



Agentúra Ministerstva školstva, vedy,
výskumu a športu SR pre štrukturálne fondy EÚ



Výstup:

Návod na tvorbu štruktúrovaných materiálov na cvičenia

Riešiteľský kolektív	Jaroslav Porubän, Michaela Bačíková
-----------------------------	-------------------------------------

História dokumentu:

Verzia	Autor(i)	Dátum	Sumár zmien
0.1	Jaroslav Porubän		Úvodná verzia.
0.2	Michaela Bačíková		Doplnenie opisu jednotlivých krokov.
0.3	Jaroslav Porubän		Oprava a doplnenie textov.
0.4	Michaela Bačíková		Doplnenie opisu elementov.
0.5	Michaela Bačíková		Doplnenie časti „Pokročilá práca s projektom“.
0.6	Jaroslav Porubän		Oprava a doplnenie textov.
0.7	Michaela Bačíková		Doplnenie opisu elementu <math>, drobné úpravy.
0.8	Michaela Bačíková	5.10.2013	Doplnenie opisu elementu <solution> podľa upravenej implementácie.
0.9	Michaela Bačíková	17.10.2013	Doplnenie časti „Zaheslovanie prístupu k učiteľským materiálom“ a časti „Vygenerované súbory“.
1.0	Michaela Bačíková	21.10.2013	Doplnenie atribútu <i>size</i> pre elementy <i>image</i> . Oprava chybného opisu elementu <i>link</i> .
1.1	Michaela Bačíková	26.10.2013	Úprava návodu pre zaheslovanie učiteľských materiálov.
1.2	Michaela Bačíková	30.10.2013	Úprava opisu elementov <solution> a <result> na základe zmien v kóde. Teraz moze v ulohe byt aj solution aj result.
1.3	Sergej Chodarev Michaela Bačíková	20.11.2013	Doplnenie opisu elementu <code>. Doplnenie opisu elementu <table>.
1.4	Michaela Bačíková	03.12.2013	Doplnenie návodu pre použitie prídavného modulu.
			Doplnenie návodu o opis štruktúry xml opisujúceho úvodnú stránku predmetu.
			Doplnenie opisu elementu <i>objective</i> .
1.5	Michaela Bačíková	04.12.2013	Oprava, doplnenie textov.

Obsah

1.	Inštalácia editora	4
2.	Stiahnutie projektu z SVN repozitára	4
2.1.	Nastavenie zobrazenia adresárov v editore	8
2.2.	Aktualizácia projektu – vždy pred prácou!	9
2.3.	Odoslanie materiálov predmetu na SVN	10
3.	Štruktúra projektu a práca s IDE.....	11
3.1.	Štruktúra projektu <i>IT4KT Practices</i>	11
3.2.	Výber profilu predmetu, generovanie materiálov a spustenie	12
3.3.	Tvorba materiálov v editore	13
4.	Publikovanie vytvorených materiálov v systéme moodle	16
5.	Štruktúra XML.....	18
5.1.	Element module	18
5.2.	Element title	18
5.3.	Element objective.....	18
5.4.	Element introduction	19
5.5.	Element step.....	19
5.6.	Element resource	19
5.7.	Element additional	19
5.8.	Element task.....	20
5.9.	Element comment.....	20
5.10.	Elementy solution a result.....	20
5.11.	Ďalšie elementy	21
	Element figure.....	21
	Element image.....	22
	Element table	23
	Element math.....	24
	Element code.....	24
5.12.	Použitie HTML elementov	26
6.	Pokročilá práca s projektom.....	27
6.1.	Zmena formy	27
	Zmena CSS štýlov.....	27
	Zmena XSLT šablón.....	27
	Zmena CSS štýlov.....	27
6.2.	Zmena štruktúry	28
6.3.	Zaheslovanie prístupu k učiteľským materiálom.....	30
7.	Vygenerované súbory.....	32
8.	Úvodná stránka k predmetu.....	34

8.1.	Element subjectInfo	34
8.2.	Element title	34
8.3.	Element objective.....	34
8.4.	Element additionalInfo	35
8.5.	Element conditions.....	35
8.6.	Element scores	35
8.7.	Element assignments.....	35
8.8.	Element assignment	35
8.9.	Element resource	36
8.10.	Úvodná stránka v prehliadači	36
9.	Prídavný modul pre prácu s XML modulmi	37
1.1.	Návod na inštaláciu	37
1.2.	Návod na použitie.....	42
	Vloženie jednoduchého elementu pomocou palety	44
	Vloženie parametrického elementu pomocou palety.....	45
	Elementy palety súboru info.xml	47
	Vytváranie nových súborov pomocou prídavného modulu.....	48

1. Inštalácia editora

Pre vytváranie učebných materiálov je potrebné si stiahnuť editor **NetBeans IDE** nakonfigurovaný pre XML z:

https://bodrog.fei.tuke.sk/it4kt/PORADY/A31_porady/20121130/netbeans.zip

Tento editor je zatiaľ možné používať len na operačnom systéme **Windows**.

Po stiahnutí je potrebné súbor **rozbalíť** pomocou komprimačného programu (napr. WinZip, WinRar, Total Commander).

Po rozbalení je potrebné spustiť dvojklikom súbor „**netbeans.bat**“ v adresári „**netbeans**“. Zobrazí sa inštaláčny dialóg. Je potrebné vykonať nasledujúce kroky:

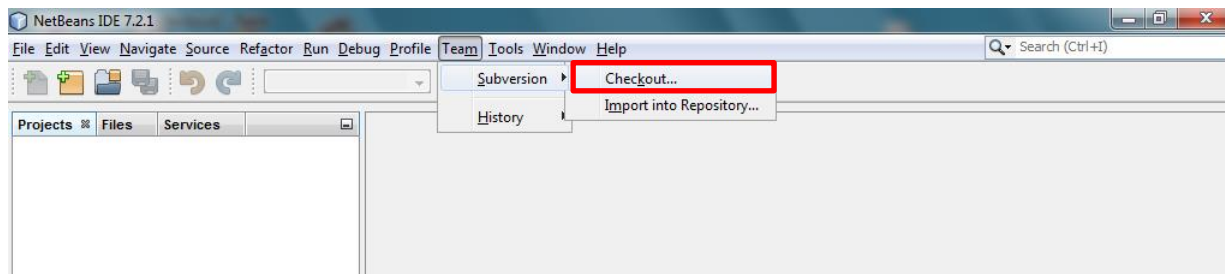
- akceptovať licenciu programu NetBeans IDE prostredníctvom „**I Accept**“,
- ak ste predtým mali nainštalovanú inú verziu NetBeans IDE, potvrdíte importovanie nastavení pomocou tlačidla „**Yes**“,
- spustí sa editor NetBeans IDE.

Spustenie editora NetBeans IDE je odteraz možné pomocou dvojkliku na súbor „**netbeans.bat**“.

2. Stiahnutie projektu z SVN repozitára

Na to, aby bolo možné vytvárať učebné materiály, je potrebné stiahnuť projekt pre vytváranie materiálov zo Subversion repozitára.

V menu NetBeans IDE vyberte „**Team -> Subversion -> Checkout...**“.

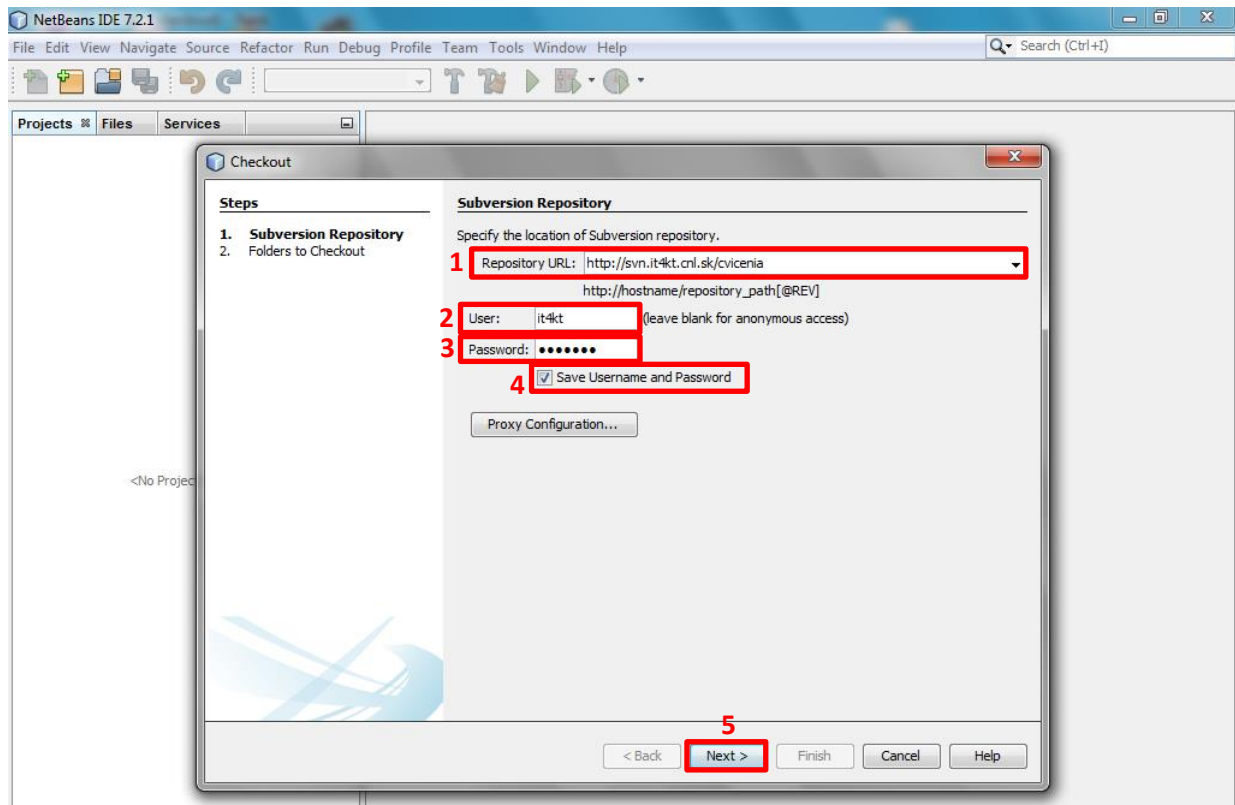


Zobrazí sa dialóg „Checkout“.

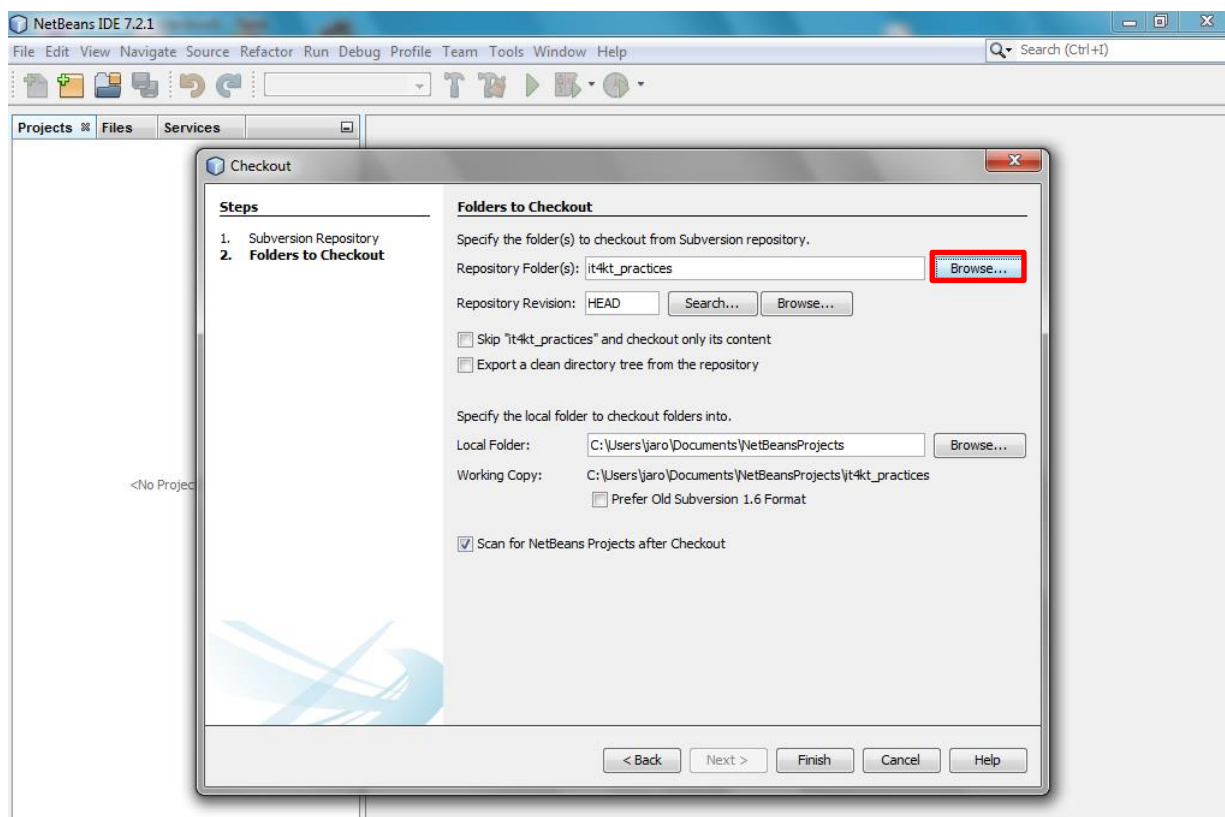
- Do poľa „**Repository URL**“ zadajte URL odkaz na repozitár projektu:

http://svn.it4kt.cnl.sk/cvicenia

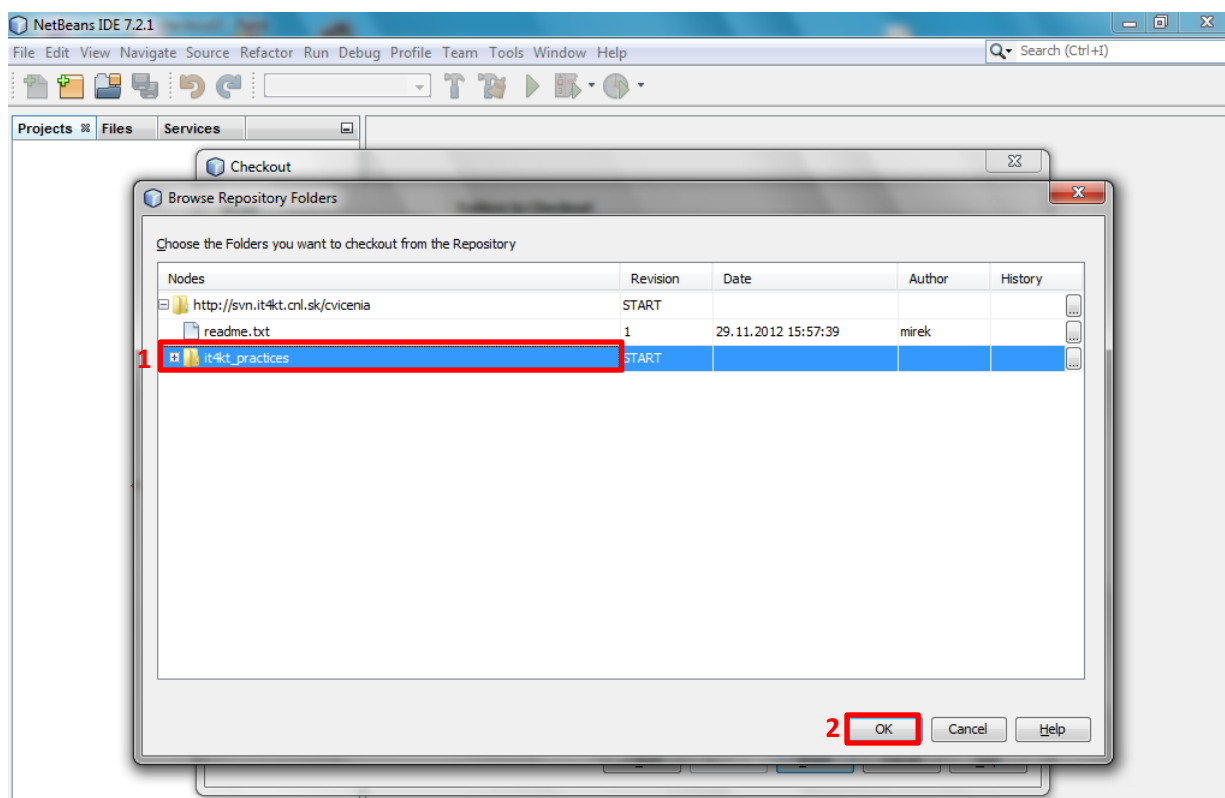
- do poľa „**User**“ zadajte login do projektu IT4KT: **it4kt**,
- do poľa „**Password**“ zadajte heslo do projektu IT4KT: **Pr0j5KT**,
- zaškrtnite políčko „**Save Username and Password**“,
- kliknite na tlačidlo „**Next**“.



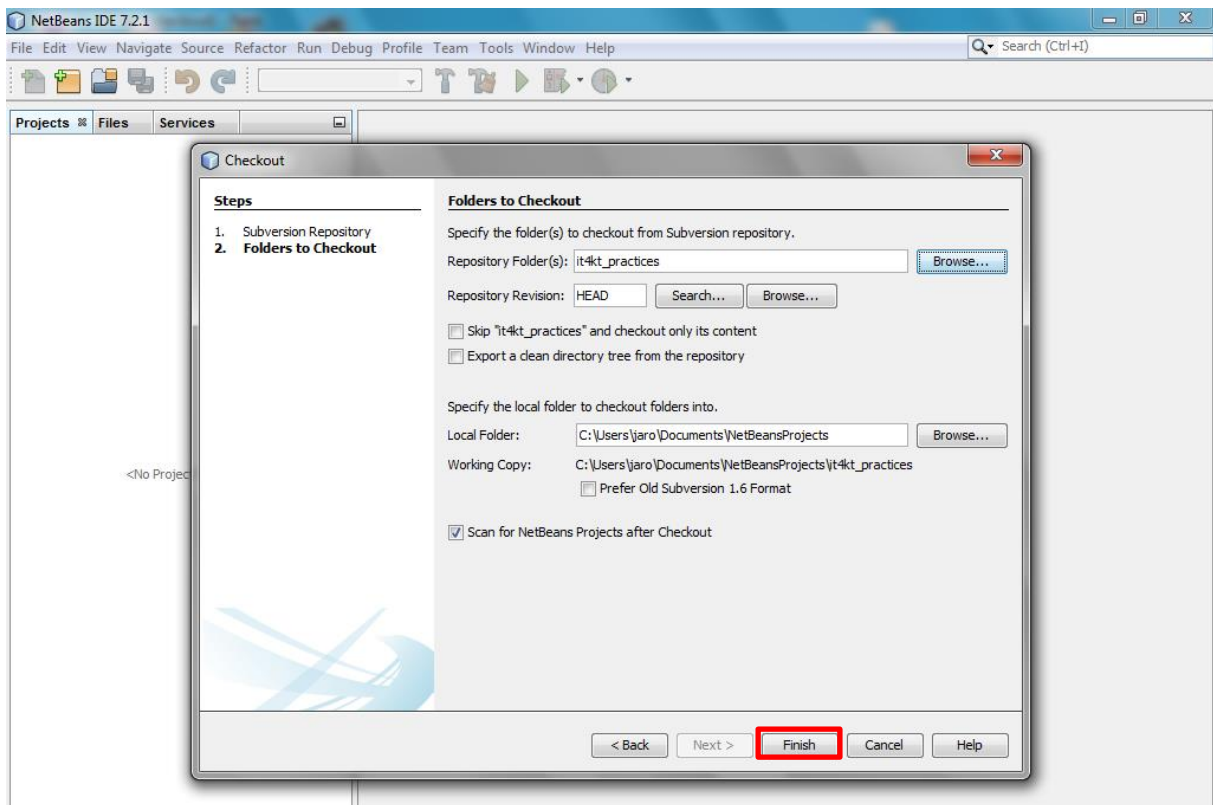
V časti **“Folders to Checkout”** vyberte adresár repozitára, ktorý sa má stiahnuť prostredníctvom tlačidla **“Browse...”** v časti **“Repository Folder(s)”**.



