

Das CT-Assist_Modul

Add-On für die SAP® ABAP® Workbench

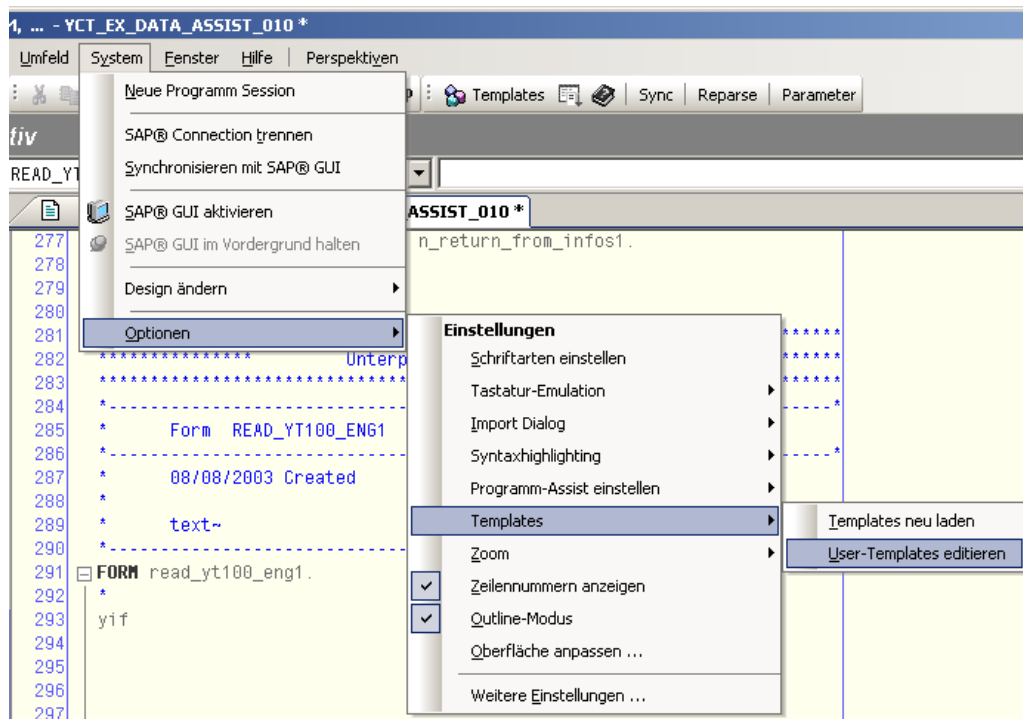
Kurzdokumentation – User-Templates

Die Verwendung von User-Templates

Im CT-Assist_Modul haben Sie die Möglichkeit, eigene User-Templates zu erstellen. Diese Templates werden automatisch in den Command-Assist übernommen, so dass Sie während der Eingabe auf diese Templates zugreifen können. Zusätzlich können Sie diese Templates auch in den Template-Assist integrieren.

1. Template-Userdatei aufrufen

Wählen Sie den Menüpunkt: **"System -> Optionen -> Templates -> User-Templates editieren"** aus, um die User-Template-Datei in das Edit-Fenster zu laden.



2. Template-Aufbau:

Der generelle Aufbau eines Templates sieht wie folgt aus:

