

# Hello World!

Dieses Kapitel enthält die folgenden Übungen:

- Programm erstellen
  - Programm mit RUN ausführen
  - Programmfehler korrigieren
  - Programm mit STOW speichern
  - Workspace-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. Markieren Sie im Library-Workspace die Library TUTORIAL.
2. Wählen Sie aus dem Kontextmenü den Befehl **New Source > Program**.

Oder:

Wählen Sie die folgende Schaltfläche in der Symbolleiste:



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.

## 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. Systemkommandos werden auch bei der Großrechnerversion von Natural benutzt. Unter Windows werden sie mit den entsprechenden Befehlen im Menü **Object** ausgeführt.
2. `CHECK` steht ebenfalls als separates Systemkommando im Menü **Object** zur Verfügung.
3. 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. Wählen Sie aus dem Menü **Object** den Befehl **Run**.

Oder:

Wählen Sie die folgende Schaltfläche in der Symbolleiste:



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

Page 1

09-06-30 12:07:25

```
Hello world!
```

2. Drücken Sie EINGABE um zum Programmeditor zurückzukehren.

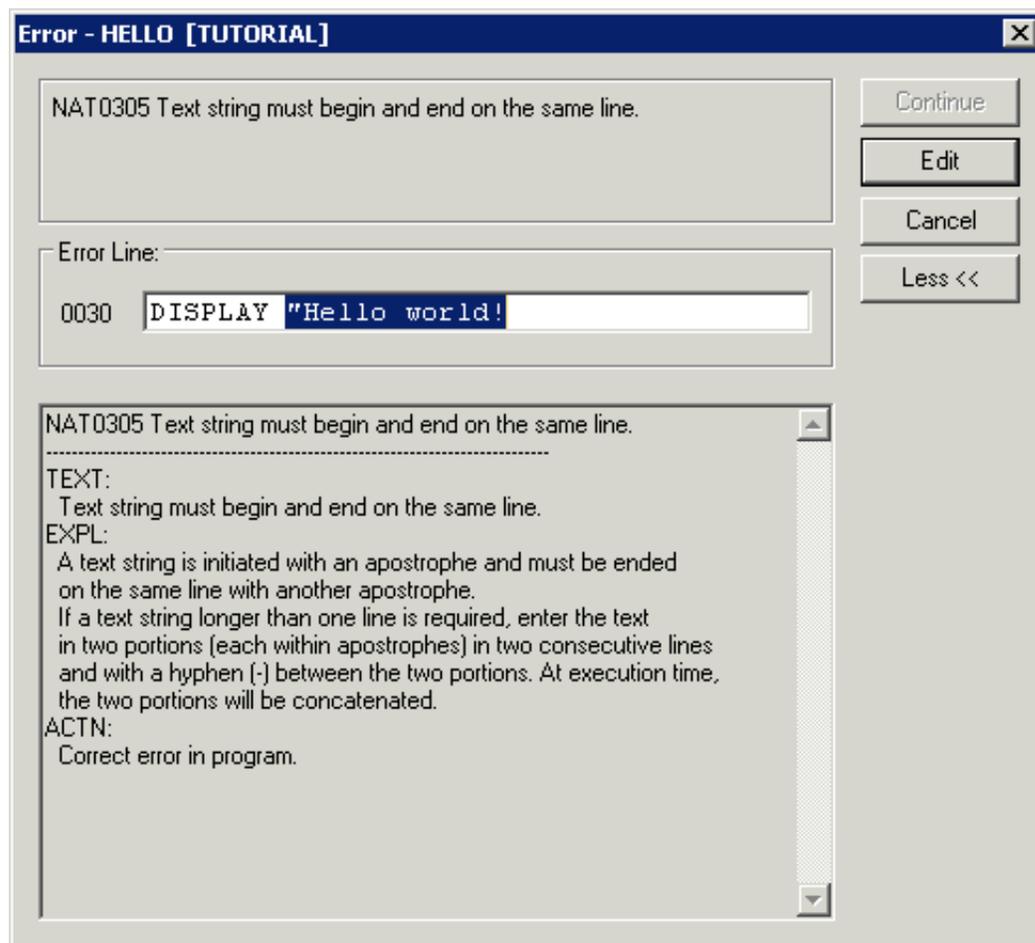
## Programmfehler korrigieren

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, erscheint ein Dialogfeld mit Hinweisen zu diesem Fehler.



3. Korrigieren Sie den Fehler im Dialogfeld. Das heißt: fügen Sie das fehlende Anführungszeichen wieder am Ende der Zeile ein.

4. Wählen Sie die Befehlsschaltfläche **Continue**, 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.

**Anmerkung:**

Statt der Befehlsschaltfläche **Continue** können Sie auch die Befehlsschaltfläche **Edit** wählen. Das Dialogfeld wird in diesem Fall geschlossen und Sie können den Fehler direkt im Programmeditor korrigieren.

## 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. Dieses Systemkommando wird mit dem entsprechenden Befehl im Menü **Object** ausgeführt.