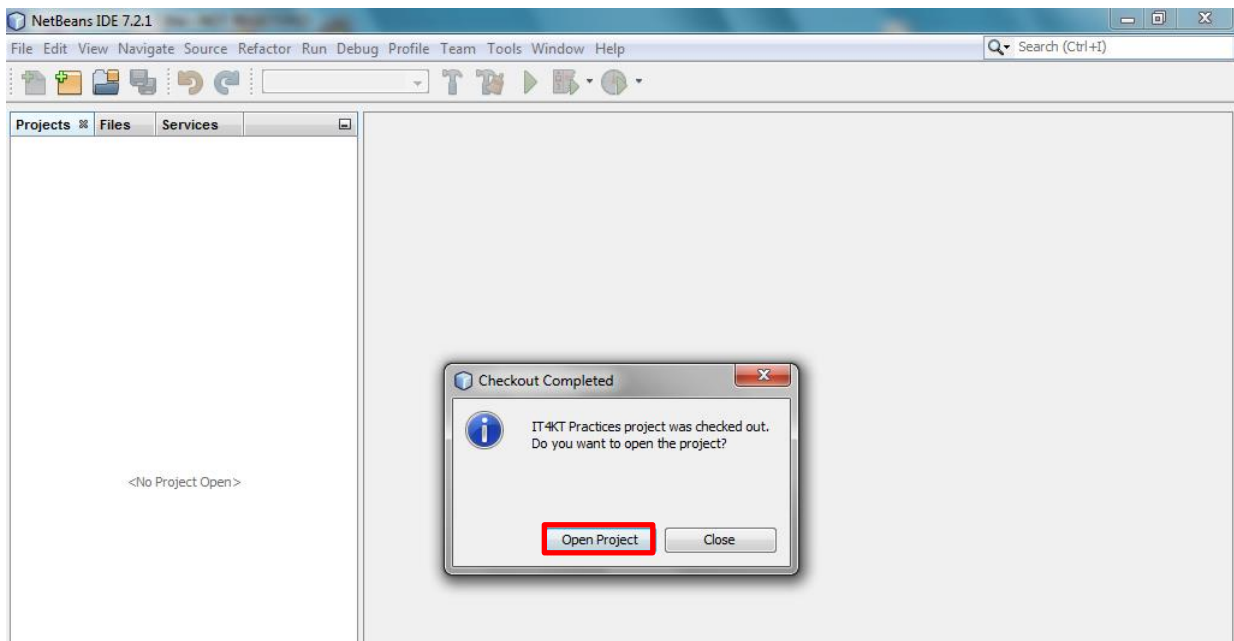
Vyberte adresár **“it4kt_practices”** a stlačte **“OK”**.



Stlačte tlačidlo „Finish“.

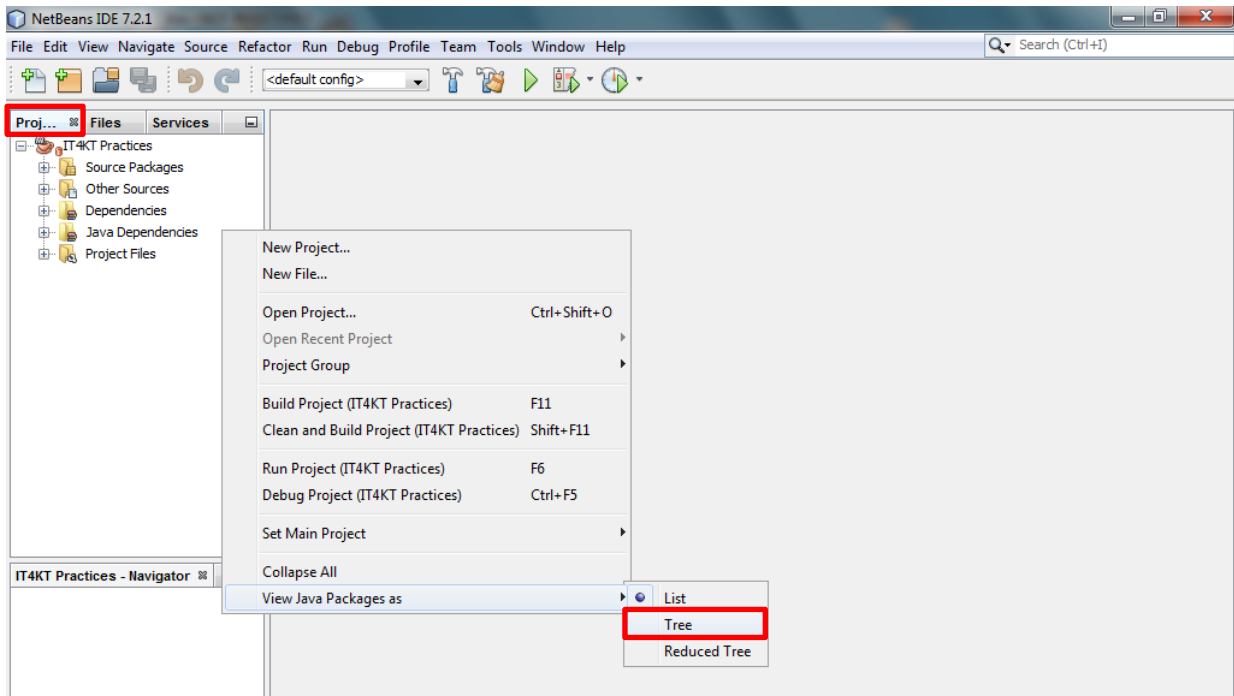


NetBeans následne stiahne projekt z repozitára SVN. Po stiahnutí sa objaví dialóg pre otvorenie projektu. Stlačte tlačidlo “Open Project”. Po kliknutí uvidíte v časti “Projects” projekt **IT4KT Practices**.



2.1. Nastavenie zobrazenia adresárov v editore

Pre správne zobrazenie adresárovej štruktúry je ešte potrebné v editore zmeniť zobrazenie adresárov. Klikne pravým tlačidlom na voľné (biele) miesto v časti „**Projects**“ a v kontextovom menu vyberte voľbu „**View Java Packages as → Tree**“.

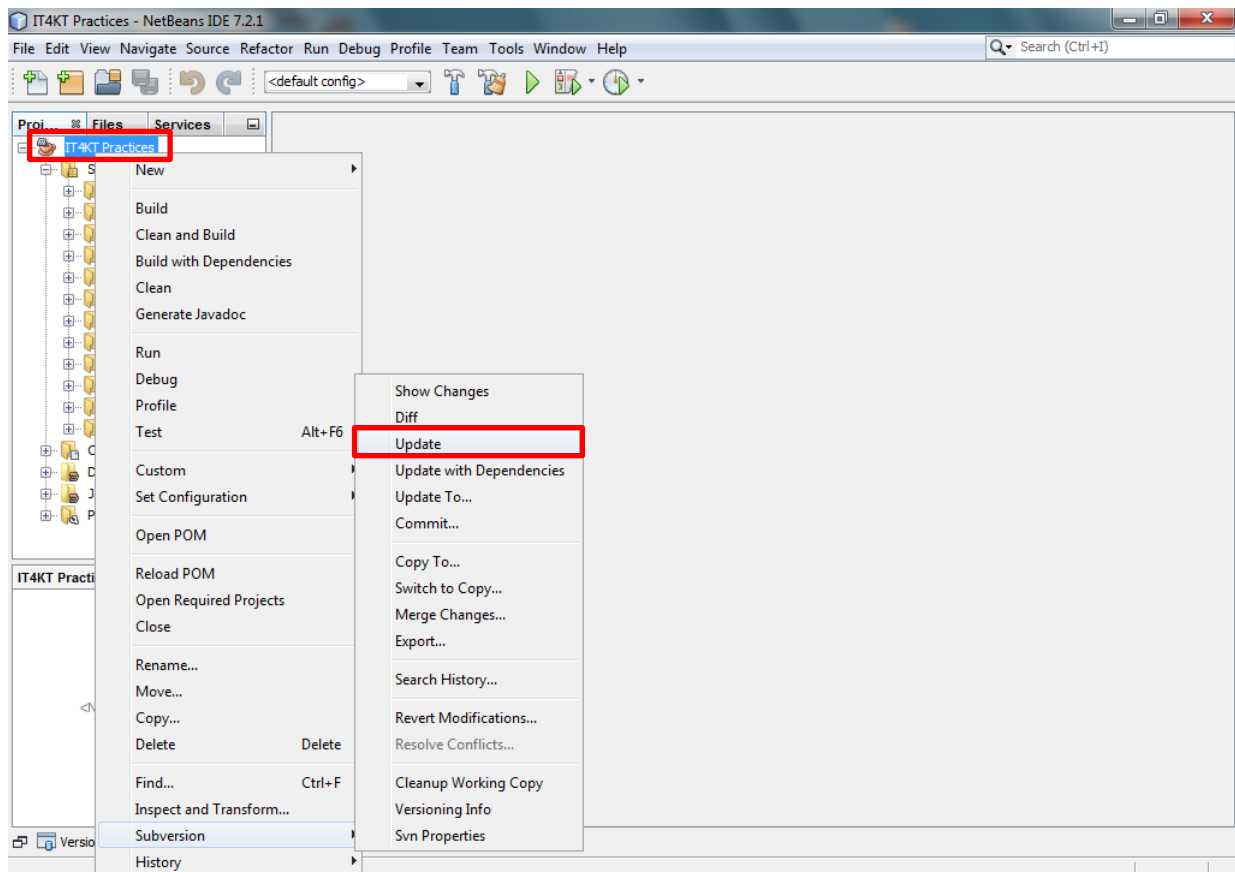


2.2. Aktualizácia projektu – vždy pred prácou!

DOLEŽITÉ: Keďže v tomto projekte bude pracovať viacero ľudí, je nutné pred každou prácou na projekte **aktualizovať projekt!**

To je možné nasledovne. Kliknite pravým tlačidlom myši na **koreňový adresár** projektu **IT4KT Practices**. V kontextovom menu vyberte „**Subversion -> Update**“.

Ak by po alebo počas procesu aktualizácie projektu nastali akékoľvek **problémy**, **ihneď kontaktujte správcov projektu!**

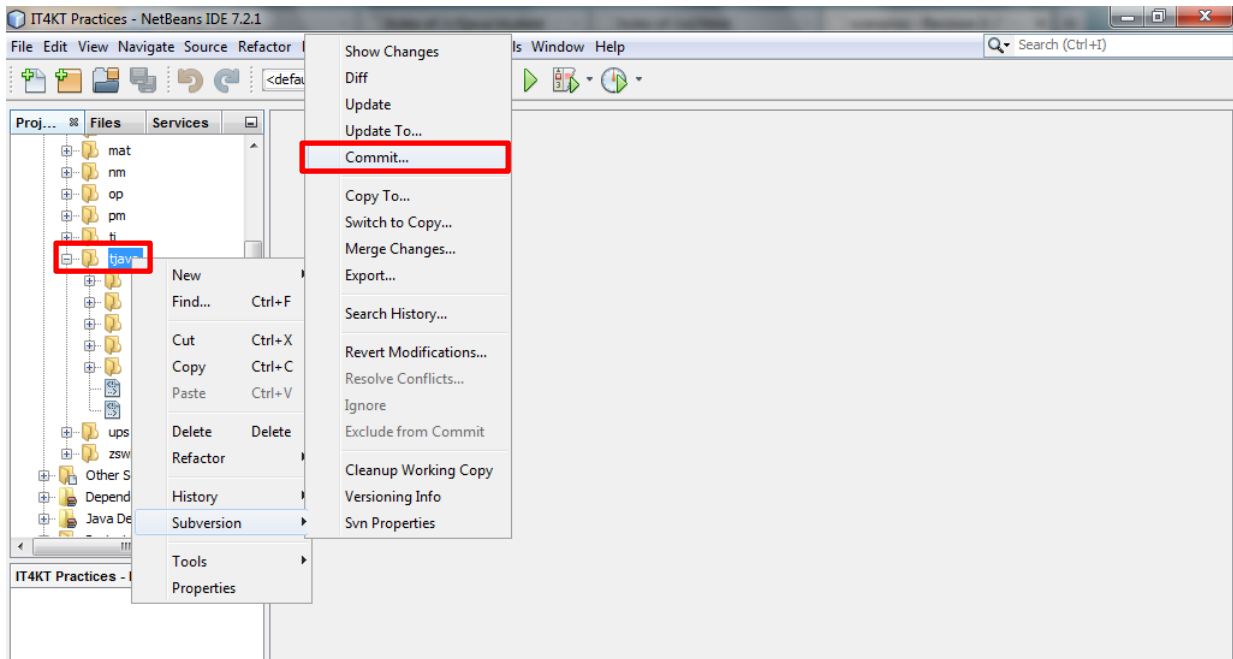


2.3. Odoslanie materiálov predmetu na SVN

Aby ste mohli **publikovať** svoj materiály pre svoj predmet, musíte ho **odoslať** na SVN repozitár.

To je možné nasledovne. Kliknite pravým tlačidlom myši na **adresár vášho predmetu**, v ktorom sa nachádzajú materiály. V kontextovom menu vyberte „**Subversion -> Commit**”.

DOLEŽITÉ: Keďže v tomto projekte bude pracovať viacero ľudí, je nutné aby ste **odosielali IBA adresár SVOJHO predmetu**, inak môžu nastať na serveri konflikty a môžete narušiť štruktúru svojho alebo cudzieho predmetu!



3. Štruktúra projektu a práca s IDE

3.1. Štruktúra projektu *IT4KT Practices*

Projekt sa skladá z dvoch hlavných adresárov:

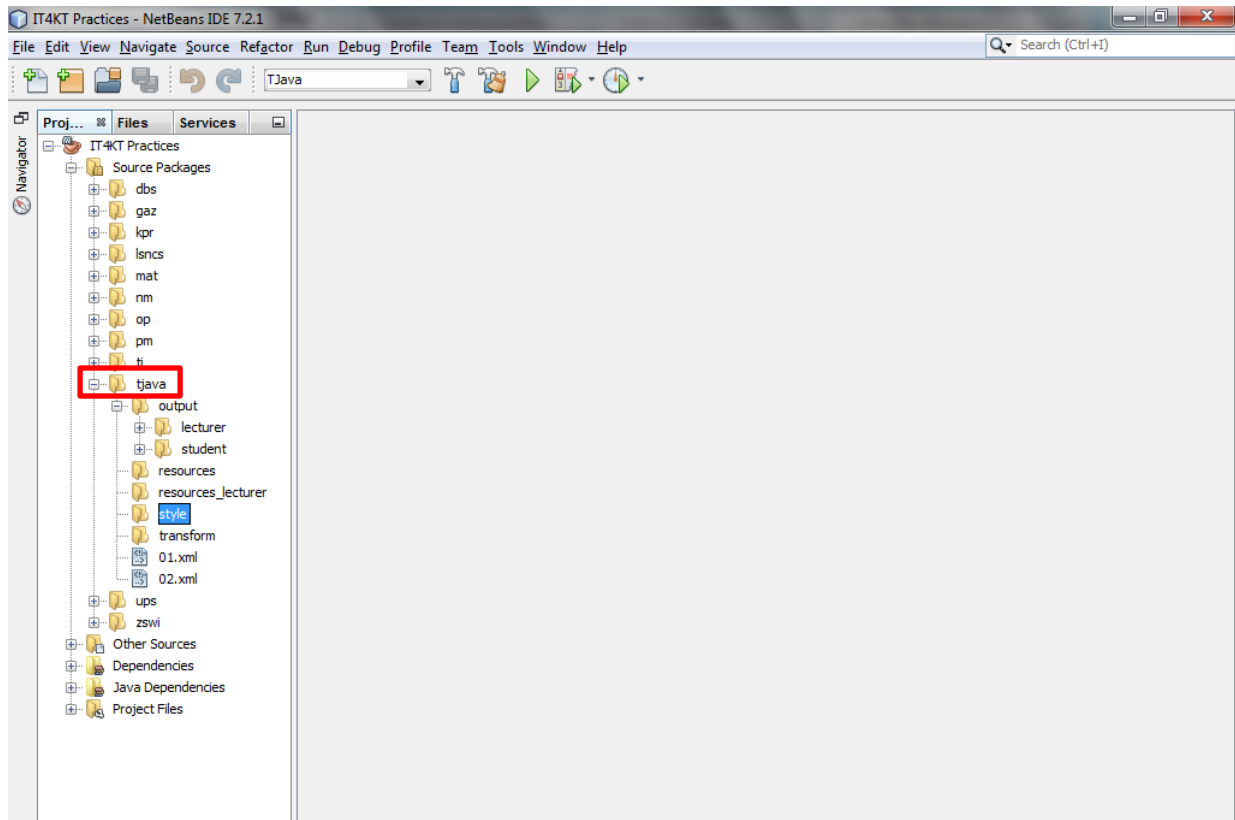
- „**Source Packages**“ – obsahuje adresáre pre predmety, v ktorých je možné vytvárať XML súbory pre cvičenia, obrázkov loga predmetu, pridávať zdroje pre cvičenia (dokumenty, obrázky, a pod.), či definovať štýly pre cvičenia. Pre každý predmet je vytvorený jeden adresár. Vzor vytvorených materiálov môžete vidieť v adresári predmetu **Technológie Java** (adresár **tjava**).
- „**Other Sources**“ – obsahuje všetko potrebné pre správne vygenerovanie Vašich HTML stránok.
- Ostatné adresáre sú konfiguračnými adresármi projektu.

Pre začatie práce s projektom si vyberte adresár vášho predmetu v zozname adresárov a rozbaľte ho pomocou „+“.