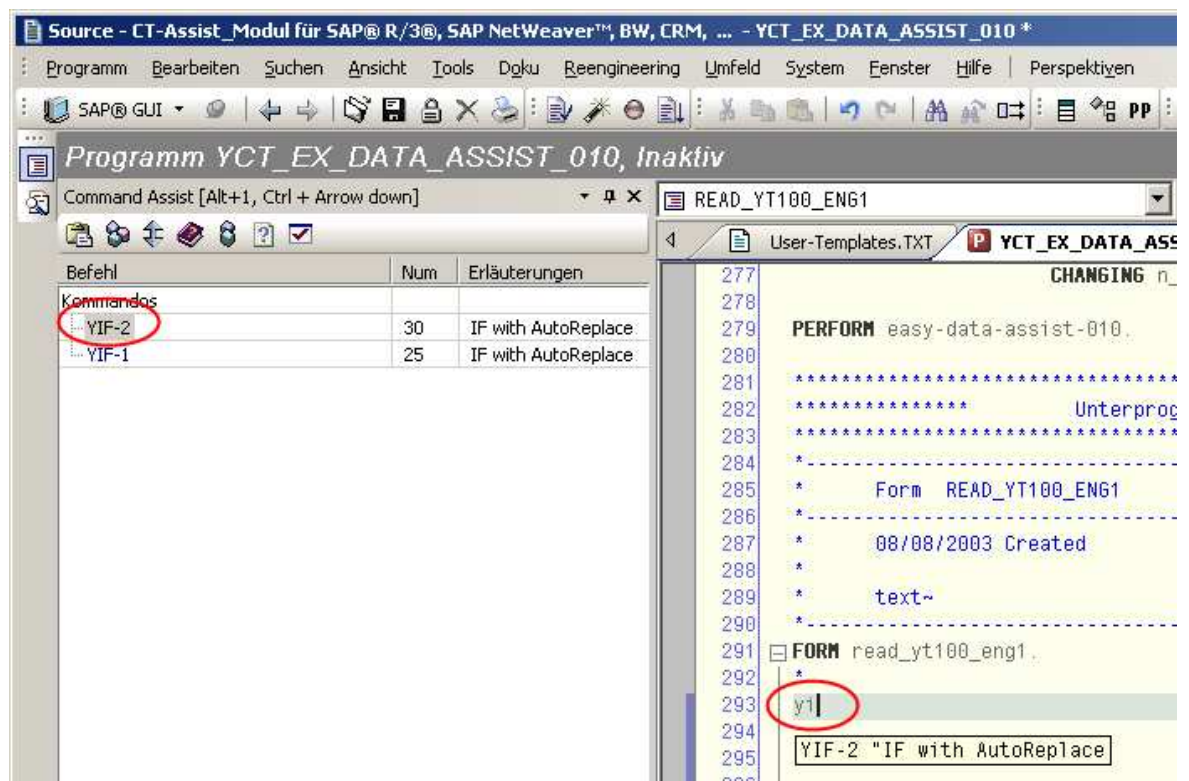
```

30 YIF-2 {Template} {AutoReplaceMode} "IF with AutoReplace
1  IF v2 LT v1. 2 3 4 "$ct-mm/dd/yyyy$"sy-datum
   v3 = v1.
ELSE.
   IF v2 GT 678. 5
   v4 = v1.
ELSE.
   CLEAR v2.
ENDIF.
ENDIF
    
```

The code is annotated with numbers 1 through 6. A red box highlights the header line: `30 YIF-2 {Template} {AutoReplaceMode} "IF with AutoReplace`. A red box highlights the date format: `"$ct-mm/dd/yyyy$"sy-datum`. A red vertical line is drawn between the code and the date format.

1. Befehlscounter

Der Befehlscounter wird benötigt, um die Position des Templates innerhalb des Command-Assist Fensters anzugeben (hier 30). Ein Beispiel:



Sie tippen die Anfangsbuchstaben **YI** ein, ein Tooltip zeigt Ihnen das Template YIF-2 als erste Auswahl, weil der Befehlscounter (NUM) von YIF-2 höher ist als der vom Template YIF-1 (NUM = 25).

Der Befehlscounter gibt also die Rangfolge des Templates innerhalb des Command-Assist-Fensters an. Je größer diese frei wählbare Zahl ist, desto höher ist die Wertigkeit bei der dynamischen Auswahl.

2. Templatename

Der Templatename wird benötigt, um das User-Template aufzurufen und einzusetzen.

3. Kennzeichnung als Template

Die Bezeichnung {Template} kennzeichnet die folgenden Zeilen als User-Template. Die geschweiften Klammern dürfen dabei nicht fehlen.

4. Funktion Autoreplace (optional)

Die Kennzeichnung {Autoreplace} ist nicht zwingend notwendig, aber sinnvoll bei dem Ersetzen der Variablennamen (z.B v1 und v2).

Ein Beispiel:

Das Template YIF-2 wird eingesetzt:

```

FORM read_yt100_eng1.
*
IF v1 LT v1-.
  v3 = v1-.
ELSE.
  IF v2 GT 678.
    v4 = v1-.
  ELSE.
    CLEAR v2-.
  ENDIF.
ENDIF.
    
```

An der Stelle **v2** wird nun **hugo** eingetippt:

```

FORM read_yt100_eng1.
*
IF hugo LT v1-.
  v3 = v1-.
ELSE.
  IF hugo GT 678.
    v4 = v1-.
  ELSE.
    CLEAR hugo.
  ENDIF.
ENDIF.
    
```

Innerhalb des blauen Bereiches wurden automatisch alle **v2** Variablen durch **hugo** ersetzt (siehe auch Menü: Bearbeiten AutoReplaceMode).