### ▶ Programm mit STOW speichern

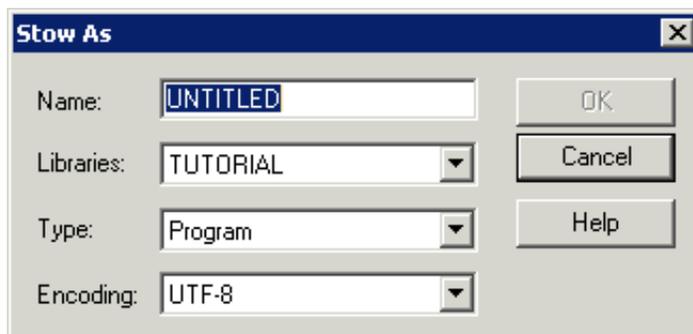
1. Wählen Sie aus dem Menü **Object** den Befehl **Stow**.

Oder:

Wählen Sie die folgende Schaltfläche in der Symbolleiste:



Da Ihr Programm noch nicht gespeichert wurde, erscheint das Dialogfeld **Stow As**.



Der Name der aktuellen Library wird automatisch im entsprechenden Dropdown-Listenfeld angezeigt.

2. Geben Sie den Namen "HELLO" im Textfeld **Name** ein.
3. Wählen Sie die Befehlsschaltfläche **OK**.

Es erscheint die Nachricht, dass das Speichern erfolgreich war. Diese Nachricht wird entweder in der Statuszeile oder in einem Dialogfeld angezeigt. Dies ist abhängig von der Einstellung einer bestimmten Workspace-Option (siehe unten).

Im Library-Workspace wird nun ein neuer Knoten mit dem Namen **Programs** angezeigt, und zwar als Unterknoten von **TUTORIAL**. Dieser Unterknoten enthält das soeben gespeicherte Programm.



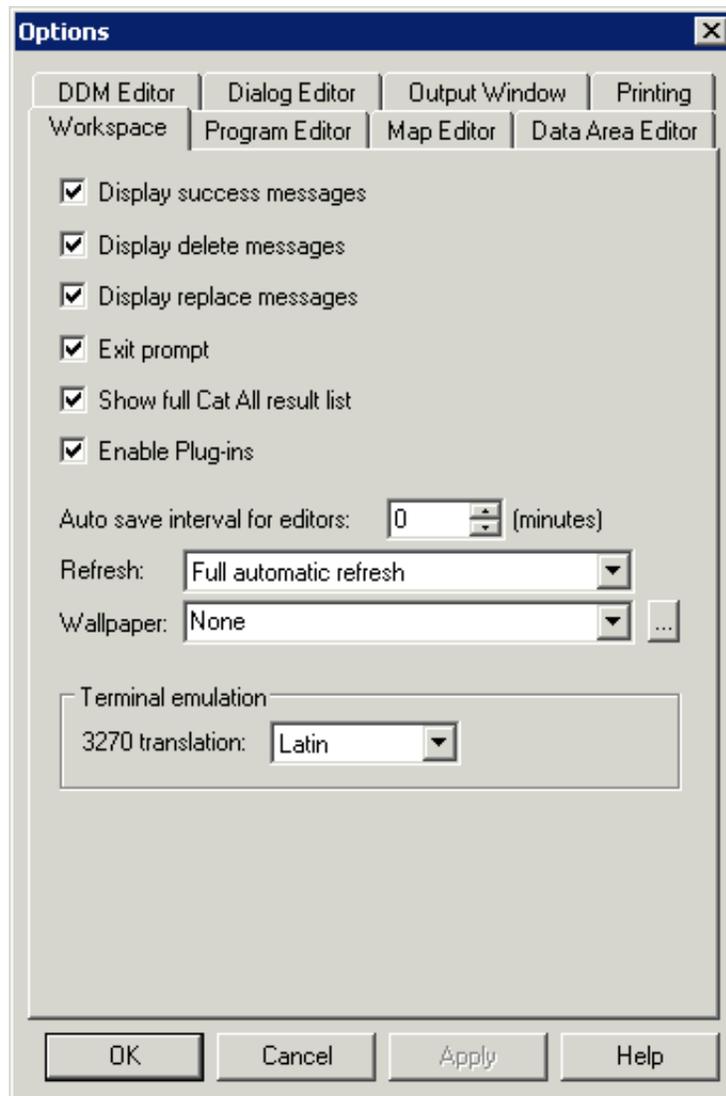
Der grüne Ball im Programm-Symbol bedeutet, dass es für dieses Objekt sowohl Sourcecode als auch ein generiertes Programm gibt.

## Workspace-Optionen angeben

Sie werden jetzt die Einstellungen Ihrer Workspace-Optionen überprüfen.

### ▶ Workspace-Optionen prüfen

1. Wählen Sie aus dem Menü **Tools** den Befehl **Options**.
2. Zeigen Sie im daraufhin erscheinenden **Options**-Dialogfeld die **Workspace**-Seite an.



3. Wenn Erfolgsmeldungen in einem Dialogfeld angezeigt werden sollen, dann muss das Kontrollkästchen **Display success messages** markiert sein.

**Anmerkung:**

Die Anzeige der Zeilennummern im Programmeditor wird durch eine Option auf der Seite **Program Editor** gesteuert.

4. Wählen Sie die Befehlsschaltfläche **OK**, um die Änderungen zu speichern und das Dialogfeld zu schließen.

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