Ak nenájdete svoj predmet v zozname medzi adresármi, **kontaktujte administrátora projektu, nevytvárajte nový adresár sami.**

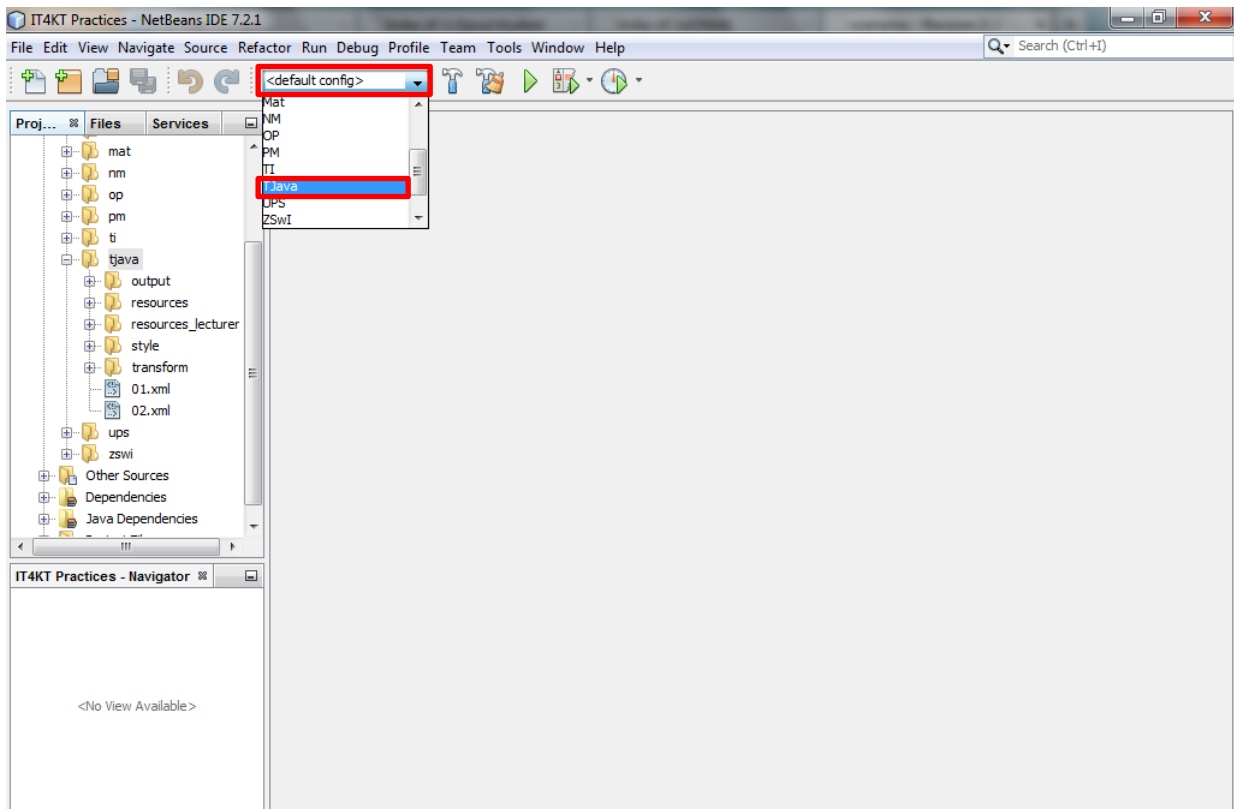
Vytváranie adresárov funguje klasicky ako v akomkoľvek prehliadači súborov: pravé kliknutie myšou na adresár, v kontextovom menu vybrať „**New -> Folder**“, zadať názov adresára a stlačiť klávesu **Enter**.

POZOR! Nové adresáre vytvárajte len v adresári svojho predmetu! Vytvorením adresárov alebo pridávaním súborov môžete narušiť štruktúru cudzieho predmetu!

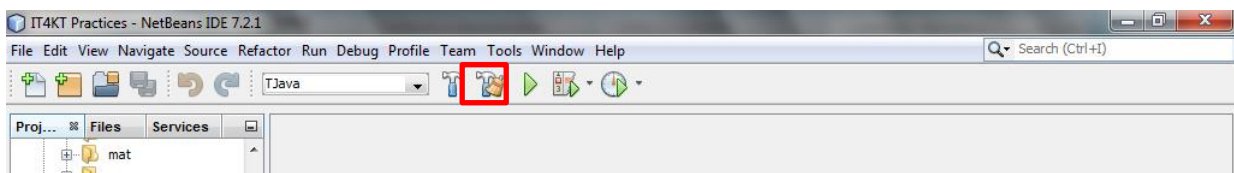


3.2. Výber profilu predmetu, generovanie materiálov a spustenie

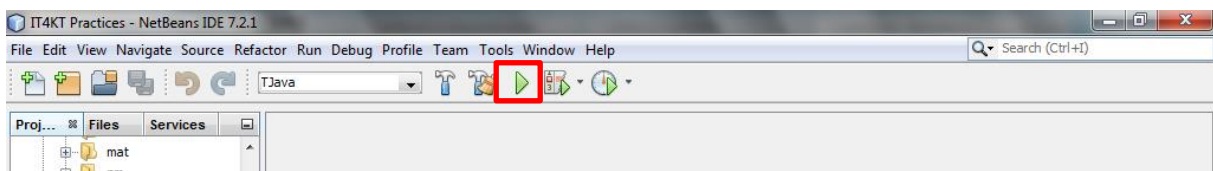
Pre správne generovanie materiálov pre predmet je potrebné **vybrať si zo zoznamu profilov** daný predmet. Je možné vybrať aj cudzí predmet za účelom inšpirácie.



Po každej zmene v projekte je potrebné projekt regenerovať stlačením tlačidla „**Clean and Build**“ v hornej lište, aby sa prejavili zmeny v materiáloch.



Spustenie projektu je možné vykonať pomocou tlačidla „**Run**“ v hornej lište.

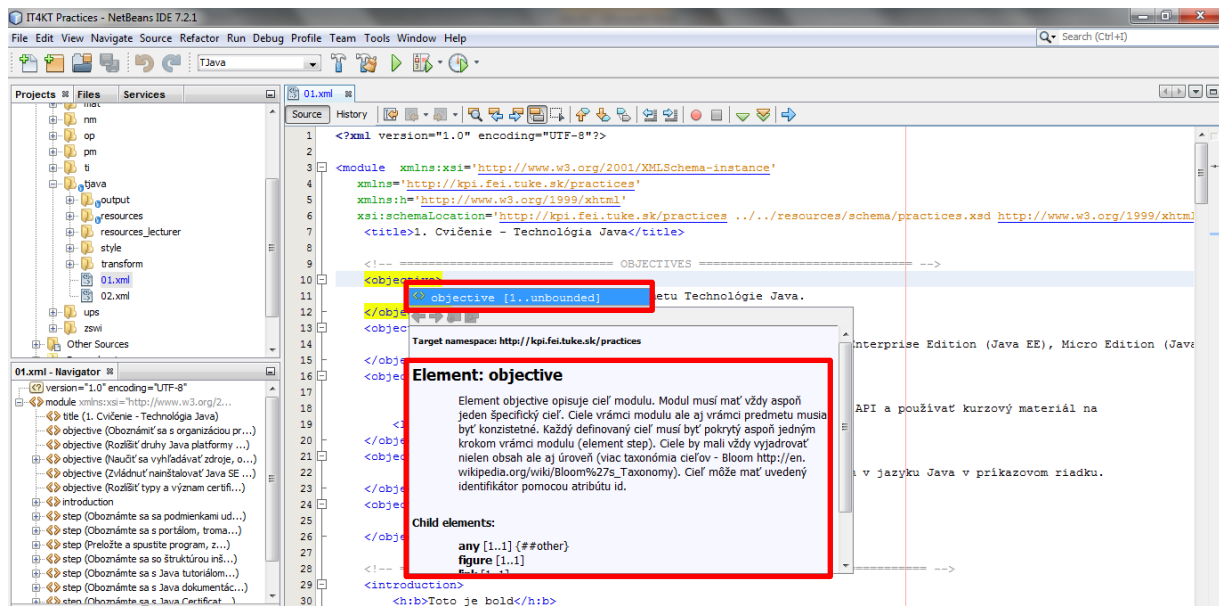


3.3. Tvorba materiálov v editore

Tvorba materiálov prebieha prostredníctvom vytvárania XML súborov. Každý XML súbor reprezentuje jeden modul (jedna HTML stránka). Názvy XML súborov je potrebné zadávať číselne v tvare: „01.xml“, „02.xml“, ... , „13.xml“.

Pre začiatok je vhodné skopírovať pripravené xml súbory „01.xml“ a „02.xml“ z už existujúcej šablóny materiálov v predmete **Technológie Java** (adresár **tjava**) do vášho predmetu.

Editor NetBeans ponúka možnosť automatického dopĺňania elementov pomocou klávesovej skratky **Alt+Space** kdekoľvek v texte. Objaví sa výber pre automatické dopĺňanie a dokumentácia k práve označenému elementu. Je možné vybrať si zo zoznamu pre automatické dopĺňanie a po stlačení klávesy **Enter** editor automaticky doplní potrebné značky.



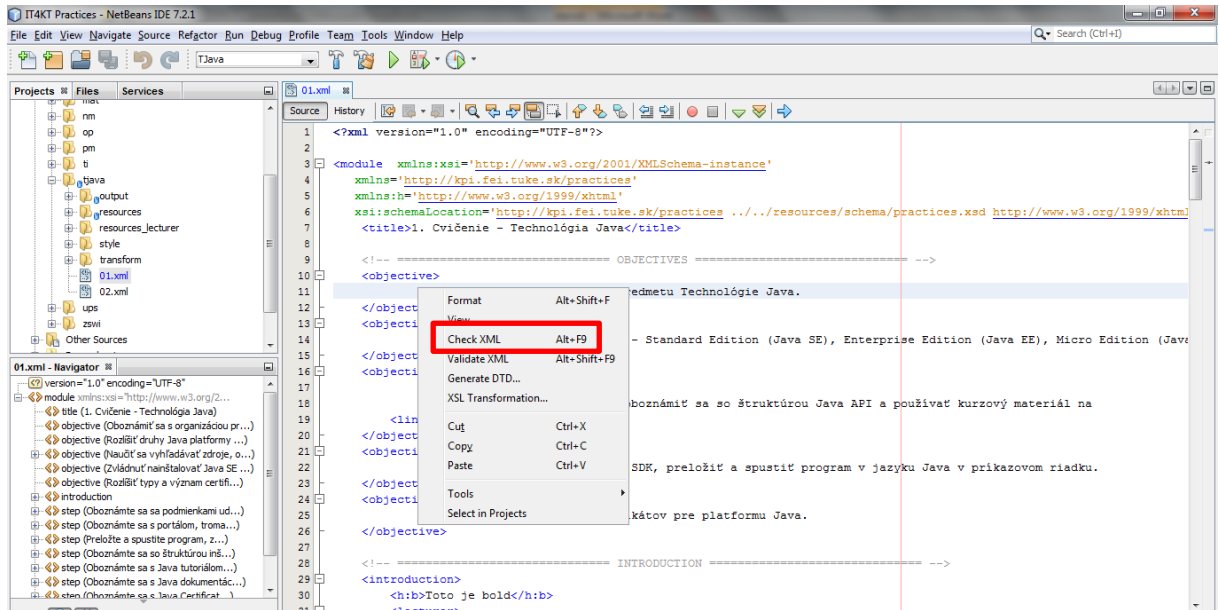
V XML majú každý element (značky v zátvorkách <>) počiatočnú a koncovú značku. Počiatočná značka sa píše v tvare <znacka> a koncová v tvare </znacka>. Atribúty značiek sa píše v úvodzovkách vo vnútri počiatočnej značky <znacka atribut="hodnota atributu">.

Do tela elementov (medzi počiatočnú a koncovú značku) je možné vložiť ďalší obsah – ďalší element alebo text.

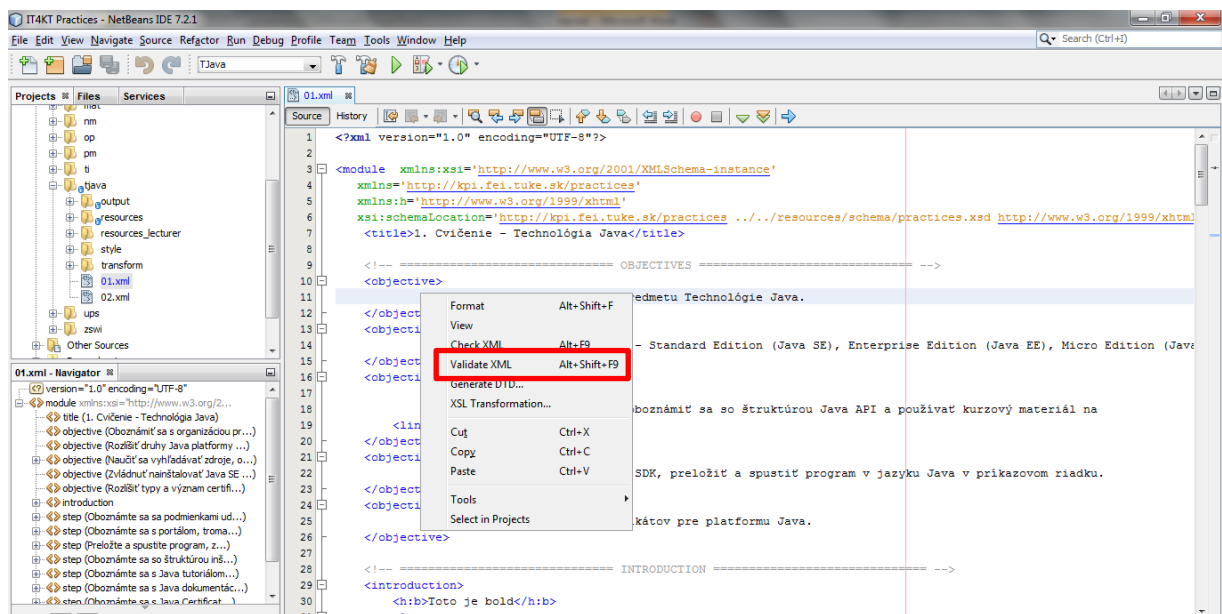
Elementy, ktoré sú z jazyka HTML je možné do XML súboru vkladať pomocou prefixu **h**; napríklad: <h:br></h:br>.

Pre bližšie oboznámenie sa s jazykom XML je k dispozícii zdroj: [Základy jazyka XML](#).

Po úprave xml súboru je možné automaticky **skontrolovať správnosť** pomocou kliknutia pravého tlačidla myši v texte a voľbou „**Check XML**“ v kontextovom menu.



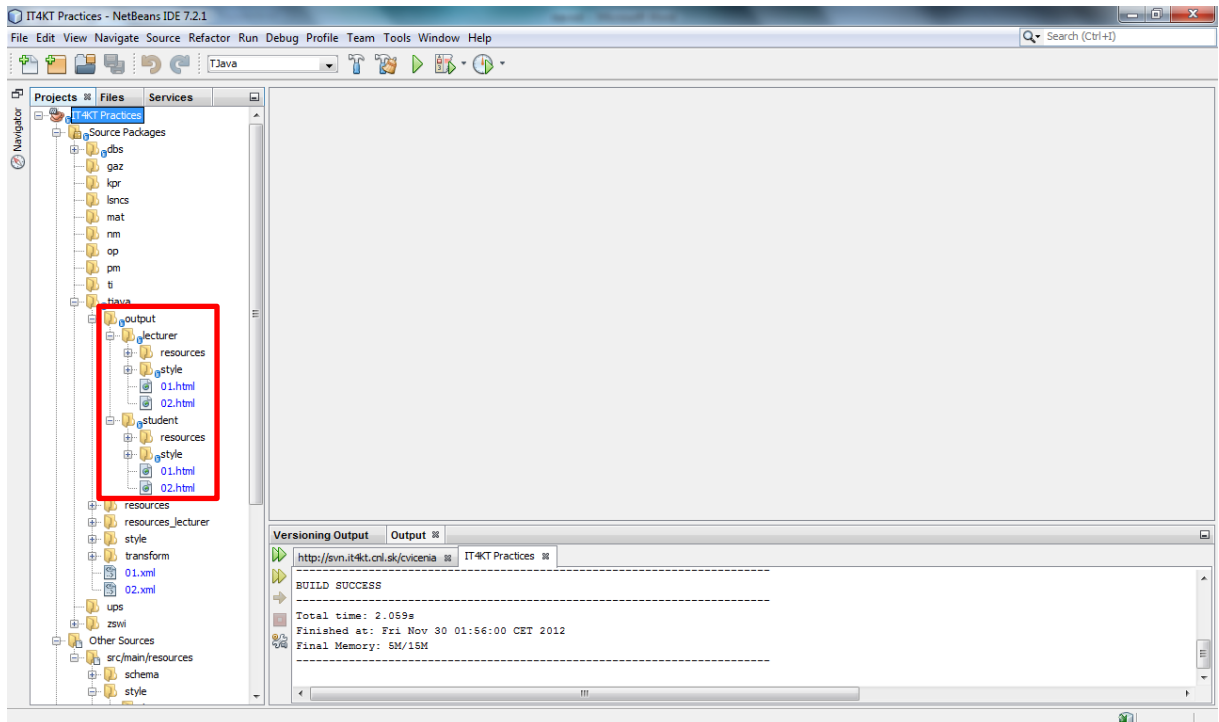
Pre kontrolu **výskytov nesprávnych elementov** v module je potrebné kliknúť pravým tlačidlom myši v texte a v kontextovom menu zvoliť „**Validate XML**“.



Po spustení projektu výsledok vygenerovania nájdete v adresári „**output**“ v adresári vášho predmetu. Obsahuje dva hlavné adresáre:

- „**student**“ – materiály pre študentov
- „**lecturer**“ – materiály pre učiteľa

V oboch adresároch sa nachádzajú HTML súbory webových stránok modulov a zdrojové súbory pre tieto stránky. Po spustení projektu (“**Run**”) sa v prehliadači automaticky otvorí úvodná stránka pre študentov (“**output/student/01.html**”).

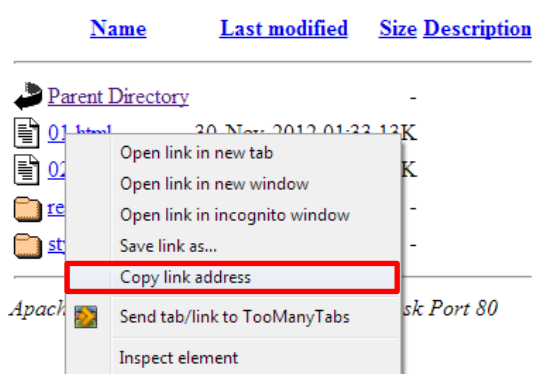


4. Publikovanie vytvorených materiálov v systéme moodle

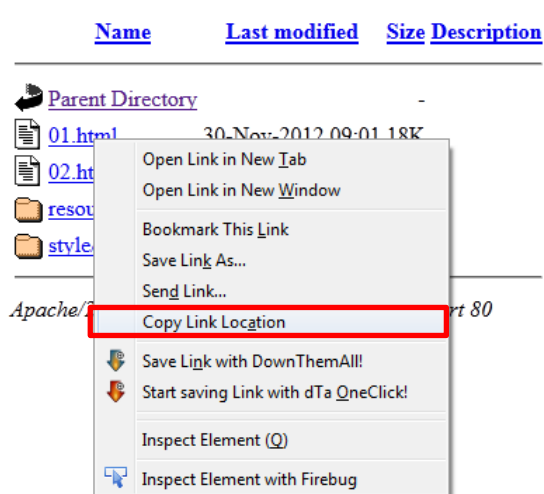
Vygenerované materiály, ktoré odošlete na SVN pomocou príkazu „Commit“ sú umiestnené na <http://it4kt.cnl.sk/c>. Rovnako ako v projekte, aj tu existuje verzia cvičení pre študenta (student) a pre učiteľa (lecturer). Verzia študent v adresári predmetu nachádza v adresári **student** a verzia učiteľ v adresári **lecturer**. Aby ste mohli publikovať cvičenie, nájdite html súbor daného cvičenia v súborovom systéme v adresári **študent**, kliknite naň pravým tlačidlom myši a v kontextovom menu vyberte voľbu:

- „Copy link address“ v prehliadači Google Chrome a Opera
- „Copy link location“ v prehliadači Mozilla Firefox
- „Copy shortcut“ v prehliadači Internet Explorer

Index of /c/tjava/student



Index of /c/tjava/student



Obr. 1: Príklad skopírovania odkazu cvičenia 1 v prehliadačoch Google Chrome (naľavo) a Mozilla Firefox (napravo)

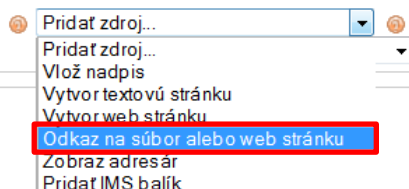
Pre publikovanie materiálov v systéme **moodle** sa prihláste (musíte mať prístup k svojmu predmetu v roli učiteľa a zapnutý mód úprav) a v danom predmete zvolte pri zodpovedajúcom cvičení voľbu „Pridať zdroj... -> Odkaz na súbor alebo web stránku“.

