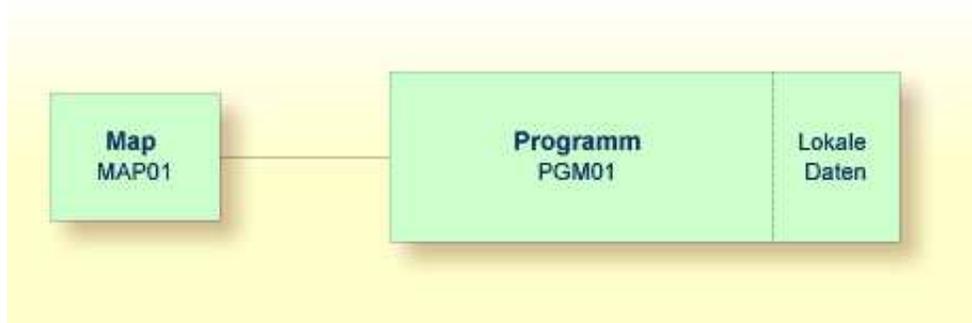


Benutzereingaben

Sie lernen nun, wie Sie einen Benutzer zur Eingabe von Daten auffordern (d.h. der Benutzer soll einen Start- und Endnamen für die Ausgabe eingeben).

Wenn Sie mit den Übungen in diesem Kapitel fertig sind, wird Ihre Beispielanwendung aus den folgenden Modulen bestehen:



Dieses Kapitel enthält die folgenden Übungen:

- Benutzereingaben ermöglichen
- Map für die Benutzereingabe gestalten
- Map aus dem Programm aufrufen
- Immer einen Endnamen benutzen

Benutzereingaben ermöglichen

Sie werden Ihr Programm jetzt so verändern, dass Eingabefelder für die Start- und Endnamen in der Ausgabe angezeigt werden. Hierzu benutzen Sie das `INPUT`-Statement.

▶ Eingabefelder definieren

1. Geben Sie Folgendes unter `END-DEFINE` ein:

```

INPUT (AD=MT)
  "Start:" #NAME-START /
  "End:  " #NAME-END
  
```

Der Session-Parameter `AD` steht für "Attribute Definition" (Attributdefinition). Sein Wert "M" steht für "Modifiable output field" (modifizierbares Ausgabefeld) und der Wert "T" steht für "translate lowercase to uppercase" (Übersetzen von Kleinbuchstaben in Großbuchstaben).

Der Wert "M" in `AD=MT` bedeutet, dass die mit `INIT` definierten Vorgabewerte (d.h.: "ADKINSON" und "BENNETT") in den Eingabefeldern angezeigt werden. Der Benutzer kann andere Werte eingeben. Wenn Sie den Wert "M" weglassen, bleiben die Eingabefelder leer, obwohl Vorgabewerte definiert wurden.

Der Wert "T" in AD=MT bedeutet, dass jede Eingabe, die in Kleinbuchstaben gemacht wurde, vor der weiteren Verarbeitung in Großbuchstaben übersetzt wird. Dies ist wichtig, weil die Namen in der Demodatenbank komplett in Großbuchstaben definiert wurden. Wenn Sie den Wert "T" weglassen, müssen Sie alle Namen komplett in Großbuchstaben angeben. Andernfalls wird der angegebene Name nicht gefunden.

"Start:" und "End:" sind Textfelder (Bezeichnungen). Sie werden in Anführungszeichen eingegeben.

#NAME-START und #NAME-END sind Datenfelder (Eingabefelder), in denen der Benutzer den gewünschten Start- und Endnamen eingeben kann.

Der Schrägstrich (/) bedeutet, dass die nachfolgenden Felder in einer neuen Zeile angezeigt werden sollen.

Ihr Programm sollte nun folgendermaßen aussehen:

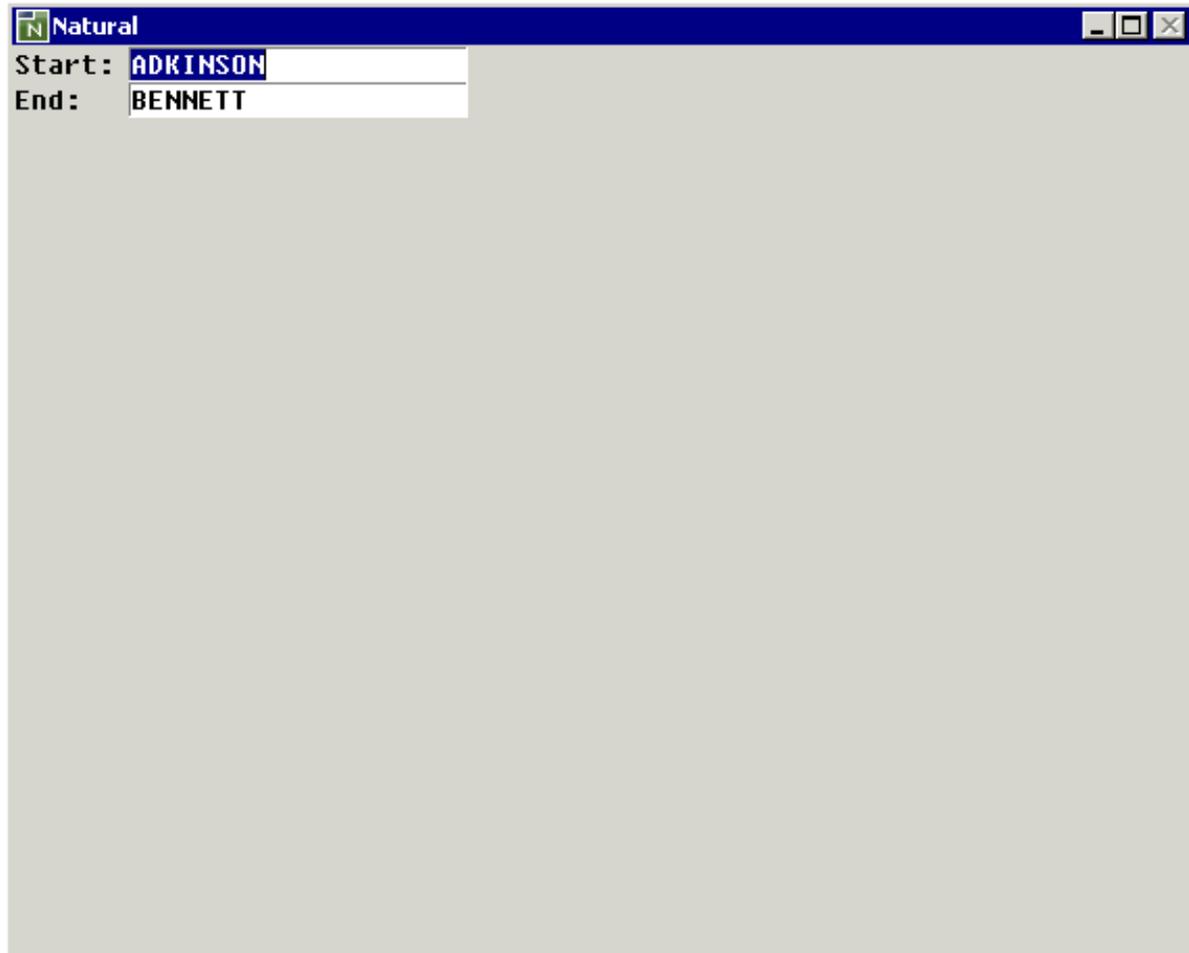
```

DEFINE DATA
LOCAL
  1 #NAME-START          (A20) INIT <"ADKINSON">
  1 #NAME-END            (A20) INIT <"BENNETT">
  1 EMPLOYEES-VIEW VIEW OF EMPLOYEES
    2 FULL-NAME
      3 NAME (A20)
    2 DEPT (A6)
    2 LEAVE-DATA
      3 LEAVE-DUE (N2)
END-DEFINE
*
INPUT (AD=MT)
  "Start:" #NAME-START /
  "End:  " #NAME-END
*
READ EMPLOYEES-VIEW BY NAME
  STARTING FROM #NAME-START
  ENDING AT #NAME-END
*
  DISPLAY NAME 3X DEPT 3X LEAVE-DUE
*
END-READ
*
END

```

2. Führen Sie das Programm mit RUN aus.

Die Ausgabe enthält die Felder, die Sie gerade definiert haben.



3. Behalten Sie die Vorgabewerte bei und drücken Sie EINGABE.

Die Liste der Mitarbeiter wird nun angezeigt.

4. Drücken Sie wiederholt EINGABE bis Sie wieder im Programmeditor sind, oder drücken Sie ESC.
5. Speichern Sie das Programm mit STOW.

Map für die Benutzereingabe gestalten

Sie lernen jetzt eine weitere Möglichkeit kennen, den Benutzer zur Eingabe aufzufordern. Sie werden mit dem Map-Editor eine Map erstellen, die dieselben Felder enthält, die Sie vorher in Ihrem Programm definiert haben. Eine Map ist ein separates Objekt; sie wird benutzt, um das Layout der Benutzeroberfläche von der Geschäftslogik der Anwendung zu trennen.

