

Výběrové řízení na pozici PROGRAMÁTOR JAVA

ZADÁNÍ PRAKTICKÉHO ÚKOLU

Abstrakt:

V konzolových aplikacích často potřebujeme zpracovávat a číst volby (options) z příkazového řádku. Vzhledem k tomu, že se v podstatě jedná o rutinní činnost, rádi bychom pro tento účel použili knihovnu, která by tuto funkcionalitu vhodnou abstrakcí kompletně pokrývala a zrychlila tak práci vývojáře. Použití knihovny (definice a čtení voleb) by mělo být intuitivní a nemělo by vyžadovat příliš konfiguračního kódu ze strany uživatele (programátora).

Specifikace parametrů:

Krátká volba (short option)

Argument začínající pomlčkou, za kterou následuje jeden znak (např. "-v")

Dlouhá volba (long option)

Argument začínající dvěma pomlčkami, za kterými následuje jeden a nebo více znaků (např. "--version")

Volba (option)

Krátká a nebo dlouhá volba

Parametr volby (option parameter)

Argument následující po volbě v případě, že daná volba akceptuje parametr (např. "f test.txt")

Funkční požadavky:

- Přehledná deklarativní definice voleb na jednom místě
- Každá volba má tuto konfiguraci
 - o povinná/nepovinná – povinná musí být součástí vstupu. Aplikovatelné pouze na volby s parametrem.
 - o popis – zobrazí se v nápovědě
 - o aliasy – pole aliasů, které danou volbu identifikují (napr. -n, --num, --number)
 - o default – hodnota, která bude použita v případě, že není součástí vstupu. Aplikovatelné pouze na nepovinné volby s parametrem.
- Typy voleb a omezení
 - o Integer – celočíselná volba s parametrem
 - Volitelné omezení: minValue, maxValue
 - o String – textová volba s parametrem
 - Volitelné omezení: minValue, maxValue
 - o Boolean –
 - Parametrický – true/false, které jsou reprezentované volitelně definovanými texty. Např. true={yes, 1, on, ...}, false={no, 0, off, ...}
 - Bezparametrický – bez parametru volby. To znamená, že pokud je taková volba na vstupu, bude automaticky zpracovaná jako true. Jinak false. Z principu nepovinný.
 - Bez omezení
 - o Enum – výčtový typ

- Bez omezení
 - Custom – vlastní typ. Knihovna musí poskytovat rozšiřitelnost pro vlastní datový typ jako například: BigDecimal, List<String>, MyCustomObject, ...
 - Knihovna musí poskytnout rozšiřitelnost pro implementaci vlastního omezení. (např. „IN“ – výběr z množiny)
- Možnost zadefinovat rezervovanou volbu. Tato volba nemusí být uživatelem (programátorem) předdefinovaná. (knihovna bude automaticky definovat rezervovanou volbu -h resp. --help, která vypíše celý seznam parametrů s jejich konfigurací)
- Uživatel knihovny je vedený výjimkami ke správnému použití

Příklad vstupu:

-n 10 --source foo.txt --dest bar.txt --locale DE --skipError --switch on

Testovací aplikace:

Funkčnost knihovny bude demonstrována na jednoduché konzolové aplikaci – kalkulačka s těmito volbami:

- -l (left operand) typu Integer, povinný
- -r (right operand) typu Integer, povinný
- -o (operator) typu Enum s hodnotami {PLUS, MINUS, MULTIPLY, DIVIDE}, povinný
- -v (verbose) typu Boolean, bezparametrický. Definuje způsob výpisu: true – vypíše se celá rovnice (např. <l>+<r>=<sum>), false – vypíše se jen výsledek

Aplikace by měla korektně odchylovat chybné aritmetické operace (např. dělení nulou).

Příklad:

Vstup1: -l 5 -r 6 -o PLUS -v

Výstup1: 5+6=11

Vstup2: -l 5 -r 6 -o PLUS

Výstup2: 11

Zpracování úkolu:

Vaším úkolem je **porozumět zadání**, připravit celkový **návrh řešení**, navrhnout rozhraní (API) knihovny a **provést implementaci**. Těžištěm Vaší práce by mělo být řešení “definice voleb” a možnost rozšíření knihovny o “vlastní” datový typ. K implementaci použijte základní knihovny JDK 11. Sestavení knihovny (JAR) doporučujeme konfigurovat pomocí nástroje Maven (<https://maven.apache.org/>)

V případě dotazů se na nás neváhejte obrátit.

Kontakty na zadavatele:

Roman Kunovský - email: roman.kunovsky@bcpraha.com, tel.: + 420 605 835 036

Šimon Kohout – email: simon.kohout@bcpraha.com