

## Dokumentace úlohy XTD: XML2DDL v PHP do IPP 2016/2017

**Jméno a příjmení:** Stoika Anastasiia

**Login:** xstoik00

### Cíl projektu

Skript na základě XML souboru s daty vytvoří sadu SQL příkazů generujících příslušnou strukturu tabulek v SQL databázi, ve které by se data obsažená ve vstupním souboru mohla nacházet.

### Postup řešení

#### Zpracování parametrů

Skript může být spuštěn s takovými parametry jak: `--help`, `--input='filename'`, `--output='filename'`, `-a`, `-b`, `-g`, `--etc=n`, `--header='header'`. Zpracování parametrů provádím pomocí PHP funkce ***getopt(\$shortopts, \$longopts)*** s předem definovanými `shortopts` a `longopts`. Při nezadání vstupního souboru, resp. výstupního souboru, za vstup se uvažuje standardní vstup `stdin` a za výstup `stdout`. Funkce parametru `-g` nebyla implementována.

#### Zpracování vstupního souboru

Načítání dat ze vstupního souboru se provede cyklem pomocí funkce ***fgetc***, která uloží data do proměnné. Následně se vytvoří `SimpleXmlElement` z načtených dat pomocí PHP funkce ***simplexml\_load\_string()*** a zavolá se funkce ***xml\_parsing***.

Funkce ***xml\_parsing*** je rekurzivní a má tři vstupné parametry, a to je vytvořený `SimpleXmlElement`, pole `array_elements`, do kterého se budou ukládat zpracované výsledky při parsování a pole se vstupními parametry. Cílem funkce je projít xml objekt, rozpoznat jednotlivé elementy, jejich podelementy a atributy, a uložit je do výsledné struktury `array_elements`.

Objekt se zpracovává pomocí třech cyklů *foreach*. První cyklus je hlavní, ze vstupního objektu načítá postupně elementy. Pro každý element se generuje sloupec daný jménem tabulky s předponou „PRK\_” a příponou „\_ID” reprezentující tzv. primární klíč. Pro každý primární klíč je nastaven datový typ INT. Druhý vnitřní cyklus slouží k zpracování atributů aktuálního elementu, pokud není nastaven parametr `-a`. Vytvoří se sloupce s názvy atributů a pomocí funkce ***set\_data\_type*** se nastaví datové typy. Třetí vnitřní cyklus je na zpracování podelementů aktuálně zpracovávaného elementu. V této části probíhá nastavování cizích klíčů pro daný

element a spočítávají se stejnojmenné podelementy. Maximální vypočtená hodnota se ukládá do položky *“count”* v poli *array\_elements*. Výpočet se provede tak, že při každém jednotlivém běhu funkce do pole *\$max\_sub* se zapíše maximální počet stejnojmenných podelementu v aktuálním elementu a porovná se s implicitně nastavenými hodnoty, a pak při následujícím zpracování stejnojmenného elementu výsledky se porovnají a uloží se větší.

*array\_elements* je třírozměrné pole, do kterého se ukládají data pro vytvoření tabulek. V podstatě celé výsledné pole už představuje tabulky vytvořené na základě načteného xml souboru. Pole bude obsahovat názvy tabulek, jednotlivé sloupce, datové typy sloupců a maximální počet stejnojmenných podelementů všech elementů.

## Datové typy

Určení datových typů se provádí pomocí funkce ***set\_data\_type***. Rozpoznání datového typu INT provádím pomocí funkcí ***is\_int(filter\_var(\$value, FILTER\_VALIDATE\_INT))*** a FLOAT pomocí funkcí ***is\_float(filter\_var(\$value, FILTER\_VALIDATE\_FLOAT))***.

Byl-li už nastaven datový typ, zavolá se funkce ***conflict\_dtype***, která vyřeší konflikt datových typů.

## Vytvoření tabulek

Po úplném zpracování vstupního souboru a vytvoření pole *array\_elements* se provede výpis jeho položek do výstupu pomocí funkce ***create\_output***.

## Zdroje informace:

- oficiální dokumentace PHP [<http://php.net/docs.php>]
- PHP 5 tutorial [<https://www.w3schools.com/php/>]