13 február - 19 február



1. Cvičenie
Java Overview

20 február - 26 február



Zobrazí sa dialóg pre pridanie internetového odkazu.

Zadajte názov cvičenia do políčka „**Meno**“, ktorý bude zobrazený ako odkaz v moodle a do políčka „**Poloha**“ vložte pomocou klávesovej skratky **Ctrl+V** skopírovaný odkaz na cvičenie.

☰ Pridávanie novej aktivity Zdroj do týždne 2 📌

Všeobecné nastavenia

Meno*

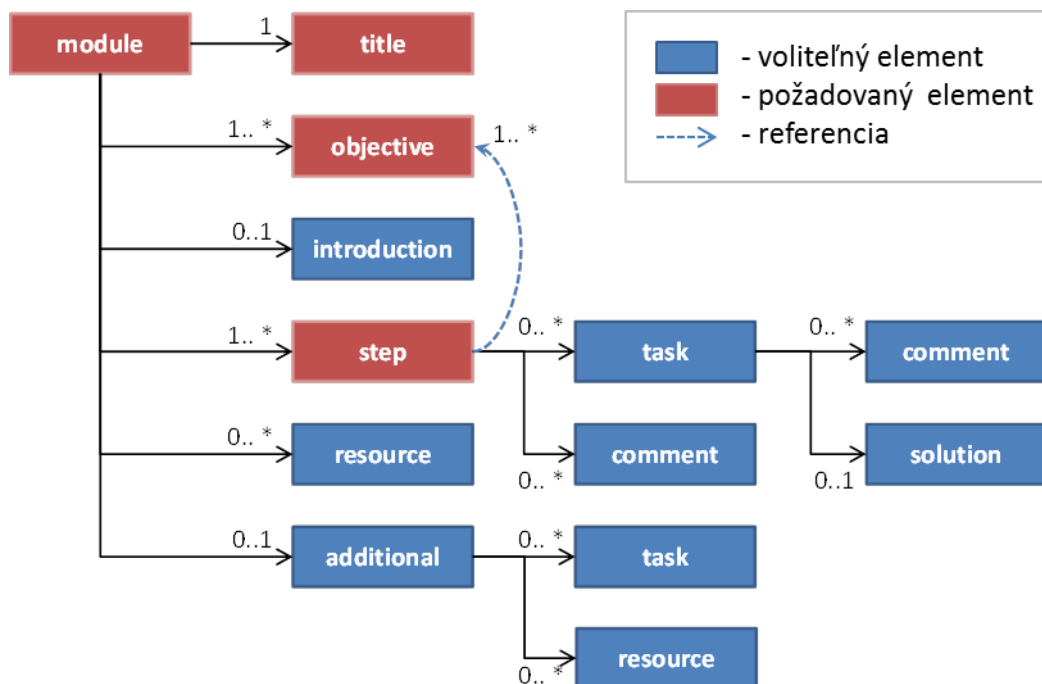
Zhrnutie 📌

Odkaz na súbor alebo web stránku

Poloha

Stlačte tlačidlo „**Uložiť a návrat do kurzu**“. Internetový odkaz s názvom, ktorý ste zadali by mal priradiť v zodpovedajúcom cvičení.

5. Štruktúra XML



Obr. 2: Štruktúra elementov v XML.

5.1. Element module

Element **module** je koreňovým elementom pre opis výučbovej jednotky (cvičenia).

Výučbová jednotka by mala predstavovať logicky súvisiaci celok. Pri prezenčnej forme štúdia je predpokladom, že táto jednotka nebude rozdelená na viac stretnutí. V študijnom predmete zvyčajne zodpovedá počet modulov počtu cvičení v rámci jedného predmetu (zvyčajne 13).

Každý modul má svoj jedinečný **názov** (element `title`), **zoznam cieľov** (element `objective`) a **krokov** (element `step`) pomocou ktorých sa tieto ciele napĺňajú.

5.2. Element title

Element **title** definuje názov modulu. Názov by mal byť stručný, no musí vystihovať obsahovú podstatu modulu.

5.3. Element objective

Element **objective** opisuje cieľ modulu.

Modul musí mať vždy aspoň jeden špecifický cieľ. Ciele v rámci modulu ale aj v rámci predmetu musia byť konzistentné.

Každý definovaný cieľ **musí byť pokrytý aspoň jedným krokom** v rámci modulu (element `step`).

Ciele by mali vždy vyjadrovať nielen obsah ale aj úroveň (viac o taxonómii cieľov: [Bloomova taxonómia](#)).

Cieľ môže mať uvedený **identifikátor** pomocou atribútu `id`.

5.4. Element introduction

Element **introduction** opisuje východiská pre konkrétny modul.

Môžu to byť základné pojmy a definície, kľúčové fakty, odkazy na výučbové materiály (prednášky, skriptá, učebnice) a pod.

Cieľom tejto časti je uviesť študenta do problematiky, vrátane vysvetlenia súvislosti tohto modulu s kontextom predmetu (prednášky, semináre, skúšanie).

5.5. Element step

Element **step** definuje množinu súvisiacich úloh, ktoré vedú k naplneniu aspoň jedného cieľa.

Pomocou atribútu `objective` je možné uviesť zoznam medzerou oddelených cieľov, ktoré budú splnené týmto krokom.

Uvedením zoznamu cieľov v atribúte `objective` je možné doceliť funkčnosť referencií na jednotlivé kroky. Po kliknutí na akýkoľvek cieľ v zozname cieľov predmetu sa vysvietia všetky kroky, ktoré je potrebné splniť na dosiahnutie tohto cieľa svetlo červenou farbou (viď obrázok).

1. **Oboznámiť sa s virtuálnym strojom Computron VM.**
2. Prakticky sa oboznámiť s fungovaním konečno-stavových akceptorov. kliknutý cieľ
3. Naučiť sa definovať regulárny výraz, ktorý definuje množinu postupností znakov akceptovateľných konečno-stavovým akceptorom.
4. Oboznámiť sa s teóriou prekladačov a prvou fázou prekladu: lexikálnou analýzou.
5. Zvládnuť upraviť kód lexikálneho analyzátora pre podporu vlastných symbolov.

Úvod

Na tomto cvičení sa naučíte pracovať s virtuálnym strojom Computron VM. Computron je program emulujúci počítač, ktorý sa bude používať ako cieľová architektúra pri preklade programov na ďalších cvičeniach. V druhej polovici cvičenia sa oboznámite s teóriou prekladačov a s prvou fázou prekladu - lexikálnou analýzou. Na základe získaných znalostí zvládnete upraviť kód lexikálneho analyzátora pre podporu vlastných symbolov.

Postup

1. Preštudujte si [návod](#) a [príručku](#) k virtuálnemu stroju **Computron VM**. vyznačené kroky, ktoré je potrebné splniť pre dosiahnutie kliknutého cieľa

Úloha: Spustíte virtuálny stroj **Computron**.

Poznámka: Pre spustenie použite príkaz:

```
java -jar Computron_VM.jar
```

Úloha: Načítajte do Computron VM predvytvorený príklad.

Úloha: Čo sa stane ak spustíte program pomocou tlačidla "Run"?

2. Spustíte príklad so správne vopred inicializovanými registrami:

Úloha: Zastavte beh príkladu pomocou tlačidla "Run" a inicializujte register Begin na hodnotu 010.

Poznámka: Nastavenie hodnôt registrov je možné vykonať v sekcii "ODFC PROGRAMMING DEVICE":

5.6. Element resource

Element **resource** slúži na opis zdroja, ktorý študenti používajú pri štúdiu modulu.

Zdrojom môže byť materiál na stiahnutie, kniha, internetový odkaz a podobne.

5.7. Element additional

Element **additional** opisuje časti, ktoré sú pripravené nad základný rámec modulu (cvičenia).

Môžu to byť **doplňujúce úlohy** (element `task`) alebo **doplňujúce zdroje** (element `resource`). Vďaka tomuto členeniu je možné pripraviť domáce úlohy, resp. úlohy pre aktívnych študentov.

Doplňujúce úlohy sú voliteľné, ale každý študent by mal splniť minimálne tie úlohy, ktoré nie sú uvedené v tejto doplňujúcej časti.

5.8. Element task

Element **task** opisuje práve jednu úlohu, ktorú majú študenti vyriešiť.

Úloha by mala byť formulovaná tak, aby bola pochopiteľná a vyriešiteľná samostatne bez širšieho kontextu.

Úloha by mala mať správne riešenie a malo by byť možné overiť, či študent úlohu vyriešil alebo nevyriešil.

5.9. Element comment

Element **comment** obsahuje doplňujúci komentár ku riešenému kroku alebo úlohe.

Komentáre by mali študentovi poskytovať dodatočné informácie k riešeniu problému, ktoré v prípade potreby študent môže použiť.

Komentár by nemal obsahovať podstatné poznámky, bez ktorých nie je možné úlohu správne vyriešiť.

Aj bez komentárov by mal ostať text modulu plnohodnotný, zvýšiť by sa mala akurát náročnosť nájdenia riešenia, keďže študenta na riešenie menej navádzame.

V budúcnosti predpokladáme dynamické zobrazovanie komentárov na základe úspešnosti/neúspešnosti riešenia.

5.10. Elementy solution a result

Element **solution** opisuje správne riešenie úlohy. Riešenie by malo byť vnorené v úlohe. Nemusí byť poskytnuté ku každej úlohe.

Pre element **result** platí to isté, čo pre element **solution**, jediný rozdiel je v sémantike. Sémantický rozdiel medzi **solution** a **result** je nasledovný:

- **solution** – riešenie je procedurálne, vyžaduje opis postupu riešenia problému. Opis bude „Riešenie:“.
- **result** – riešenie bude typu „výsledok“, používa sa pre matematické úlohy alebo úlohy, ktoré neobsahujú opis postupu riešenia problému. Opis bude „Výsledok:“

Poznámka: Úloha môže mať aj postup riešenia aj výsledok.

Tip: ak sa element <solution> alebo <result> vnorí do bloku <lecturer>, bude riešenie viditeľné len pre učiteľa.

Dobrovoľné atribúty elementov solution a result sú:

- **hidden** – prepínač pre skryté riešenie

Atribút hidden

Atribút **hidden** slúži ako prepínač grafického zobrazenia riešenia, kedy je možné nastaviť riešenie alebo výsledok ako **skrytý**. Je možné zadať jednu z nasledovných hodnôt (pričom predvolená hodnota je *false*):

- *false* – riešenie bude zobrazené vždy
- *true* – riešenie bude skryté v prípade prvého zobrazenia stránky a zobrazí sa po stlačení tlačidla „Zobraziť riešenie“ (resp. „Zobraziť výsledok“ pri type výsledok).

Príklad pridania riešenia alebo výsledku úlohy:

```

<task>
  Zmeňte farbu písma tohto riadka.
  <solution>
    Kliknite na tlačidlo „Farba písma“.
  </solution>
</task>

```

```

<task>
  3+4=?
  <result hidden="true">
    7
  </result>
</task>

```

Výsledok:

Riešená úloha: Zmeňte farbu písma tohto riadka.
Riešenie: Kliknite na tlačidlo "Farba písma".

Riešená úloha: 3+4=? [Zobraziť výsledok](#)

Riešená úloha: 3+4=?
Výsledok: 7 [Zobraziť výsledok](#)

Obr.: Naľavo príklad klasickej úlohy a úlohy typu výsledok s možnosťou zobrazenia/skrytia, napravo úloha po kliknutí na tlačidlo „Zobraziť výsledok“. Po opätovnom kliknutí na tlačidlo „Zobraziť výsledok“ bude výsledok schovaný.

Použitie postupu riešenia aj výsledku v jednej úlohe:

```

<task>
  Vyrátajte sústavu rovníc:
  3x + 2y = 15
  10x - 15y = -5
  <solution>
    ... postup riešenia ..
  </solution>
  <result hidden="true">
    ... výsledok ...
  </result>
</task>

```

5.11. Ďalšie elementy

Element figure

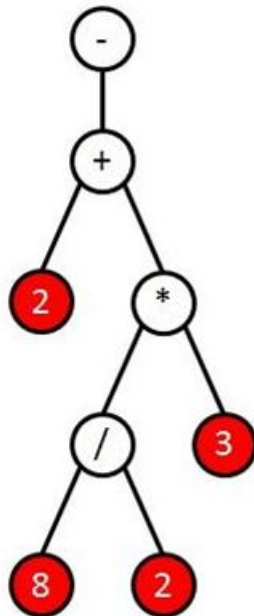
Element **figure** zaobaluje centrovanej obrázok s názvom.

Element môže obsahovať buď preformátovaný text (html pre), respektíve element **image**. Môže obsahovať textový opis obrázku, ktorý sa vkladá pomocou značky **caption**.

Príklad pridania obrázku:

```
<figure>
  <image>ast_uzly.png</image>
  <caption>
    Abstraktný syntaktický strom s označením listových
    uzlov.
  </caption>
</figure>
```

Výsledok:



Obr.: Abstraktný syntaktický strom s označením listových uzlov

Na obrázok je možné nastaviť veľkosť obrázka pomocou atribútu **scale**. Príklad:

```
<image scale="50%">prechodovy-diagram.png</image>
```

Element image

Element **image** slúži na vloženie obrázku do modulu. Obrázok bude zobrazený priamo na mieste vloženia.

Odporúčame vložiť element **image** do elementu **figure**, ktorý umožňuje pridať k obrázku aj textový popis.

Tento element je analógia elementu **img** z HTML.

Element table

Element **table** zaobaluje centrovajúcu tabuľku s popisom. Element môže obsahovať riadky (elementy **row**) a v nich sa môžu nachádzať stĺpce (**col**).

Tabuľka môže obsahovať aj hlavičkový riadok - element **header**.

Príklad pridania tabuľky:

```
<table>
  <caption>Toto je tabulka</caption>
  <header>
    <col>Meno</col>
    <col>Priezvisko</col>
    <col>Vek</col>
  </header>
  <row>
    <col>John</col>
    <col>Snow</col>
    <col>28</col>
  </row>
  <row>
    <col>Mike</col>
    <col>Shinoda</col>
    <col>30</col>
  </row>
  <row>
    <col>Peter</col>
    <col>Parker</col>
    <col>25</col>
  </row>
</table>
```

Výsledok:

Tab.: Toto je tabulka

Meno	Priezvisko	Vek
John	Snow	28
Mike	Shinoda	30
Peter	Parker	25

Element **table** slúži ako zjednodušená verzia tabuľky pre neprogramátorov. Pre vytvorenie tabuľky je ale samozrejme možné použiť aj klasické html elementy `<h:table>`, `<h:td>`, `<h:tr>`, `<h:th>` a pod (viď kapitola **Použitie HTML elementov**).

Element math

Element **math** slúži na zápis matematických vzorcov vo formáte Tex, Latex, MathML, AsciiMath.

Dobrovoľné atribúty elementu math sú:

- **display** – prepínač grafického zobrazenia vzorca
- **type** – prepínač formátu zápisu vzorcov

Atribút display

Atribút **display** slúži ako prepínač grafického zobrazenia vzorca. Je možné zadať jednu z nasledovných hodnôt (pričom predvolená hodnota je *inline*):

- *block* – zobrazenie vzorca v samostatnom boku.
- *inline* – zobrazenie vzorca priamo v texte.

Atribút type

Atribút **type** slúži ako prepínač formátu zápisu vzorcov. Je možné zadať jednu z nasledovných hodnôt (pričom predvolená hodnota je *Tex*):

- *Tex* – pre zápis vzorcov vo formáte Tex.
- *MathML* – pre zápis vzorcov vo formáte MathML.
- *AsciiMath* – pre zápis vzorcov vo formáte AsciiMath.

Príklad použitia elementu math:

```
<step>
```

```
  Toto: <math>x^2</math> je vzorec v texte.
```

```
  <task>
```

```
    Toto: <math display="block">x = \frac{-b \pm \sqrt{b^2-4ac}}{2a}</math> je vzorec v bloku.
```

```
  </task>
```

```
</step>
```

Vzorový príklad je možné vidieť v prvom cvičení predmetu NM.

Element code

Element **code** slúži na zápis zdrojového kódu v rôznych programovacích jazykoch. Element umožňuje farebné zvýrazňovanie niektorých prvkov kódu.

Dobrovoľné atribúty elementu code sú:

- **display** – prepínač zobrazenia kódu,
- **lang** – označenie programovacieho jazyka pre zvýrazňovanie syntaxe.

Atribút display

Atribút **display** slúži ako prepínač zobrazenia kódu v samostatnom bloku alebo v texte. Je možné zadať jednu z nasledujúcich hodnôt (predvolená hodnota je *inline*):

- *inline* – zobrazenie kódu priamo v texte (použije sa neproporcionálne písmo),
- *block* – zobrazenie kódu v samostatnom boku (použije sa neproporcionálne písmo a zachovajú sa medzery a rozdeľovače riadkov).

Atribút lang

Atribút **lang** slúži na označenie programovacieho jazyka pre zvýrazňovanie syntaxe. Je to nepovinný atribút, ak sa nepoužije, nebude použité žiadne zvýrazňovanie syntaxe v texte. Je možné ho použiť iba pri zobrazení do bloku. Podporované hodnoty atribútu sú: *bash*, *cpp* (pre C++), *cs* (pre C#), *css*, *diff*, *http*, *ini*, *java*, *javascript*, *json*, *haskell*, *matlab*, *perl*, *php*, *python*, *ruby*, *sql*, *xml* (tiež pre HTML).

Na zvýrazňovanie sa používa program highlight.js: <http://highlightjs.org/>.

5.12. Použitie HTML elementov

Pridávanie akýchkoľvek **HTML elementov** je možné použitím prefixu **h:** pred značkou elementu.
Príklady:

Príklad (formátovanie):

```
<h:b>Syntaxou riadený interpretátor</h:b> je programový nástroj, ktorý  
vyhodnocuje <h:i>význam (sémantiku)</h:i> postupnosti príkazov, ktoré  
<h:u>dostane na vstupe</h:u>.
```

Výsledok:

Syntaxou riadený interpretátor je programový nástroj, ktorý vyhodnocuje *význam (sémantiku)* postupnosti príkazov, ktoré dostane na vstupe.

