby bylo možné žákům ukázat, jak má jejich práce vypadat a tím je správně směřovat v jejich další činnosti. Je ale třeba důrazně žáky upozornit, že jejich výsledky se mohou lišit.

5.2.4 Jazyk Markdown

Dalším krokem při plnění úlohy *4.2 Formátovaný dokument* bylo vysvětlit žákům pravidla syntaxe jazyka Markdown. Žáci dostali pomůcku podle přílohy *I Markdown pomůcka*, která byla vytištěna na papíru formátu A5 (2 stránky na stranu A4), s tím, že jim zůstane a mohou si do ní zapisovat svoje vlastní poznámky. Po zkušenostech z výuky byla pomůcka doplněna o formátování odstavců. Upravená pomůcka je uvedena v příloze *J Markdown pomůcka – verze 2*.

Žáci se vrátili k vytištěnému textu se značkami jazyka Markdown a na základě nově nabytých informací o jejich významu opravili formátování svého textu v textovém editoru. Při této úpravě mohli žáci vidět, kterým značkám jazyka Markdown intuitivně porozuměli.

Někteří žáci si u této úlohy nebyli jisti tím, jak mají postupovat. Proto bylo nutné jasně a srozumitelně popsat pravidla jazyka Markdown, jeho využití a uvést opět příklad se psaním čistě textové zprávy, kde bylo třeba zachovat formátování. Vhodné se jeví přidat další úlohu, než se začne více pracovat s jazykem Markdown. V této úloze by žáci měli za

úkol vytvořit vlastní značky pro označení tučného písma a kurzívy. Takové cvičení je uvedeno v kapitole 4.12.1 *Značkování*.

Na ZŠ Donovalská bylo zapotřebí výrazně vyšší osobní asistence při formátování dokumentu, kdy bylo nutné žáky upozorňovat, že všechny značky ve zdrojovém textu mají nějaký význam, který by žáci měli přenést do formátování v textovém editoru.

Nakonec všichni žáci úspěšně zvládli naformátovat dokument. Podle očekávání bylo při kontrole formátovaného dokumentu nutné dbát zvýšené pozornosti při ověřování používání číselovaného seznamu a odrážek. Někteří žáci se snažili toto formátování obejít ručním doplněním znaků nebo čísel místo toho, aby použili formát číselovaný seznam nebo odrážky.

Tato úloha měla zhruba podobnou obtížnost jako předchozí úloha. Někteří žáci opět bojovali se zadáním práce a nepochopili, že návod, co která značka znamená, mají vytištěn na pomůcce před sebou, a začali do textového editoru přepisovat značky jazyka Markdown. Problémy s formátováním nadpisů se navzdory očekávání neobjevily. Konkrétní rozdíly mezi sedmým a devátým ročníkem nebyly zřejmé.

5.2.5 Příprava nákupního seznamu

Ve cvičení podle kapitoly 4.3 *Nákupní seznam* měli žáci za úkol nejdříve vytvořit nákupní seznam, kde budou dvě kategorie. Každá kategorie bude mít minimálně 3 položky. Žádné další požadavky nejsou dány.

Žáci ZŠ Gutova byli předem upozorněni, aby si dali záležet na výběru položek, jelikož se seznamem budou pracovat i v dalších hodinách. Seznam měli tvořit v textovém editoru Notepad (Poznámkový blok), který nepodporuje formátování.

Na ZŠ Donovalská měli někteří žáci problém se spuštěním poznámkového bloku, jelikož v operačním systému Microsoft Windows 10 se po otevření nabídky Start a napsání textu „notepad“ aplikace poznámkový blok nevyhledala a bylo třeba hledat „poznámkový blok“.

Úloha nebyla obtížná, zvláště poté, co byl uveden příklad dvou kategorií, žáci si sami vytvořili vlastní dvě kategorie. Dodržení přesného zadání ohledně nadpisů, počtu kategorií a položek v každé kategorii bylo třeba ještě individuálně zdůraznit, a to při výuce na obou školách.

5.2.6 Nákupní seznam v jazyce Markdown

V této lekci měli žáci za úkol upravit vytvořený nákupní seznam doplněním nadpisu celého seznamu a upravením formátování pomocí jazyka Markdown. Na ZŠ Gutova žáci doplňovali značky jazyka Markdown přímo do textového editoru. Po vytvoření seznamu v editoru Notepad žáci vložili zdrojový text do online editoru jazyka Markdown a opravili si případné chyby podle výsledného formátování.

Na ZŠ Donovalská vzhledem k problémům žáků se základní obsluhou programů byla úloha upravena tak, že žáci začali formátovat text přímo v online editoru jazyka Markdown. Tento editor v pravé části zobrazuje již výsledný formátovaný dokument a tím pomáhá při tvorbě a formátování zdrojového dokumentu.

Přímá editace v tomto online nástroji sebou přinesla komplikace ohledně záměny datově-orientovaného značkovacího jazyka ve prospěch jazyka prezentačního. Někteří žáci měli Nadpis 1 podtržen jak znaky =, tak na dalším řádku znaky -. Tento způsob vytvořil elegantní vzhled dokumentu, kdy Nadpis 1 byl ve výsledku správně formátován a zobrazen ve správné velikosti a následně byl na celém řádku podtržen vodorovnou čarou, protože použití znaku mínus (-) na dalším řádku vedlo k tomu, že v dokumentu byla vykreslena horizontální čára. Prvek horizontální čára slouží k oddělení částí dokumentů a tudíž jeho použití v tomto místě bylo chybné. Správným postupem by byl export zdrojového kódu z jazyka Markdown do HTML, kde je možné upravit vzhled nadpisů a přidat jim spodní ohraničení.

V případě přímé editace v online nástroji je třeba vést žáky k tomu, aby i zdrojový kód byl čitelně formátovaný. Především nadpisy by měly být podtrženy v celé své délce, i když pro výsledný formát stačí jen jeden symbol. Tento problém se nevyskytl na ZŠ Gutova, kde žáci nejdříve prováděli formátování v Notepadu a kde viděli výsledný formátovaný dokument až na konci své práce.

Při formátování číslovaných seznamů, resp. odrážek bylo třeba žákům připomenout, že je nutné napsat číslo, tečku a mezeru, resp. symbol a mezeru, aby číslování, resp. odrážky fungovaly správně.

Vzhledem k samostatné tvůrčí povaze této úlohy byla úloha pro žáky náročnější. Vhodné je procházet třídou a na případné chyby upozornit nejlépe celou třídu, jelikož stejnou chybu může dělat i někdo další.

Žáci ZŠ Gutova na tomto místě dostali za úkol anonymně zhodnotit dosavadně proběhlou výuku. Všichni byli rádi za to, že se naučili něco nového. Někteří si stěžovali na téma, které je obtížnější na pochopení. Celkově se všichni žáci shodli, že se jim hodina líbila.

