

# Hello World!

Dieses Kapitel enthält die folgenden Übungen:

- Programm erstellen
  - Programm mit RUN ausführen
  - Programmfehler korrigieren
  - Programm mit STOW speichern
  - Informationen über ein Programm anzeigen
  - Inhalt der aktuellen Library anzeigen
  - Editorprofil-Optionen angeben
- 

## Programm erstellen

Sie werden nun Ihr erstes kurzes Programm schreiben, das "Hello World!" anzeigt. Es wird in der Library gespeichert, die Sie zuvor erstellt haben.

### ▶ Neues Programm erstellen

1. Achten Sie darauf, dass Sie in der Library TUTORIAL angemeldet sind.
2. Geben Sie unten im Menü **Development Functions** die folgenden Informationen ein und drücken Sie EINGABE:

Code .. C	Type .. P	Name .. HELLO_____
Command ==>		
Enter-PF1---	PF2---	PF3---
PF4---	PF5---	PF6---
PF7---	PF8---	PF9---
PF10--	PF11--	PF12---
Help	Menu	Exit
		Canc

"C" steht für die Funktion **Create Object** (Objekt erstellen), "P" steht für den Objekttyp Programm und "HELLO" ist der Name des zu erstellenden Programms.

### **Tipp:**

Wenn Sie den Funktionscode C eingeben, können Sie auch einen Stern (\*) im **Type**-Feld angeben. Wenn Sie EINGABE drücken, erscheint eine Liste aller Objekttypen und aller Buchstaben, die für diese Objekttypen benutzt werden können.

Der Programmeditor erscheint. Er ist zurzeit leer.

3. Geben Sie den folgenden Code im Programmeditor ein:

```
* The "Hello world!" example in Natural.  
*  
DISPLAY "Hello world!"  
END /* End of program
```

Eine Kommentarzeile beginnt mit einem Stern (\*), dem mindestens ein Leerzeichen oder ein zweiter Stern folgt. Wenn Sie das Leerzeichen oder den zweiten Stern vergessen, geht Natural davon aus, dass Sie eine Systemvariable angegeben haben; dies verursacht einen Fehler.

Wenn Sie Leerzeilen in Ihr Programm einfügen möchten, dann sollten Sie sie als Kommentarzeilen definieren. Dies ist hilfreich, wenn Sie Ihr Programm auf einer anderen Plattform (Windows, Großrechner, UNIX oder OpenVMS) weiterbearbeiten wollen. Mit der Großrechnerversion von Natural werden beispielsweise alle leeren Zeilen automatisch gelöscht, sobald Sie EINGABE drücken (Standardeinstellung).

Sie können auch einen Kommentar am Ende einer Statement-Zeile einfügen. In diesem Fall muss der Kommentar mit einem Schrägstrich beginnen, dem ein Stern folgt (/).

Der Text, der in der Ausgabe erscheinen soll, wird mit dem Statement DISPLAY angegeben. Er steht in Anführungszeichen.

Das Statement END markiert das physische Ende des Natural-Programms. Jedes Programm muss mit END abgeschlossen werden.

Wenn Sie EINGABE drücken, kann es passieren, dass alle Kleinbuchstaben in Großbuchstaben umgesetzt werden. Dieses Verhalten ist im Editorprofil definiert (dies wird später erklärt).

## Programm mit RUN ausführen

Das Systemkommando RUN ruft automatisch das Systemkommando CHECK auf, welches das Programm auf Fehler überprüft. Wenn kein Fehler gefunden wird, wird das Programm im Flug kompiliert und ausgeführt.

### Anmerkungen:

1. CHECK steht ebenfalls als separates Systemkommando zur Verfügung.
2. Es gibt bei Natural auch das Systemkommando EXECUTE, welches die mit dem Systemkommando STOW gespeicherte Version Ihres Programms benutzt (dieses Kommando wird später in diesem Tutorial erklärt). Im Gegensatz zu EXECUTE werden bei RUN immer die letzten Programmänderungen berücksichtigt.

### ▶ Programm mit RUN ausführen

1. Geben Sie eines der folgenden Kommandos in der Kommandozeile des Programmeditors ein:

```
RUN
```

```
R
```

Systemkommandos können abgekürzt werden. R ist die Abkürzung für RUN.