Príklad (odrážkový zoznam):

Oboznámte sa s pokynmi pre prípravu na test a spracovanie zadání:

```
<h:ul>  
  <h:li>  
    <link href="04.html">Test A</link>  
  </h:li>  
  <h:li>  
    Zadanie B (<link href="07.html">pokyny</link>)  
  </h:li>  
  <h:li>  
    Zadanie C (<link href="11.html">pokyny</link>)  
  </h:li>  
</h:ul>
```

Výsledok:

Oboznámte sa s pokynmi pre prípravu na test a spracovanie zadání:

- ◊ [Test A](#)
- ◊ Zadanie B ([pokyny](#))
- ◊ Zadanie C ([pokyny](#))

Podobným spôsobom je možné pridávať akýkoľvek HTML element. Je potrebné ale dávať pozor, aby **každý element mal koncovú značku**.

Viac o HTML na: <http://videotutorialy.sk/category/html/> a <http://www.w3schools.com/html/default.asp>.

6. Pokročilá práca s projektom

Projekt generátora materiálov má **preddefinovaný formát HTML stránky**, ktorý môžete používať. Ak ale chcete tento formát **rozsíriť alebo zmeniť**, je to možné urobiť v adresári vášho predmetu pridaním CSS štýlov, XSLT šablón alebo definovaním nových XML elementov prostredníctvom XSD schémy.

6.1. Zmena formy

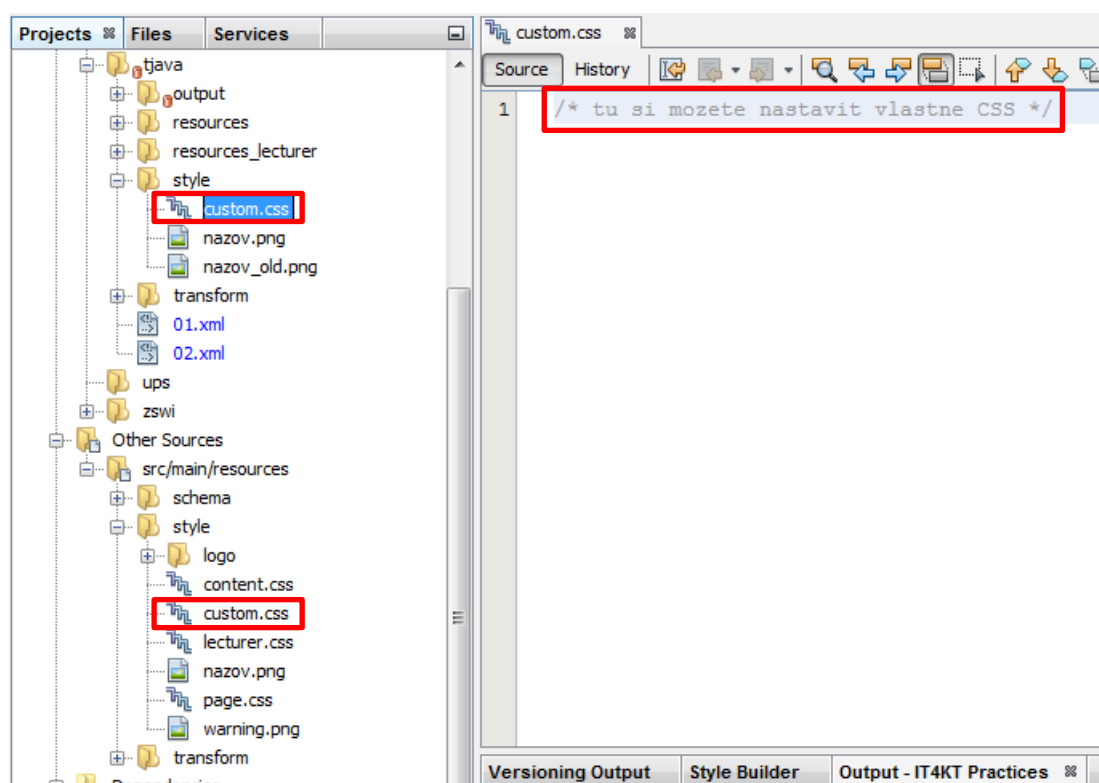
Forma je v projekte definovaná pomocou **CSS štýlov** a pomocou **XSLT transformačnej šablóny**.

Zmena CSS štýlov

V adresári „**Other sources/src/main/resources/style**“ sa nachádzajú predvolené CSS štýly a logá.

Pre úpravu existujúcich štýlov skopírujte súbor „**custom.css**“ do adresára „**style**“ v adresári vášho predmetu. Do tohto CSS súboru môžete nastaviť vlastné CSS štýly.

Štýly vytvorené v tomto CSS súbore prepíšu predvolené štýly.



Zmena XSLT šablón

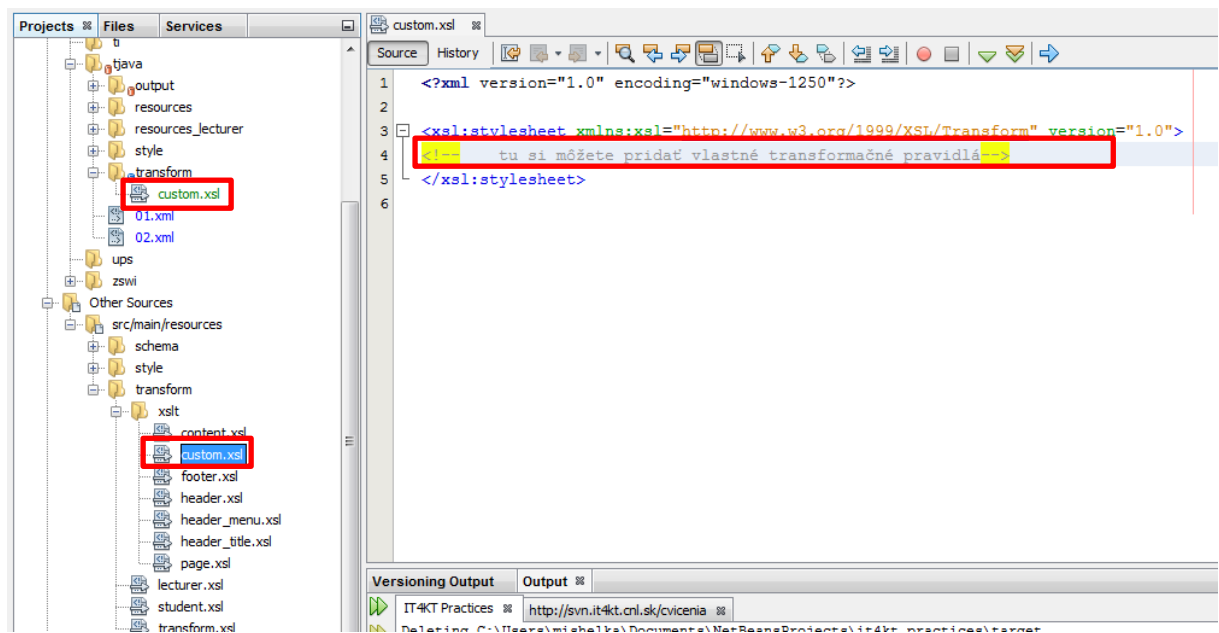
Štruktúru stránky (transformačné pravidlá) je možné meniť pomocou **XSLT šablóny**.

Zmena CSS štýlov

V adresári „**Other sources/src/main/resources/transform/**“ sa nachádzajú predvolené CSS štýly a logá.

Pre úpravu existujúcich štýlov skopírujte súbor „**custom.xsl**“ do adresára „**transform**“ v adresári vášho predmetu. Do tohto XSL súboru môžete vkladať vlastné transformačné pravidlá.

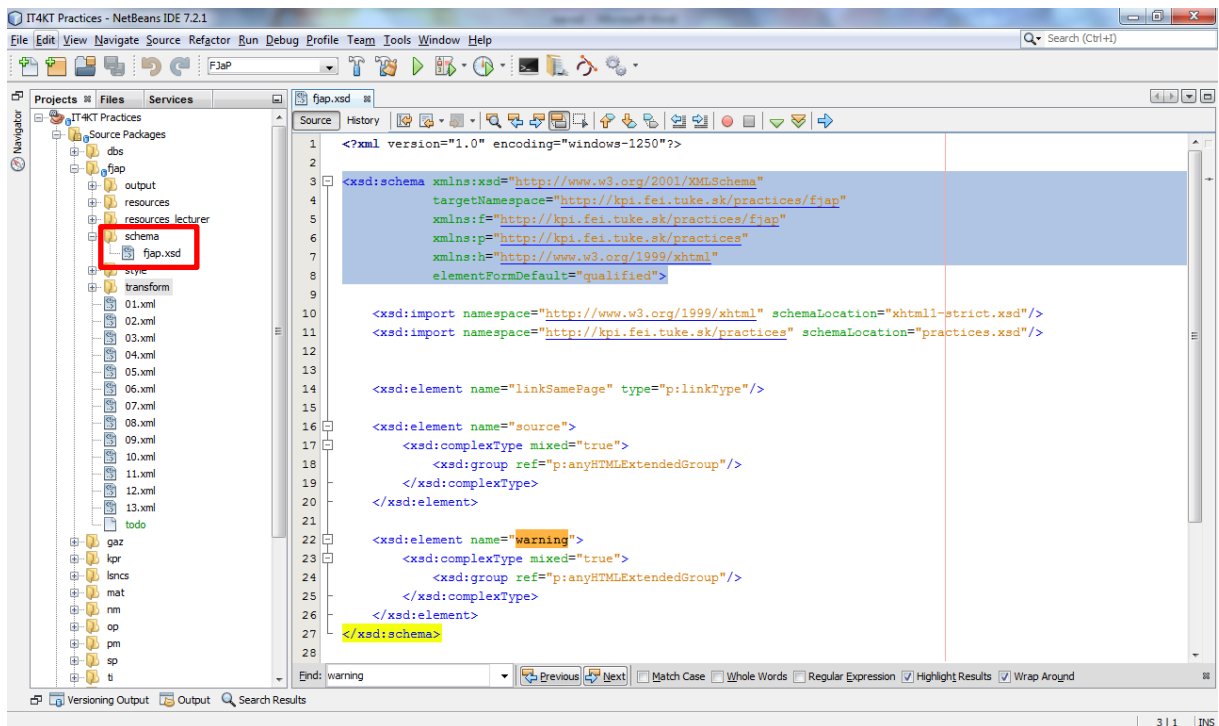
Transformačné pravidlá vytvorené v tomto XSL súbore prepíšu predvolené transformačné pravidlá.



6.2. Zmena štrukt ry

Prid vanie nových XML elementov je mo n  vytvoren m nových XSD sch m.

XSD sch my je mo n  vytv rať v adres ri **schema** v adres ri v šho predmetu.



V novej sch me je potrebn  vytvoriť vlastn  priestor mien a nastaviť ho ako cieľov  pre t tu sch mu, napr.:

```
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
```

```

targetNamespace="http://kpi.fei.tuke.sk/practices/fjap"
xmlns:f="http://kpi.fei.tuke.sk/practices/fjap"
xmlns:p="http://kpi.fei.tuke.sk/practices"
xmlns:h="http://www.w3.org/1999/xhtml"
elementFormDefault="qualified">

```

Tento priestor mien budete používať ako prefix pre svoje nové elementy v xml súboroch. Ak chcete použiť existujúce XSD typy a elementy, je potrebné importovať ich pomocou elementu import na začiatku vášho xsd súboru hneď po deklarácii priestorov mien:

```

<xsd:import namespace=http://www.w3.org/1999/xhtml
              schemaLocation="xhtml1-strict.xsd"/>
<xsd:import namespace=http://kpi.fei.tuke.sk/practices
              schemaLocation="practices.xsd"/>

```

Potom je možné definovať nové elementy a typy, napr.:

```

<xsd:element name="warning">
  <xsd:complexType mixed="true">
    <xsd:group ref="p:anyHTMLExtendedGroup"/>
  </xsd:complexType>
</xsd:element>

```

V xml súboroch je takýto element možné potom použiť v tvare:

```

<step objectives="id_ciel3">
  <f:warning>
    Odložte si príklad pre ďalšie cvičenia.
  </f:warning>
</step>

```

Je však nutné do xml súboru vložiť na do hlavičky váš vytvorený priestor mien:

```

<module xmlns:xsi='http://www.w3.org/2001/XMLSchema-instance'
        xmlns='http://kpi.fei.tuke.sk/practices'
        xmlns:f='http://kpi.fei.tuke.sk/practices/fjap'
        xmlns:h='http://www.w3.org/1999/xhtml'
        xsi:schemaLocation='
http://kpi.fei.tuke.sk/practices ../../resources/schema/practices.xsd
http://www.w3.org/1999/xhtml ../../resources/schema/xhtml1-strict.xsd
http://kpi.fei.tuke.sk/practices/fjap schema/fjap.xsd'>

```

Príklad vytvorenia vlastnej xsd schémy, vlastných transformačných pravidiel (alebo rozšírenia/upravenia existujúcich) a vlastných štýlov je možné vidieť v adresári predmetu **fjap** v adresároch **schema**, **transform/xslt** a **style**.

6.3. Zaheslovanie prístupu k učiteľským materiálom

Kvoli jednoduchosti práce s generátorom materiálov a kontroly vygenerovaných materiálov, zaheslovanie prístupu k učiteľským materiálom odporúčame vykonať **až po dokončení materiálov** vtedy, keď už je potrebné poskytnúť cvičenia študentom. Zaheslovanie prístupu je možné tak, že do adresára **output/lecturer** vložíme súbory **.htaccess** a **.htpasswd**.

Najprv je ale potrebné nastaviť Netbeans tak, aby boli tieto dva súbory viditeľné. To je možné nasledovne:

V menu kliknite na: **Tools->Options->Miscellaneous->Files**

V textovom poli zmeňte zápis tak, aby akceptoval aj htpasswd súbory (doplňte text `“htpasswd/”`):

```
^(CVS|SCCS|vssver.?\|.scc|#|*#|*%|*_svn)$|~$|^\. (?!htpasswd|htaccess$) .*$
```

Potvrďte tlačidlom **OK**. Teraz by ste mali vidieť súbory vytvorené v nasledujúcom postupe.

Postup pre vytvorenie súborov **.htaccess** a **.htpasswd**:

Najprv si vytvoríme oba súbory. V prostredí Netbeans je možné nové súbory v adresári lecturer vytvárať nasledovne: pravý klik na adresár lecturer -> **New** -> **Other...** -> v okne vyberte **Other** -> **Empty File** -> **Next** -> zadajte názov „**htpasswd**“ alebo „**htaccess**“ -> **Finish**.

Vytvorte nový súbor s názvom **.htaccess** a vložte do neho nasledujúci obsah:

```
AuthUserFile /opt/it4kt/www/c/nazov_adr_predmetu/lecturer/.htpasswd
AuthGroupFile /dev/null
AuthName "The Secret Page"
AuthType Basic

<Limit GET POST>
require valid-user
</Limit>
```

pričom text **nazov_adr_predmetu** je potrebné nahradiť za názov adresára príslušného predmetu, (napr. fjap). Uložte a zatvorte súbor **.htaccess**.

Ďalej vytvorte nový súbor **.htpasswd**. Obsah súboru **.htpasswd** je možné vytvoriť pomocou [tohto nástroja](#), kde do políčka **“Username”** napíšete login a do políčka **“Password”** zadajte heslo pre predmet (odporúčame písmená aj čísla, napr. fjapko2013). Po stlačení tlačidla **“Create .htpasswd file”** sa zobrazí vygenerovaný obsah, mal by to byť jeden riadok v tvare **login:\$zakryptovane_heslo**.

Tento riadok skopírujte do novo vytvoreného súboru **.htpasswd**.

Môže sa stať, že po nakopírovaní bude v adresárovej štruktúre v nástroji Netbeans vidno súbor **.htpasswd** sivou farbou. Chyba je v SVN, ktorého predvoleným nastavením je ignorovanie tohto súboru. Pre odignorovanie je potrebné **kliknúť na .htpasswd pravým tlačidlom -> Subversion -> Unignore**.

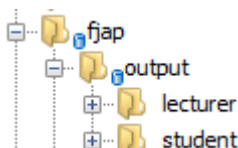
V nasledujúcom kroku je potrebné presunúť tieto dva súbory na server, aby bolo možné zamedziť prístupu k učiteľským materiálom. To urobíme prostredníctvom funkcie **Commit** nad adresárom projektu podľa návodu v kapitole [Odoslanie materiálov predmetu na SVN](#).

UPOZORNENIE: Odteraz už je možné využívať LEN tlačidlo **Build**. Ak sa pomýlite a stlačíte namiesto neho **Clean & Build**, obsah adresára bude premazaný a oba súbory pre zaheslovanie budete musieť nakopírovať znova. To isté sa stane, ak omylom zmažete adresár **output/lecturer**. Preto odporúčame oba súbory si zálohovať.

Po tomto je dobré overiť, či všetko prebehlo správne. Upload materiálov sa udeje až po niekoľkých sekundách. Ak si teraz zobrazíte adresár svojho predmetu v prehliadači (napr. <http://it4kt.cnl.sk/c/fjap/>) a znovu načítate obsah (Reload), adresár **lecturer** už nebude viditeľný. Po zadaní adresy pre tento adresár do prehliadača (napr. <http://it4kt.cnl.sk/c/fjap/lecturer>) už bude prehliadač vyžadovať heslo pre prístup do adresára. To isté platí aj pre zobrazenie akejkoľvek stránky z učiteľských materiálov. Overte si, že vami zadané meno a heslo korektne umožní prístup k učiteľským materiálom.

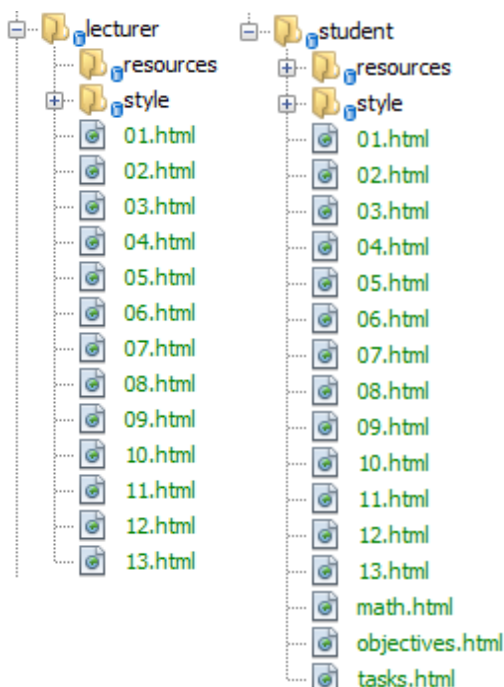
7. Vygenerované súbory

Všetky vygenerované súbory sa nachádzajú v adresári **output** vášho predmetu. Tento adresár obsahuje nasledujúce podadresáre:



- **student** – obsahuje html stránky a zdroje, ktoré sú viditeľné verejne, pre študentov aj učiteľov.
- **lecturer** – obsahuje html stránky a zdroje, ktoré by mali byť viditeľné len pre učiteľa.