Die Map, die Sie jetzt erstellen werden, wird folgendermaßen aussehen:

```
XXXXXXXXXX                                     TT:TT:TT

Start: [XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX]
End:   [XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX]
```

Die erste Zeile der Map enthält Systemvariablen für das aktuelle Datum und die aktuelle Uhrzeit. Es gibt zwei Datenfelder (Eingabefelder), in denen der Benutzer einen Startnamen und einen Endnamen eingeben kann. Vor den Datenfeldern befinden sich Textfelder (Bezeichnungen).

Für die oben gezeigte Map sind die folgenden Schritte erforderlich:

- Map erstellen
- Textfelder definieren
- Bezeichnungen für die Textfelder definieren
- Datenfelder definieren
- Namen und Attribute für Datenfelder definieren
- Systemvariablen einfügen
- Map testen
- Map mit STOW speichern

Map erstellen

Sie werden jetzt den Map-Editor aufrufen, mit dem Sie das Layout Ihrer Map gestalten.

Lassen Sie den Programmeditor im Hintergrund offen.

Map erstellen

1. Markieren Sie im Library-Workspace die Library, die auch Ihr Programm enthält (d.h. markieren Sie den Knoten **TUTORIAL**).
2. Wählen Sie aus dem Kontextmenü den Befehl **New Source > Map**.

Oder:

Wählen Sie die folgende Schaltfläche in der Symbolleiste:



Ein leeres Map-Editor-Fenster erscheint.

Textfelder definieren

Sie werden jetzt zwei Textfelder (auch Konstanten oder Bezeichnungen genannt) in der Map definieren.

► Textfelder definieren

1. Wählen Sie aus dem Menü **Insert** den Befehl **Text Constant**.

Anmerkung:

Wenn der Map-Editor aktiv ist, werden andere Menüs in der Menüleiste angezeigt. Jetzt werden die Menüs **Insert**, **Field** und **Map** angezeigt (statt des Menüs **Programs**, welches zu sehen war als der Programmeditor aktiv war).

Oder:

Wählen Sie die folgende Schaltfläche in der Symbolleiste:



Anmerkung:

Standardmäßig wird die Symbolleiste, in der diese Schaltfläche enthalten ist, in vertikaler Ausrichtung rechts neben dem Map-Editor-Fenster angezeigt. Als der Programmeditor aktiv war, wurde dort eine andere Symbolleiste angezeigt.

2. Gehen Sie mit der Maus an die Stelle im Map-Editor-Fenster, an der die Textkonstante eingefügt werden soll.

Der Mauszeiger verändert sich. Er zeigt jetzt ein Kreuz und das Symbol für eine Textkonstante.

3. Drücken Sie die Maustaste und halten Sie sie gedrückt.
4. Ziehen Sie die Maus solange nach rechts bis das Feld die gewünschte Länge hat. Für unser Feld ist eine Länge von etwa 10 Zeichen ausreichend.

Während Sie die Maus ziehen, wird die aktuelle Länge in der Statusleiste des Anwendungsfensters angezeigt. Format und Position in der Map werden dort ebenfalls angezeigt. Ein Textfeld hat immer das Format A (alphanumerisch).

5. Lassen Sie die Maustaste los.

Jetzt wird ein Textfeld mit einer Standardbezeichnung angezeigt. Der Anfasser lässt erkennen, dass das Textfeld markiert ist. Der Mauszeiger zeigt noch immer ein Kreuz und das Symbol für eine Textkonstante; Sie können sofort ein weiteres Textfeld erstellen.

Anmerkung:

Wenn Sie diesen Modus verlassen wollen, können Sie eine beliebige Position in der Map anklicken.

- Erstellen Sie ein zweites Textfeld unter dem ersten Textfeld.

Falls das Feld in der falschen Spalte beginnt, können Sie es verschieben, indem Sie den Mauszeiger auf dieses Feld stellen, die Maustaste drücken und gedrückt halten, und die Maus dann an die gewünschte Position ziehen. Wenn Sie die Maustaste loslassen, wird wieder der normale Mauszeiger angezeigt.

Bezeichnungen für die Textfelder definieren

Die Textfelder, die Sie eben erstellt haben, haben noch nicht die richtigen Bezeichnungen. Diese werden Sie jetzt definieren.

▶ Bezeichnungen definieren

- Markieren Sie das erste Textfeld und wählen Sie aus dem Kontextmenü den Befehl **Definition**.

Oder:

Klicken Sie das Textfeld doppelt an.

Der bestehende Text wird markiert.

- Geben Sie "Start:" als Bezeichnung für das erste Textfeld ein und drücken Sie EINGABE.

Das Textfeld (für das Sie vorher eine Länge von 10 Zeichen definiert haben) wird automatisch auf die Länge dieser Zeichenkette gebracht.

- Wiederholen Sie die Schritte oben und geben Sie "End:" als Bezeichnung für das zweite Textfeld ein.

