

Príklad práce s tabulkou (akcie DB_ ...)

V příklade sa predpokladá existenciu objektov [SD.DbStruct](#), [TB.Table](#). alej musí by spustený príslušný proces [D2000 DBManager](#) a v operanom systéme je v ODBC korektne konfigurovaná databáza s príslušným [DSN](#) menom.

```

INT _handle ; handle pre prístup k tabuके
INT _retCode
TEXT _where
RECORD (SD.DbStruct)_dbRow ; pomocné premenné
RECORD (SD.DbStruct) _dbRowSave

; otvorenie tabuके pre modifikáciu
DB_CONNECT TB.Table, _DB_MODIFY, _handle, _retCode
IF _retCode # _ERR_NO_ERROR THEN ; nepodarilo sa spoji s tabuके
    END
ENDIF

; ítanie podla kľúovej položky
_dbRow[1]^Int := 1
DB_READ _handle, _dbRow[1], _retCode
IF _retCode # _ERR_NO_ERROR THEN ; riadok sa v tabuके nenachádza
    END
ENDIF

; odloží si pôvodný riadok z tabuके do pomocnej premennej
SET _dbRowSave[1] WITH _dbRow[1] ; zmením text
_dbRow[1]^Text := _dbRow[1]^Text + "zmena"
; zapíšem riadok naspä do tabuके podla kľúovej položky
DB_UPDATE _handle, _dbRow[1], _retCode
IF _retCode # _ERR_NO_ERROR THEN ; nepodaril sa update
    END
ENDIF

; znova preítam riadok, ale cez klauzulu WHERE
_where := "Int="+%IToStr(_dbRowSave[1]^Int)
DB_READ _handle, _dbRow[1], _retCode WHERE _where
IF _retCode # _ERR_NO_ERROR THEN ; musí tam by
    END
ENDIF

; znova preítam riadok, ale cez klauzulu WHERE s parametrizovanou podmienkou
_where := "Int=#PAR#"
DB_READ _handle, _dbRow[1], _retCode WHERE _where BINDIN _dbRowSave[1]^Int
IF _retCode # _ERR_NO_ERROR THEN ; musí tam by
    END
ENDIF

; zruším ho
DB_DELETE _handle, _dbRow[1], _retCode
IF _retCode # _ERR_NO_ERROR THEN ; musí tam by
    END
ENDIF

; alebo cez klauzulu WHERE
_where := "Int="+%IToStr(_dbRowSave[1]^Int)
;DB_DELETE _handle, _retCode WHERE _where
;IF _retCode # _ERR_NO_ERROR THEN ; musí tam byt
; END
;ENDIF

; alebo cez klauzulu WHERE a parametrizovanu podmienku
_where := "Int=#PAR#"
;DB_DELETE _handle, _retCode WHERE _where BINDIN _dbRowSave[1]^Int
;IF _retCode # _ERR_NO_ERROR THEN ; musí tam byt
; END
;ENDIF

```

```

; vložím pôvodný obsah
DB_INSERT _handle, _dbRowSave[1], _retCode
IF _retCode # _ERR_NO_ERROR THEN ; malo by sa to podari
    END
ENDIF
; zatvorím tabuku
DB_DISCONNECT _handle

```

V nasledujúcom príklade sa predpokladá existencia objektov [SD.BlobStruct](#), [TB.BlobTable](#). alej musí by spustený príslušný proces [D2000 DBManager](#) a v operanom systéme je v ODBC korektne konfigurovaná databáza s príslušným [DSN](#) menom. Tabuka *BlobTable* v databáze musí ma okrem stpcov *ID* a *Meno* (ktoré sú definované v [SD.BlobStruct](#)) aj stpec *MyFile* typu BLOB (Binary Large Object). Databáza Sybase tento typ nazýva *LONG BINARY*, Oracle pre zmenu *BLOB* a MS SQL Server zase *IMAGE*. Kompatibilita je skrátka skvelá vec.

```

INT _handle ; handle pre prístup k tabuke
INT _retCode
TEXT _where
RECORD (SD.BlobStruct)_dbRow ; pomocná premenná

; otvorenie tabuky pre modifikáciu
DB_CONNECT TB.BlobTable, _DB_MODIFY, _handle, _retCode
IF _retCode # _ERR_NO_ERROR THEN; nepodarilo sa spoji s tabukou
    END
ENDIF

; vložím do tabuky riadok
_dbRow^ID := 99
_dbRow^Meno := "My inserted file"
DB_INSERT _handle, _dbRow, _retCode
IF _retCode # _ERR_NO_ERROR THEN ; malo by sa to podari
    END
ENDIF

; vložím do tabuky súbor ako BLOB objekt
DB_UPDATE_BLOB _handle, "MyFile", "C:\File99.txt", _retCode, 99
IF _retCode # _ERR_NO_ERROR THEN ; malo by sa to podari
    END
ENDIF

; načítam z tabuky iný súbor (z riadku s ID=11) a uloží ho na disk C:\
DB_READ_BLOB _handle, "MyFile", "C:\File11_new.txt", _retCode, 11

; keby nebol stpec ID definície štruktúry SD.BlobStruct definovaný
; ako kúový, bolo by treba špecifikova podmienku WHERE:
DB_READ_BLOB _handle, "MyFile", "C:\File11_new.txt", _retCode WHERE "ID=11"

; zatvorím tabuku
DB_DISCONNECT _handle

; práca bez otvárania tabuky (funkcie DBS_)

; načítam alší súbor bez otvárania tabuky
DBS_READ_BLOB TB.BlobTable, "MyFile", "C:\File22_new.txt", _retCode, 22

; uloží ho do tabuky
DBS_UPDATE_BLOB TB.BlobTable, "MyFile", "C:\File22_new.txt", _retCode, 33

```



Súvisiace stránky:

[Akcie v skriptoch](#)