Spoločné adresáre a súbory pre učiteľa aj pre študenta sú:



- **resources** – adresár so zdrojmi pre projekt
- **style** – adresár pre štýly html stránok
- **01-13.html** – súbory html stránok, pre každé cvičenie jeden súbor

Adresár **student** navyše obsahuje aj tri ďalšie súbory:

- **math.html** – zbierka všetkých matematických vzorcov použitých vo všetkých cvičeniach predmetu. V zozname vzorcov sa zobrazia iba tie vzorce, ktoré sú v xml súboroch zapísané v bloku $$$. Tento výstup je možné použiť ako zbierku vzorcov pre študentov na naučenie resp. ako oficiálny ťahák pri testoch a skúškach.$
- **objectives.html** - zbierka všetkých cieľov vo všetkých cvičeniach. Tento výstup je možné využiť napríklad pri tvorbe úvodnej stránky k predmetu.
- **tasks.html** – zbierka všetkých riešených aj neriešených úloh vo všetkých cvičeniach (okrem doplňujúcich). Tento výstup je možné použiť ako zbierku príkladov alebo úloh pre študentov.

Predvolené nastavenie je zatiaľ také, že pre tieto tri súbory v projekte neexistuje odkaz a je potrebné ho pridať ručne podľa potreby. Každý zo súborov je možné zobraziť v prehliadači. Príklad:

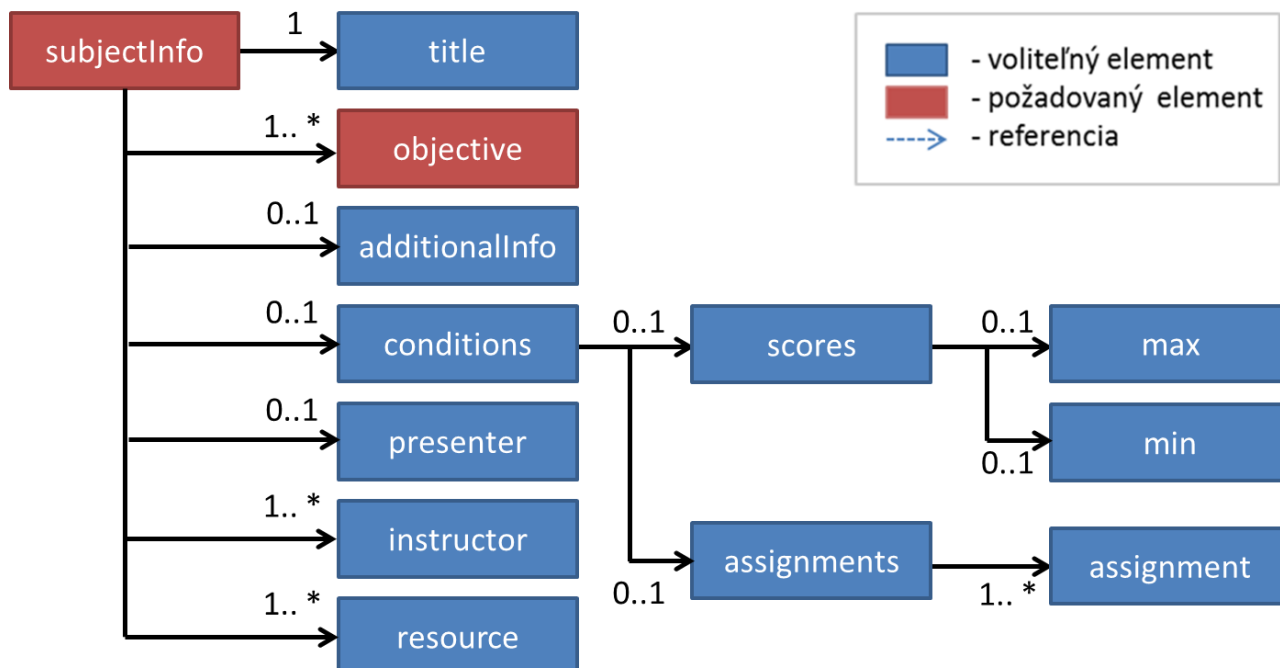
<http://it4kt.cnl.sk/c/nm/student/math.html>

<http://it4kt.cnl.sk/c/fjap/student/objectives.html>

<http://it4kt.cnl.sk/c/fjap/student/tasks.html>

8. Úvodná stránka k predmetu

V každom predmete sa nachádza súbor **info.xml**, ktorý reprezentuje úvodnú stránku k predmetu. Tento súbor má nasledujúcu štruktúru:



Obr. 3: Štruktúra elementov v súboroch info.xml

8.1. Element subjectInfo

Element **subjectInfo** je koreňovým elementom pre opis informácií k predmetu.

Opis predmetu by mal obsahovať všeobecné informácie o predmete a cvičeniach. Môže mať svoj názov (ak nie je, bude automaticky generovaný), zoznam cieľov, doplňujúce informácie, podmienky pre získanie zápočtu, zoznam prednášajúcich a cvičiacich a zoznam zdrojov k predmetu.

8.2. Element title

Element **title** definuje názov stránky obsahujúcej informácie o predmete. Ak nie je zadaný, bude automaticky vygenerovaný názov v tvare „Informácie o predmete“. Môže sa uviesť aj názov „Podmienky pre získanie zápočtu“.

8.3. Element objective

Element **objective** opisuje všeobecný cieľ predmetu.

Každý predmet musí mať vždy aspoň jeden špecifický cieľ. Ciele v rámci predmetu musia byť konzistentné.

Ciele by nemali byť konkrétne, mali by opisovať všeobecné ciele predmetu, čo sa študent naučí a čo dosiahne na tomto predmete. Tieto ciele by mali byť rozložiteľné na čiastkové ciele, ktoré predstavujú ciele modulov.

Ciele by mali vždy vyjadrovať nielen obsah ale aj úroveň (viac o taxonómii cieľov: [Bloomova taxonómia](#)).

8.4. Element additionalInfo

Element **additionalInfo** opisuje prídavné informácie pre predmet.

Môže to byť napr. opis priebehu cvičení, kľúčové fakty a pod.

8.5. Element conditions

Element **conditions** definuje množinu podmienok pre získanie zápočtu z daného predmetu. Mal by opisovať bodové hodnotenie predmetu a typy zadaní, ktoré študent musí na cvičeniach počas semestra zvládnuť.

Pomocou atribútu `generatePresenceConditions="true"` je možné povoliť vygenerovanie dvoch základných podmienok: (i) „Ani jedna neospravedlnená neúčast“ a (ii) „Maximálne 3 ospravedlnené neúčasti - spôsob náhrady určí cvičiaci“.

8.6. Element scores

Element **scores** slúži na určenie počtu bodov, ktoré sú podmienkou pre získanie zápočtu z cvičení.

Obsahuje dva pod-elementy **max** a **min** definujúce maximálnu a minimálnu bodovú hodnotu.

Príklad:

```
<scores>
  <max>30</max>
  <min>16</min>
</scores>
```

8.7. Element assignments

Element **assignments** obsahuje zoznam zadaní, ktoré musí študent zvládnuť pre získanie zápočtu z cvičení.

8.8. Element assignment

Element **assignment** opisuje práve jedno zadanie, ktoré musí študent zvládnuť pre získanie zápočtu.

Element má tri nepovinné atribúty:

- **type** – typ zadania
- **maxPoints** – maximálny možný počet bodov za zadanie
- **week** – týždeň, ktorý je posledným pre odovzdanie zadania

Atribút type

Atribút **type** slúži na určenie typu zadania. Povolené hodnoty atribútu sú: *test* (pre test), *program* (pre programovacie zadanie), *document* (pre dokument – napr. referát), *activity* (pre body za aktivitu), *presentation* (pre prezentáciu), *undefined* (pre žiadny typ). Predvolená hodnota atribútu je *undefined*.

Ak nie je zadaný atribút **type**, text elementu **assignment** by mal vysvetľovať typ zadania a účel zadania.

Atribút maxPoints

Atribút **maxPoints** slúži na zadanie číselnej hodnoty reprezentujúcej maximálny možný počet za dané zadanie. Tento atribút je povinný.

Atribút week

Atribút **week** slúži na zadanie číselnej hodnoty reprezentujúcej posledný termín pre odovzdanie zadania. Ak nie je zadaný, je potrebné termín zadať v texte elementu assignment.

Príklad použitia elementu assignment:

```
<assignment type="test" maxPoints="10" week="4">  
    Test A  
</assignment>
```

Vzorové príklady je možné vidieť v šablóne súboru info.xml a v predmete FJaP.

8.9. Element resource

Element **resource** slúži na opis zdroja, ktorý študenti používajú pri štúdiu predmetu.

Zdrojom môže byť materiál na stiahnutie, kniha, internetový odkaz a podobne.

Zoznam zdrojov na úvodnej stránke by nemal obsahovať všetky zdroje z modulov, ale len najdôležitejšie, všeobecné zdroje týkajúce sa predmetu a cvičení všeobecne, ktoré je možné použiť pri štúdiu počas celého semestra.

8.10. Úvodná stránka v prehliadači

Úvodnú stránku je možné v internetovom prehliadači zobraziť kliknutím na **Cvičenia**: v hornom menu.



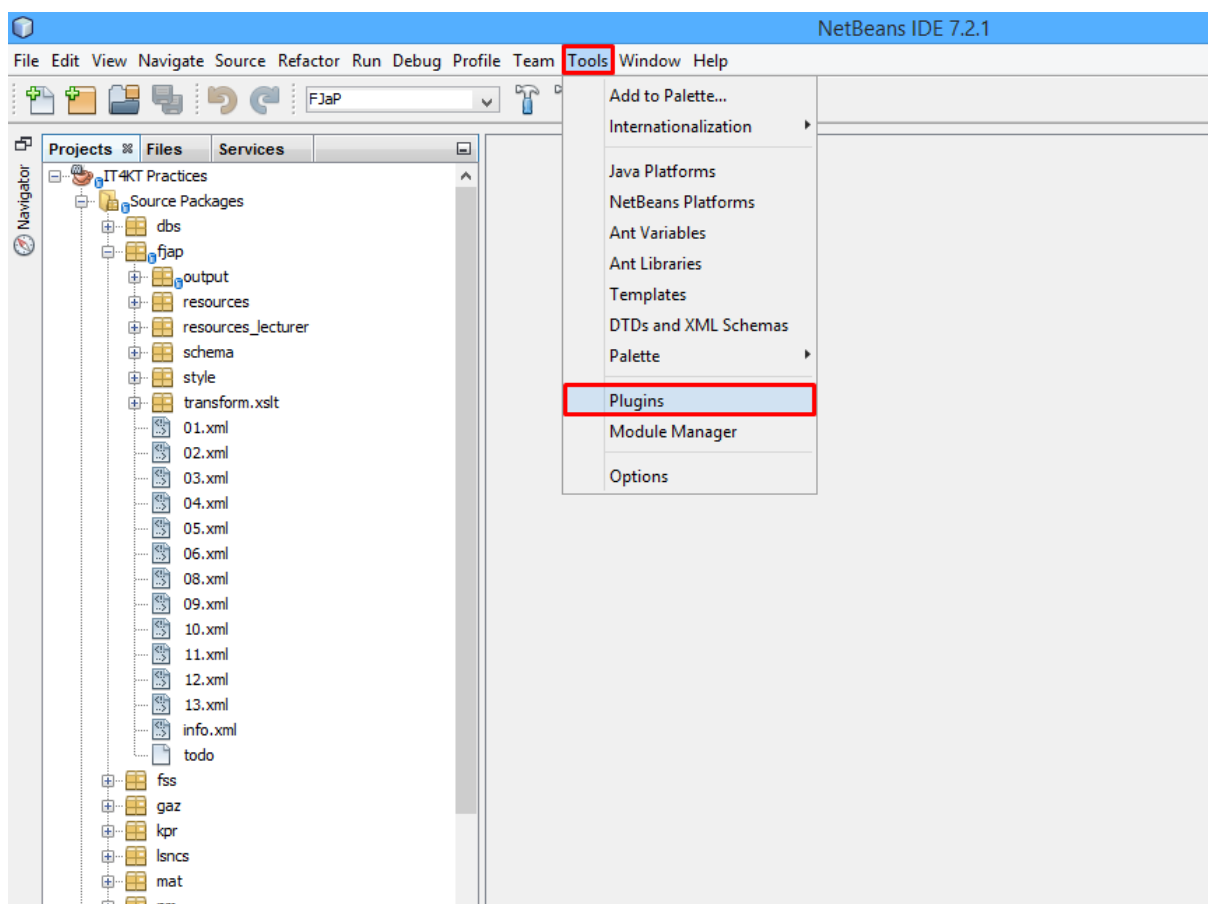
9. Prídavný modul pre prácu s XML modulmi

Prídavný modul pre prácu s modulmi umožňuje zjednodušenie a urýchlenie vytvárania materiálov pre cvičenia. Poskytuje jednoduchú, ľahko použiteľnú paletu elementov, ktoré je možné jednoduchým presunutím myši vložiť do XML súboru modulu.

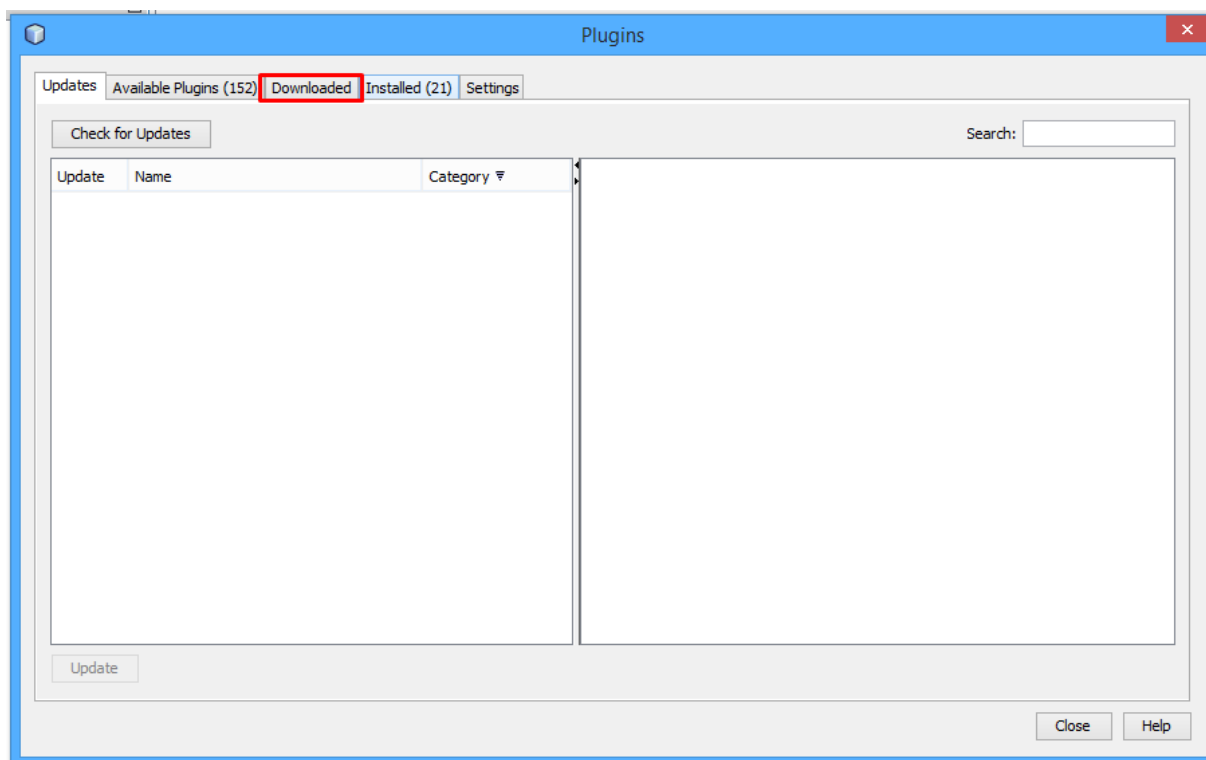
Prídavný modul je možné si stiahnuť z <http://bodrog.fei.tuke.sk/it4kt/services/e1/2013-12-03-sk-tuke-it4kt-plugin-v1.0.nbm>.

9.1. Návod na inštaláciu

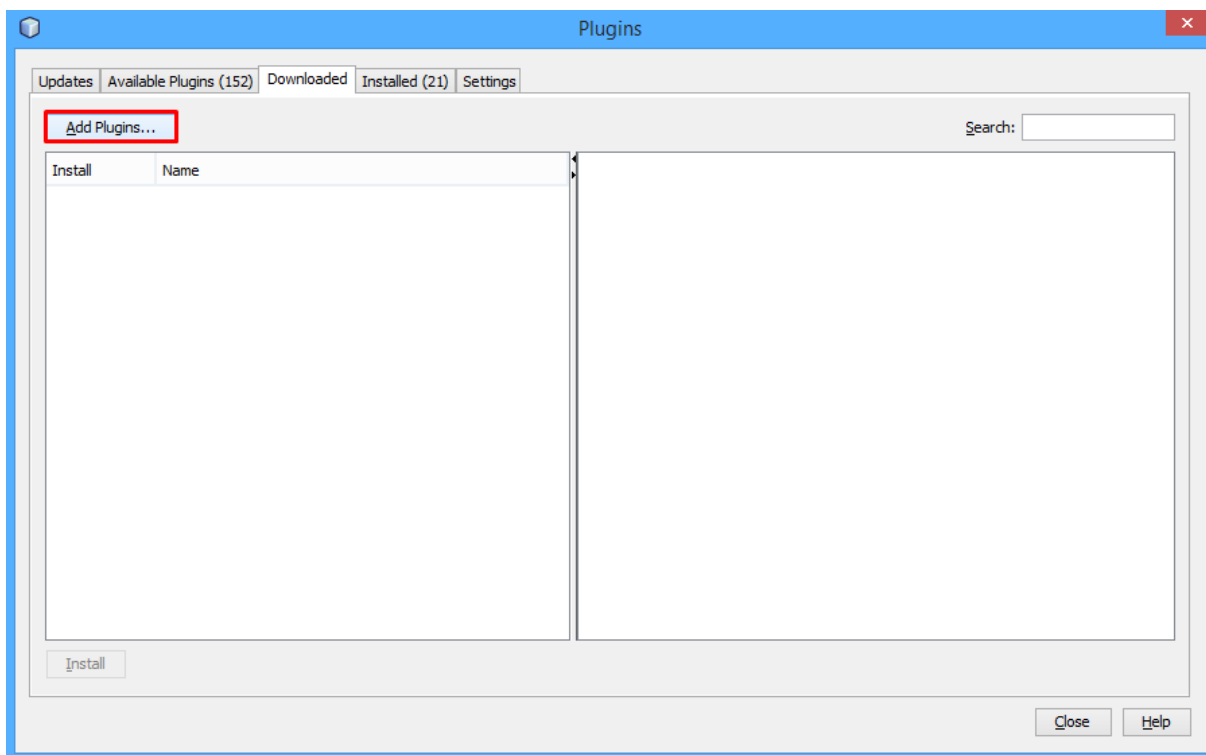
Stiahnite si prídavný modul z vyššie uvedeného odkazu. V hlavnom menu v nástroji Netbeans kliknite na **Tools->Plugins**.