Datenfelder definieren

Sie werden jetzt zwei Datenfelder in der Map definieren. Dies sind die Eingabefelder, in denen der Benutzer den Start- und Endnamen eingeben kann.

Sie können die Datenfelder auf zwei verschiedene Arten definieren: mit dem Befehl **Data Field**, wobei es in Ihrer Verantwortung liegt, das Format und die Länge des Datenfelds korrekt anzugeben, oder mit dem Befehl **Import > Data Field**, wobei Sie das Datenfeld einfach aus einer Liste auswählen und das Format und die Länge bereits korrekt definiert sind. Diese beiden Arten sind unten beschrieben.

▶ Datenfeld definieren und dabei die Länge selbst angeben

- Wählen Sie aus dem Menü **Insert** den Befehl **Data Field**.

Oder:

Wählen Sie die folgende Schaltfläche in der Symbolleiste:



- Erstellen Sie das Datenfeld rechts neben dem zuvor eingegebenen Textfeld für den Startnamen. Sie erstellen das Datenfeld auf dieselbe Weise wie ein Textfeld. Für das Datenfeld ist eine Länge von 20 Zeichen erforderlich (falls Ihr Feld zu kurz oder zu lang ist, erfahren Sie in einer späteren Übung, wie Sie dies ändern können). Das Datenfeld wird automatisch mit mehreren "X"-Zeichen gefüllt.

► Datenfeld importieren

1. Wählen Sie aus dem Menü **Insert** den Befehl **Import > Data Field**.

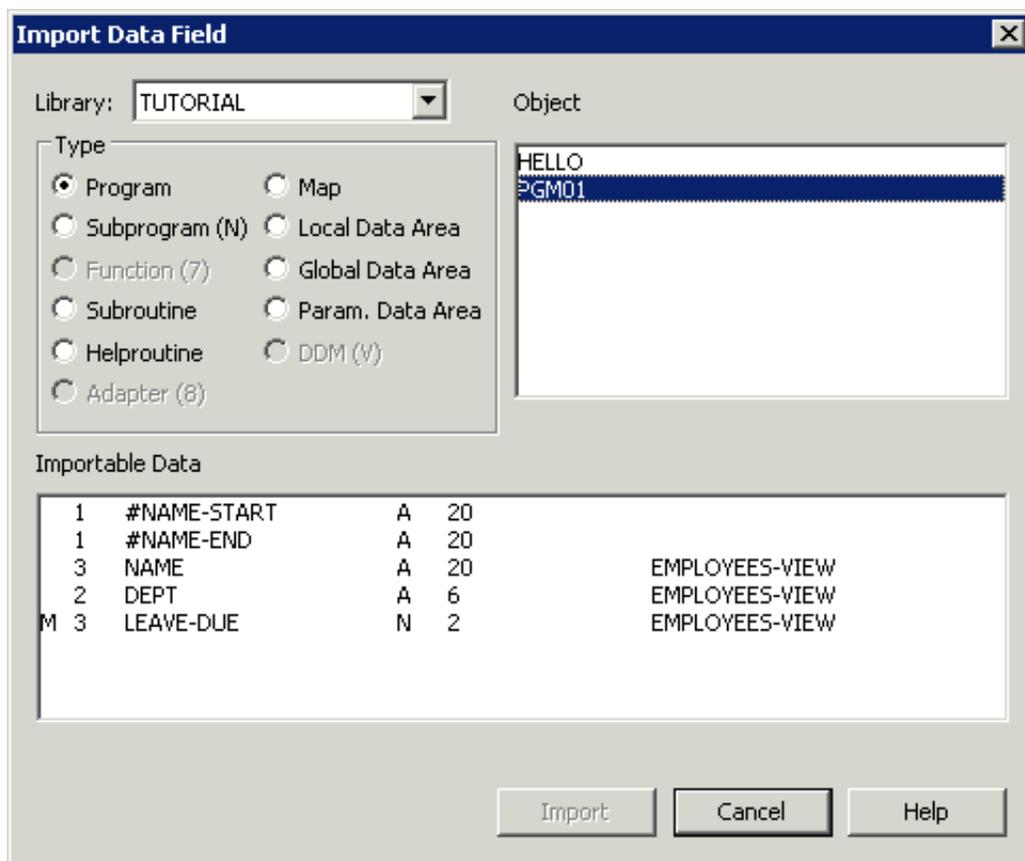
Das Dialogfeld **Import Data Field** erscheint.

2. Wählen Sie aus dem Dropdown-Listenfeld **Library** den Eintrag **TUTORIAL**.
3. Markieren Sie das Optionsfeld **Program**.

Alle Programme, die zurzeit in Ihrer Library definiert sind, werden jetzt im Listenfeld **Object** angezeigt.