In Abhängigkeit von Ihren Umgebungseinstellungen befindet sich die Kommandozeile entweder am oberen oder unteren Rand des Bildschirms.

```
> RUN                                     > + Program   HELLO   Lib TUTORIAL
```

Wenn Ihr Code syntaktisch korrekt ist, wird der von Ihnen definierte Text ausgegeben.

```
MORE
Page      1                                05-03-11  12:07:25

Hello world!
```

2. Drücken Sie EINGABE um zum Programmeditor zurückzukehren.

## Programmfehler korrigieren

Sie werden jetzt einen Fehler in Ihr Programm einbauen und das Programm anschließend noch einmal mit RUN ausführen.

### Fehler korrigieren

1. Löschen Sie das zweite Anführungszeichen in der Zeile, die das DISPLAY-Statement enthält.
2. Führen Sie das Programm noch einmal wie oben beschrieben aus.

Wenn ein Fehler gefunden wird, wird eine Fehlermeldung angezeigt.

```

NAT0305 Text string must begin and end on the same line.
>                                     > + Program   HELLO   Lib TUTORIAL
All  ....+....1....+....2....+....3....+....4....+....5....+....6....+....7..
      0010 * The "Hello world!" example in Natural.
      0020 *
E 0030 DISPLAY "HELLO WORLD!"
      0040 END /* End of program
      0050
      0060
      0070
      0080
      0090
      0100
      0110
      0120
      0130
      0140
      0150
      0160
      0170
      0180
      0190
      0200
      ....+....1....+....2....+....3....+....4....+....5....+.... S 4   L 1

```

Die fehlerhafte Zeile ist hervorgehoben und am Anfang mit einem "E" gekennzeichnet.

3. Korrigieren Sie den Fehler. Das heißt: fügen Sie das fehlende Anführungszeichen wieder am Ende der Zeile ein.
4. Führen Sie das Programm erneut aus, um den nächsten Fehler zu finden.

In diesem Fall werden keine weiteren Fehler gefunden und die Ausgabe wird angezeigt.

5. Drücken Sie EINGABE, um zum Programmeditor zurückzukehren.

## Programm mit STOW speichern

Wenn Sie ein Programm mit STOW speichern, wird es kompiliert und der Sourcecode sowie das generierte Programm werden in der Systemdatei abgelegt.

Wie beim Systemkommando RUN wird auch beim Systemkommando STOW automatisch das Systemkommando CHECK aufgerufen. Ein Programm kann nur dann mit STOW gespeichert werden, wenn es syntaktisch korrekt ist.

### Anmerkung:

Wenn Sie Ihre Programmänderungen speichern wollen, obwohl das Programm syntaktische Fehler enthält (wenn Sie Ihre Arbeit zum Beispiel bis zum nächsten Arbeitstag unterbrechen wollen), dann können Sie das Systemkommando SAVE benutzen.

### ▶ Programm mit STOW speichern

- Geben Sie Folgendes in der Kommandozeile des Programmeditors ein:

```
STOW
```

## Informationen über ein Programm anzeigen

Das Systemkommando LIST ist hilfreich, wenn Sie herausfinden wollen, ob für ein Objekt nur der Sourcecode vorhanden ist oder ob beides, der Sourcecode und das generierte Programm, vorhanden ist.

### ▶ Informationen über ein Programm anzeigen

1. Geben Sie eines der folgenden Kommandos in der Kommandozeile des Programmeditors ein:

```
LIST DIR HELLO
```

```
L DIR HELLO
```

Der folgende Bildschirm erscheint. Die Informationen bei **Cataloged on** stehen nur dann zur Verfügung, wenn das Objekt mit STOW gespeichert wurde.

```

13:15:45          ***** NATURAL LIST COMMAND *****          2007-03-20
User SAG          - List Directory -          Library TUTORIAL

Directory of Program HELLO          Saved on ... 2007-03-20 13:15:36
-----
Library ... TUTORIAL  User-ID ..... SAG      Mode ..... Structured
TP-System .. COMPLETE Terminal-ID .. DAEFTCA9
Op-System .. MVS/ESA  Transaction .. NATvr
NAT-Ver ... v.r.s
Source size ..... 100 Bytes

Directory of Program HELLO          Cataloged on 2007-03-20 13:15:36
-----
Library ... TUTORIAL  User-ID ..... SAG      Mode ..... Structured
TP-System .. COMPLETE Terminal-ID .. DAEFTCA9
Op-System .. MVS/ESA  Transaction .. NATvr
NAT-Ver ... v.r.s
Used GDA ...
Size of global data ... 0 Bytes  Size in DATSIZE ..... 560 Bytes
Size in buffer pool ... 2620 Bytes

Size of OPT-Code ..... 0 Bytes
Initial OPT string ...

ENTER to continue

```

#### Anmerkung:

Die Bezeichnungen *vr* und *v.r.s* im Beispielbildschirm oben stehen für die aktuelle Versionsnummer von Natural. Siehe auch die Definition von *Version* im *Glossary*.

