Hello World!

This chapter contains the following exercises:

- Creating a Program
- Running a Program
- Correcting Program Errors
- Stowing a Program
- Displaying Information about a Program
- Displaying the Content of the Current Library
- Setting the Editor Profile Options

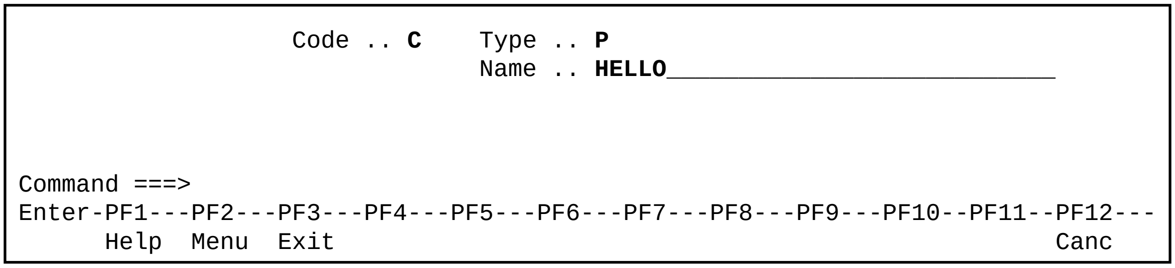
Creating a Program

You will now write your first short program which displays "Hello World!". It will be stored in the library you have created previously.

To create a new program

1. Make sure that you have logged on to the library named `TUTORIAL`.

2. At the bottom of the **Development Functions** menu, enter the following information and press ENTER:



```
                   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" stands for the function **Create Object**, "P" stands for the object type program, and "HELLO" is the name of the program to be created.

Tip:
When you enter the function code `C`, you can also enter an asterisk (*) in the **Type** field. When you press ENTER, a list of all object types and the letters that correspond to these object types is shown.

The program editor appears. It is currently empty.

3. Enter the following code in the program editor:

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

Comment lines start with an asterisk (*) followed by at least one blank or a second asterisk. When you forget to enter the blank or second asterisk, Natural assumes that you have specified a system variable; this will result in an error.

If you want to insert empty lines in your program, you should define them as comment lines. This is helpful, if you want to access your program from different platforms (Windows, mainframe, UNIX or OpenVMS). With the mainframe version of Natural, for example, the default is that empty lines are automatically deleted when you press ENTER.

You can also insert comments at the end of a statement line. In this case, the comment starts with a slash followed by an asterisk (/*).

The text that is to be shown in the output is defined with the `DISPLAY` statement. It is enclosed in quotation marks.

The `END` statement is used to mark the physical end of a Natural program. Each program must end with `END`.

When you press ENTER, it may happen that all of your lower-case characters are translated to upper-case characters. This behavior is defined in the editor profile (which is explained later).

Running a Program

The system command `RUN` automatically invokes the system command `CHECK` which checks the program code for errors. If no error is found, the program is compiled on the fly and then executed.

Notes:

1. `CHECK` is also available as a separate command.
2. Natural also provides the system command `EXECUTE` which uses the stowed version of your program (stowing a program is explained later in this tutorial). In contrast to this, the `RUN` command always uses your latest modifications to the program.

To run a program

1. In the program editor's command line, enter one of the following:

```
RUN
```

```
R
```

System commands may be abbreviated. `R` is the abbreviated form of `RUN`.

Depending on the definitions in your environment, the command line is located either at the top or bottom of the screen.

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

When your code is syntactically correct, the output contains the text you have defined.

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


Hello world!
```