4. Markieren Sie das Programm mit dem Namen PGM01.

Die importierbaren Felder werden jetzt unten im Dialogfeld angezeigt.



5. Markieren Sie das Feld **#NAME-END** und wählen Sie die Befehlsschaltfläche **Import**.

Der Name der Befehlsschaltfläche **Cancel** im Dialogfeld **Import Data Field** ändert sich in **Quit**.

6. Wählen Sie die Befehlsschaltfläche **Quit**, um das Dialogfeld **Import Data Field** zu schließen.

Das Datenfeld wird jetzt oben links in der Map angezeigt. Es ist mit "X"-Zeichen gefüllt. Anfassers weisen darauf hin, dass das Datenfeld markiert ist.

7. Verschieben Sie das Datenfeld, so dass es rechts neben dem zuvor eingegebenen Textfeld für den Endnamen erscheint. Hierzu stellen Sie den Mauszeiger auf das Feld, drücken die Maustaste und halten sie gedrückt, ziehen die Maus an die gewünschte Position und lassen die Maustaste wieder los.

Namen und Attribute für Datenfelder definieren

Folgendes gilt nur für das Datenfeld für den Startnamen, das Sie manuell definiert haben. Es gilt nicht für das Datenfeld für den Endnamen, das Sie importiert haben: Wenn Sie ein neues Datenfeld für eine Benutzervariable erstellen, wird von Natural ein Name für dieses Feld vergeben. Dieser Feldname enthält eine Nummer. Sie müssen die Namen der neu erstellten Felder an die Namen anpassen, die in Ihrem Programm definiert wurden.

Sie werden jetzt dafür sorgen, dass dieselben Namen wie in Ihrem Programm benutzt werden: #NAME-START und #NAME-END. Die Ausgabe dieser Felder (d.h. die Benutzereingabe) wird an die entsprechenden Benutzervariablen in Ihrem Programm übergeben.

Sie werden auch dafür sorgen, dass für #NAME-START und #NAME-END dieselben Attribute definiert sind.

▶ Namen und Attribute für Datenfelder definieren

1. Markieren Sie das Datenfeld für den Startnamen und wählen Sie aus dem Kontextmenü den Befehl **Definition**.

Oder:

Klicken Sie das Datenfeld doppelt an.

Das Dialogfeld **Field Definition** erscheint.

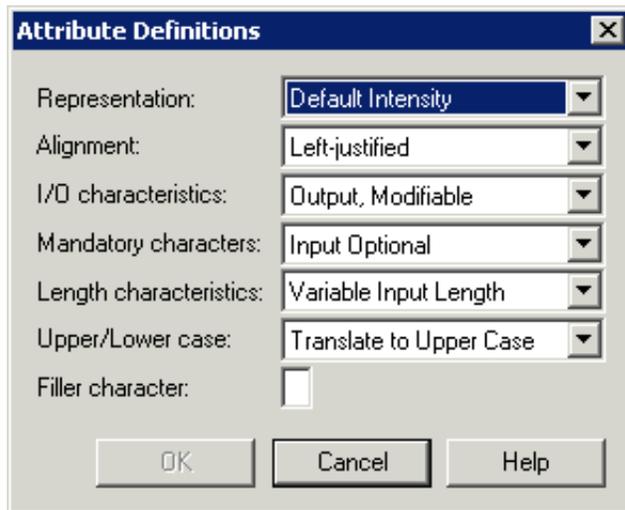
Das Textfeld **Field** enthält den Feldnamen, der von Natural vergeben wurde: #FIELD_#1.

2. Geben Sie "#NAME-START" im Textfeld **Field** ein.

Das Format muss **A** ein. Dies wird automatisch vorgegeben.

3. Geben Sie "20" im Textfeld **Length** ein (falls Ihr Feld eine andere Länge hat).
4. Wählen Sie die Befehlsschaltfläche **Attributes**.

Das Dialogfeld **Attribute Definitions** erscheint.



5. Achten Sie darauf, dass im Dropdown-Listenfeld **I/O characteristics** der Eintrag **Output, Modifiable** ausgewählt ist.

Hiermit wird das Feld als modifizierbares Ausgabefeld definiert.

6. Achten Sie darauf, dass im Dropdown-Listenfeld **Upper/Lower case** der Eintrag **Translate to Upper Case** ausgewählt ist.

Dies ermöglicht es dem Benutzer, den Namen in Kleinbuchstaben einzugeben. Bis jetzt konnten die Namen in der Demodatenbank nur gefunden werden, wenn die Namen komplett in Großbuchstaben eingegeben wurden.

7. Geben Sie im Textfeld **Filler character** einen Unterstrich (_) ein.

Standardmäßig ist ein Leerzeichen als Füllzeichen definiert. Deshalb müssen Sie das Leerzeichen zuerst löschen, bevor Sie den Unterstrich eingeben können.

