

HE - Helproutine

Mit diesem Session-Parameter können Sie den Namen einer Helproutine angeben, die einem Feld zugewiesen werden soll.

Mögliche Werte		Siehe <i>HE-Parameter-Syntax</i> weiter unten.	
Standard-Einstellung	keine		
Spezifikation in Session	ja	Gültige Statements:	INPUT
		Gültiges Kommando:	keines

Helproutinen können mit dem Natural-Programm-Editor erstellt werden, Help Maps mit dem Natural-Map-Editor.

Die zugewiesene Helproutine kann dann bei der Verarbeitung des betreffenden INPUT-Statements oder der betreffenden Map aufgerufen werden, indem der Benutzer ein Fragezeichen (?) (das ist die Voreinstellung) in das Feld eingibt oder den Cursor in das Feld plaziert und die mit dem SET KEY-Statement definierte Hilfe-Funktionstaste drückt.

Wenn ein Wert von einer Helproutine an ein Eingabefeld übergeben werden soll, muss das Feld als änderbar (AD=M) definiert sein.

Die folgenden Themen werden nachfolgend erörtert:

- HE-Parameter-Syntax
- Ausführung von Helproutinen
- Beispiele

HE-Parameter-Syntax

Der Parameter hat die folgende Syntax:

$\mathbf{HE}=\text{operand1} \left[, \left\{ \begin{array}{l} \text{operand2} \\ = \\ \text{nX} \end{array} \right\} \right] \dots 20$

Operanden-Definitionstabelle:

Operand	Mögliche Struktur			Mögliche Formate												Referenzierung erlaubt	Dynam. Definition			
<i>operand1</i>	C	S					A											nein	nein	
<i>operand2</i>	C	S	A				A	U	N	P	I	F	B	D	T	L	C	O	nein	nein

Syntax-Beschreibung:

<i>operand1</i>	<p><i>operand1</i> ist der Name der Helproutine oder Map, die aufgerufen werden soll. Der Name kann eine 1 bis 8 Zeichen umfassende alphanumerische Konstante oder Benutzervariable sein. Wenn eine Variable verwendet wird, muss sie vorher definiert worden sein. Die Groß-/Kleinschreibung des Namens wird nicht verändert. Der Name kann ein Kaufmännisches Und (&) enthalten; zur Ausführungszeit wird dieses Zeichen durch den aus einem Zeichen bestehenden Code ersetzt, der dem aktuellen Wert der Systemvariablen *LANGUAGE entspricht. Diese Funktion ermöglicht die Verwendung von mehrsprachigen Helproutinen oder Maps.</p> <p>Weitere Informationen zum <i>operand1</i> in einer Map siehe die HE Helproutine-Option im Abschnitt <i>Extended Field Editing in Map Editor</i> in der <i>Editors</i>-Dokumentation.</p>
<i>operand2</i>	<p>Nach dem Helproutinen-Namen können Sie 1 bis 20 Parameter (<i>operand2</i>) angeben, die an die Helproutine übergeben werden. Diese können Konstanten oder Benutzervariablen sein, die die Parameterwerte enthalten.</p> <p>Weitere Informationen zum <i>operand2</i> in einer Map siehe die HE Helproutine-Option im Abschnitt <i>Extended Field Editing in Map Editor</i> in der <i>Editors</i>-Dokumentation.</p>
=	<p>Geben Sie ein Gleichheitszeichen (=) als Parameter an, wird der Name des Feldes (unter dem es in der Map definiert ist) an die Helproutine übergeben. Ist die Helproutine nicht einem Feld, sondern einer Map zugeordnet, wird mit = der Map-Name übergeben.</p> <ul style="list-style-type: none"> • Wird das Gleichheitszeichen bei HE= auf Statementebene angegeben, wird der Name des zurzeit ausgeführten Objekts (so wie er in der Systemvariablen *PROGRAM enthalten ist) an die Helproutine oder Help-Map übergeben. In Beispiel 3 ist der übergebene Objektname PROGRAM1. • Wird das Gleichheitszeichen bei HE= auf Elementebene (Feldebene) angegeben, wird der Name des Feldes an die Helproutine oder Help-Map übergeben. In Beispiel 3 ist der übergebene Feldname #PARM1. <p>Wird mit = ein Feld- bzw. Map-Name übergeben, so muss der entsprechende Parameter in der Helproutine mit Format/Länge A65 definiert werden.</p>
<i>nX</i>	<p>Die Notation <i>nX</i> können Sie verwenden, um wegzulassende Parameter anzugeben, das heisst, Parameter, für die keine Werte übergeben werden sollen. Die entsprechenden empfangsseitigen Parameter im DEFINE DATA PARAMETER-Statement der aufgerufenen Helproutine müssen als OPTIONAL definiert werden.</p>

