

# SET AS [DIRECT]

## Akcia SET AS [DIRECT]

### Funkcia

Zmena priradenia (asociovania) objektu s:

- [lokálnou premennou](#) typu *ALIAS*,
- položkou štruktúry typu *Objekt* (odkaz na objekt):
  - pre lokálne premenné typu *Record* alebo typový *ALIAS*,
  - pre objekty typu [Štruktúrovaná premenná](#).

### Deklarácia

```
SET dstIdent AS srcIdent [DIRECT]
```

### Parametre

dstIdent	in	Identifikátor hodnoty typu <i>ALIAS</i> (alebo typu <i>Objekt</i> v prípade položky štruktúry), <a href="#">identifikátor riadku</a> alebo <a href="#">celej štruktúry</a> - cie hodnôt a asociácií.
srcIdent	in	Identifikátor hodnoty typu <i>ALIAS</i> (alebo typu <i>Objekt</i> v prípade položky štruktúry), <a href="#">identifikátor riadku</a> alebo <a href="#">celej štruktúry</a> - zdroj hodnôt a asociácií.

### Popis

Typ štruktúr definovaných parametrami *dstIdent* a *srcIdent* musí by vždy rovnaký (iná akcia generuje chybu `_ERR_RECORD_NO_COMP`) a taktiež musia mať rovnaký rozmer. Môžu byť typu:

- Record
- ALIAS
- objekt typu [Štruktúrovaná premenná](#)

Akcia nastavuje hodnoty položiek v *dstIdent\_Rec* hodnotami zo zodpovedajúcich položiek *srcIdent\_Rec*. V prípade, že *dstIdent\_Rec* je odkaz na riadok, *srcIdent\_Rec* musí byť taktiež odkaz na riadok. Podobne, ak *dstIdent\_Rec* je odkaz na celú hodnotu, tak aj *srcIdent\_Rec* musí byť odkaz na celú hodnotu.

Akcia má podobné pravidlá ako akcia [SET WITH](#).

- Pre položky štruktúr iného typu ako *Objekt* (odkaz na objekt) má aj rovnakú funkciu.
- Pre položku typu *Objekt* však nemá význam priradenia hodnoty, ale zmenu asociovaného (pripojeného) objektu.

Akcia umožňuje použiť index riadku 0 v nasledovnom prípade:

SET *dstIdent*[0]^Col1 AS *srcIdent*[0]^Col2 priom

1. *dstIdent* je lokálna štruktúrovaná premenná a *Col1* je typu *Objekt*
2. *srcIdent* je:

- lokálna štruktúrovaná premenná a *Col2* je typu *Objekt*,
- typový *ALIAS* (*ALIAS* (structDef) *\_rA*) a *Col2* je typu *Objekt*,
- objekt typu štruktúrovaná premenná; je možné použiť kúové slovo *DIRECT*, (ak *Col2* nie je typu odkaz na objekt, *DIRECT* je povinný).

Akcia interne vykoná predpísané priradenie (SET AS) pre všetky riadky daného stpca. Preto je potrebné, aby rozmery oboch štruktúr (*dstIdent* aj *srcIdent*) boli rovnaké.

### Príklad

```
ALIAS _a
INT _i

SET _a AS U.INT

; prvý test
IF U.Int = _a THEN
ENDIF

; druhý test
_i := U.Int
IF U.Int = _i THEN
ENDIF
```

Prvý test na zhodu hodnôt bude vždy pravdivý, lebo lokálna premenná `_a` typu *ALIAS* je "nasmerovaná" na objekt *U.Int*.

Výsledok druhého testu závisí od toho, i objekt *U.Int* nezmenil svoju hodnotu od momentu jej priradenia do `_i` a vykonania testu.

Zmena hodnoty:

```
ALIAS _a

SET _a AS U.INT
_a := 1
```

Priradenie je skriptom interpretované ako priradenie hodnoty 1 do *U.Int* !!!

Vlastnosti lokálnej premennej typu *ALIAS* sú zhodné pre položky typu *Objekt*:

- objektov typu *Štruktúrovaná premenná*,
- lokálnych premenných typu *Record* (bez modifikátora **NOALIAS**).

Napríklad:

```
ALIAS _a

SET _a AS U.Int
_a := 1 ; prvé priradenie

RECORD (SD.RecordDef) _lArr1
SET _lArr1[1]^Object AS U.Int
_lArr1[1]^Object := 1 ; druhé priradenie
```

Prvé aj druhé priradenie majú vzhľadom na objekt *U.Int* rovnaký účinok.

Zmena smerovania položky typu *Objekt* objektu typu *Štruktúrovaná premenná*:

```
SET SV.Struktura[1]^Object AS U.Int
```

Akcia nasmeruje položku *SV.Struktura[1]^Object* na objekt *U.Int*. Od tohoto okamihu bude hodnota položky vždy zhodná s hodnotou objektu *U.Int*.

Priradenie:

```
SV.Struktura[1]^Object := 1
```

má za následok (podobne ako pri predchádzajúcich prípadoch) priradenie hodnoty 1 do objektu *U.Int*.

```
RECORD (SD.RecordDef) _lArr1

REDIM _lArr1[2]
SET _lArr1[1]^Object AS U.Int ; priradenie 1
SET _lArr1[2]^Object AS U.Int2 ; priradenie 2
SET _lArr1[1]^Object AS _lArr1[2]^Object ; priradenie 3
```

Význam priradení 1 a 2 bol vysvetlený v predchádzajúcom texte. Priradenie 3 má v danom kontexte význam zhodný s akciou **SET \_lArr1[1]^Object AS U.Int2**.