Füllzeichen werden in der Map benutzt, um leere Positionen in den Eingabefeldern zu füllen, damit der Benutzer bei der Dateneingabe die exakte Position und Länge eines Feldes erkennen kann.

8. Wählen Sie die Befehlsschaltfläche **OK**, um das Dialogfeld **Attribute Definitions** zu schließen.
9. Wählen Sie die Befehlsschaltfläche **OK**, um das Dialogfeld **Field Definition** zu schließen.
10. Wiederholen Sie die oben aufgeführten Schritte für den Endnamen. Achten Sie darauf, dass der richtige Feldname (#NAME-END), das richtige Format (A) und die richtige Länge (20) definiert sind. Achten Sie darauf, dass dieselben Attributdefinitionen wie bei #NAME-START benutzt werden.

Systemvariablen einfügen

Natural-Systemvariablen enthalten Information über die aktuelle Natural-Session, wie zum Beispiel die aktuelle Library, Benutzer, oder Datum und Uhrzeit. Diese Informationen können an jedem beliebigen Punkt in einem Natural-Programm referenziert werden. Alle Systemvariablen beginnen mit einem Stern (*).

Sie werden jetzt Systemvariablen für Datum und Uhrzeit in der Map einfügen. Wenn das Programm mit RUN ausgeführt wird, werden das aktuelle Datum und die aktuelle Uhrzeit in der Map angezeigt.

▶ Systemvariablen einfügen

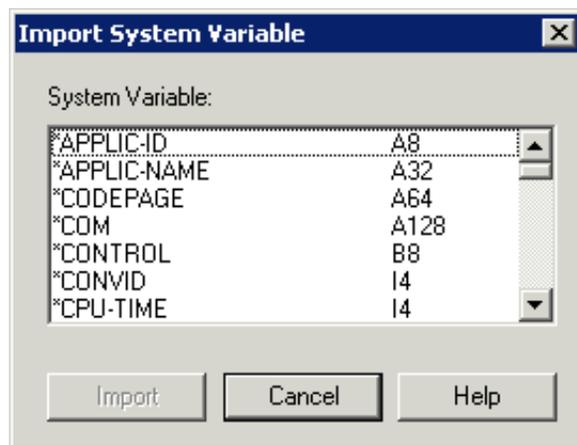
1. Wählen Sie aus dem Menü **Insert** den Befehl **Import > System Variable**.

Oder:

Wählen Sie die folgende Schaltfläche in der Symbolleiste:



Das Dialogfeld **Import System Variable** erscheint.



2. Rollen Sie die Liste bis zu *DAT4I und markieren Sie diesen Eintrag.
3. Rollen Sie die Liste bis zu *TIMX, drücken Sie STRG und markieren Sie diesen Eintrag.
4. Wählen Sie die Befehlsschaltfläche **Import**, um die markierten Variablen zu importieren.

Die Befehlsschaltfläche **Cancel** im Dialogfeld **Import System Variable** trägt nun den Namen **Quit**.

5. Wählen Sie die Befehlsschaltfläche **Quit**, um das Dialogfeld zu schließen.

Beide Systemvariablen sind nun oben links in der Map zu sehen.

6. Markieren Sie **TT:TT:TT** (d.h. die Systemvariable für die Uhrzeit) und verschieben Sie dies an das Ende der ersten Zeile.

Anmerkung:

Es ist unter Umständen erforderlich, das Map-Editor-Fenster zu vergrößern, damit Sie das Ende der Zeile sehen können.

Map testen

Um zu überprüfen, ob Ihre Map wie beabsichtigt funktioniert, werden Sie sie jetzt testen.

