DB_DELETE

Akcia DB_DELETE a DBS_DELETE

Funkcia

Zrušenie jedného alebo viacerých riadkov z tabuky.

Deklarácia

```
 DB_DELETE handleIdent_Int, rowIdent,retCodeIdent_Int [ORAHINT
hintIdent_Str]

 DBS_DELETE dbObjIdent, rowIdent, retCodeIdent_Int [TRANS transHandle_Int]
[ORAHINT hintIdent_Str]
```

alebo

```
 DB_DELETE handleIdent_Int, retCodeIdent_Int WHERE strExpression_Str
[BINDIN varIdent1, varIdent2, ... ] [ORAHINT hintIdent_Str]

 DB_DELETE handleIdent_Int, retCodeIdent_Int WHERE strExpression_Str
[BINDIN structRowIdent] [ORAHINT hintIdent_Str]

 DBS_DELETE dbObjIdent, retCodeIdent_Int  WHERE strExpression_Str [BINDIN
varIdent1, varIdent2, ... ] [TRANS transHandle_Int] [ORAHINT
hintIdent_Str]

 DBS_DELETE dbObjIdent, retCodeIdent_Int  WHERE strExpression_Str [BINDIN
structRowIdent] [TRANS transHandle_Int] [ORAHINT hintIdent_Str]
```

Parametre

handleIdent_Int	in	Identifikátor typu *Int* (handle) spojenia s tabukou (DB_CONNECT).
dbObjIdent	in	Odkaz na objekt typu Tabuka.
retCodeIdent_Int	out	Identifikátor návratového kódu.
rowIdent	in	Identifikátor jedného riadku štruktúry.
strExpression_Str	in	Výraz typu String, ktorý identifikuje rušené riadky. Pokia je výraz parametrizovaný, musí by uvedené aj kúové slovo **BINDIN** nasledované hodnotami parametrov (*structRowIdent* alebo *varIdent1*, *varIdent2*, ...).
varIdent1, varIdent2, ...	in	Zoznam objektov, konštánt alebo lokálnych premenných, ktoré sa použijú na mieste parametrov parametrizovaného výrazu *strExpression_Str*.
structRowIdent	in	Odkaz na riadok lokálnej premennej typu *Record* alebo na riadok štruktúrovanej premennej. Hodnoty z tohto riadku sa použijú na mieste parametrov parametrizovaného SQL výrazu *strExpression_Str*.
transHandle_Int	in	Identifikátor Pripojenia na databázu.
hintIdent_Str	in	Výraz typu *String*, ktorý definuje Oracle SQL hint. Ten je použitý ako inštrukcia pre optimalizátor výkonu SQL príkazu. Zapisuje hodnotu bez obaujúcich znakov /*+ <*orahint*> */. Príklad použitia je uvedený tu.

Návratový kód

Hodnota parametra *retCodeIdent_Int* - pozri tabuku chybových stavov. Je možné získa rozšírenú informáciu o chybe.

Popis

Tabuka musí by otvorená s prístupom *_DB_MODIFY*.

V prvom prípade je rušený riadok urený hodnotou kúového políka v hodnote typu Štruktúra (*rowIdent*). Poda neho sa v tabuke nájde príslušný riadok a zruší sa.

V druhom prípade je podmienkou pre rušenie výraz typu *String* (*strExpression_Str*), ktorého výsledná hodnota je použitá ako klauzula **WHERE** pre SQL príkaz **DELETE**. V tomto prípade je hodnota kúovej

položky v štruktúre nepoužitá a nemusí by nastavená. Ak je v tabuke viacej riadkov ktoré vyhovujú podmienke (*strExpression_Str*), budú zrušené všetky.

Výhodou akcie **DBS_DELETE** pri práci s tabukou je možnos vynecha jej otvorenie a zatvorenie (jednoduchší zápis).

Pre D2000 verziu 5: Nevýhoda akcie **DBS_DELETE** je v rýchlosti. Každé volanie akcie **DBS_DELETE** má za následok v rámci jej obsluhy v procese D2000 DBManager nutnos databázu otvori a následne zatvori, o môže by asovo náročná innos a predstavuje z databázového hadiska pomerne neštandardný postup. Nutnos otvárania a zatvárania databázy je možné potlai v rámci transakného spracovania tak, že za príkaz uvedieme parameter **TRANS**.

Pre D2000 verziu 6 a vyššie: Optimalizácia procesu **D2000 DBManager** (recyklovanie spojení, predpripravené spojenia) spôsobila, že akcia **DBS_DELETE** sa vykoná rovnako rýchlo ako akcia **DB_DELETE** a navyše sa ušetrí as potrebný na spracovanie explicitného príkazu na otvorenie databázy **DB_DELETE**.

Upozornenie

Variant príkazu

```
DB_DELETE WHERE
```

umožuje zruši celý obsah tabuky.

Ak je podmienka (*strExpression_Str*) splnená pre všetky riadky tabuky (napríklad "1=1"), bude zmazaný jej celý obsah.

Príklad

Príklad práce s tabukou (akcie DB_ ...).

Súvisiace odkazy

DB_CONNECT
DB_DISCONNECT
DB_INSERT
DB_INSUPD
DB_READ
DB_READ_BLOB
DB_UPDATE
DB_UPDATE_BLOB

DB_TRANS_OPEN
DB_TRANS_COMMIT
DB_TRANS_ROLLBACK
DB_TRANS_CLOSE

Všetky databázové akcie

Súvisiace stránky:

Akcie v skriptoch