5.2.7 Struktura dokumentu a XML

Podle zadání úlohy 4.6 *Strukturování dokumentu* dostali žáci vytištěné nákupní seznamy se dvěma kategoriemi. Žákům bylo řečeno, kterou část si mají zakroužkovat, a byli vyzváni, aby se pokusili najít obecné nadřazené slovo označující obsah této části. Nejdříve bylo nutné žáky navést uvedením příkladu, pak byli, zvláště žáci deváté třídy, schopni sami nabídnout vhodná pojmenování. Takto byly postupně zakroužkovány a pojmenovány všechny části jedné kategorie nákupního seznamu. Části druhé kategorie nákupního seznamu měli žáci pojmenovat samostatně za využití předchozí společné práce.

Podle úlohy 4.8 *Seznam do XML* byla žákům rozdána pomůcka k vysvětlení pravidel pro tvorbu značek XML a pravidla jim byla vysvětlena. Pomůcka je v příloze *K XML pomůcka*. Na základě získaných informací o pravidlech tvorby názvů značek v XML bylo žákům zadáno, aby upravili názvy částí dokumentu odpovídajícím způsobem.

Žákům ZŠ Donovalská byl ukázán výsledný nákupní seznam, který je uveden v úloze 4.7 *Pojmenování částí značkami*. Žáci byli vyzváni, aby na tomto příkladě našli chyby v pojmenování. Vzhledem k tomu, že příklad v této úloze ještě není upraven podle pravidel a doporučení pro značky XML, obsahuje několik chyb. Tímto byla výuka na ZŠ Donovalská ukončena a žáci byli vyzváni k vyplnění závěrečných dotazníků.

Na ZŠ Gutova žáci získané znalosti ohledně formátování XML dokumentu, tvorby názvů značek a jejich použití včetně jejich párovosti, aplikovali na své nákupní seznamy. Dále byli žáci poučeni o možnosti využívat mezer nebo tabulátorů pro přehlednější členění XML dokumentu. Toto pravidlo již žáci znali z práce se zdrojovým kódem webové stránky, se kterým pracovali již v dřívějších hodinách s paní učitelkou.

Po provedené úpravě svých seznamů do značkovacího jazyka XML byli žáci vyzváni, aby své nákupní seznamy vložili do online XML validátoru na stránce

www.xmlvalidation.com. I přes to, že validátor je dostupný pouze v anglickém jazyce, žáci byli schopni s drobnou pomocí diagnostikovat chyby a provést jejich opravu.

Bylo vidět, zvláště v případě žáků sedmé třídy, že toto téma je velmi abstraktní a strukturování dat je náročná operace. Pro nižší ročníky druhého stupně by proto bylo vhodné přidat ještě další úlohy a více rozebrat důvody a důležitost strukturování dokumentu. Využít by bylo možné například úlohu z kapitoly *4.12.3 Datově orientovaný dokument*, kde mají žáci za úkol vymyslet způsob zápisu dat z tabulky do textového dokumentu.

5.2.8 XML dokument do tabulky

Další úloha byla inspirována úlohou *4.10 XML do tabulky*, kde byli žáci vyzváni k vyplnění tabulky. Na rozdíl od zadání v původní úloze dostali žáci nyní elektronickou verzi tabulky, která se nachází v příloze v kapitole *G Tabulka Menu* a zdrojový XML kód anglického menu, který je v příloze v kapitole *H Menu – Zdrojový kód XML*. Žáci měli za úkol se ve zdrojovém kódu dokumentu zorientovat a podle něj vyplnit tabulku.

Žáci úkol plnili s nadšením. Se záznamem menu z XML kódu do tabulky se nevyskytly žádné komplikace. Lze to přisuzovat i jednoduché struktuře zdrojového kódu, kterému by někteří porozuměli i v případě, že by jednotlivé značky nebyly srozumitelné. Touto úlohou bylo téma XML ukončeno.

5.2.9 Markdown do HTML

Dalším krátkým tématem byl export zdrojového kódu z jazyka Markdown do HTML. Vzhledem k tomu, že žáci ZŠ Gutova již znali jazyk HTML, byla úloha zvolena k procvičení a zopakování jazyka HTML a některých jeho značek. Žákům byl vysvětlen export z nástroje Dillinger podle kapitoly *4.4 Z Markdownu do HTML*.

Žáci vzhledem k jasnému a jednoduchému postupu a znalosti práce se zdrojovým kódem HTML neměli s exportem problémy.

5.2.10 SVG česká vlajka

Poslední úlohou vyučovanou na ZŠ Gutova byla úloha *4.11 Česká vlajka v SVG*. Žákům byl rozdán čtverečkový papír o velikosti A5. Úkolem žáků bylo nakreslit českou vlajku. Při kreslení se měli žáci zvláště soustředit, aby důležité body (např. rohy), byly na

mřížových bodech souřadnicové sítě. Žáci byli seznámeni s pravidly vzhledu vlajky, což jim připadalo vtipné. Pravděpodobně je pobavila představa, že by někdo nevěděl, jak vypadá česká vlajka. Bylo jim vysvětleno, že se jedná o státní symbol, který má pevně stanovený poměr stran a umístění pruhů a modrého klínu.

Pro větší názornost byla vlajka načrtnuta na tabuli. Při jejím načrtávání bylo cílem společně určit správné rozměry vlajky podle daných pravidel, protože práce s poměrem 2:3 by mohla slabším žákům činit potíže.

Žáci dostali na začátku pomůcku k jazyku XML a SVG, která je v příloze *K XML pomůcka* a v příloze *L SVG pomůcka*. Žákům byly stručně vysvětleny základní příkazy jazyka SVG a na základě ukázkové vlajky nakreslené na tabuli využity její souřadnice a příkazy SVG k ukázkovému zapsání vlajky. Vlajka byla na závěr zobrazena v internetovém prohlížeči. Vzhledem k blížícímu se konci hodiny žáci již neměli možnost více procvičit zápis kódu jazyka SVG a práci s ním. I tak byli žáci z jazyka SVG nadšení, jak vyplývá ze závěrečného dotazníku.

5.3 Závěrečný dotazník

Na závěr výuky dostaly obě třídy za úkol vyplnit závěrečný dotazník, jeho zadání je uvedeno v příloze *F Závěrečný dotazník*. Dotazník pro ZŠ Donovalská byl vzhledem k menšímu obsahu probraného tématu upraven a nezmiňoval jazyk SVG. Při vyhodnocování dotazníku je nutné přihlídnout ke skutečnosti, že objem a přístup k výuce značkovacího jazyka XML byl odlišný mezi ZŠ Gutova a ZŠ Donovalská.