Alebo:

```

SET SV.Struktura[1]^Object AS U.Int                ; priradenie 1
WAIT
SET SV.Struktura[2]^Object AS U.Int2              ; priradenie 2
WAIT
SET SV.Struktura[1]^Object AS SV.Struktura[2]^Object ; priradenie 3
WAIT

```

Podobne ako v predchádzajúcom prípade, posledné priradenie je zhodné s priradením **SET SV.Struktura[1]^Object AS U.Int2**. Ako vidie z príkladu, ide o kopírovanie odkazu(ov), ktoré sú obsiahnuté na pravej strane (identifikátor *srcIdent*).

Takáto konštrukcia neumožňuje nasmerova (vytvori odkaz) políko štruktúry na iné políko štruktúry. Preto bol zavedený modifikátor **DIRECT**, ktorý toto umožňuje. Akcia **SET AS DIRECT** obmedzuje možné typy parametrov na:

- *dstIdent* - políko typu *Objekt* [lokálnej premennej](#) typu *Record*, lokálnej premennej *typový ALIAS* alebo objektu typu [Štruktúrovaná premenná](#).
- *srcIdent* - políko objektu typu [Štruktúrovaná premenná](#).

Napríklad:

```

RECORD (SD.RecordDef) _lArr1

REDIM _lArr1[2]
SET _lArr1[1]^Object AS U.Int DIRECT                ; priradenie 1
SET _lArr1[2]^Object AS U.Int2 DIRECT              ; priradenie 2
SET _lArr1[1]^Object AS _lArr1[2]^Object DIRECT    ; priradenie 3
SET _lArr1[1]^Object AS SV.Struktura[2]^Int DIRECT ; priradenie 4

```

Priradenia 1, 2, 3 sú neprípustné. Priradenie 4 nasmeruje odkaz na políko hodnoty objektu typu [Štruktúrovaná premenná](#).

Ak je známe len **HOBJ** štruktúrovanej premennej, nastavenie odkazu na jej políko sa môže vykona nasledovným spôsobom:

**SET \_locRec[2]^Object AS (HOBJ, ROW, COL) DIRECT**

Napríklad:

```

; použité objekty
; SD.Active: Definícia štruktúry s jedným stpcom typu Objekt. Meno stpca
je "Object".
; SV.Active: dvojriadková štruktúrovaná premenná typu SD.Active
RECORD (SD.Active) _b
BOOL _bOpenOk

; vytvorenie odkazu z políka SV.Active[1]^Object na políko v riadku 1,
stpec 1 objektu s jednoznacným identifikátorom 2131 (HOBJ)
; ak objekt 2131 nie je typu štruktúrovaná premenná alebo index riadku
alebo stpca je mimo rozsah, akcia neprebehne úspešne
SET SV.Active[1]^Object AS (2131, 10, 10) DIRECT
; neúspešný priebeh predchádzajúcej akcie generuje Run-Time error
WAIT

; vytvorenie odkazu z políka _b[1]^Object na políko v riadku 2, stpec 1
objektu s jednoznacným identifikátorom 24623 (HOBJ)
; na rozdiel od predchádzajúceho príkladu, musí by objekt 24623 otvorený
_bOpenOk := %OpenRefToObject(24623, @TRUE)
IF _bOpenOk THEN
SET _b[1]^Object AS (24623, 2, 1) DIRECT
ENDIF

```

!!! Akcia **SET SV.Struktura[1]^Object AS U.Int** (podobne ako priradenie hodnoty objektu) je spracovaná na strane procesu [D2000 Server](#). Jej efekt sa prejaví po ase, v závislosti od zaaženia systému. Na jej skutonú realizáciu je možné v skripte poka akciou [WAIT](#) !!!

Akcia umožňuje zmenu hodnoty typového ALIASu.

```
ALIAS (SD._System_Redundancy) _aT
ALIAS (SD._System_Redundancy) _aT1
SET _aT AS SV._System_Redundancy
SET _aT1 AS _aT
```

Kontrola prípustnosti priradenia vzhľadom na typ štruktúry prebehne počas ukladania ESL skriptu. Zmena hodnoty typového ALIASu bez kontroly typu počas ukladania je možná napríklad nasledovne:

```
BOOL _bOk
ALIAS (SD._System_Redundancy) _aT
INT _hbj

; príklad 1
_hbj := SV._System_Redundancy\HBJ
SET _aT AS (_hbj)

; príklad 2
_hbj := %StrToHBJ("SV._System_Redundancy")
_bOk := %OpenRefToObject(_hbj, @TRUE)
SET _aT AS (_hbj)

; alebo príklad 3
ALIAS _a
_hbj := %StrToHBJ("SV._System_Redundancy")
_bOk := %OpenRefToObject(_hbj, @TRUE)
SET _a AS (_hbj)
SET _aT AS _a
```

Kontrola priradenia prebehne až v ase vykonávania akcie a je možná len v prípade, ak príslušný ESL skript (resp. objekt, ktorý ESL skript obsahuje) má referenciu na priraovanú štruktúrovanú premennú alebo túto referenciu skript vytvoril sám volaním funkcie [%OpenRefToObject](#).

Pozri kapitoly:

- [Lokálne premenné v skripte](#)
- [Prípustné kombinácie parametrov identSrc a identDst](#)
- Akcia **SET WITH**



Súvisiace stránky:

[Akcie v skriptoch](#)