

# DB\_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, ... ]]  
[TRANStransHandle_Int] [ORAHINT hintIdent_Str]
```

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

### Parametre

handleIdent_Int	in	Identifikátor typu <i>Int</i> - (handle) spojenia s tabuľkou ( <b>DB_CONNECT</b> ).
dbObjectIdent	in	Odkaz na objekt typu <i>Tabuľka</i> .
rowIdent_Int	in/out	Identifikátor jedného riadku štruktúry (výsledok íťania).
resultCodeIdent_Int	output	Identifikátor návratovej hodnoty - úspešnosť akcie.
localStructIdent	output	Identifikátor celej lokálnej štruktúry.
maxRows_Int	in	Identifikátor - maximálny počet íťaných riadkov.
strExpression_Str	in	Výraz, ktorý definuje íťaný riadok tabuľky (riadky). Výraz môže obsahovať aj podmienku triedenia (napr. " <i>ID&lt;100 ORDER BY Name</i> "). Pokiaľ je výraz <b>parametrizovaný</b> , musí byť uvedený aj kľúčové slovo <b>BIND</b> nasledované hodnotami parametrov ( <i>structRowIdent</i> alebo <i>varIdent1</i> , <i>varIdent2</i> , ...).
varIdent1, varIdent2, ...	in	Zoznam objektov, konštánt alebo <b>lokálnych premenných</b> , ktoré sa použijú na mieste parametrov <b>parametrizovaného</b> výrazu <i>strExpression_Str</i> .
structRowIdent	in	Odkaz na riadok lokálnej premennej typu <i>Record</i> alebo na riadok <b>štruktúrovanej premennej</b> . Hodnoty z tohto riadku sa použijú na mieste parametrov <b>parametrizovaného</b> SQL výrazu <i>strExpression_Str</i> .
transHandle_Int	in	Identifikátor <b>Pripojenia</b> na databázu.
hintIdent_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>"/+ &lt;orahint&gt; */</i> . Príklad použitia je uvedený <b>tu</b> .

## Návratový kód

Hodnota parametra *resultCodeIdent\_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**).

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 prečítať jeden riadok z tabuľky do jedného riadku štruktúry.

Ak pri výbere vyhovuje podmienke pre výber viac ako jeden riadok tabuľky, akcia **DB\_READ(DBS\_READ)** načítava prvý riadok. Ak nevyhovuje ani jeden riadok, akcia vráti chybu prostredníctvom *resultCodeIdent\_Int*. Typ štruktúry (*rowIdent*) musí byť zhodný s typom tabuľky.

Druhá forma zápisu umožňuje prečítať jeden, alebo viac riadkov z tabuľky. Maximálny počet riadkov určuje hodnota parametra *maxRows*. Ak hodnota *maxRows* je nastavená na -1, systém sa snaží prečítať všetky riadky zodpovedajúce zadanej podmienke. Maximálny počet riadkov je obmedzený konfiguračným parametrom databázy **Maximum vrátených riadkov**. Pri prekroení tohto počtu riadkov bude do lokálnej štruktúry *localStructIdent* načítaných prvých **Maximum vrátených riadkov** a v *resultCodeIdent\_Int* sa vráti chyba **\_ERR\_DATABASE\_ROWS\_LIMIT**.

Ak je hodnota *maxRows* väčšia ako **Maximum vrátených riadkov**, načítava sa najviac **Maximum vrátených riadkov** riadkov. Pokiaľ je k dispozícii viac riadkov, v *resultCodeIdent\_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 opanom prípade naíta zistený počet riadkov. Tento spôsob môže byť z optimalizovaných 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 počtu 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 počtu 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 to 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\\_INSERTUPD](#)  
[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](#)