Na ZŠ Gutova se dotazníku zúčastnilo 12 žáků. Žáci v otázce, která témata byla nejlehčí, nejčastěji odpovídali, že téma Markdown, následovalo téma SVG. Pro Markdown hlasovalo 10 žáků, pro SVG 5 žáků. Téma Markdown bylo pro žáky snadné především „jednoduchostí a zapamatovatelností příkazů“. Dále téma žáky potěšilo lehkostí pochopení a svojí zajímavostí.

Na ZŠ Donovalská se dotazníku zúčastnilo pouze 16 žáků, 2 žáci po první hodině odešli. Jako nejlehčí téma zvolilo 5 žáků téma Markdown. Mnoho žáků neoznačilo za snadné žádné téma. U jazyka Markdown žáci ocenili lehké ovládání, snadné pochopení a „měla jsem u sebe návod, na kterém bylo vše napsáno.“ 2 žáci zvolili téma XML, kde se jim líbila forma práce (kroužkování na papíře).

Jako nejtěžší téma na ZŠ Gutova zvolilo 6 žáků jazyk XML, 5 žáků jazyk SVG. Pro 2 žáky nebylo obtížné nic, všechna témata jim připadala lehká, jelikož se s nimi již setkali, nebo s nimi již pracují. XML přišlo žákům obtížné hlavně z důvodu celkové srozumitelnosti a někteří žáci si stěžovali, že „nestíhal jsem a moc jsem to nepochopil“. Na ZŠ Donovalská 13 žáků zvolilo jako nejobtížnější téma jazyk XML. Obtížnost žáci přičítali nepozornosti a nezábavnosti tématu. Dva žáci zvolili jako obtížný jazyk Markdown, a to z důvodů obtížného formátování.

K otázce „Která témata ti přinesla něco nového a případně co?“ žáci ZŠ Gutova často odpovídali, že „všechno“. Na ZŠ Donovalská 8 žáků volilo Markdown, 4 žáci odpověděli „nic“. K otázce „Která témata si myslíš, že bys mohl využít i v budoucnu“ žáci na ZŠ Gutova často uváděli jazyk SVG, někteří „všechno se může vždy hodit“. Na ZŠ Donovalská žáci nejčastěji volili jazyk Markdown. Žáci na ZŠ Donovalská se s jazykem SVG neseťkali.

Na otázku „Byla výuka pochopitelná?“ 7 žáků ZŠ Gutova odpovědělo jednoznačné ano, pro 5 žáků byla pochopitelná s výhradami. Mezi nejčastější výhrady uváděli žáci „ano, ale na mě možná trochu rychlá“ a „Jak co. Něco bylo těžší a něco lehčí na pochopení...“. Na ZŠ Donovalská byla pro 12 žáků výuka jednoznačně pochopitelná. Pro 3 žáky byla výuka částečně pochopitelná a na jednoho žáka byla výuka příliš rychlá.

5.4 Vyhodnocení ověření

Podle očekávání, úloha 4.1 *Formátování textu podle značek* nepřinesla žákům žádné problémy a jevila se žákům lehká. U úlohy 4.2 *Formátovaný dokument* měli žáci problémy s pochopením zadání úlohy, kdy docházelo k záměně zdrojového dokumentu a výsledného formátovaného dokumentu. Podle očekávání bylo nutné v rámci této úlohy připomenout práci se styly v textovém editoru, v případě třídy ZŠ Donovalská ještě navíc ovládání textového editoru. Při formátování výsledného dokumentu podle zdrojové kódu jazyka Markdown si někteří žáci nebyli jisti tím, jak mají postupovat. Vhodné se jeví přidat další úlohu pro lepší pochopení principu značkování. Taková úloha je v kapitole 4.12.1 *Značkování*. Podle očekávání se někteří žáci snažili formátování číslovaných seznamů obejít ručním doplněním čísel, místo toho, aby použili formát číslovaný seznam.

Problémy s formátováním nadpisů se navzdory očekávání neobjevily. Konkrétní rozdíly mezi sedmým a devátým ročníkem nebyly zřejmé.

Při tvorbě nákupního seznamu podle úlohy 4.3 *Nákupní seznam* se nevyskytly zásadní nedostatky. Při formátování nákupního seznamu pomocí jazyka Markdown bylo nutné zdůraznit pravidla pro psaní seznamů.

Podle očekávání bylo téma XML pro žáky náročné. Bylo vidět, zvláště v případě žáků sedmé třídy, že toto téma je velmi abstraktní a strukturování dat je pro žáky náročná operace. Pro nižší ročníky druhého stupně by proto bylo vhodné přidat ještě další úlohy a více rozebrat důvody a důležitost strukturování dokumentu.

Podle závěrečného dotazníku, jazyk SVG žáky zaujal. Potvrdilo se očekávání, že žáci budou mít problémy se souřadnicovým systémem. Jednalo se více o problémy práce v souřadnicovém systému obecně, než že by se jednalo o komplikace způsobené otočením orientace sítě souřadnic. Oproti očekávání, anglické příkazy ani operace s atributy práci nekomplikovali.

6 Závěr

V rámci této diplomové práce vznikla metodika pro výuku značkovacích jazyků Markdown, HyperText Markup Language (HTML), eXtensible Markup Language (XML) a Scalable Vector Graphics (SVG). Součástí metodiky jsou úkoly pro žáky, které probírané téma rozvíjí. Při ověřování metodiky vznikly pomůcky s informacemi ke značkovacím jazykům Markdown, XML a SVG, které je možné vytisknout a rozdat žákům pro jejich snazší orientaci v tématu.

Součástí metodiky je teoretická část, která umožňuje vyučujícímu, aby se seznámil s problematikou značkovacích jazyků, aniž by musel tyto informace hledat v externích zdrojích informací. Teoretická část v kapitole 2.2 *Syntaxe a značky* zahrnuje všechna důležitá pravidla včetně syntaxe značkovacích jazyků, která jsou potřebná pro použití této metodiky.

Na základě analýzy vybraných publikací z kapitoly 3 *Učebnice a skripta* se ukázalo, že značkovacím jazykům se obecně nevěnuje žádná publikace. Publikace se většinou zabývají konkrétně HTML. Přístup jednotlivých publikací se liší od pouhého vyjmenování značek HTML přes jejich detailnější popis až po práci ve specializovaném WYSIWYG⁶ editoru. Vzhledem k datu vydání některých publikací některé informace nebyly již aktuální.

Metodika začíná jednodušší úlohou s obecným značkovacím jazykem a dále navazujícími příklady postupně pokračuje ke složitějším značkovacím jazykům. Další jazyk, který metodika uvádí, je jazyk Markdown, který byl pro žáky nový a zajímavý. Při ověřování metodiky se objevily obtíže s pochopením zadání úloh. To mohlo být zapříčiněno zejména nedostatečným zdůrazněním, co je považováno za zdrojový kód a co je výsledný formátovaný dokument. Úlohy s jazykem Markdown vyžadují, aby byli žáci seznámeni se základy formátování v textovém editoru. Podle očekávání byl jazyk XML pro žáky náročný jak na pochopení, tak na představitost. Jazyk SVG přišel žákům zajímavý a v dotazníku uvedli, že ho využijí i v budoucnu.