2. Drücken Sie EINGABE um zum Programmeditor zurückzukehren.

## Inhalt der aktuellen Library anzeigen

Das Systemkommando LIST kann dazu benutzt werden, eine Liste aller Natural-Objekte in der aktuellen Library anzuzeigen. Dies ist zum Beispiel dann hilfreich, wenn Sie im Laufe dieses Tutorials eines oder mehrere Ihrer Natural-Objekte löschen wollen, um wieder von vorne zu beginnen.

### ▶ Liste der Natural-Objekte anzeigen

1. Geben Sie eines der folgenden Kommandos in der Kommandozeile des Programmeditors ein:

```
LIST *
```

```
L *
```

Der folgende Bildschirm erscheint. Das Programm, das sie eben erstellt haben, wird aufgelistet.

```
13:34:27          ***** NATURAL LIST COMMAND *****          2007-03-20
User SAG          - LIST Objects in a Library -          Library TUTORIAL

Cmd  Name          Type          S/C  SM  Version  User ID  Date          Time
---  *          *          *   *   *          *          *          *
___  HELLO          Program      S/C  S   v.r.s    SAG      2007-03-20    13:15:36

                                                    1 Objects found

Top of List.
Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Print Exit  Sort          --   -   +   ++          >   Canc
```

2. Um herauszufinden, welche Kommandos zur Verfügung stehen, geben Sie ein Fragezeichen (?) in der **Cmd**-Spalte neben Ihrem Programm ein.

Das folgende Fenster erscheint.



```

13:35:43          ***** NATURAL EDITORS *****          2007-03-20
                    - Editor Profile -

Profile Name .. SYSTEM__

PF and PA Keys
PF1 ... HELP_____ PF2 ... _____ PF3 ... EXIT_____
PF4 ... _____ PF5 ... _____ PF6 ... _____
PF7 ... -_____ PF8 ... +_____ PF9 ... _____
PF10 .. SC=_____ PF11 .. _____ PF12 .. CANCEL_____
PF13 .. _____ PF14 .. _____ PF15 .. MENU_____
PF16 .. _____ PF17 .. _____ PF18 .. _____
PF19 .. --_____ PF20 .. ++_____ PF21 .. _____
PF22 .. _____ PF23 .. _____ PF24 .. _____
PA1 ... _____ PA2 ... SCAN_____ PA3 ... _____

Automatic Functions
Auto Renumber .. Y   Auto Save Numbers .. 0__   Source Save into .. EDITWORK

Additional Options .. N

Command ==>
Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help      Exit  AddOp Save  Flip                                Del  Canc

```

Wenn es für einen Benutzer noch kein Editorprofil gibt, dann wird das Standardprofil SYSTEM angezeigt. Mit diesem Standardprofil kann ein Benutzerprofil erstellt werden. Wenn ein Benutzerprofil bereits existiert, dann wird es anstelle des Profils SYSTEM angezeigt.

- Geben Sie "Y" im Feld **Additional Options** (zusätzliche Optionen) ein und drücken Sie EINGABE.

Oder:

Drücken Sie PF4.

Das folgende Fenster erscheint.

```

+-----+-----+-----+-----+-----+-----+
!                                     !
!                                     !
!   + Editor Defaults ..... N       !
!   + General Defaults ..... N      !
!   + Colour Definitions ..... N    !
!                                     !
!                                     !
!                                     !
!                                     !
!                                     !
!                                     !
+-----+-----+-----+-----+-----+-----+

```

- Geben Sie "Y" in den Feldern **Editor Defaults** (Standardeinstellungen für den Editor) und **General Defaults** (Allgemeine Standardeinstellungen) ein und drücken Sie EINGABE.

Zuerst erscheint das folgende Fenster.



```

+----- EDITOR DEFAULTS -----+
!                                     !
!   Escape Character for Line Command .. .   !
!   Empty Line Suppression ..... Y         !
!   Source Size Information ..... N         !
!   Source Status Message ..... N         !
!   Absolute Mode for SCAN/CHANGE ..... N  !
!   Range Mode for SCAN/CHANGE ..... N    !
!   Direction Indicator ..... +          !
!                                     !
!                                     !
+-----+

```

Hinter **Escape Character for Line Command** sehen Sie das Umschaltzeichen, das für die Zeilenkommandos definiert ist. Dieses Tutorial geht davon aus, dass das Standardzeichen verwendet wird; dies ist der Punkt (.).