2. Press ENTER to return to the program editor.

Correcting Program Errors

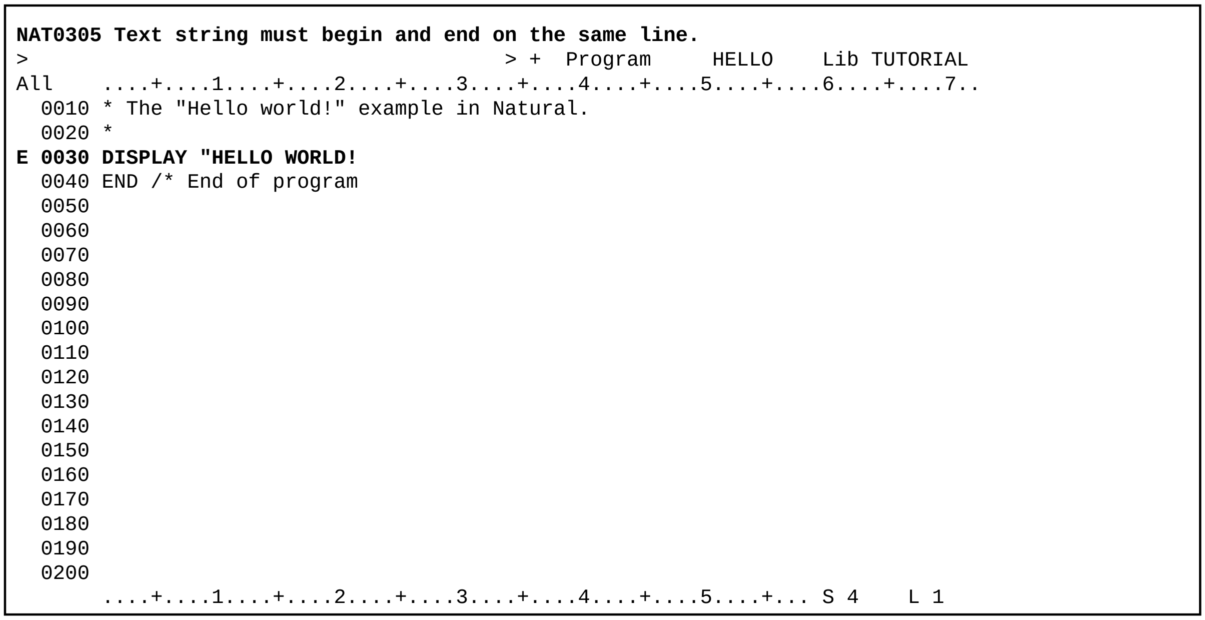
You will now create an error in your Hello World program and then run the program once more.

To correct an error

1. Delete the second quotation mark in the line containing the `DISPLAY` statement.

2. Run the program once more as described above.

   When the error is found, an error message is displayed.



```
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
```

The statement line that contains the error is highlighted and marked with an "E".

3. Correct the error, that is: insert the missing quotation mark at the end of the line.
4. Run the program once more to find the next error.

   In this case, no more errors are found and the output is shown.
5. Press ENTER to return to the program editor.

Stowing a Program

When you stow a program, it is compiled and both source code and a generated program are stored in the Natural system file.

Like the `RUN` command, the system command `STOW` automatically invokes the `CHECK` command. A program is only stowed when it is syntactically correct.

Note:
If you want to save the changes to your program, even if the program contains a syntactical error (for example, if you want to suspend your work until the next day), you can use the system command `SAVE`.

To stow a program

- In the program editor’s command line, enter the following:

```
STOW
```

Displaying Information about a Program

The `LIST` command is useful to find out whether only the source code or both source code and a generated program are available for an object.