Očekávané rozdíly ve výuce ve prospěch devátého ročníku se nepotvrdily. Mnohem větší roli než věk žáků hrála jejich motivace. Odpovídalo to rozdílnému zaměření předmětů.

6 Zkratka WYSIWYG zkracuje větu What You See Is What You Get, v překladu Co vidíte, to dostanete. Mezi takové editory HTML může být zařazen například Microsoft FrontPage.

Zatímco pro žáky devátého ročníku byl předmět povinný, žáci sedmého ročníku mohli vybírat z povinně-volitelných předmětů, a tedy se jednalo o jejich svobodnou volbu.

Jako nejtěžší se pro žáky ukázalo pochopení zadání úloh. Proto je potřeba při využití této metodiky dbát na formulaci zadání úloh a dalších pokynů pro žáky. Zejména je nutné pečlivě rozlišovat mezi zdrojovým kódem obsahujícím značky a výsledným formátovaným dokumentem.

Ověřilo se, že výuka značkovacích jazyků vedená podle této metodiky je pro druhý stupeň základní školy pochopitelná a zvládnutelná.

7 Seznam literatury

About W3C. In: *World Wide Web Consortium (W3C)* [online]. Last modified Fri, 22 Jan 2016 13:44:43 GMT [cit. 2016-02-13]. Dostupné v: <https://www.w3.org/Consortium/>

ANDRIES, Patrick, Samira CUNY and François YERGEAU, BRAY, Tim, Jean PAOLI a C. M. SPERBERG-MCQUEEN (eds.). Langage de balisage extensible (XML) 1.0: Recommandation du W3C, 10 février 1998. In: *World Wide Web Consortium (W3C)* [online]. 10 février 1998 [cit. 2016-02-13]. Dostupné v: <http://pages.videotron.com/fyergeau/w3c/xml10/REC-xml-19980210.fr.html>

BARTELS, Angela: *Internet History : HTML Code Evolution 1.0 to 5.0* [online]. May 3rd, 2011 [cit. 2016-01-09]. Dostupné v: <http://blog.rackspace.com/internet-history-html-evolution/>

BRAY, Tim and C. M. SPERBERG-MCQUEEN (eds.). *Extensible Markup Language (XML) : W3C Working Draft 14-Nov-96*. In: *World Wide Web Consortium (W3C)* [online]. 14-Nov-96 [cit. 2016-02-13]. Dostupné v: <http://www.w3.org/pub/WWW/TR/WD-xml-961114.html>

BRAY, Tim et al (eds.). *Extensible Markup Language (XML) 1.0 (Fifth Edition) : W3C Recommendation 26 November 2008*. In: *World Wide Web Consortium (W3C)* [online]. 2008 [cit. 2017-04-01]. Dostupné v: <https://www.w3.org/TR/REC-xml/#NT-NameStartChar>

CRHONEK, Vítězslav. *Historie značkovacích jazyků* [online]. Last modified Fri, 18 Apr 2003 12:47:11 GMT [cit. 2014-03-04]. Dostupné v: <https://www.fi.muni.cz/usr/jkucera/pv109/2003/xcrhonek.htm>

GRUBER, John. *Daring Fireball : Markdown* [online]. 2004 [cit. 2017-01-17]. Dostupné v: <http://daringfireball.net/projects/markdown/>

KLEMENT, Milan. *Výpočetní technika : Hardware a software*. Olomouc : Univerzita Palackého v Olomouci, 2001. ISBN 80-244-0316-1.

KOSEK, Jiří. *XML pro každého : podrobný průvodce*. Praha : Grada, 2000. 163 s. ISBN 8071698601.

- KOVÁŘOVÁ, Libuše. *Informatika pro základní školy : 3. díl*. Kralice na Hané: Computer Media, 2004. ISBN 80-866-8624-8.
- NAVRÁTIL, Pavel. *S počítačem nejen k maturitě : 2. díl*. Vydání páté. Kralice na Hané : Computer Media, 2004. ISBN 80-866-8619-1.
- PROCHÁZKA, David. *CSS a XHTML : tvorba dokonalých WWW stránek krok za krokem – 2., aktualizované vydání*. První vydání. Praha : Grada, 2011. ISBN 978-80-247-3897-0.
- RAMBOUSEK, Vladimír. *Praktické činnosti pro 6. – 9. ročník základních škol : Práce s počítačem (2)*. Praha: Nakladatelství Fortuna, 2003. ISBN 80-716-8874-6.
- RAMBOUSEK, Vladimír a kol. *Rozvoj informačně technologických kompetencí na základních školách : Výzkum stavu a struktury informačně technologické gramotnosti*. V Praze : České vysoké učení technické, 2013. 352 s. ISBN 978-80-01-05407-9.
- Rámcový vzdělávací program pro obor vzdělání 18-20-M/01 Informační technologie. Ministerstvo školství, mládeže a tělovýchovy, 29. 5. 2008. Čj. 6 907/2008-23.
- Rámcový vzdělávací program pro obor vzdělání 34-41-L/51 Polygrafický průmysl. Ministerstvo školství, mládeže a tělovýchovy, 19. 4. 2010. Č. j. 1606/2010-23.
- Rámcový vzdělávací program pro obor vzdělání 34-53-L/01 Reprodukční grafik pro média. Ministerstvo školství, mládeže a tělovýchovy, 29. 5. 2008. Čj. 6 907/2008-23.
- ROUBAL, Pavel. *Informatika a výpočetní technika pro střední školy : Teoretická učebnice*. Brno : Computer Press, 2010. ISBN 978-80-251-3228-9.
- Scalable Vector Graphics (SVG) 1.1 (Second Edition) : W3C Recommendation 16 August 2011. In: *World Wide Web Consortium (W3C)* [online]. 2011 [cit. 2017-03-05]. Dostupné v: <https://www.w3.org/TR/SVG11/>
- Systém souřadnic. HOLICA, Pavel. *Tutoriál pro vytváření SVG grafiky* [online]. Technické lyceum VOŠ a SPŠ elektrotechnické Olomouc, 2007 [cit. 2017-04-08]. Dostupné v: <http://jabber.spseol.cz/svg/texty/svg/souradnice.xhtml>
- Šokové stavy. In: *První pomoc Zásady první pomoci* [online]. [cit. 2017-03-12]. Dostupné v: http://www.ezsp6.estranky.cz/clanky/sokove-stavy_-sok.html

VANÍČEK, Jiří a Petr ŘEZNÍČEK. *Informatika pro základní školy : 1. díl Základy práce s PC*. Brno : Computer Press, 2004. ISBN 80-251-0196-7.