5. Template-Inhalt

Diese Zeilen werden nach dem Aufruf des Templates an die aktuelle Codingstelle eingesetzt.

6. Templatevariablen

Die folgenden SY-Variablen können Sie z.B als Kommentar innerhalb Ihres Templates verwenden:

- \$ct-mm/dd/yyyy\$ "sy-datum
- \$ct-hh/mm/ss\$ "sy-uzzeit
- \$ct-user\$ "sy-uname
- \$ct-system\$ "sy-sysid
- \$ct-host\$ "sy-host
- \$ct-mandant\$ "sy-mandt
- \$ct-release\$ "sy-saprl

3. Template-Aufbau (Auto-Structure-Fill):

Neben dem oben beschriebenen User-Template-Aufbau gibt es noch ein weiteres Spezialtemplate zum dynamischen Ausfüllen von Strukturen.

Als Basis kann entweder eine programminterne Struktur/Tabelle dienen oder eine Struktur/Tabelle aus dem SAP® Data Dictionary.

Die Templateangaben werden für jedes Struktur-Element durchlaufen, Punkte und Kommas werden entsprechend entfernt oder hinzugefügt.

Ein Beispiel:

```
6      YSORT-tabelle      {AutoStructureFill}      "SORT Tab f1 f2 Ascending

      SORT tabname BY
        {elementname      description<CR>}
      ASCENDING .
```

1. Befehlscounter

Zunächst wird wieder der Befehlscounter (hier 6) angegeben

2. Templatename

Dann folgt der Templatename (hier YSORT-tabelle)

3. Templatebezeichnung {AutoStructureFill}

Für die dynamische Generierung einer Struktur/Tabelle muss die Bezeichnung {AutoStructureFill} hinter dem Templatenamen stehen

4. Name der Struktur (tabname)

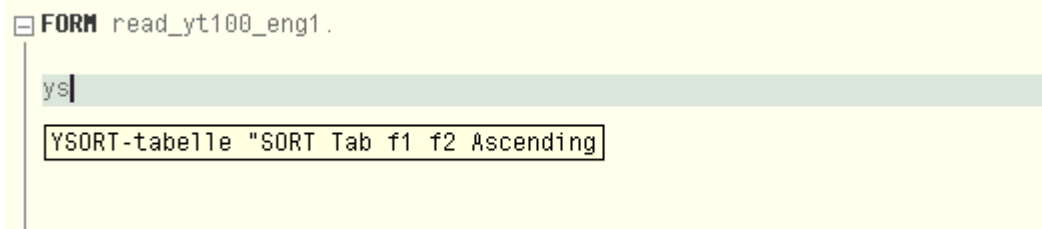
Das Wort 'tabname' dient als Platzhalter für die spätere Struktur oder Tabelle.

5. Strukturelement, Beschreibung und Zeilenwechsel

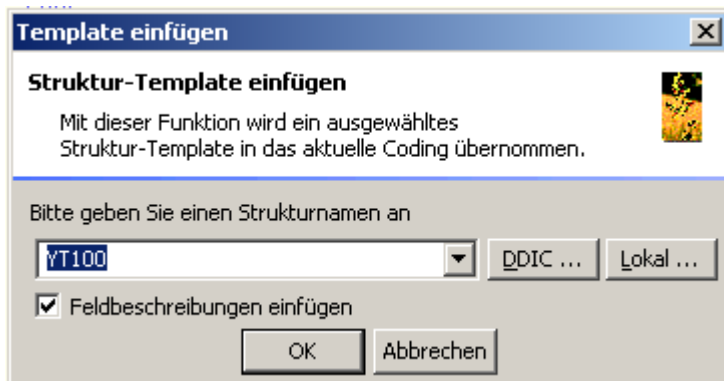
In geschweiften Klammern wird angegeben wie die Elemente generiert werden sollen. Das Wort '**elementname**' dient als Platzhalter für das spätere Element. Das Wort '**description**' bedeutet, dass zu dem Element (falls vorhanden) eine Beschreibung als Kommentar hinzugefügt wird (ist nicht zwingend erforderlich). Das Steuerelement <CR> steht für den Zeilenwechsel.

... zurück zum Beispiel:

Es wird 'ys' eingetippt. Ein Tooltip mit dem Template wird angezeigt.



Mit TAB wird das Template ausgeführt. In der folgenden Dialogbox geben Sie die Struktur/Tabelle ein und klicken auf den Ok-Button.



