

Úvod

Při tvorbě projektu nebylo využito objektového návrhu. Skript by se dal dělit do několika následujících částí: zpracování parametrů ze vstupu, ošetření parametrů-jdou-li korektní vzhledem k zadání, zpracování SQL dotazu, sestavení cesty a následný výpis výsledků.

Implementace

Zpracování parametrů

Ze začátku byla využita funkce `getopt()`, ale kvůli jejím nedostatkům byla nahrazena za zpracování pomocí regulárních výrazů. Cyklem `foreach()` je brán parametr po parametru a porovnáván s povolenými možnostmi. Pokud se zde vyskytuje jakýkoli nepovolený parametr, program je ukončen s chybou. Roztříděné parametry jsou dále předány ve formě pole `$options` funkci `parse_options()`.

Ošetření parametrů

Funkce `parse_options()` slouží k ošetření parametrů dle zadání. Probíhá zde např. kontrola, zda nedošlo k zadání dvou SQL dotazů zároveň, načtení vstupního XML souboru, vytisknutí nápovědy atd. Tato funkce byla schválně vytvořena zvlášť pro lepší přehlednost kódu, oproti možnosti zabudovat ji do předešlé funkce. Po vykonání je vráceno pole `$glob` a s tímto polem volána následně funkce `parse_query()`.

Zpracování SQL dotazu

Funkce `parse_query()` jedním dlouhým regulárním výrazem zkontroluje správnost zadaného SQL dotazu. Proběhne-li všechno v pořádku, výsledkem bude pole `$QR` s jednotlivými částmi dotazu. Jestliže tak nenastane, program je ukončen s chybou. Dále je testován rozsah klauzule `LIMIT`. Kontrola zda se jedná o integer a také zda to není nula (při bezproblémovém vykonání programu a nalezení výsledků, by nedošlo k vypsání výsledků, jen hlavičky a obalení kořenovým elementem, jestliže je tak nastaveno, viz. dále). K této klauzuli jsem podle fóra pochopil, že záporné hodnoty nebudou testovány a tak testuji na kladný rozsah konstanty `PHP_INT_MAX`. Následně probíhá roztřídění klauzule `FROM`, zda obsahuje element, atribut, element s atributem, `ROOT` nebo prázdný řetězec.

Dotaz doposud není vyhodnocen celý, zbývá ošetřit klauzuli `WHERE`. K tomu poslouží funkce `parse_where()`, která má jako vstup proměnou `$QR` z předešlé funkce vyhodnocování dotazu. Tato funkce zjistí, jestli se vyskytuje v dotazu klauzule `NOT` (popřípadě kolikrát, podle gramatiky může totiž dojít k zanoření) a vyhodnotí, jestli výsledek má být záporný či nikoli. Následuje další dlouhý regulární výraz pro celou klauzuli `WHERE`, při nesplnění je program ukončen s chybou. Jako další krok je kontrola zda `WHERE` obsahuje `CONTAINS` či naopak operátor. Při jakémkoli negativním stavu je program ukončen s chybou, jinak funkce končí s vrácením pole `$where`.

Sestavení cesty

Jak už tomu nadpis napovídá, bude následovat sestavení cesty na základě získaných a roztříděných údajů, které máme k dispozici v proměnných `$QR` a `$where`. Tyto proměnné máme na vstupu u funkce `search_action()`. Provede se kontrola, jestli klauzule `FROM` není prázdná, v tom případě i cesta bude prázdná. Řazení pro cestu je následující: `FROM` → `SELECT` → `WHERE`. Funkce navrací sestavenou cestu v proměnné `$path`.

Výpis výsledků

Nakonec je volána funkce `final_print()` s předešlými proměnnými. Vyhledání výsledků z výsledné cesty provede funkce `xpath()`. Na základě proměnných se vyhodnocuje jak postupovat dál:

- Zda se má vypsát do souboru či na standartní výstup.
- Vypsání hlavičky.
- Vypsání kořenových elementů.
- V případě prázdné klauzule `FROM` vypsání prázdného souboru (popř. vypsání dvou bodů výš).
- Oříznutí klauzulí `LIMIT` (v případě hodnoty „0“ bude výsledek stejný jako o bod výš).