

# Opis szablonu SQL dla eksportu do systemów księgowych.

## Informacje dla programistów.

Plik szablonu jest poprawnym składniowo programem w języku T-SQL MS SqlServer 2005.

Program posiada 4 parametry wejściowe oraz 11 parametrów wyjściowych o ściśle określonych typach danych. Oprócz możliwości przekazywania parametrów musi posiadać pojedynczą instrukcję SELECT opisującą wyjściową tabelę eksportowanych danych. Kolumny wyjściowe muszą być nazwane (unikatowo), nawet jeśli zmienna @Header zostanie ustawiona na 0.

Wartości parametrów przekazywane są za pomocą znaków pytajnika ? (specyfikacja ADO 2.0), kolejność wystąpienia poszczególnych parametrów (a więc tym samym znaków pytajnika) jest ściśle określona.

| Lp | Nazwa zmiennej    | Typ danych   | Kierunek          | Znaczenie                                                                                             |
|----|-------------------|--------------|-------------------|-------------------------------------------------------------------------------------------------------|
| 1  | @Data             | datetime     | z programu (In)   | data pobierana z pola 'Data wydruku'                                                                  |
| 2  | @OdDaty           | datetime     | z programu (In)   | data pobierana z pola 'Od daty'                                                                       |
| 3  | @DoDaty           | datetime     | z programu (In)   | data pobierana z pola 'Do daty'                                                                       |
| 4  | @Filia            | tinyint      | z programu (In)   | znacznik pobierany z pola 'Filie'.<br>0 - wszystkie filie lub numeryczny identyfikator bieżącej filii |
| 5  | @FileName         | varchar(200) | do programu (Out) | nazwa wyjściowego pliku eksportu (bez ścieżki)                                                        |
| 6  | @Header           | bit          | do programu (Out) | 0 – bez nagłówka kolumn,<br>1 – z nagłówkiem kolumn                                                   |
| 7  | @Separator        | varchar(2)   | do programu (Out) | określa znakowy separator kolumn w pliku wyjściowym                                                   |
| 8  | @Delimiter        | varchar(1)   | do programu (Out) | określa znakowy ogranicznik danych tekstowych                                                         |
| 9  | @DateFormat       | varchar(20)  | do programu (Out) | określa formatowanie dat                                                                              |
| 10 | @DateDelimiter    | varchar(1)   | do programu (Out) | określa znakowy ogranicznik danych typu data                                                          |
| 11 | @DecimalSeparator | varchar(1)   | do programu (Out) | określa znak przed częścią dziesiętną liczby (kropka albo przecinek)                                  |
| 12 | @TrueValue        | varchar(10)  | do programu (Out) | określa znakową reprezentację logicznej prawdy                                                        |
| 13 | @FalseValue       | varchar(10)  | do programu (Out) | określa znakową reprezentację logicznego fałszu                                                       |
| 14 | @NullValue        | varchar(20)  | do programu (Out) | określa znakową reprezentację wartości NULL                                                           |
| 15 | @Encode           | varchar(20)  | do programu (Out) | Określa kodowanie polskich znaków.<br>NULL – Windows-1250<br>'MAZOVIA' – format Mazovia               |

### Przykład programu SQL:

```
DECLARE @OdDaty datetime, @DoDaty datetime, @Data datetime, @Filia tinyint
DECLARE @FileName varchar(200), @Header bit
DECLARE @Separator varchar(2), @Delimiter varchar(1)
DECLARE @DateFormat varchar(20), @DateDelimiter varchar(1), @DecimalSeparator varchar(1)
DECLARE @TrueValue varchar(10), @FalseValue varchar(10), @NullValue varchar(20), @Encode varchar(20)
SET @FileName = 'vat_r.txt'
SET @Header = 0
SET @Separator = char(9)
SET @Delimiter = ''
SET @DateFormat = 'yy/MM/dd'
SET @DateDelimiter = ''
SET @DecimalSeparator = '.'
SET @TrueValue = '1'
SET @FalseValue = '0'
SET @NullValue = ''
SET @Encode = 'MAZOVIA'
SET @Data = ?
SET @OdDaty = ?
SET @DoDaty = ?
SET @Filia = ?
SET ? = @FileName
SET ? = @Header
SET ? = @Separator
SET ? = @Delimiter
SET ? = @DateFormat
SET ? = @DateDelimiter
SET ? = @DecimalSeparator
SET ? = @TrueValue
SET ? = @FalseValue
SET ? = @NullValue
SET ? = @Encode
SELECT ... --tu właściwa instrukcja select
```

Powyższy przykład utworzy plik tekstowy rozdzielany *znakiem tabulacji* o nazwie *VAT\_r.txt* z tekstami ograniczonymi znakiem cudzysłowu " o dacie w formacie *"11/05/27"*, z separatorem dziesiętnym *123.20* gdzie logiczna prawda reprezentowana jest tekstem *1* a logiczny fałsz reprezentowany jest znakiem *0* a wartości *null* znakiem pustym. Kodowanie polskich znaków jest w formacie *Mazovia*

### Kodowanie liczb zmiennoprzecinkowych.

Typ `money` formatowany jest w pliku wyjściowym z dwoma miejscami dziesiętnymi (`format: #0.00`). Pozostałe typy zmiennoprzecinkowe formatowane są z czterema miejscami dziesiętnymi (`format: #0.0000`), przy czym znak kropki dziesiętnej definiowany jest w zmiennej/parametrze `@DecimalSeparator`

### Wartości domyślne parametrów.

Ustawienie zmiennej parametru wyjściowego na wartość NULL spowoduje zastosowanie przez program wartości domyślnej dla tego parametru (nie dotyczy parametru `@FileName`, parametr ten jest wymagany.)

Lista wartości domyślnych:

```
@Header = 0
@Separator = char(9)
@Delimiter = ''
@DateFormat = 'yyyy-MM-dd'
@DateDelimiter = ''
@DecimalSeparator = ','
@TrueValue = '1'
@FalseValue = '0'
@NullValue = ''
@Encode = 'Windows-1250'
```

### Lokalizacja plików szablonu SQL.

Pliki szablonów powinny być umieszczone w katalogu `c:\motoFirma\khSQL\`

Nazwa pliku powinna mieć rozszerzenie `.sql`

Plik powinien być zapisany z kodowaniem `ANSII`.