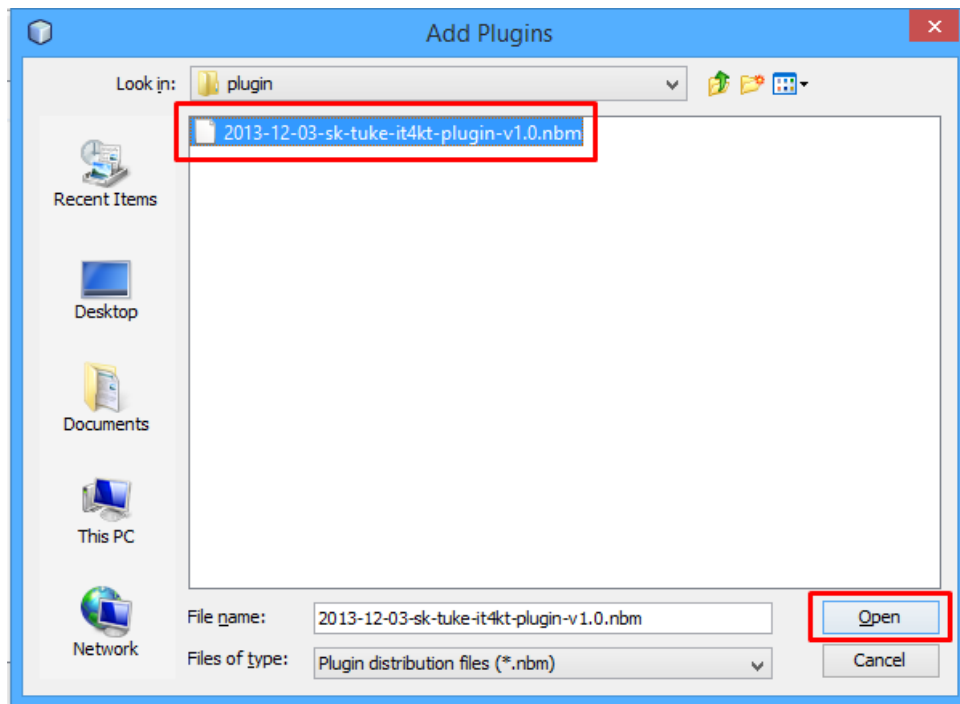
V novo otvorenom okne **Plugins** vyberte záložku **Downloaded**.



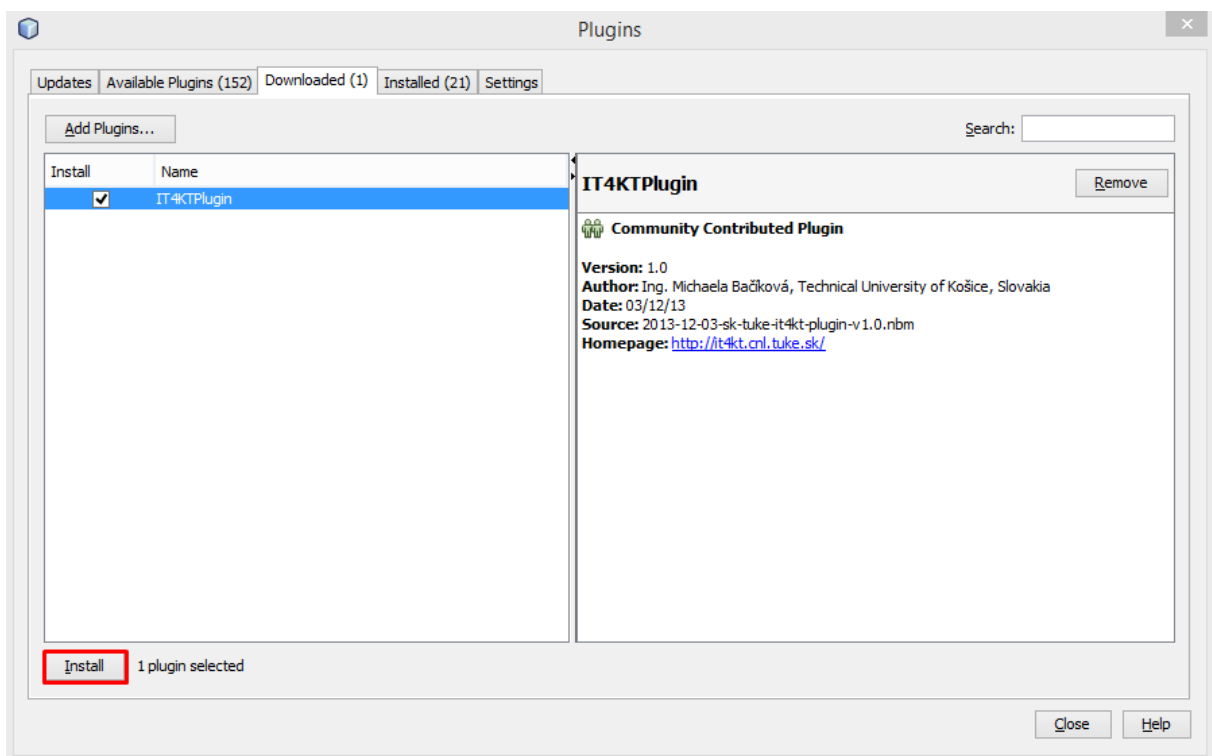
V tejto záložke kliknite na **Add Plugins...**



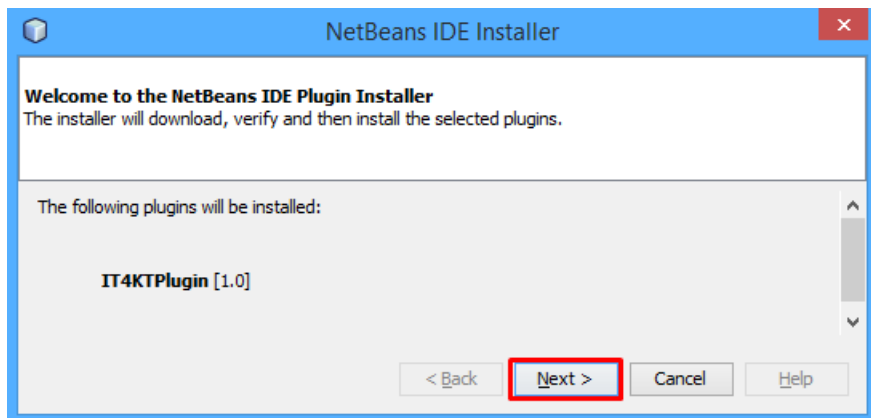
V dialógu vyberte súbor prídavného modulu, ktorý ste stiahli z odkazu a kliknite na **Open**.



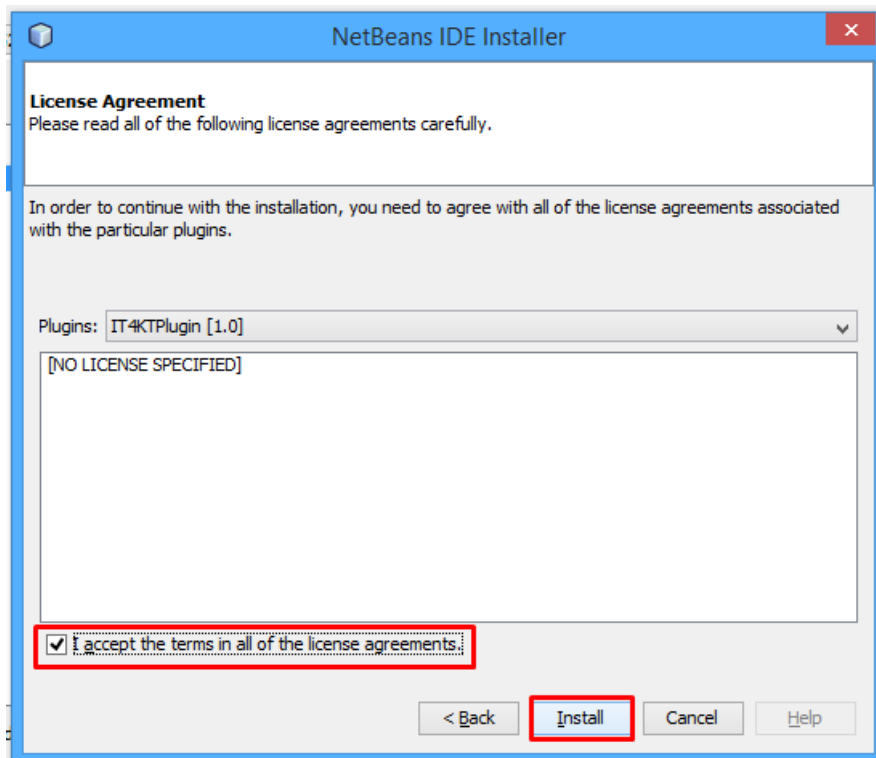
V okne **Plugins** by sa v zozname stiahnutých prídavných modulov mala objaviť zaškrtnutá položka **IT4KTPlugin**. kliknite na tlačidlo **Install**.



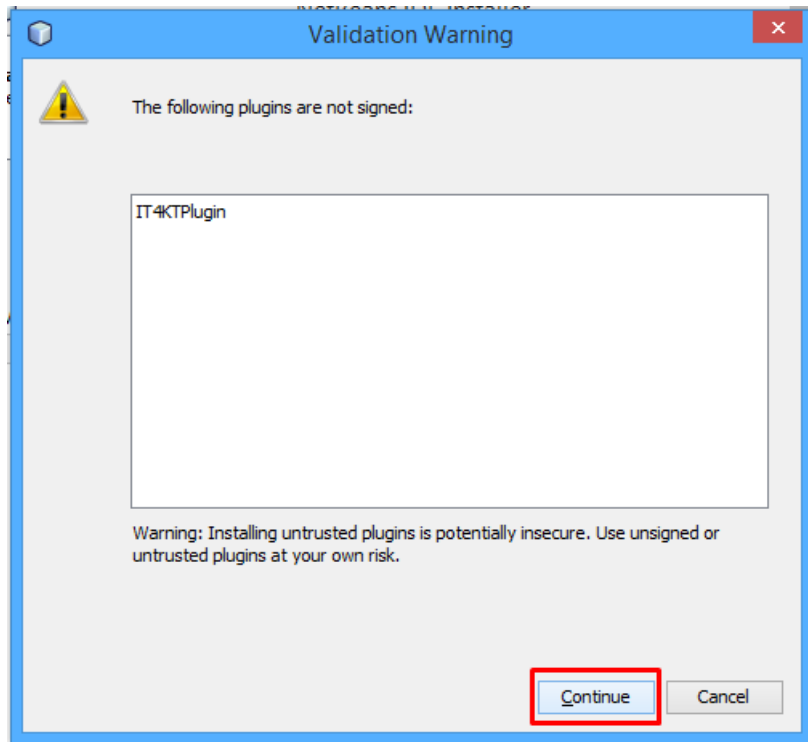
Zobrazí sa inštalátor prídavného modulu. Kliknite na tlačidlo **Next**.



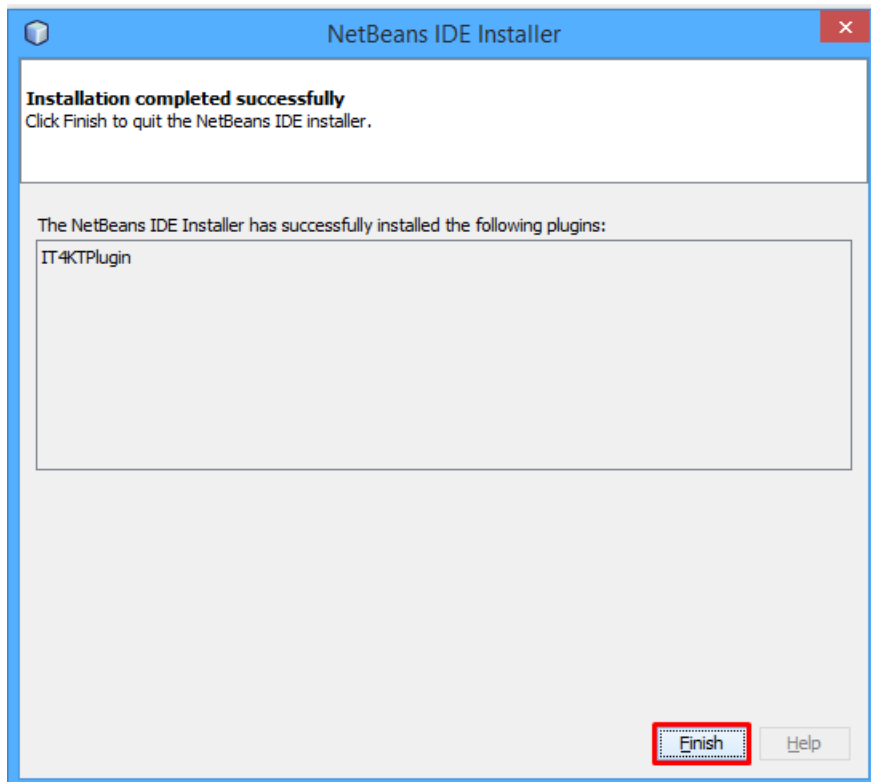
Potvrďte, že súhlasíte s pravidlami zaškrtnutím zaškrtačacieho tlačidla a kliknite na tlačidlo **Install**.



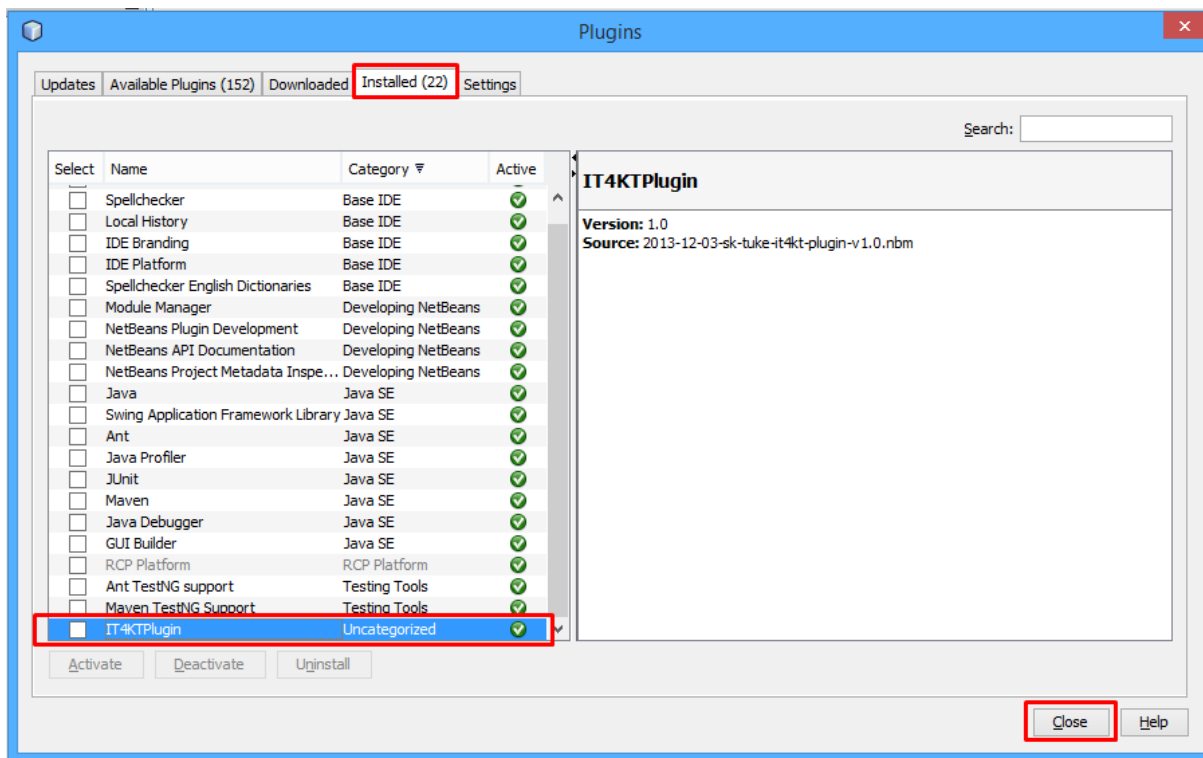
Zobrazí sa okno, ktoré Vás upozorní na to, že obsah, ktorý inštalujete, nie je oficiálne overený. Kliknite na tlačidlo **Continue**.



Po úspešnej inštalácii sa zobrazí správa o výsledku inštalácie. Kliknite na tlačidlo **Finish**.

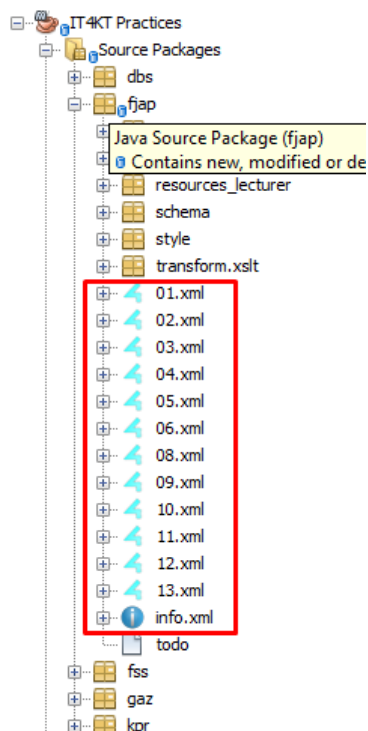


Pred použitím skontrolujte, či bol prídavný modul naozaj pridaný do zoznamu nainštalovaných modulov kliknutím na záložku **Installed**. Následne zatvorte okno tlačidlom **Close**.

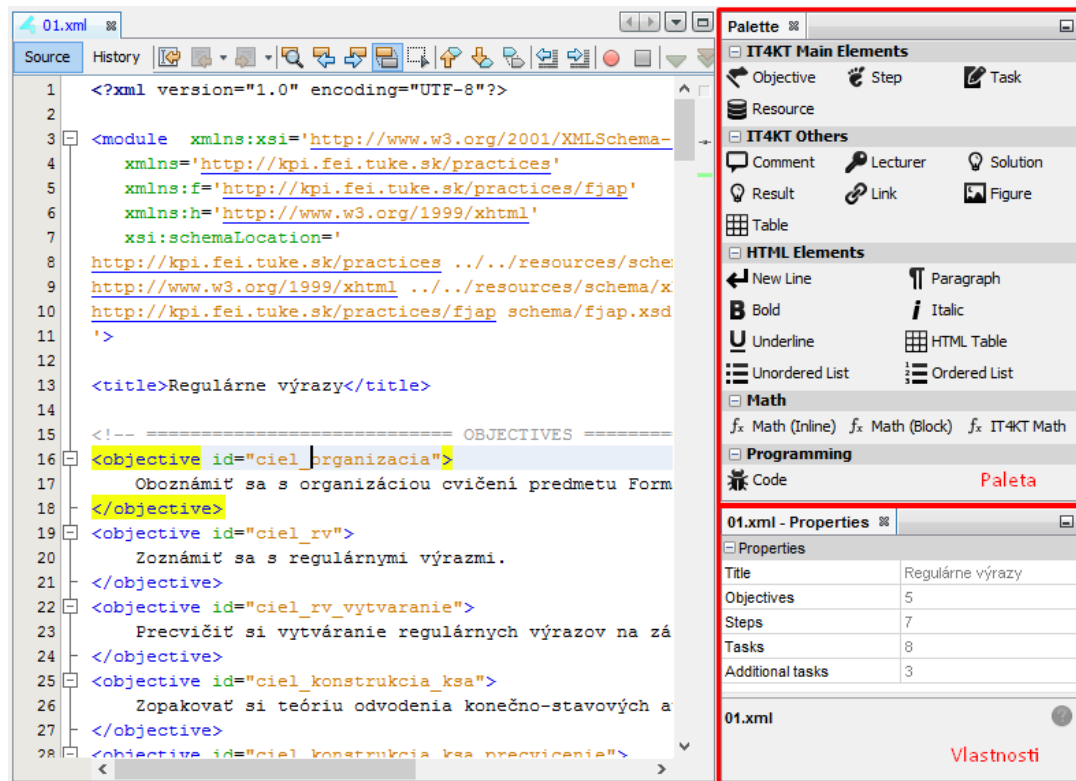


9.2. Návod na použitie

Po nainštalovaní prídavného modulu je možné v projekte ihneď vidieť zmenu – editor rozpoznáva jednotlivé typy xml súborov.