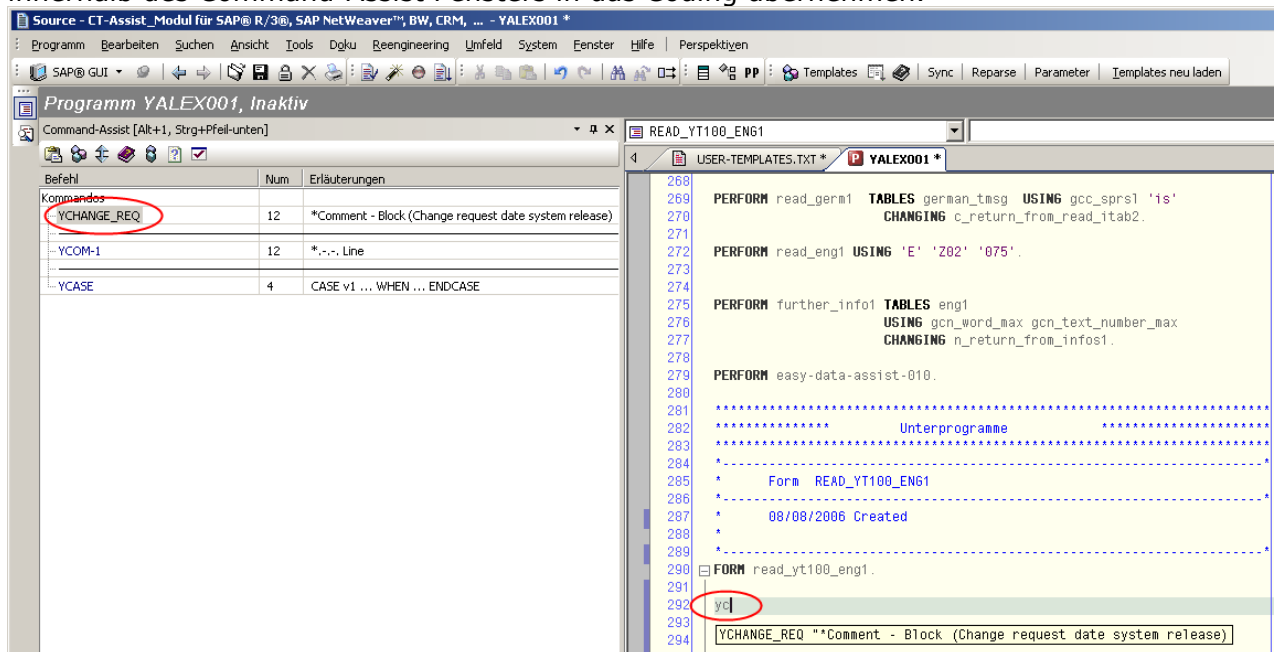
Ergebnis:

```
FORM read_yt100_eng1.
SORT| YT100 BY
  ZSPRSL " Language Key
  ZARBGB " Application Area
  ZMSGNR " Message number
  ZTEXT1 " Message text
  ZANSWER2 " Message Return
  ZNOTE3 " Character field with length 73
ASCENDING .
```

