

Zadání

Cílem bylo vytvořit skript, který dle vstupního XML souboru s daty vytvoří sadu SQL příkazů, kterými je možné vygenerovat databázi, do které se dají vložit data ze vstupního souboru. Toto je dále upraveno parametry předanými při spuštění skriptu.

Řešení

Zpracování argumentů

Ke zpracování argumentů jsem použil funkci `getopt`. Také dochází ke kontrole, zda všechny předané argumenty byly správné. Poté se nastaví globální proměnné, které reprezentují informace předané argumenty. Dále pak ke kontrole možných konfliktů mezi argumenty. Pokud byl zadán parametr `--input` nebo parametr `--output`, dochází také ke kontrole (zda existují, zda je povolen zápis) příslušných souborů a následně k jejich otevření.

Zpracování XML souboru

Zpracování XML souboru probíhá za pomoci knihovny SimpleXML. Skript rekurzivně načítá elementy, pro každý vytváří objekt mé vlastní třídy `element`, přidává mu atributy a podelementy. Tyto elementy jsou vkládány do asociativního pole, kde klíčem je jejich jméno. Atributy jsou v elementu reprezentovány jako asociativní pole, kde klíčem je jejich název a hodnotou jejich typ. Podelementy jsou vkládány taktéž do asociativního pole, klíčem je opět název, ale hodnotou je počet. Dále má každý `element` proměnné pro jméno, jméno rodiče, a typ, který je využit, má-li daná XML položka textový obsah.

Pokud se již v poli nachází `element` se stejným jménem, dochází k jejich sloučení pomocí funkce `merge_elements`.

Rozpoznání datového typu

Funkce `get_type` rozpozná a vrátí typ dat, které se nachází v textovém řetězci předaném funkci. Datovými typy jsou `BIT`, `INT`, `FLOAT`, `NVARCHAR` a `NTEXT`. Funkce nerozlišuje mezi `NVARCHAR` a `NTEXT`, k tomuto dochází až při tisku DDL.

Ke kontrole konfliktu datových typů a k jeho vyřešení je použita funkce `compare_types`, která vrací "silnější" datový typ dle rovnice: `BIT < INT < FLOAT < NVARCHAR < NTEXT`

Parametr `--etc`

Pokud je nastaven parametr `--etc` a není nastaven parametr `-g`, dochází ke kontrole elementů pomocí funkce `check_max`, která projde všechny elementy a prověří, zda některý nemá počet podelementů se stejným jménem, který přesahuje hodnotu předanou s argumentem `--etc`. Pokud takový element objeví, odstraní z jeho podelementů příslušný podelement a naopak ho nastaví jako podelement onoho podelementu. V případě, že je zároveň nastaven parametr `-g`, k této funkci nedochází.

Parametr `-g`

Z pole elementů se funkcí `create_relations` vytvoří pole polí, které reprezentuje vztahy mezi tabulkami. Pokud existuje položka `[a][b]` pak je mezi tabulkou `a` a tabulkou `b` vztah vyjádřený hodnotou na této položce v tomto poli. Vztah je vyjádřen jako `1:1`, `1:N`, `N:1` nebo `N:M`. Dochází ke zpracování kardinality vztahů.

Výstup

Na výstup se tiskne sada SQL příkazů nebo XML.

Závěr

Správnost řešení jsem testoval pomocí oficiálních testů, které jsem mírně rozšířil o automatickou kontrolu pomocí JExamXML.