

DB_SET_PROCESS_PARAMS

Akcia DB_SET_PROCESS_PARAMS

Funkcia

Akcia pre nastavenie hodnôt pomenovaných parametrov, alebo zrušenie parametrov "kontextu".

Deklarácia

```
DB_SET_PROCESS_PARAMS paramsRec
```

Parametre

paramsRec	in	Identifikátor celej lokálnej štruktúry . Štruktúrovaná hodnota má definované dva stpce: <ul style="list-style-type: none">• paramName : String(64)• paramValue : String(64)
-----------	----	--

Hodnota **paramName** definuje názov parametra. Hodnota **paramValue** definuje hodnotu parametra. Prípustné kombinácie hodnôt a ich interpretáciu znázorňuje nasledujúca tabuľka.

Vplyv na výsledok	paramName	paramValue
Vloží alebo aktualizuje hodnotu parametra	zadané	zadané
Zruší parameter	zadané	(neplatná hodnota)
Ignozuje (nie je súčasťou požiadavky na zmenu)	(neplatná hodnota)	zadané
Ignozuje (nie je súčasťou požiadavky na zmenu)	(neplatná hodnota)	(neplatná hodnota)

Upozornenie: Ak nastane situácia, že názvy parametrov sa pri jednom volaní akcie opakujú, tak dochádza k nedefinovanému správaniu!

Popis

"Kontext" je jednoznačne priradený **skupine procesov**. Skupina procesov je tvorená:

- vždy samostatným procesom typu [D2000 EventHandlerler](#) (bez použitia parametra `--batch_mode`) alebo
- v prípade aktívnych schém trojicou procesov [D2000 HI](#) (HIP), [HIS](#) a [D2000 Server Event Executor](#) (SEE).

Ak je proces **D2000 EventHandlerler** spustený s parametrom `--batch_mode`, tak "Kontext" je jednoznačne priradený skupine eventov, ktoré sú otvorené s rovnakým íslom inštancie ([OPENEVENT](#)), nie celému procesu. Parameter umožní beh viacerých eventov na jednom Event Handleri, pričom každý môže používať vlastný "Kontext". V prípade rovnakých ísel inštancií je "Kontext" medzi nimi zdieľaný (je spoločný) a podmienky vykonávania databázových operácií sú preto rovnaké.

Kontext predstavuje skupinu jednoznačne pomenovaných parametrov, pričom každý parameter má definovanú hodnotu. Vzniká prvým použitím akcie **DB_SET_PROCESS_PARAMS**. Spravuje ho proces [D2000 Server](#). Kontext zaniká ukončením posledného procesu z príslušnej skupiny. Obsah kontextu je automaticky replikovaný na všetky bežiacie procesy [D2000 DBManager](#).

D2000 DbManager pred vykonaním ubovoného databázového príkazu, ktorý bol iniciovaný v rámci kontextu, vkladá jeho obsah do tabuľky `D2000_PROCESS_PARAMS`. Meno parametra vkladá do stpca `PARAM_NAME` a jeho hodnotu do stpca `PARAM_VALUE`. Tabuľku `D2000_PROCESS_PARAMS` DbManager vytvára pri jej prvom použití.

Poznámka 1

Akcii je možné využiť na tvorbu parametrizovaných pohadov, pričom parametre pre pohad pochádzajú práve z tabuľky `D2000_PROCESS_PARAMS`, ktorú automaticky nastavuje proces DbManager podľa kontextu, v ktorom je databázový príklad použitý.

Poznámka 2

Tabuľka `D2000_PROCESS_PARAMS` je Global Temporary Table. Jednotlivé sessions v nej vidia iba riadky, ktoré tam zapísali. Po zrušení session, sa riadky viazané na danú session z tabuľky automaticky odstránia. Bližší popis, ako aj jej DLL, je v dokumentácii ku Global Temporary Table.

Príklad

```

; príklad overí výsledok akcie DB_SET_PROCESS_PARAMS preítaním tabuky
D2000_PROCESS_PARAMS
; insert/update parametra viazaného na proces (aktualizuje tabuku
D2000_PROCESS_PARAMS)

RECORD NOALIAS (SD.Params) _params

TEXT _cmd ; SQL select
INT _retCode ; návratová hodnota

_params[1]^PARAM_NAME := "NAME"
_params[1]^PARAM_VALUE := "SELF.DBM"

DB_SET_PROCESS_PARAMS _params
REDIM _params[0]

_cmd := "SELECT PARAM_NAME, PARAM_VALUE FROM D2000_PROCESS_PARAMS WHERE
PARAM_NAME = 'NAME'"

; preíť obsah tabuky
SQL_SELECT DBC_TEST_DSN, _retCode, 1000, _cmd BIND _params
IF _retCode = _ERR_NO_ERROR THEN
; štruktúra _params je naplnená, obsahuje jeden riadok
ELSE
; chyba
ENDIF

```

Súvisiace odkazy

[Všetky databázové akcie](#)



Súvisiace stránky:

[Akcie v skriptoch](#)