Anmerkungen:

1. Die Operanden müssen entweder mit dem Input-Delimiterzeichen (wie mit dem Session-Parameter ID definiert) oder mit einem Komma voneinander getrennt werden. Ein Komma darf hierzu allerdings nicht verwendet werden, falls das Komma als Dezimalkomma (mit dem Session-Parameter DC) definiert ist.
2. Wenn Parameter angegeben werden, so muss die Helprouine mit einem DEFINE DATA PARAMETER-Statement beginnen, in dem Felder definiert werden, die in Format und Länge den übergebenen Parametern entsprechen.
3. Der Wert des Feldes, dem die Helprouine zugeordnet ist, kann in der Helprouine referenziert werden. Hierzu muss im DEFINE DATA PARAMETER-Statement der Helprouine ein Feld definiert werden, das in Format und Länge dem ursprünglichen Feld entspricht. Werden in dem DEFINE DATA PARAMETER-Statement noch andere Felder definiert, so muss dieses Feld immer als letztes definiert werden.
4. Ist das Feld, für das die Helprouine angegeben wird, ein Element eines Arrays, so können die Ausprägungen dieses Feldes von der Helprouine referenziert werden; hierzu müssen Sie Index-Parameter mit Format I (Integer), N (numerisch ungepackt) oder P (gepackt numerisch) am Schluss des DEFINE DATA PARAMETER-Statements definieren. Entsprechend der Array-Dimensionen können Sie bis zu drei Index-Parameter angeben.

Ausführung von Helprouinen

Wenn eine Helprouine — durch Eingabe eines Fragezeichens (?) in das Feld oder durch Drücken der (mit einem SET KEY-Statement definierten) Hilfetaste, oder über ein REINPUT USING HELP-Statement — aufgerufen wird, werden alle in andere Felder eingegebenen Werte erst verarbeitet, nachdem die Ausführung der Helprouine beendet ist.

Anmerkung:

Pro INPUT-Statement ist jeweils nur eine Hilfe-Anforderung möglich. Wenn für mehrere Felder gleichzeitig Hilfe angefordert wird (z.B. durch Eingabe von Fragezeichen in mehrere Felder), wird nur die erste Hilfe-Anforderung ausgeführt.

Beispiele

Beispiel 1:

```
/* MAIN PROGRAM
DEFINE DATA
1 #A(A20/1:3)
END-DEFINE
...
SET KEY PF1=HELP
...
INPUT #A (2) (HE='HELPA',=)
...
END
```

Beispiel 2:

```
/* HELP-ROUTINE 'HELPA'
DEFINE DATA PARAMETER
1 #VARNAME (A65)
1 #PARM1 (A20)
1 #VARINDEX (I2)
END-DEFINE
...
```

Beispiel 3:

```
* Program 'PROGRAM1'
*
DEFINE DATA LOCAL
1 #PARM1 (A65) INIT <'valueparm1'>
END-DEFINE
SET KEY PF1 = HELP
FORMAT KD=ON
*
INPUT (AD=M HE='HELP1',=)
  'Enter ? for name of executed object:'
  / #PARM1
*
INPUT (AD=M)
  'Enter ? for field name:'
  / #PARM1 (HE='HELP1',=)
*
END
```

Parameter Data Area in Beispiel-Helproutine HELP1:

```
* Helproutine 'HELP1'
*
DEFINE DATA PARAMETER
1 #FLD1 (A65)
END-DEFINE
...
```