VRBÍK, Václav. *Internet a tvorba jednoduchých WWW stránek*. Plzeň : Západočeská univerzita v Plzni, 2001. ISBN 80-7082-724-6.

W3C Mission. In: *World Wide Web Consortium (W3C)* [online]. Last modified Fri, 22 Jan 2016 13:44:43 GMT [cit. 2016-02-13]. Dostupné v:

<https://www.w3.org/Consortium/mission>

XML Elements. In: *W3schools.com: The world's largest web developer site* [online]. [cit. 2017-04-04]. Dostupné v: https://www.w3schools.com/xml/xml_elements.asp

XML pro začátečníky – 2. část. BLAŽEK, Michal. In: *Programujte.com* [online]. 10. 7. 2007 [cit. 2017-03-05]. Dostupné v:

<http://programujte.com/clanek/2007062201-xml-pro-zacatecniky-2-cast/>

8 Seznam příloh

Příloha A: 10 návrhových cílů pro jazyk XML (orig. EN).....	78
Příloha B: Zdrojový text První pomoc.....	79
Příloha C: Pravidla pro názvy značek XML.....	80
Příloha D: Úvodní dotazník.....	80
Příloha E: Vzhled úvodního dotazníku.....	81
Příloha F: Závěrečný dotazník.....	81
Příloha G: Tabulka Menu.....	82
Příloha H: Menu – Zdrojový kód XML.....	82
Příloha I: Markdown pomůcka.....	83
Příloha J: Markdown pomůcka – verze 2.....	84
Příloha K: XML pomůcka.....	86
Příloha L: SVG pomůcka.....	87
Příloha M: Data: Úvodní dotazník ZŠ Gutova.....	89
Příloha N: Data: úvodní dotazník ZŠ Donovalská.....	90
Příloha O: Data: Závěrečný dotazník ZŠ Gutova.....	91
Příloha P: Data: Závěrečný dotazník ZŠ Donovalská.....	92

Příloha A: 10 návrhových cílů pro jazyk XML (orig. EN)

The design goals for XML are: (Bray, 1996)

1. XML shall be straightforwardly usable over the Internet.
2. XML shall support a wide variety of applications.
3. XML shall be compatible with SGML.
4. It shall be easy to write programs which process XML documents.
5. The number of optional features in XML is to be kept to the absolute minimum, ideally zero.
6. XML documents should be human-legible and reasonably clear.
7. The XML design should be prepared quickly.
8. The design of XML shall be formal and concise.
9. XML documents shall be easy to create.
10. Terseness is of minimal importance.

Příloha B: Zdrojový text První pomoc

Zdrojový text (Šokové stavy, 2017) vhodný k elektronické distribuci se nachází v souboru zdroj-prvni_pomoc.doc ve formátu Microsoft Word 97, který je kompatibilní i s novějšími verzemi Microsoft Office a sadami OpenOffice a LibreOffice.

První pomoc

První pomoc jsou úkony a opatření, která při náhlém ohrožení nebo postižení zdraví člověka omezují rozsah a důsledky tohoto ohrožení či postižení. Poskytnout první pomoc je v rámci svých schopností a sil povinen každý občan České republiky starší 18 let, pokud tím neohrozí svoje zdraví či život.

Postup resuscitace

Nehrozí nebezpečí?

Bez známek života? (Nereaguje, nedýchá)

Přivolejte pomoc, telefon 155 nebo 112

Resuscitujte – Mačkejte propnutýma rukama střed hrudní kosti 100 – 120 x za minutu do hloubky 5 - 6 cm.

Pokračujte do příjezdu záchranky

Postup v případě šoku

Pravidlo 5T

Teplo: Zajistit dostatečný tepelný komfort, ale ne přehřátí (v létě do stínu)

Ticho: Zajistit klidné prostředí, ale s postiženým komunikovat

Transport: Zajistit odsun k lékaři

Tišení bolesti: V žádném případě nepodávat léky, spíše uklidňovat, vše vysvětlovat

Tekutiny: V žádném případě nedávat napít, ale pouze otírat a zvlhčovat rty, obličej, maximálně několik kávových lžiček vody, nebo čaje.

Příloha C: Pravidla pro názvy značek XML

Název značky se skládá z prvního znaku (NameStartChar) a libovolného počtu dalších znaků (NameChar). Délka není omezena. (Bray, 2008)

NameStartChar může nabývat jedné z těchto hodnot (údaje v hranaté závorce udávají rozsah):

: [A-Z] _ [a-z] [#xC0-#xD6] [#xD8-#xF6] [#xF8-#x2FF] [#x370-#x37D]
[#x37F-#x1FFF] [#x200C-#x200D] [#x2070-#x218F] [#x2C00-#x2FEF]
[#x3001-#xD7FF] [#xF900-#xFDCF] [#xFDF0-#xFFFD] [#x10000-#xEFFFF]

NameChar může nabývat stejných hodnot jako NameStartChar a ještě těchto dalších:

- . [0-9] #xB7 [#x0300-#x036F] [#x203F-#x2040]

Příloha D: Úvodní dotazník

Otázka (typ odpovědí)	Možné odpovědi
Jméno příjmení (text)	
V budoucnu bych chtěl umět (vícenásobný výběr)	Programovat; Vytvářet webové stránky; Řešit problémy s počítači
Jazyk Markdown (výběr)	Znám; Slyšel jsem; Nemám tušení
Tvorba webových stránek (výběr)	Dělám běžně; Už jsem tvořil; Tuším, co to je; Netuším, co to je
HTML (výběr)	Vím, co to je; Tuším, co to je; Slyšel jsem, ale víc o tom nevím; Netuším, co to je
Zdrojový kód webové stránky (výběr)	Viděl jsem; Neviděl jsem; Netuším
XML (výběr)	Vím, co to je; Tuším, co by to mohlo být; Netuším, co to je
SVG (výběr)	Vím, co to je; Tuším, co by to mohlo být; Netuším, co to je

Příloha E: Vzhled úvodního dotazníku

Značkovací jazyky

Vyplňte podle skutečnosti

Jméno a příjmení *

Short-answer text

V budoucnu bych chtěl umět

- programovat
- vytvářet webové stránky
- řešit problémy s počítači

Jazyk Markdown *

- Zním
- Slyšel jsem
- Nemám tušení

Tvorba webových stránek *

- Dělán běžně
- Už jsem tvořil
- Tuším, co to je
- Netuším, co to je

Příloha F: Závěrečný dotazník

Otázka (typ odpovědi)	Možné odpovědi
Jméno a příjmení (text)	
Které téma bylo nejlehčí (vícenásobný výběr)	Markdown; XML; SVG
Čím pro tebe bylo snadné? (dlouhý text)	
Které téma bylo nejtěžší? (vícenásobný výběr)	Markdown; XML; SVG
Čím pro tebe bylo nejtěžší? (dlouhý text)	