Dieses Tutorial geht auch davon aus, dass die Option **Empty Line Suppression** (Unterdrückung von Leerzeilen) auf "Y" gesetzt ist. In diesem Fall werden alle Leerzeilen im Programmeditor automatisch gelöscht, sobald Sie EINGABE drücken. Die Leerzeilen werden nicht gelöscht, wenn diese Option auf "N" gesetzt ist.

4. Achten Sie darauf, dass für dieses Tutorial alle Optionen so gesetzt sind, wie oben angezeigt. Drücken Sie EINGABE, um das nächste Fenster anzuzeigen.

Das folgende Fenster erscheint.

```

+----- GENERAL DEFAULTS -----+
!                                     !
!   Editing in Lower Case ..... N         !
!   Dynamic Conversion of Lower Case ... Y  !
!   Position of Message Line ..... TOP    !
!   Cursor Position in Command Line .... N !
!   Stay on Current Screen ..... N       !
!   Prompt Window for Exit Function .... Y !
!   ISPF Editor as Program Editor ..... N !
!   Leave Editor with Unlock ..... N     !
!                                     !
+-----+

```

Wenn die Option **Editing in Lower Case** (Editieren in Kleinbuchstaben) auf "Y" gesetzt ist und die Option **Dynamic Conversion of Lower Case** (dynamische Umsetzung in Kleinbuchstaben) auf "N" gesetzt ist, dann bleibt der Sourcecode so erhalten, wie Sie ihn eingeben. Diese Eigenschaft ist jedoch abhängig von den Einstellungen in Ihrer Systemumgebung; wenn diese Einstellungen so gewählt sind, dass eine Umsetzung in Großbuchstaben erzwungen wird, dann kann dies nicht von Natural beeinflusst werden.

5. Wenn Sie es möchten, können Sie die oben genannten Optionen für die Umsetzung in Kleinbuchstaben ändern und EINGABE drücken. Drücken Sie noch einmal EINGABE, um zum Fenster **Additional Options** zurückzukehren, und drücken Sie dann noch einmal EINGABE, um dieses Fenster zu schließen.
6. Wenn ein Benutzerprofil noch nicht erstellt wurde, überschreiben Sie den Profilnamen SYSTEM mit Ihrer User-ID und drücken Sie EINGABE.

Wenn ein Benutzerprofil bereits existiert, fahren Sie fort mit dem nächsten Schritt.

7. Drücken Sie PF5, um Ihre Änderungen in der Datenbank zu speichern, und drücken Sie dann PF3, um das Editorprofil zu verlassen.

**Anmerkung:**

Statt eine PF-Taste zu drücken können Sie auch das entsprechende Kommando in der Kommandozeile eingeben. Im hier beschriebenen Fall können Sie zum Beispiel die Kommandos `SAVE` und `EXIT` eingeben.

Oder:

Wenn Sie Ihre Änderungen nicht speichern möchten, drücken Sie PF3 um das Editorprofil zu verlassen.

Beim Verlassen des Editorprofils wird ein Fenster mit verschiedenen Optionen angezeigt. Es erscheint auch dann, wenn Sie Ihre Änderungen eben erst gespeichert haben.

```
+----- EXIT Function -----+
!                               !
!  _ Save and Exit              !
!  _ Exit without Saving       !
!  _ Resume Function           !
!                               !
!                               !
+-----+
```

8. Wenn Sie Ihre Änderungen bereits unmittelbar vor dem Verlassen des Editorprofils gespeichert haben, können Sie die Option **Exit without Saving** (Verlassen ohne Speichern) wählen. Drücken Sie EINGABE, um zum Programmeditor zurückzukehren.

**Vorsicht:**

Falls Sie nach dem Drücken von PF5 oder nach der Eingabe des Kommandos `SAVE` weitere Änderungen gemacht haben und jetzt die Option **Exit without Saving** wählen, dann gelten Ihre zuletzt gemachten Änderungen nur in der aktuellen Session; sie werden nicht in der Datenbank gespeichert.

Ihr Programm wird wieder angezeigt. Falls Sie das Editorprofil geändert haben, dann werden die neuen Einstellungen jetzt im Programmeditor benutzt (und auch im Data-Area-Editor, der später in diesem Tutorial beschrieben wird).

Sie können nun mit den nächsten Übungen fortfahren: *Datenbankzugriff*.