

Dokumentace úlohy SYN: Zvýraznění syntaxe v Pythonu 3 do IPP 2016/2017  
jméno a příjmení: Roman Kiselevič  
Login: xkisel00

## Zadání

Cílem úlohy bylo vytvořit skript, který zvýrazňuje různé části textu s použitím html značek. Informace o zvýraznění se nachází ve formátovacím souboru, který obsahuje regulární výrazy a seznamy parametrů a to tak, že ke každému výrazu patří jeden seznam parametrů, kde každý parametr reprezentuje html značku.

## Práce se vstupními argumenty

Za zpracování argumentů příkazové řádky je zodpovědní funkce `get_args()`, která používá funkci `getopt()` ze standardní knihovny Python, což mnohem usnadňuje syntaktickou analýzu. Výsledkem funkce `get_args()` je slovník, ve kterém klíče a hodnoty jsou řetězec reprezentující vstupní argument a jeho hodnota. V případě syntaktické nebo sémantické chyby `get_args()` vyvolá výjimku `GetoptError`.

## Práce se soubory

Pro operace zápisu a čtení nad soubory (případně standardním vstupem nebo výstupem) se používají standardní funkce resp. `file.read()` a `file.write()`. Pro otevření souboru se používá standardní funkce `file = open()` s parametrem `encoding='utf-8'`.

## Formátovací soubor

Všechny třídy a funkce potřebné pro práci s formátovacím souborem se nacházejí v modulu `ipp_syn.format_file`. Samotný formátovací soubor reprezentuje třída `FormatFile`, jejíž konstruktor se stará o syntaktickou analýzu a má jediný parametr, což je název souboru. Syntaktická analýza se provádí pomocí regulárních výrazů ze standardního modulu `re` a v případě chyby se vyvolá výjimka `FormatFileError`. Analýza se provádí takovým způsobem, že na začátku se načítá regulární výraz, který se konvertuje do objektu `RegexPython`. Potom se načítá seznam parametrů, a pro každý z nich se vytváří objekt `RegexParam`. Na konci výsledkem analýzy je slovník, klíči kterého jsou objekty `Regex` a hodnoty jsou seznamem objektů `RegexParam`. V každém objektu třídy `FormatFile` tento slovník je zapouzdřený a přístup k němu uskutečňuje pomocí přetížených magických metod (`__setitem__()`, `__getitem__()` atd.) Taktéž jsou přetížení iterované metody (`__iter__()`, `__next__()`)

## Závěr

Skript byl otestován sadou testů přiloženou k zadání projektu a několika vlastními testy. Testování proběhlo na operačním systému Arch Linux 4.9.11-1 a na školním serveru Merlin s operačním systémem CentOS. Všechny tyto testy dopadly úspěšně