

# EDA\_CheckFunctionRec

Funkcia pre syntaktickú kontrolu EDA-L skriptu.

## Deklarácia

```
%EDA_CheckFunctionRec(  
    TEXT in funcBody,  
    REC in params,  
    REC out errors,  
    INT out errorCode  
)
```

## Parametre

	Parameter	Dátový typ	Typ	Popis
1	<b>funcBody</b>	TEXT	IN	Predpis funkcie v EDA-L.
2	<b>params</b>	REC	IN	Štruktúra volitených parametrov.
2	<b>errors</b>	REC	OUT	Popis chýb, ktoré sa našli pri komplikácii. (kód chyby, popis chyby, íslo riadku v EDA-L skripte) [TEXT, TEXT, INT].
3	<b>errorCode</b>	INT	OUT	Chybový (návratový) kód.

## Obsah štruktúry volitených parametrov verzie 1

	Parameter	Dátový typ	Prednastavená hodnota	Popis
1	<b>structVersion</b>	INT		Verzia štruktúry - 1.
2	<b>typeStrict</b>	BOOL	@TRUE	Povolí alebo zakáže typovo strikný režim komplikácie predpisu.

## Príklad použitia

```
BEGIN  
    INT _errorCode  
    TEXT _function  
    RECORD NOALIAS (SD.EDA_CheckFunction_Parms_V1) _params  
    RECORD NOALIAS (SD.EDA_CompilationErrors) _errors  
  
    REDIM _errors[0]  
    _function := "VEKTOR _v;" + %Chr(10)  
    _function := _function + "INT _i;" + %Chr(10)  
    _function := _function + "%CreateVektor(_v, 10);" + %Chr(10)  
    _function := _function + "FOR _i = 1 TO 10 LOOP" + %Chr(10)  
    _function := _function + " IF _i <= 5 THEN" + %Chr(10)  
    _function := _function + " _v[_i] := 1.0;" + %Chr(10)  
    _function := _function + " ELSE" + %Chr(10)  
    _function := _function + " _v[_i] := 10 / _i;" + %Chr(10)  
    _function := _function + " END IF;" + %Chr(10)  
    _function := _function + "END LOOP;" + %Chr(10)  
    _function := _function + "RETURN _v;"  
    CALL %EDA_CheckFunctionRec(_function, _params, _errors, _errorCode)  
    IF _errorCode != 0 THEN  
        RETURN  
    ENDIF  
END
```



Súvisiace stránky:

[Chybové kódy](#)  
[EDA konštanty](#)  
[Tvorba externých funkcií](#)