▶ Map testen

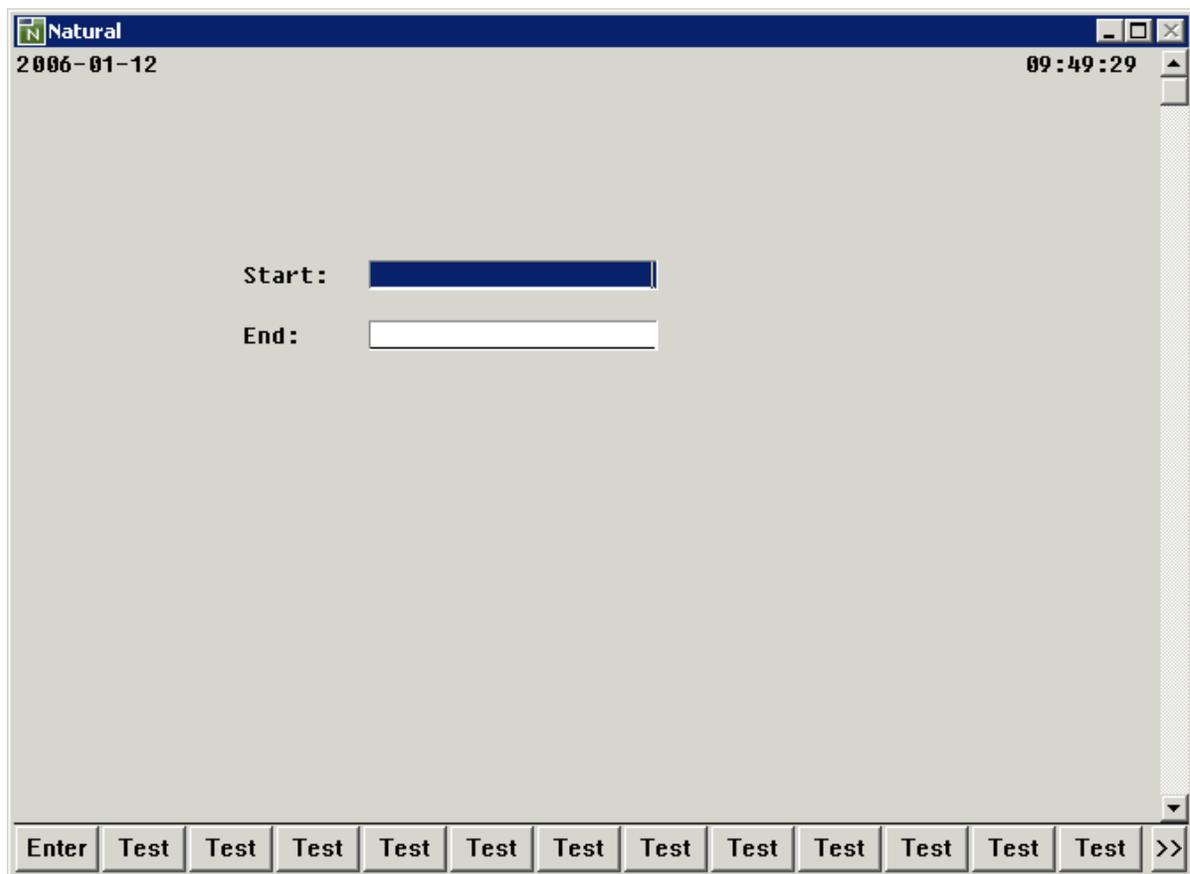
1. Wählen Sie aus dem Menü **Object** den Befehl **Test**.

Oder:

Wählen Sie die folgende Schaltfläche in der Symbolleiste:



Die folgende Ausgabe wird angezeigt.



Anmerkung:

Es ist eventuell erforderlich, das Ausgabefenster zu vergrößern, damit Sie das Datum oben rechts sehen können.

Das Eingabefeld für den Startnamen ist automatisch markiert, da es das erste Eingabefeld in der Map ist. Beide Eingabefelder enthalten Füllzeichen.

Anmerkung:

Wenn der Einfügemodus aktiv ist, muss der Benutzer die Füllzeichen löschen, damit die Eingabe von Text möglich ist. Im Überschreibemodus ist dies nicht erforderlich.

2. Drücken Sie EINGABE, um zum Map-Editor zurückzukehren.

Map mit STOW speichern

Wenn die Map erfolgreich getestet wurde, müssen Sie sie mit STOW speichern, damit sie von Ihrem Programm gefunden werden kann.

▶ Map mit STOW speichern

1. Speichern Sie die Map auf dieselbe Weise mit STOW, wie Sie ein Programm speichern.
2. Wenn Sie nach einem Namen für die Map gefragt werden, geben Sie "MAP01" ein.

Im Library-Workspace erscheint ein neuer Knoten mit dem Namen **Maps** als Unterknoten von **TUTORIAL**. Dieser Unterknoten enthält die Map, die Sie eben gespeichert haben.

Lassen Sie den Map-Editor für spätere Änderungen offen.

Map aus dem Programm aufrufen

Sobald eine Map mit STOW gespeichert wurde, kann Sie mit einem WRITE- oder INPUT-Statement aus einem Natural-Programm heraus aufgerufen werden.

▶ Map aus dem Programm aufrufen

1. Kehren Sie zum Programmeditor zurück.

Falls Sie den Programmeditor nicht sehen können (weil Sie vorher zum Beispiel das Fenster für den Map Editor vergrößert haben), können Sie zu einem offenen Programmeditorfenster zurückkehren, indem Sie den entsprechenden Befehl für PGM01 aus dem Menü **Window** wählen.

