

1 Zadání

Úkolem tohoto projektu bylo vytvořit skript v jazyce PHP 5, který na základě XML souboru s daty vytvoří sadu SQL příkazů generujících příslušnou strukturu tabulek v SQL databázi, v níž by se data obsažená ve vstupním souboru mohly nacházet.

2 Popis řešení úlohy

2.1 Zpracování argumentů

Argumenty příkazové řádky se zpracovávají ve funkci `ziskajArgumenty()` (je dostupná v souboru `argumenty.php`), která se zavolá jako první, jakmile se skript spustí. Funkce zjistí správnost zadaných argumentů, možné kolize argumentů a podle toho buď skript pokračuje, nebo zobrazí příslušnou chybovou hlášku a ukončí program s odpovídajícím návratovým kódem.

2.2 Vstup skriptu

Skript je schopen zpracovávat XML soubor zadaný přepínačem `--input` nebo text zadaný na standardní vstup. Skript předpokládá, že daný XML soubor je validní. V případě nevalidity XML souboru skript vygeneruje pouze prázdný soubor.

2.3 Zpracování XML souboru

Funkce pro zpracování XML souboru se nacházejí v souboru `xml.php`. Skript zpracovává XML soubor pomocí knihovny SimpleXML, která z jednotlivých elementů a atributů vytvoří objekty a skript pomocí rekurze generuje přímo řádky SQL příkazů, které se ukládají do pole polí. Nadřazené pole je indexované názvem tabulek, které toto nadřazené pole obsahuje. Podřazené pole již obsahují na každém indexu jeden řádek SQL příkazu. Sloupce vznikající z elementů generuje funkce `generatorPodelementu()`, sloupce z atributů generuje zase funkce `generatorAtributu()`. Obě funkce pracují nezávisle na sobě.

2.4 Určení datového typu

K určení datového typu sloupců vznikajících z elementů nebo atributů slouží funkce `datovyTyp()` (nachází se v souboru `xml.php`). Funkce se na základě pravidel rozhodne o jaký datový typ jde. Pokud je analyzovaný řetězec prázdný, obsahuje pouze `True/False` nebo `0/1` funkce rozhodne, že jde o datový typ `BIT`. Pokud je string číslo, které obsahuje nějaký znak specifický pro datový typ `double` (standart C99) funkce rozhodne, že jde o typ `FLOAT`, jinak pokud je string číslo rozhodne, že jde o `INT`. Jinak pokud obsahuje cokoli jiného, funkce rozhodne, že se jedná o datový typ `NVARCHAR` (pokud string pochází z atributu elementu) nebo `NTEXT` (pokud string pochází z obsahu elementu).

2.5 Konflikty datových typů

Konflikty datových typů se řeší přímo při ukládání řádků SQL příkazu do podřazených polí, kdy se prověřuje, zda pole už náhodou neobsahuje prvek se stejným názvem, ale jiným datovým typem. Pokud obsahuje zkontroluje se, zda má menší datový typ podle pravidla `BIT<INT<FLOAT<NVARCHAR<NTEXT` jako znovunalezený prvek, a když má, tak stávající datový typ přepíše novým vyšším datovým typem.

2.6 Přepínač --etc

Funkcionalita přepínače --etc je zpracována ve funkci `generatorPodelementu()`, která si nejprve pomocí funkce `spocitajElementy()`, spočítá, kolik podelementů stejného názvu se v elementu nachází a podle toho, zda byl nebo nebyl počet větší jako zadaný maximální počet přepínačem --etc, umístí cizí klíč.

2.7 Přepínač -g

Funkcionalita přepínače -g je samostatně zpracovaná v soubore `prepinacg.php`. Ze vstupu se nejprve vygeneruje klasickým způsobem databázová struktura, která se uloží do pole a pak se spustí funkce `xmlDoG()`, která volá funkce pro generování jednotlivých vztahů. Nejprve se funkcí `vztahPodradenost()` vygenerují vztahy N:1, pak se pomocí funkce `vztahNadradenost()` vygenerují k těmto vztahům, symetrické vztahy 1:N a nakonec se funkcí `vztahRovnocennost()`, vygenerují zbývající vztahy N:M.

3.0 Rozšíření

Validace souboru zadaného přepínačem --isvalid je zpracována v souboru `validacia.php`. Soubor se nejprve klasicky zpracuje a vygeneruje se z něj databázová struktura, která se následně porovnává s databázovou strukturou vygenerovanou ze souboru daného přepínačem --input. Kontroluje se, zda databázová struktura určená k validaci nemá náhodou více tabulek, zda stávající tabulky mají stejný název, jako tabulky ze struktury --input, zda se všechny sloupce ze struktury --isvalid nacházejí také ve struktuře --input a zda souhlasí datové typy.

2.9 Výstup skriptu

Vytvořené SQL příkazy se na konci vytisknou buď do souboru, který je specifikován přepínačem --output, nebo v případě absence přepínače, na standardní výstup. Všechny úpravy a ošetření jsou provedeny před samotným výpisem.

3 Závěr

Funkcionalita skriptu byla řádně otestována na systému Linux Deepin 2014.2 a na systému CentOS prostřednictvím veřejně dostupné oficiální testovací sady k projektu a také mnou vytvořenou testovací sadou, která porovnávala výstupy s očekávanými výstupy pomocí nástrojů Apgdiff a JExamXML. Všechny testy dopadly úspěšně. Celý skript byl tvořen pouze v jednoduchém textovém editoru Sublime Text 3.