Otázka (typ odpovědi)	Možné odpovědi
Která témata ti přinesla něco nového a případně co? (dlouhý text)	
Která témata si myslíš, že bys mohl využít i v budoucnu? (dlouhý text)	
Byla výuka pochopitelná? (dlouhý text)	

Příloha G: Tabulka Menu

Jméno	Cena	Popis	Kalorie

Příloha H: Menu – Zdrojový kód XML

XML Tutorial. *W3schools.com* [online]. [cit. 2017-04-06]. Dostupné z:

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

```
<?xml version="1.0" encoding="UTF-8"?>
<breakfast_menu>
<food>
  <name>Belgian Waffles</name>
  <price>$5.95</price>
  <description>
    Two of our famous Belgian Waffles with plenty of real
    maple syrup
  </description>
  <calories>650</calories>
</food>
<food>
  <name>Strawberry Belgian Waffles</name>
  <price>$7.95</price>
```

```

    <description>
        Light Belgian waffles covered with strawberries and
whipped cream
    </description>
    <calories>900</calories>
</food>
</breakfast_menu>

```

Příloha I: Markdown pomůcka

Text vhodný k tisku 2 stránky na 1 je v přílohách v souboru markdown_sheet.pdf.

Nadpisy

Pro formátování nadpisů je možné použít dva rozdílné způsoby zápisu. První zápis, graficky více oddělený ve zdrojovém kódu, umožňuje formátování pouze stylů *Nadpis 1* a *Nadpis 2*. *Nadpis 1*, resp. *Nadpis 2* se formátuje podtržením na dalším novém řádku pomocí znaku rovná se (=), resp. spojovník (-). Druhou možností je odsadit text příslušným počtem křížků (#). Například pro *Nadpis 1* je to jeden křížek a pro *Nadpis 6* se jedná o šest křížků. Alternativně a pro lepší vzhled zdrojového kódu může být použito označení křížky i na konci nadpisu.

<pre>Toto je Nadpis 1 ===== Toto je Nadpis 2 -----</pre>	<pre># Toto je Nadpis 1 # Toto je Nadpis 2 ##### Toto je Nadpis 6</pre>	<pre># Toto je Nadpis 1 # ## Toto je Nadpis 2 ## ##### Toto je Nadpis 6 #####</pre>
--	---	---

Seznamy

Markdown podporuje dva typy seznamů. Řazené číslované seznamy a nečíslované seznamy (odrážky). Pro vytvoření číslovaného seznamu je třeba, aby každá položka byla na vlastním řádku a začínala číslovkou následovanou tečkou a mezerou.

1. Otevřít
2. Upravit
3. Uložit

Nečíslovaný seznam také požaduje, aby každá položka byla na vlastním řádku. Řádek musí začínat hvězdičkou (*), plusem (+) nebo spojovníkem (-) a poté musí následovat mezera.

* Modrá	+ Modrá	- Modrá
* Červená	+ Červená	- Červená
* Zelená	+ Zelená	- Zelená

Tučné a kurzíva

Zvýrazňování textu **tučně** nebo *kurzívou* se použije ohraničení daného textu hvězdičkou (*) nebo podtržítkem (_). Pro **tučné** písmo se uvádějí dva znaky za sebou, pro *kurzívu* jen jeden.

kurzíva označení hvězdičkou nebo _kurzíva označená podtržítkem_

tučné písmo nebo __tučné písmo__

Příloha J: Markdown pomůcka – verze 2

Text vhodný k tisku 2 stránky na 1 je v přílohách v souboru markdown_sheet.v2.pdf.

Odstavec

Odstavce se oddělují jedním prázdným řádkem. Větší počet mezer za sebou ve výsledku vykresluje vždy jen jednu mezeru. Ostatní mezery jsou ignorovány.

Nadpisy

Pro formátování nadpisů je možné použít dva rozdílné způsoby zápisu. První zápis, graficky více oddělený ve zdrojovém kódu, umožňuje formátování pouze stylů *Nadpis 1* a *Nadpis 2*. *Nadpis 1*, resp. *Nadpis 2* se formátuje podtržením na dalším novém řádku pomocí znaku rovná se (=), resp. spojovník (-). Druhou možností je odsadit text příslušným počtem křížků (#). Například pro *Nadpis 1* je to jeden křížek a pro *Nadpis 6* se jedná o šest

křížků. Alternativně a pro lepší vzhled zdrojového kódu může být použito označení křížky i na konci nadpisu.

```
Toto je Nadpis 1 # Toto je Nadpis 1 # Toto je Nadpis 1 #
=====
Toto je Nadpis 2 ## Toto je Nadpis 2 ## Toto je Nadpis 2 ##
-----

##### Toto je Nadpis 6 ##### Toto je Nadpis 6
6 #####
```

Seznamy

Markdown podporuje dva typy seznamů. Řazené číslované seznamy a nečíslované seznamy (odrážky). Pro vytvoření číslovaného seznamu je třeba, aby každá položka byla na vlastním řádku a začínala číslovkou následovanou tečkou a mezerou.

1. Otevřít
2. Upravit
3. Uložit

Nečíslovaný seznam také požaduje, aby každá položka byla na vlastním řádku. Řádek musí začínat hvězdičkou (*), plusem (+) nebo spojovníkem (-) a poté musí následovat mezera.

* Modrá	+ Modrá	- Modrá
* Červená	+ Červená	- Červená
* Zelená	+ Zelená	- Zelená

Tučné a kurzíva

Zvýrazňování textu **tučně** nebo *kurzívou* se použije ohraničení daného textu hvězdičkou (*) nebo podtržítkem (_). Pro **tučné** písmo se uvádějí dva znaky za sebou, pro *kurzívu* jen jeden.

kurzíva označení hvězdičkou nebo _kurzíva označená podtržítkem_

tučné písmo nebo __tučné písmo__

Příloha K: XML pomůcka

Text vhodný k tisku je v souboru xml_svg_sheet.pdf na straně jedna. Text je určen k tisku současně s pomůckou SVG, kdy se provede tisk 2 stránky na 1 stranu A4 a každý žák dostane celý list A4, kde bude pomůcka pro XML a SVG.

eXtensible Markup Language

Značky XML se uvádějí do **lomených závorek** „<“ a „>“ stejně jako v HTML. Část dokumentu je uvedena **počáteční** značkou a ukončena **koncovou značkou**, která má stejný název a její název je uvozen lomítkem „/“. Je třeba dodržet **stejnou velikost písmen** v názvu značky. Značky **<nazev>** a **<Nazev>** jsou dvě různé značky. Všechny značky jsou **povinně párové**, tedy každá značka musí být i ukončena. Pro značku, která nemá žádný obsah, lze použít zkrácení zápisem lomítka za mezeru v uvozující značce, např. **<nazev />**. Stejně jako v HTML se značky jazyka XML **nesmějí křížit**. Každá značka může obsahovat atributy. Atributy se zapisují za název značky oddělený mezerou stále uvnitř lomených závorek. Ukázka značky s atributem a prázdné značky.

```
<nazev_znacky atribut="hodnota">Obsah značky</nazev_kapitoly>  
<podnazev />
```

Pravidla pro tvorbu názvu značky nebo atributu:

- Název může obsahovat všechna písmena, číslice a znaky dvojtečka (:), podtržítka (_), mínus (-) a tečka (.).
- Název nesmí začínat číslicí, mínusem (-) nebo tečkou (.).
- Název nesmí začínat XML v libovolné velikosti písmen (Xml, xMl, atp.).