Sie können auch PGM01 im Library-Workspace doppelt anklicken (oder wenn Sie mit der Tastatur arbeiten, können Sie es markieren und EINGABE drücken). Wenn das Programm zuvor geschlossen wurde, wird es dadurch wieder geöffnet. Wenn es im Hintergrund noch immer geöffnet ist, wird das Editorfenster dadurch wieder in den Vordergrund geholt.

2. Ersetzen Sie die vorher definierten INPUT-Zeilen durch die folgende Zeile:

```
INPUT USING MAP 'MAP01'
```

Hiermit wird die von Ihnen erstellte Map aufgerufen.

Der Name der Map muss in einfache Anführungszeichen gesetzt werden, um die Map von einer Benutzervariablen zu unterscheiden.

Ihr Programm sollte nun folgendermaßen aussehen:

```
DEFINE DATA
LOCAL
  1 #NAME-START          (A20) INIT <"ADKINSON">
  1 #NAME-END            (A20) INIT <"BENNETT">
  1 EMPLOYEES-VIEW VIEW OF EMPLOYEES
    2 FULL-NAME
      3 NAME (A20)
```

```

      2 DEPT (A6)
      2 LEAVE-DATA
        3 LEAVE-DUE (N2)
END-DEFINE
*
INPUT USING MAP 'MAP01'
*
READ EMPLOYEES-VIEW BY NAME
  STARTING FROM #NAME-START
  ENDING AT #NAME-END
*
  DISPLAY NAME 3X DEPT 3X LEAVE-DUE
*
END-READ
*
END

```

3. Führen Sie das Programm mit RUN aus.

Ihre Map wird nun angezeigt.

4. Drücken Sie wiederholt EINGABE bis Sie zum Programmeditor zurückkehren, oder drücken Sie ESC.
5. Speichern Sie das Programm mit STOW.

Immer einen Endnamen benutzen

So wie Ihr Programm jetzt kodiert ist, werden keine Daten gefunden, wenn kein Endname angegeben wird.

Sie werden jetzt die Anfangswerte für Start- und Endname entfernen; danach müssen diese Namen immer vom Benutzer angegeben werden. Um zu gewährleisten, dass ein Endname immer benutzt wird, auch wenn er vom Benutzer nicht eingegeben wurde, werden Sie Ihrem Programm nun ein entsprechendes Statement hinzufügen.

Den Endnamen benutzen

1. Gehen Sie zum DEFINE DATA-Block und entfernen Sie die Anfangswerte (INIT) für die Felder #NAME-START und #NAME-END, so dass die entsprechenden Zeilen folgendermaßen aussehen:

```

1 #NAME-START      (A20)
1 #NAME-END        (A20)

```

2. Geben Sie Folgendes unter INPUT USING MAP 'MAP01' ein:

```

IF #NAME-END = ' ' THEN
  MOVE #NAME-START TO #NAME-END
END-IF

```

Wenn das Feld #NAME-END leer ist (d.h. wenn der Benutzer keinen Endnamen angibt), wird der Startname automatisch als Endname benutzt.

Anmerkung:

Statt des Statements MOVE #NAME-START TO #NAME-END können Sie auch die folgende Variante des ASSIGN- oder COMPUTE-Statements benutzen: #NAME-END := #NAME-START.

Ihr Programm sollte nun folgendermaßen aussehen:

```

DEFINE DATA
LOCAL
  1 #NAME-START          (A20)
  1 #NAME-END            (A20)
  1 EMPLOYEES-VIEW VIEW OF EMPLOYEES
    2 FULL-NAME
      3 NAME (A20)
    2 DEPT (A6)
    2 LEAVE-DATA
      3 LEAVE-DUE (N2)
END-DEFINE
*
INPUT USING MAP 'MAP01'
*
IF #NAME-END = ' ' THEN
  MOVE #NAME-START TO #NAME-END
END-IF
*
READ EMPLOYEES-VIEW BY NAME
  STARTING FROM #NAME-START
  ENDING AT #NAME-END
*
  DISPLAY NAME 3X DEPT 3X LEAVE-DUE
*
END-READ
*
END

```

3. Führen Sie das Programm mit RUN aus.
4. Geben Sie in der daraufhin erscheinenden Map "JONES" in dem Feld für den Startnamen ein und drücken Sie EINGABE.

Anmerkung:

Da **Translate to Upper Case** für dieses Feld definiert wurde, können Sie den Namen jetzt auch in Kleinbuchstaben angeben.

In der daraufhin erscheinenden Liste werden jetzt nur die Mitarbeiter mit dem Namen "Jones" angezeigt.

5. Drücken Sie EINGABE, um zum Programmeditor zurückzukehren.
6. Speichern Sie das Programm mit STOW.

Sie können nun mit den nächsten Übungen fortfahren: *Verarbeitungsschleifen und Labels*.