

EDA_SetFunctionRec

Uloží predpis pre vypoítaný vektor.

V prípade zápisu do verziovaných vektorov sú zmeneným dátam automaticky pridelené asové znaky s aktuálnym asom. V prípade potreby je možné asový znak zada explicitne - nastavením parametra `forceVersionTime` na `@TRUE` a nastavením atribútu parametra `version` (`versionTIM`) na požadovaný as. Existujúce dáta s rovnakou asovou znakovou ako nové dáta budú prepísané.

Funkciou je možné vynútiť predpoítanie vektora parametrom `precalcRq`. Potom, v prípade, že pri ítaní vektora je potrebné náíta aj nepredpoítaný interval, je do logu generované upozornenie. Aplikane je možné zobrazí upozornenie užívateľovi (akcia ON "EDA_WARNING").

Deklarácia

```
%EDA_SetFunctionRec(  
  INT/TEXT in _vector,  
  INT in _funcType,  
  TEXT in _funcBody,  
  REC in _params,  
  INT out _errorCode  
);
```

Parametre

	Parameter	Dátový typ	Typ	Popis
1	<code>vector</code>	INT/TEXT	IN	ID / kód vektora.
2	<code>funcType</code>	INT	IN	Typ výpotu (podporený len 1 - na požiadanie).
3	<code>funcBody</code>	TEXT	IN	Predpis funkcie v EDA-L.
4	<code>params</code>	REC	IN	Štruktúra volitených parametrov.
5	<code>errorCode</code>	INT	OUT	Návratová hodnota.

Obsah štruktúry volitených parametrov verzie 1:

	Parameter	Dátový typ	Default	Popis
1	<code>structVersion</code>	INT		Verzia štruktúry - 1.
2	<code>period</code>	INT		Nepodporené
3	<code>offset</code>	INT		Nepodporené
4	<code>desc</code>	TEXT	bez popisu	Popis k funkcii.
5	<code>pod</code>	TIME		Nepodporené
6	<code>pdo</code>	TIME		Nepodporené
7	<code>userName</code>	TEXT	"" (nezapisuje logovaciu informáciu)	Login užívateľa, ktorý vykonal zmenu (volitený).
8	<code>deletePrecalculated</code>	BOOL	TRUE	Príznak, i sa majú predpoítané dáta zmaza alebo ponecha.
9	<code>version</code>	INT/TEXT	"" (nenastavené)	Identifikátor (id/kód) verzie, do ktorej sa má vykona zápis.
10	<code>forceVersionTime</code>	BOOL	@FALSE (asová peiatka sa generuje - aktuálny as)	Pre zapisované dáta vynúti použitie asovej peiatky z parametra <code>version</code> .
11	<code>precalcRq</code>	BOOL	@FALSE	Príznak vynucujúci predpoítanos vektora pri jeho ítaní.
12	<code>cached</code>	INT	0 (default read cache)	Identifikátor cache.
13	<code>checkFunction</code>	BOOL	@FALSE	Príznak, i sa pred uložením funkcie, má vykona aj jej syntaktická kontrola. Systém v tomto prípade neumožní uloží predpis s chybami.
14	<code>batchId</code>	INT		Identifikátor dávky, do ktorej sa má príkaz vloží.
15	<code>userLog</code>	TEXT		Užívateľský popis akcie.

Príklad použitia

```

BEGIN
INT _vectorId = 1000000
TEXT _vectorName = "vector.test"
INT _errorCode
TEXT _funcBody
RECORD NOALIAS (SD.EDA_CreateVector_Params_V1) _createParams
RECORD NOALIAS (SD.EDA_SetFunction_Params_V1) _funcParams

; vytvori vypocitany vektor
_createParams[1]^structVersion := 1
_createParams[1]^periodBeginTime := %TimeFromItems(2015, 1, 1, 0, 0, 0, 0)
_createParams[1]^periodStepBase := 1
_createParams[1]^periodStepCount := 3600
CALL %EDA_CreateVectorRec(_vectorName, _vectorId, _vectorName, 13, _createParams, _errorCode)
IF _errorCode != 0 THEN
RETURN
ENDIF

; nastavi vektoru predpis
_funcBody := "VEKTOR _v;" + %Chr(10)
_funcBody := _funcBody + "INT _i;" + %Chr(10)
_funcBody := _funcBody + "%CreateVektor(_v, _BT, _ET, _STEP);" + %Chr(10)
_funcBody := _funcBody + "FOR _i = 1 TO %ValuesNr(_v) LOOP" + %Chr(10)
_funcBody := _funcBody + "  _v[_i] := _i;" + %Chr(10)
_funcBody := _funcBody + "END LOOP;" + %Chr(10)
_funcBody := _funcBody + "RETURN _v;"
_funcParams[1]^structVersion := 1
CALL %EDA_SetFunctionRec(_vectorId, 1, _funcBody, _funcParams, _errorCode)
IF _errorCode != 0 THEN
RETURN
ENDIF
END

```

i Súvisiace stránky:

- [Externé funkcie](#)
- [Chybové kódy](#)
- [EDA konštanty](#)
- [Tvorbá externých funkcií](#)