Dobré návyky při tvorbě názvů značek XML:

- Znak mezera není povolen, ale může být nahrazen znakem podtržítka (_).
- Znak dvojtečka (:) má speciální význam, její použití se řídí dalšími pravidly.
- Znaky tečka (.) a mínus (-) mohou být v názvu značky některými aplikacemi chybně interpretovány, proto tyto znaky nepoužívat.
- Nepoužívat v názvech diakritiku, některé aplikace s ní neumí pracovat.

Struktura XML dokumentu

Celý dokument XML musí být uzavřen v jedné značce. Stejně jako je tomu u HTML, kde tuto funkci plní značka **<html>**.

Pro správné nastavení kódové stránky XML dokumentu by mělo být na první řádce předcházející celému dokumentu definována verze XML a nastaveno kódování.

```
<?xml version="1.0" encoding="UTF-8"?>
<dokument>
...
</dokument>
```

Příloha L: SVG pomůcka

Text vhodný k tisku je v souboru xml_svg_sheet.pdf na straně dvě. Text je určen k tisku současně s pomůckou XML, kdy se provede tisk 2 stránky na 1 stranu A4 a každý žák dostane celou list A4, kde bude pomůcka pro XML a SVG.

Scalable Vector Graphics

SVG je jazyk sloužící k popisu dvourozměrné vektorové grafiky v XML. Celá grafika je uzavřena ve značce `<svg>`. Takto může být vložena i do dokumentu HTML, moderní prohlížeče ji umí zobrazit.

Kružnice

Značkou pro nakreslení kružnice je `<circle>`.

Atributy:

- `cx, cy` Souřadnice X a Y středu kružnice v pixelech
- `r` Poloměr kružnice v pixelech
- `fill` Barva kružnice, lze použít anglické označení (red, blue, yellow, ...)

```
<circle cx="100" cy="100" r="50" fill="yellow" />
```

Obdélník

Značkou pro nakreslení obdélníku slouží značka `<rect>`.

Atributy:

- `x, y` Souřadnice levého horního rohu v pixelech
- `width` Šířka v pixelech
- `height` Výška v pixelech
- `fill` Barva obdélníku, lze použít anglické označení (red, yellow, ...)

```
<rect x="20" y="40" width="30" height="10" fill="yellow" />
```

Polygon (n-úhelník, jako trojúhelník, čtyřúhelník pětiúhelník, ...)

Značkou pro nakreslení polygonu slouží značka **<polygon>**.

Atributy:

- fill Barva polygonu, lze použít anglické označení (green, yellow, ...)
- points Seznam vrcholů oddělených mezerou. Vrchol je zapsán jako dvojice X,Y.

<polygon fill="red" points="0,0 150,100 0,200" />

Příloha M: Data: Úvodní dotazník ZŠ Gutova

V budoucnu bych chtěl umět	Jazyk Markdown	Tvorba webových stránek	HTML	Zdrojový kód webové stránky	XML	SVG
řešit problémy s počítači	Slyšel jsem	Tuším, co to je	Vím, co to je	Viděl jsem	Netuším, co to je	Netuším, co to je
programovat	Slyšel jsem	Dělám běžně	Vím, co to je	Viděl jsem	Vím, co to je	Vím, co to je
vytvářet webové stránky	Nemám tušení	Už jsem tvořil	Tuším, co to je	Viděl jsem	Netuším, co to je	Netuším, co to je
programovat	Nemám tušení	Tuším, co to je	Vím, co to je	Viděl jsem	Netuším, co to je	Netuším, co to je
řešit problémy s počítači	Nemám tušení	Už jsem tvořil	Vím, co to je	Viděl jsem	Netuším, co to je	Netuším, co to je
programovat	Nemám tušení	Už jsem tvořil	Vím, co to je	Viděl jsem	Tuším, co by to mohlo být	Tuším, co by to mohlo být
programovat, řešit problémy s počítači	Slyšel jsem	Už jsem tvořil	Vím, co to je	Viděl jsem	Netuším, co to je	Netuším, co to je
programovat, vytvářet webové stránky	Nemám tušení	Už jsem tvořil	Vím, co to je	Viděl jsem	Netuším, co to je	Netuším, co to je
programovat	Nemám tušení	Už jsem tvořil	Vím, co to je	Viděl jsem	Netuším, co to je	Netuším, co to je
vytvářet webové stránky	Nemám tušení	Už jsem tvořil	Vím, co to je	Viděl jsem	Netuším, co to je	Netuším, co to je
vytvářet webové stránky	Nemám tušení	Tuším, co to je	Tuším, co to je	Viděl jsem	Netuším, co to je	Netuším, co to je
řešit problémy s počítači	Nemám tušení	Už jsem tvořil	Tuším, co to je	Viděl jsem	Netuším, co to je	Netuším, co to je
vytvářet webové stránky, řešit problémy s počítači	Nemám tušení	Netuším, co to je	Slyšel jsem, ale víc o tom nevím	Viděl jsem	Netuším, co to je	Netuším, co to je
řešit problémy s počítači	Nemám tušení	Už jsem tvořil	Vím, co to je	Viděl jsem	Netuším, co to je	Netuším, co to je

