

DBS_READ

Akcia DB_READ a DBS_READ

Funkcia

ítanie jedného alebo viacerých riadkov z tabučky.

Deklarácia

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

```
DB_READ handleIdent_Int, rowIdent, retCodeIdent_Int [WHERE  
strExpression_Str [BINDIN structRowIdent ]] [ORAHINT hintIdent_Str]
```

```
DBS_READ dbObjIdent, rowIdent, retCodeIdent_Int [WHERE strExpression_Str  
[BINDIN varIdent1, varIdent2, ... ]] [TRANS transHandle_Int] [ORAHINT  
hintIdent_Str]
```

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

alebo

```
DB_READ handleIdent_Int, localStructIdent, retCodeIdent_Int, maxRows_Int  
[WHERE strExpression_Str [BINDIN varIdent1, varIdent2, ... ]][ORAHINT  
hintIdent_Str]
```

```
DB_READ handleIdent_Int, localStructIdent, retCodeIdent_Int, maxRows_Int  
[WHERE strExpression_Str [BINDIN structRowIdent ]][ORAHINT hintIdent_Str]
```

```
DBS_READ dbObjIdent, localStructIdent,retCodeIdent_Int, maxRows_Int [WHERE  
strExpression_Str [BINDIN varIdent1, varIdent2, ... ]][TRANS  
transHandle_Int] [ORAHINT hintIdent_Str]
```

```
DBS_READ dbObjIdent, localStructIdent,retCodeIdent_Int, maxRows_Int [WHERE  
strExpression_Str [BINDIN structRowIdent ]][TRANS transHandle_Int]  
[ORAHINT hintIdent_Str]
```

Parametre

handleIdent_Int	in	Identifikátor typu <i>Int</i> - (handle) spojenia s tabuľkou (DB_CONNECT).
dbObjectIdent	in	Odkaz na objekt typu <i>Tabuľka</i> .
rowIdent_Int	in / out	Identifikátor jedného riadku štruktúry (výsledok itania).
returnCodeIdent_Int	out	Identifikátor návratovej hodnoty - úspešnosť akcie.
localStructureIdent	out	Identifikátor celej lokálnej štruktúry.
maxRows_Int	in	Identifikátor - maximálny počet itaných riadkov.
strExpression_Str	in	Výraz, ktorý definuje itaný riadok tabuľky (riadky). Výraz môže obsahovať aj podmienku triedenia (napr. " <i>ID<100 ORDER BY Name</i> "). Pokiaľ je výraz parametrizovaný , musí byť uvedený aj kľúčové slovo BINDIN nasledované hodnotami parametrov (<i>structRowIdent</i> alebo <i>varIdent1</i> , <i>varIdent2</i> , ...).
varIdent1, varIdent2, ...	in	Zoznam objektov, konštánt alebo lokálnych premenných , ktoré sa použijú na mieste parametrov parametrizovaného výrazu <i>strExpression_Str</i> .
structRowIdent	in	Odkaz na riadok lokálnej premennej typu <i>Record</i> alebo na riadok štruktúrovanej premennej . Hodnoty z tohto riadku sa použijú na mieste parametrov parametrizovaného SQL výrazu <i>strExpression_Str</i> .
transHandle_Int	in	Identifikátor Pripojenia na databázu.
hint_Ident_Str	in	Výraz typu <i>String</i> , ktorý definuje Oracle SQL hint. Ten je použitý ako inštrukcia pre optimalizátor výkonu SQL príkazu. Zapisuje hodnotu bez úvodných a koncových znakov <i>/*+ <orahint> */</i> . Príklad použitia je uvedený tu .

Návratový kód

Hodnota parametra *returnCodeIdent_Int* - pozri tabuľku [chybových stavov](#). Je možné získať [rozšírenú informáciu o chybe](#).

Popis

Klauzula **WHERE** pre SQL príkaz **SELECT**, ktorý vykoná výber z tabuľky, je definovaná buď hodnotou výrazu *strExpression_Str*, alebo, ak táto hodnota nie je zadaná, tak hodnotami definovaných (platných) kľúčových položiek (položky označené ako [kľúčové](#) pri definícii objektu typu *Tabuľka*) v prvom riadku štruktúry (*rowIdent* alebo *localStructIdent[1]*). Nastavenie týchto hodnôt musí byť realizované pred akciou **DB_READ (DBS_READ)**.

Pozn.: ak nie je zadaná ani podmienka, ani tabuľka nemá kľúčové stĺpce, načítajú sa všetky dáta z tabuľky (ak ich nie je viac ako špecifikované maximum).

Pozn.: ak sú itané dáta do jedného riadku štruktúry (verzie s parametrom *rowIdent_Int*) s použitím kľúčových položiek, tak hodnoty všetkých kľúčových položiek v *rowIdent_Int* musia byť platné, inak sa itanie neuskutí, skoní s chybou a [%GetLastExtErrorCode\(\)](#) vráti chybu 667.