4. Aufruf eines Templates:

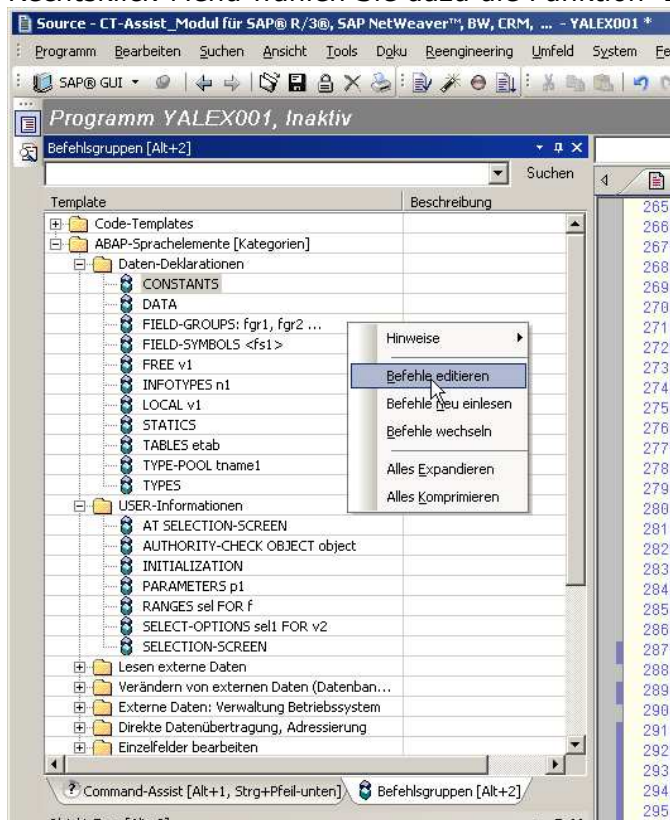
1. Command-Assist

Während der Eingabe können Sie das vorgeschlagene Template aus dem Tooltip-Fenster mit TAB übernehmen. Alternativ können Sie das Template mit Doppelklick innerhalb des Command-Assist Fensters in das Coding übernehmen.



2. Befehlsgruppenbaum (Alt + 2)

Ihre User-Templates können in den Befehlsgruppenbaum kopiert werden. Über das Rechtsklick-Menü wählen Sie dazu die Funktion **'Befehle editieren'** aus.



Anschließend wird eine Datei in das Edit-Fenster geladen. Tragen Sie hier den Templatenamen nach dem vorgegeben Muster ein (siehe YSORT-Tabelle).

```

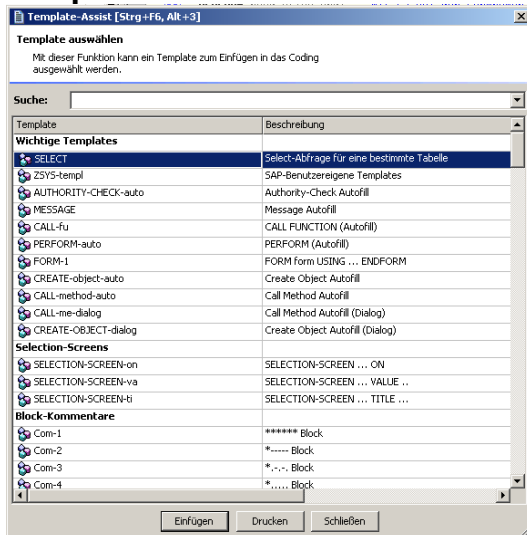
1 | *****
2 | ; Code-Templates
3 | *****
4 |
5 | + f: Code-Templates
6 | + f: Wichtige Templates
7 | - SELECT |Select-Abfrage für eine bestimmte Tabelle |{InsertCommand}}
8 | - USER |SAP-Benutzereigene Templates |{InsertCommand}}
9 | - AUTHORITY-CHECK-auto |Authority-Check Autofill |{InsertCommand}}
10 | - MESSAGE |Message Autofill |{InsertCommand}}
11 | - CALL-fu |CALL FUNCTION (Autofill) |{InsertCommand}}
12 | - CALL-me |Call Method Autofill |{InsertCommand}}
13 | - PERFORM-auto |PERFORM (Autofill) |{InsertCommand}}
14 | - CREATE-object-auto |Create Object |{InsertCommand}}
15 | - FORM-ue |*----- FORM form USING ENDFORM |{InsertCommand}}
16 | - YSORT-tabelle |Eine Tabellenstruktur dynamisch erzeugen |{InsertCommand}}
    
```

Speichern Sie die Datei und führen die Funktion **'Befehle neu einlesen'** im Rechtsklick-Menü des Befehlsgruppen-Baums aus. Das Template wird daraufhin im Befehlsgruppenbaum angezeigt.

CREATE-object-auto	Create Object
FORM-ue	*----- FORM form USING ... ENDFORM
YSORT-tabelle	Eine Tabellenstruktur dynamisch erzeugen
Selection-Screens	
SELECTION-SCREEN-on	SELECTION-SCREEN ... ON
SELECTION-SCREEN-va	SELECTION-SCREEN ... VALUE ..

3. Befehls-Templates (Template-Assist)

Wenn Ihre User-Templates im Template-Assist angezeigt werden sollen, (Menü: Tools -> Befehls-Templates) editieren Sie bitte die Datei **TemplateBrowser.txt**.



Öffnen Sie die Datei **TemplateBrowser.txt** über die Funktion '**Menü: Datei -> Lokale Datei**'. Die Datei befindet sich im Installationspfad des CT-Assist_Moduls (z.B. C:\Programme\CT-Software\CT-Assist_Modul\templates_default\TemplateBrowser.txt). Nach dem Sie die Datei editiert und abgespeichert haben, wird die Änderung innerhalb des Template-Baums nach dem nächsten Start des CT-Assist_Moduls sichtbar.