Po otvorení napr. súboru 01.xml sa na pravej strane editora objaví paleta elementov. Ak sa paleta nezobrazila, je možné ju otvoriť pomocou kliknutia na položky **Window->Palette** v hlavnom menu editora. Podobným spôsobom je možné zobraziť aj vlastnosti dokumentu pomocou kliknutia na položky **Window->Properties** v hlavnom menu editora.




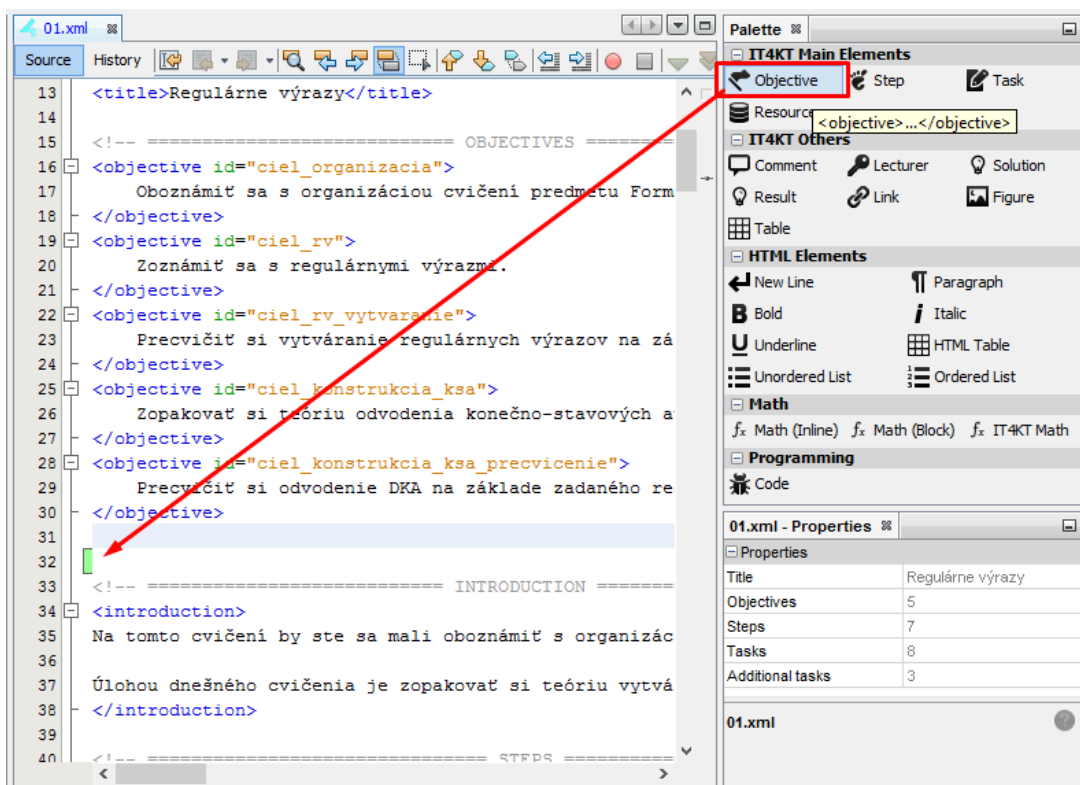
Okno **Properties** (Vlastnosti) má informačný účel. Je v ňom možné vidieť základné informácie o module, ako je názov, počet cieľov, celkový počet krokov, celkový počet úloh a celkový počet prídavných úloh.

Okno **Palette** (Paleta) slúži na pridávanie jednotlivých typov elementov priamo do modulu.

Vloženie jednoduchého elementu pomocou palety

Návod na vloženie jednoduchého elementu do xml súboru (štýlom drag&drop):

- 1 Kliknite napr. na element **Objective** v palete, podržte tlačidlo myši
- 2 Presuňte tento element do súboru na miesto, kde chcete vložiť ďalší cieľ
- 3 Pustite tlačidlo myši
- 4 Text elementu sa automaticky vloží na určené miesto
- 5 Pri chybnom vložení je možné pre vrátenie zmeny stlačiť tlačidlo **Späť**  alebo použiť klávesovú skratku **Ctrl+Z**.



Zoznam jednoduchých elementov:

- **Objective** – vloží blok `<objective> ... </objective>`
- **Step** – vloží blok `<step> ... </step>`
- **Task** – vloží blok `<task> ... </task>`
- **Resource** – vloží blok `<resource> ... </resource>`
- **Comment** – vloží blok `<comment> ... </comment>`
- **Lecturer** – vloží blok `<lecturer> ... </lecturer>`
- **New Line** – vloží nový riadok
- **Paragraph** – vloží HTML paragraf
- **Bold** – tučné písmo
- **Italic** – šikmé písmo
- **Underline** – podčiarknuté písmo
- **Math** elementy – slúžia na vloženie matematických vzorcov pomocou `\(\)` a `[[\]]`
- **Code** – vloží blok `<code> ... </code>`

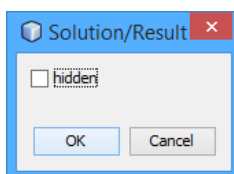
Vloženie parametrického elementu pomocou palety

Parametrické elementy sa vkladajú rovnako ako jednoduché elementy, ale po ich vložení sa zobrazí dialóg, ktorý umožňuje nastavenie parametrov.

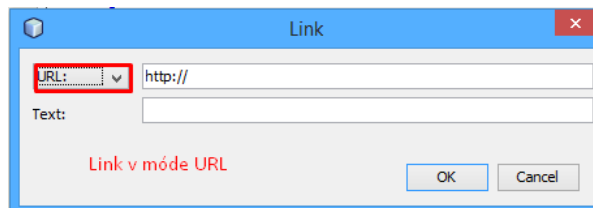
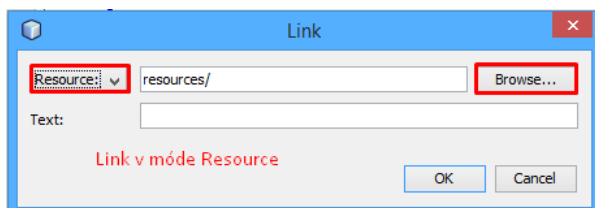
Po zobrazení dialógu je možné nastaviť hodnoty, alebo nechať hodnoty prednastavené a jednoducho potvrdiť tlačidlom **OK**.

Zoznam parametrických elementov:

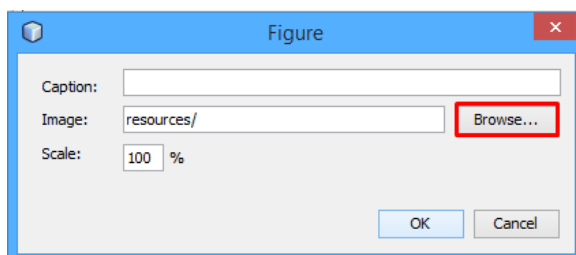
- **Solution** – vloží blok `<solution> ... </solution>`. Je možné nastaviť, či bude riešenie viditeľné alebo skryté (predvolené je skryté).



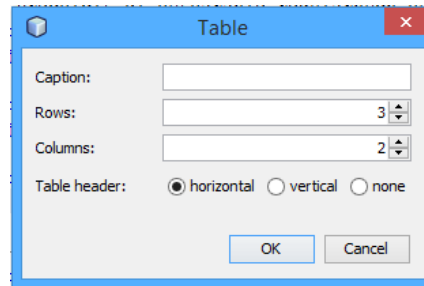
- **Result** – vloží blok `<result> ... </result>`. Funguje rovnako ako blok Solution, rozdiel je len sémantický (resp. v názve vložených značiek).
- **Link** – vloží blok `<link> ... </link>`. Vloží odkaz na interný alebo externý zdroj. V prípade, že chceme vybrať zdroj z adresára **resources** v projekte, je možné použiť tlačidlo **Browse**. V prípade vloženia odkazu je potrebné prepnúť mód z **Resource** na **URL**. Text odkazu nie je povinné nastaviť, v tom prípade bude textom zdroj resp. URL.



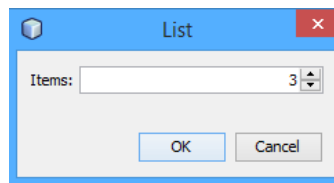
- **Figure** – vloží blok `<figure> ... </figure>`. Pre obrázok je možné nepovinne nastaviť popis obrázka (**Caption**), vybrať jeho zdroj pomocou tlačidla **Browse...** a nastaviť jeho veľkosť v percentách (**Scale**). Pre použitie tohto elementu je nutné, aby súbor s obrázkom bol v adresári **resources**.



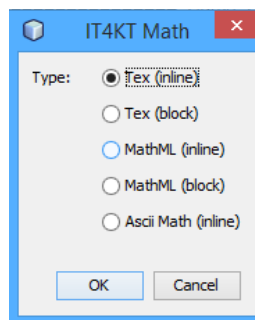
- **Table** – vloží blok `<table> ... </table>`, teda tabuľku. Pre tabuľku je nepovinne možné nastaviť popis tabuľky (**Caption**), počet riadkov (**Rows**) a stĺpcov (**Columns**) a smerovanie hlavičky tabuľky (**Table header**) na: horizontálne, vertikálne a bez hlavičky.



- **Unordered List** – vloží blok `<h:ul> ... </h:ul>`, teda neusporiadaný zoznam. Pre zoznam je možné nastaviť počet položiek.

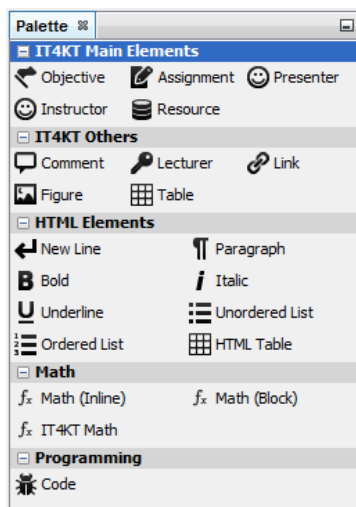


- **Ordered List** – vloží blok `<h:ol> ... </h:ol>`, teda usporiadaný zoznam. Funguje rovnako ako **Unordered list**: je možné nastaviť počet položiek.
- **HTML Table** – vloží HTML tabuľku, nastavenia sú rovnaké ako pri **Table**.
- **IT4KT Math** – vloží blok `$...$`. Slúži ako alternatíva k `\[\]` a `\(\)`, pričom vzorce označené týmto blokom budú zaradené do vygenerovanej zbierky vzorcov. Pri výbere tohto elementu z palety je možné nastaviť typ matematického vzorca (viď kapitolu Element math).



Elementy palety súboru info.xml

Súbor **info.xml** predstavuje úvodné informácie k predmetu. Jeho štruktúra je odlišná od modulov, preto má samostatnú paletu.



Odlišné elementy na palete súboru **info.xml** sú nasledujúce:

- **Assignment** – vloží blok `<assignment> ... </assignment>` reprezentujúci zadanie alebo podmienku pre splnenie zápočtu. Je možné nastaviť text, typ zadania (**test** pre test, **program** pre programovacie zadanie, **activity** pre body za aktivitu, **presentation** pre nutnosť odprezentovania výsledkov práce, **document** pre dokument, **none** pre základný typ), maximálny počet bodov, ktorý je možné získať za dané zadanie a dobrovoľne týždeň, v ktorom študent musí zadanie najneskôr odovzdať.

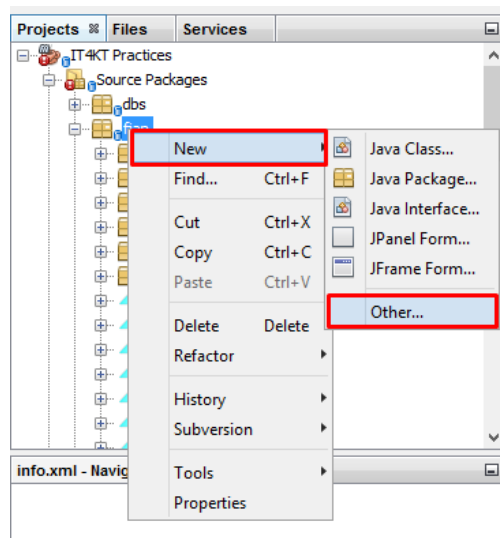
- **Presenter** – vloží blok `<presenter> ... </presenter>` reprezentujúci informácie o prednášajúcom na prednáškach z daného predmetu. U prednášajúceho je možné nastaviť jeho meno, e-mailový kontakt a url na domovskú webovú stránku prednášajúceho.

- **Instructor** – vloží blok `<instructor> ... </instructor>` reprezentujúci informácie o cvičiacom na cvičeniach z daného predmetu. Nastavenia sú rovnaké ako pri elemente **Presenter**.

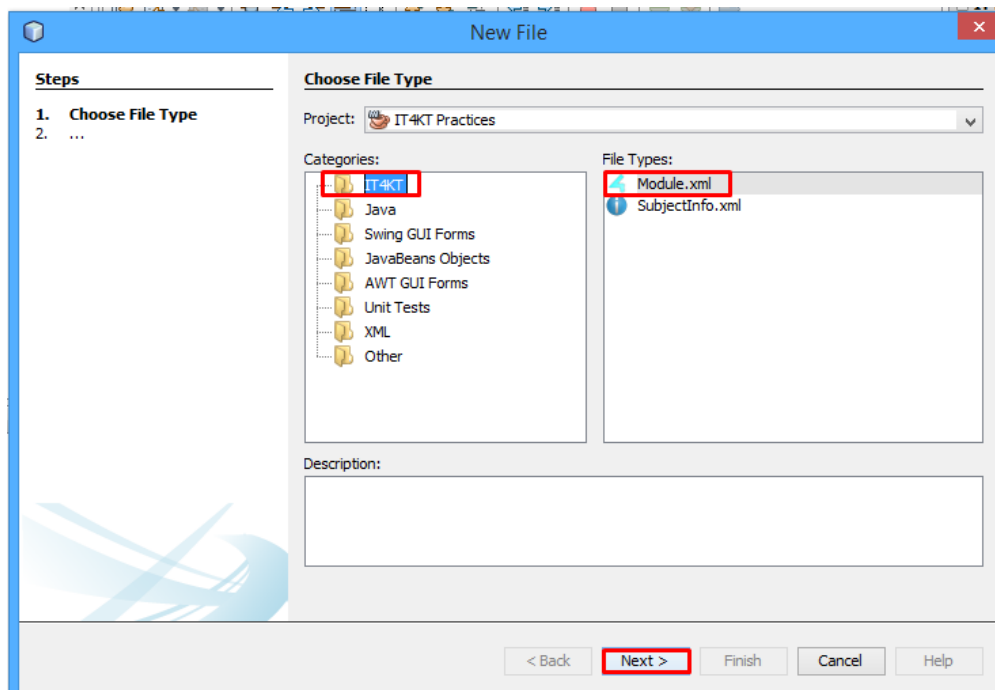
Vytváranie nových súborov pomocou prídavného modulu

Prídavný modul je implementovaný tak, aby poskytoval aj možnosť rýchleho vytvorenia prázdneho modulu alebo informácií k predmetu (info.xml).

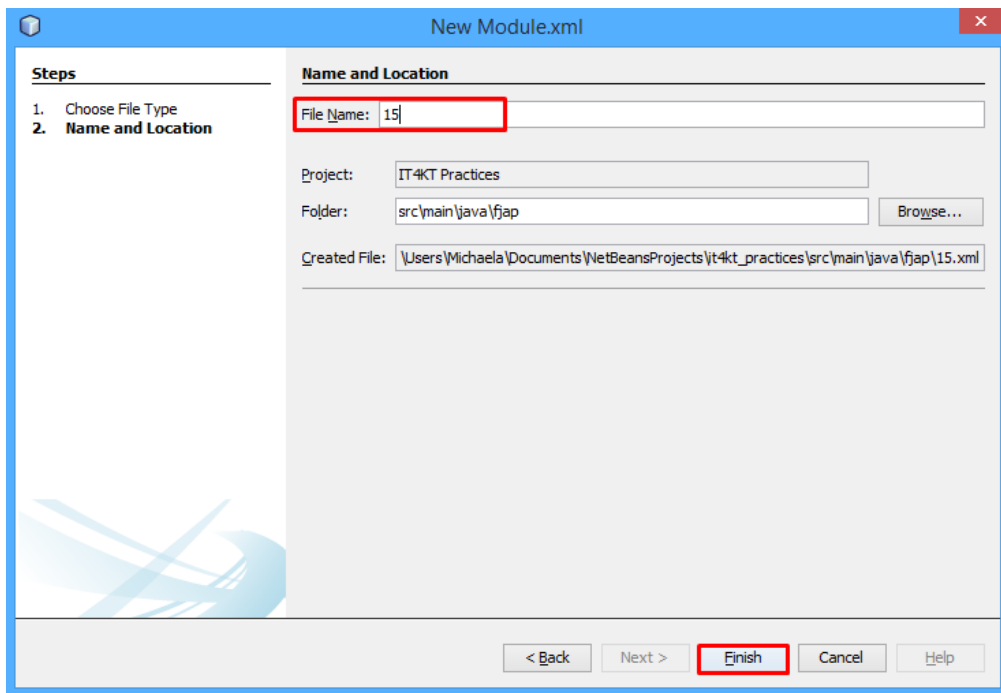
Pre vytvorenie nového modulu kliknite pravým tlačidlom na adresár vášho predmetu a vyberte možnosť **New -> Other...**



V zozname kategórií vyberte kategóriu **IT4KT** a v zozname typov súborov vyberte **Module.xml**. Kliknite na tlačidlo **Next**.



Zadajte názov nového súboru a kliknite na tlačidlo **Finish**.



Vytvorí sa nový prázdny modul s predvytvorenou šablónou pre vpisovanie obsahu. Podobným spôsobom je možné vytvoriť aj súbor **info.xml**. V zozname kategórií vyberte **IT4KT**, v zozname typov súbor vyberte **SubjectInfo.xml** a ako názov súboru uveďte napr. **info**. Vytvorí sa nový súbor info.xml s predvyplnenou šablónou pre vpisovanie obsahu.