Z toho vyplýva, že klauzula **WHERE** nie je súčasťou SQL príkazu **SELECT**, ak nie je definovaná hodnota výrazu *strExpression_Str* a

- neexistujú kľúčové položky (položky označené ako [kľúčové](#) pri definícii objektu typu *Tabuľka*),
- alebo nie je definovaná (platná) ani jedna z hodnôt kľúčových položiek,

- alebo štruktúra má nulovú dimenziu (v prípade *localStructIdent*).

Prvá forma zápisu akcie umožňuje preíta jeden riadok z tabuky do jedného riadku štruktúry.

Ak pri výbere vyhovuje podmienke pre výber viac ako jeden riadok tabuky, akcia **DB_READ(DBS_READ)** naíta prvý riadok. Ak nevyhovuje ani jeden riadok, akcia vráti chybu prostredníctvom *retCodeIdent_Int*. Typ štruktúry (*rowIdent*) musí byť zhodný s typom tabuky.

Druhá forma zápisu umožňuje preíta jeden, alebo viac riadkov z tabuky. Maximálny počet riadkov určuje hodnota parametra *maxRows*. Ak hodnota *maxRows* je nastavená na -1, systém sa snaží preíta všetky riadky zodpovedajúce zadanej podmienke. Maximálny počet riadkov je obmedzený konfiguránnym parametrom databázy [Maximum vrátených riadkov](#). Pri prekroení tohto potu riadkov bude do lokálnej štruktúry *localStructIdent* naítaných prvých [Maximum vrátených riadkov](#) a v *retCodeIdent_Int* sa vráti chyba [_ERR_DATABASE_ROWS_LIMIT](#).

Ak je hodnota *maxRows* väčšia ako [Maximum vrátených riadkov](#), naíta sa najviac [Maximum vrátených riadkov](#) riadkov. Pokiaľ je k dispozícii viac riadkov, v *retCodeIdent_Int* sa vráti chyba [_ERR_DATABASE_ROWS_LIMIT](#).

Ak je hodnota *maxRows* nastavená na -2, systém najskôr zistí počet riadkov zodpovedajúcich danej podmienke. Ak ich je 0 alebo viac ako [Maximum vrátených riadkov](#), nenaíta žiadne riadky, v opakom prípade naíta zistený počet riadkov. Tento spôsob môže byť z optimalizačných dôvodov rýchlejší ako pre *maxRow=-1*. Počet riadkov sa zisuje pomocou výberu "SELECT COUNT(*) FROM table_name [WHERE where_condition]".

Ak je zadaná podmienka *StrExpression_Str*, pred zistením potu riadkov sa ešte zisuje, či podmienka obsahuje klauzulu **ORDER BY**. Ak klauzulu obsahuje, táto sa aj s nasledujúcim textom odstráni. Dôvodom je, že pri zisovaní potu riadkov napr. Sybase neakceptuje "SELECT COUNT(*) FROM aaaa WHERE bbbb = cccc ORDER BY dddd" a vracia chybu *Function or column reference to 'dddd' in the select list must also appear in a GROUP BY*.

Parameter *localStructIdent* musí predstavovať meno lokálnej štruktúry zodpovedajúceho typu (t.j. rovnakého, aký je použitý v konfigurácii objektu *dbObjIdent* typu [Tabuka](#)). Pokiaľ chcete mať istotu, že výstup nebude riadený klauzulou **WHERE** vyskladanou na základe kúových položiek, je vhodné pred volaním akcie **DB_READ(DBS_READ)** redimenzovať lokálnu štruktúru na 0.

Po výbere riadkov z tabuky sú tieto prenesené do premennej *localStructIdent*. V prípade potreby je jej zmenený rozmer (**REDIM**). Ak výberovej podmienke nevyhovuje ani jeden riadok, akcia sa ukoní úspešne (*retCodeIdent_Int* = [_ERR_NO_ERROR](#)) a rozmer štruktúry *localStructIdent* sa zmení na 0.

Výhodou akcie **DBS_READ** pri práci s tabukou je možnosť vynechať jej otvorenie a zatvorenie (jednoduchší zápis).

Pre D2000 verziu 5: Nevýhoda akcie **DBS_READ** je v rýchlosti. Každé volanie akcie **DBS_READ** má za následok, v rámci jej obsluhy v procese [D2000 DBManager](#), nutnosť databázu otvoriť a následne zatvoriť, o čom môže byť asovo náročné. Predstavuje z databázového hľadiska pomerne neštandardný postup. Nutnosť otvárania a zatvárania databázy je možné potlačiť v rámci transakčného 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_READ** sa vykoná rovnako rýchlo ako **DB_READ** a navyše sa ušetrí čas potrebný na spracovanie explicitného príkazu na otvorenie databázy **DB_CONNECT**.

Príklad

Súvisiace odkazy

[Príklad práce s tabukou \(akcie DB_ ...\).](#)

[DB_CONNECT](#)
[DB_DELETE](#)
[DB_DISCONNECT](#)
[DB_INSERT](#)
[DB_INSUPD](#)
[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](#)