Příloha N: Data: úvodní dotazník ZŠ Donovalská

V budoucnu bych chtěl umět	Jazyk Markdown	Tvorba webových stránek	HTML	Zdrojový kód webové stránky	XML	SVG
programovat	Nemám tušení	Tuším, co to je	Slyšel jsem, ale víc o tom nevím	Neviděl jsem	Netuším, co to je	Netuším, co to je
programovat	Nemám tušení	Tuším, co to je	Tuším, co to je	Viděl jsem	Netuším, co to je	Netuším, co to je
ani jedno z nabízených	Nemám tušení	Tuším, co to je	Slyšel jsem, ale víc o tom nevím	Netuším	Netuším, co to je	Netuším, co to je
řešit problémy s počítači	Nemám tušení	Tuším, co to je	Tuším, co to je	Neviděl jsem	Netuším, co to je	Netuším, co to je
vytvářet webové stránky	Nemám tušení	Tuším, co to je	Netuším, co to je	Netuším	Netuším, co to je	Netuším, co to je
řešit problémy s počítači	Nemám tušení	Už jsem tvořil	Slyšel jsem, ale víc o tom nevím	Netuším	Netuším, co to je	Netuším, co to je
ani jedno z nabízených	Nemám tušení	Dělám běžně	Slyšel jsem, ale víc o tom nevím	Netuším	Netuším, co to je	Netuším, co to je
ani jedno z nabízených	Nemám tušení	Netuším, co to je	Netuším, co to je	Netuším	Netuším, co to je	Netuším, co to je
vytvářet webové stránky	Nemám tušení	Tuším, co to je	Slyšel jsem, ale víc o tom nevím	Neviděl jsem	Netuším, co to je	Netuším, co to je
ani jedno z nabízených	Nemám tušení	Tuším, co to je	Netuším, co to je	Netuším	Netuším, co to je	Netuším, co to je
ani jedno z nabízených	Nemám tušení	Tuším, co to je	Slyšel jsem, ale víc o tom nevím	Netuším	Netuším, co to je	Netuším, co to je
ani jedno z nabízených	Nemám tušení	Tuším, co to je	Slyšel jsem, ale víc o tom nevím	Neviděl jsem	Netuším, co to je	Netuším, co to je
ani jedno z nabízených	Nemám tušení	Tuším, co to je	Slyšel jsem, ale víc o tom nevím	Neviděl jsem	Netuším, co to je	Netuším, co to je
ani jedno z nabízených	Nemám tušení	Tuším, co to je	Netuším, co to je	Netuším	Netuším, co to je	Netuším, co to je
ani jedno z nabízených	Nemám tušení	Už jsem tvořil	Vím, co to je	Netuším	Netuším, co to je	Netuším, co to je
ani jedno z nabízených	Nemám tušení	Tuším, co to je	Netuším, co to je	Viděl jsem	Netuším, co to je	Netuším, co to je
vytvářet webové stránky	Nemám tušení	Tuším, co to je	Slyšel jsem, ale víc o tom nevím	Viděl jsem	Tuším, co by to mohlo být	Netuším, co to je
řešit problémy s počítači	Nemám tušení	Tuším, co to je	Slyšel jsem, ale víc o tom nevím	Neviděl jsem	Netuším, co to je	Netuším, co to je

Příloha O: Data: Závěrečný dotazník ZŠ Gutova

Které téma bylo nejjednodušší	Čím pro tebe bylo snadné?	Které téma bylo nejtěžší?	Čím pro tebe bylo nejtěžší?	Která témata ti přinesla něco nového a případně co?	Která témata si myslíš, že bys mohl využít i v budoucnu?	Byla výuka pochopitelná?
Markdown	Jednoduchost znaků	XML, SVG	srozumitelností	všechno	Myslím, že v mém případě je nevyužiju	z 90%
Markdown	Zapamatovat si příkazy	SVG	Zapamatovat si příkazy	Všechna témata a všechno	SVG	Ano
Markdown	Nejsem si jistý	XML	Nejsem si jistý	Hodně věcí	Nejsem si jistý	Ano
Markdown	Všechno mi přišlo lehké. Pracuji s tím 2 roky.	Markdown	Všechno mi přišlo lehké. Pracuji s tím 2 roky.	Markdown je nové.	SVG	Ano, určitě!!!!
Markdown, SVG	zda se mi to hodne srozumitelne protoze uz umim html	Markdown	musim neco zvolit jinak to bylo lehke	XML markdown	svg	ano moc
Markdown	Bylo nejjednodušší na pochopení	SVG	Bylo těžké na psaní	SVG - protože jsem se naučila dělat tvary	Asi XML a SVG	Jak co. Něco bylo těžší a něco lehčí na pochopení...
Markdown, SVG	Přišlo mi to fajn a rychle jsem to pochopila	XML	Moc jsem nepochopila	Všechna něco super a nového	Všechna	Samozřejmě
SVG	celkově to bylo zajímavé celé to téma	XML	celé	SVG	SVG	někdy ne
SVG	nevím	XML	nevím	markdown, xml, svg	nevím	ano
SVG	pochopil jsem	XML	nestíhal jsem a moc jsem to nepochopil	všechny	všechny	ano
Markdown	nemuseli se psát dlouhé zápisy	SVG	celkově	všechna byla pro mě nová a zajímavá	SVG	ano, ale na mě možná trochu rychlá
Markdown	bylo prostě nejjednodušší:-)	SVG	trošku složitější jazyk i na psaní	všechna, jen jsem o nich slyšel	všechno se může vždy hodit:-)	celkem ano ale já jsem pomalejší takže nestíhání

Příloha P: Data: Závěrečný dotazník ZŠ Donovalská

Které téma bylo nejlehčí	Čím pro tebe bylo snadné?	Které téma bylo nejtěžší?	Čím pro tebe bylo nejtěžší?	Která témata ti přinesla něco nového a případně co?	Která témata si myslíš, že bys mohl využít i v budoucnu?	Byla výuka pochopitelná?
Markdown	Bylo lehčí	XML	Bylo složitější zadání	Obě	Markdown	Docela ano
Markdown	víc me to bavilo	XML	nedával jsem pozor	markdown	markdown	castecne
Markdown	ovládání	XML		markdown	markdown	ano
Markdown	ničím	XML	vším	nic	já žádné	ano
Markdown	ničím	XML	všechno	asi nic	žádné	teoreticky
Markdown		XML		markdown	v psaní na internetu	ano
Markdown	ez life			Spíš utvrdila .	poflexime	yy
Markdown	Lehké na pochopení	XML	Už mě tolik nebavilo	Už vím,jak s tím pracovat	Markdown	Ano
XML	krouškování na papír	Markdown	všechno	nic	nic	dobry
Markdown		XML		markdown	nejspíš žádné	ano
Markdown		XML		Markdown	Žádná	Ano
Markdown		XML		markdown	asi žádné	ano
Markdown	Měla jsem u sebe navod ,na kterém bylo vše napsáno.	XML	Ani nevím, ale připadalo mi jen trochu složilo	Markdown - uprava textu	markdown	ano
XML	povídání	Markdown	formátování	markdown	markdown	ano
Markdown		XML		obě dvě	asi ani jedno	ANO, byla.
Markdown	Kvůli dřívější práci s např s Excelem ... a upravování textových dokumentů není pro mě moc těžký .. ^-^	XML	Naprostá předchozí neznalost	Editování pomocí znaků na klávesnici byla ... ehmm dost zajímavá ^-^	Markdown ^-^	SPíše Markdown jsem rychle pochopil ale XML ... neznal jsem to což znamená že jsem to pomaleji pochopil ^-^ :D