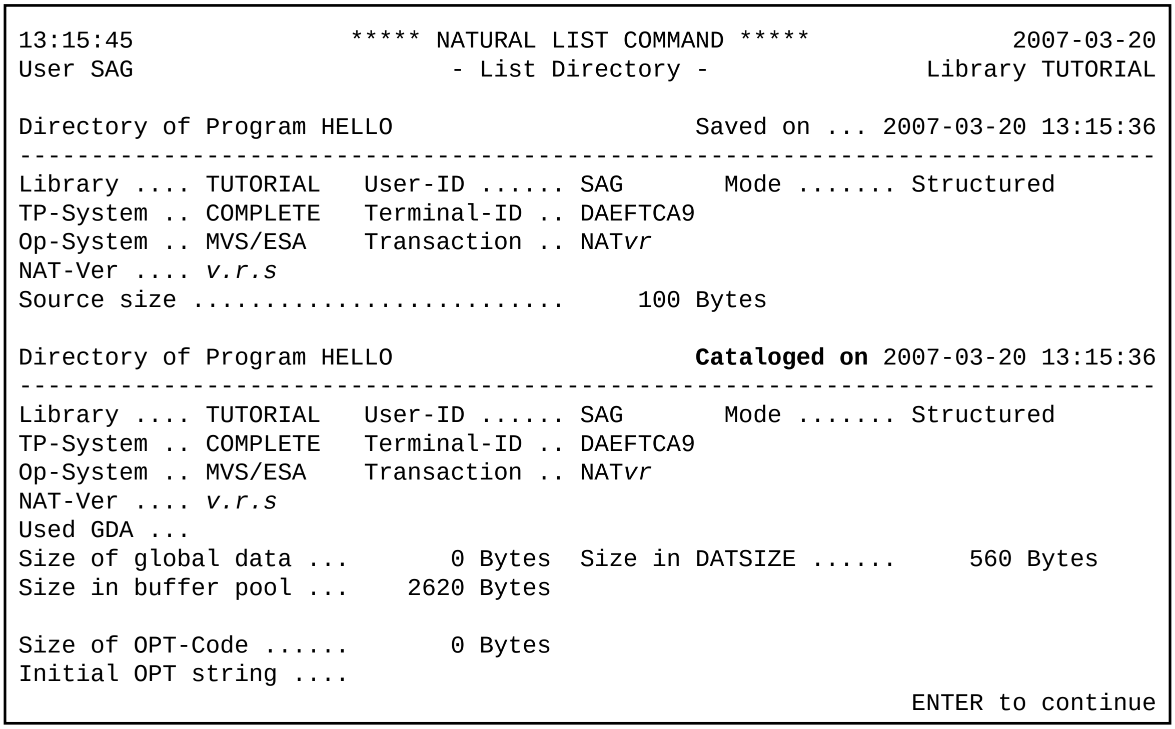
To display information about a program

1. In the program editor’s command line, enter one of the following:

```
LIST DIR HELLO
```

```
L DIR HELLO
```

The following screen appears. The information provided with **Cataloged on** is only available when the object has been stowed.



```
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
```

Note:
In the above example, the notations *vr* and *v.r.s* stand for the current version number of Natural. See also the definition of *Version* in the *Glossary*.

2. Press ENTER to return to the program editor.

Displaying the Content of the Current Library

The `LIST` command can also be used to display a list of all Natural objects in the current library. This is helpful, for example, if you decide at some point during this tutorial that you want to delete one or more of your Natural objects in order to start again from the very beginning.

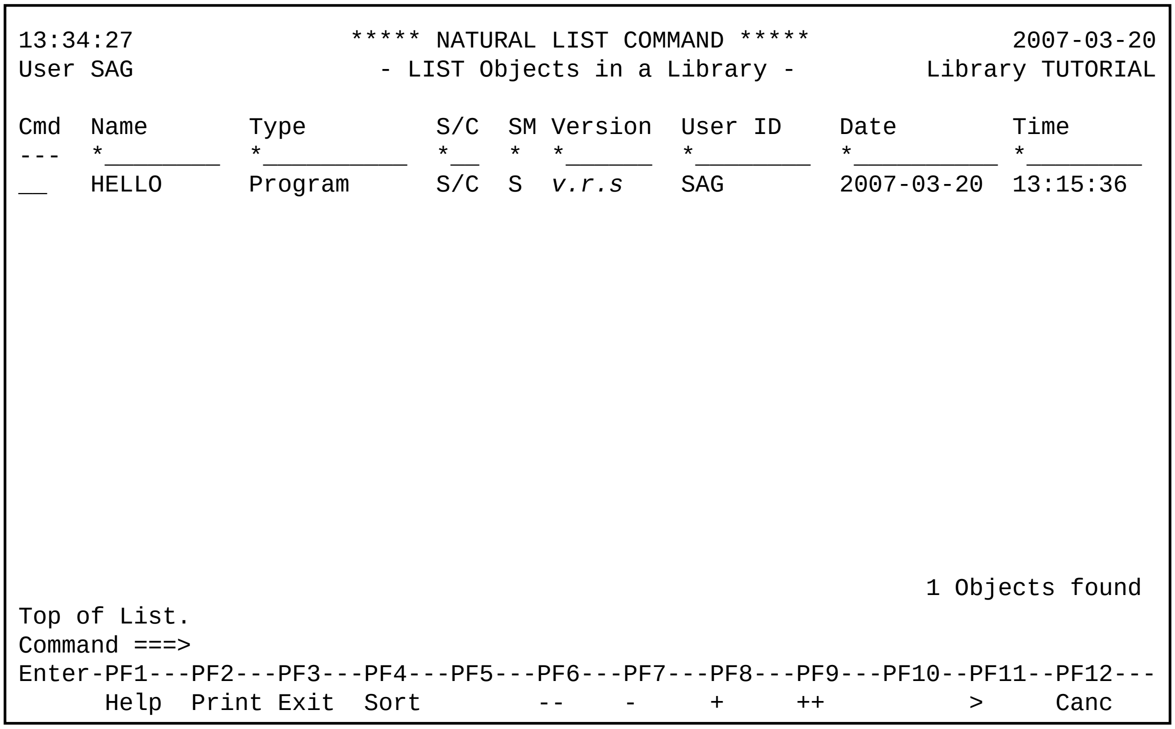
To display a list of Natural objects

1. In the program editor's command line, enter one of the following:

   ```
   LIST *
   ```

   ```
   L *
   ```

   The following screen appears. It lists the program you have just created.

   ```
   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. To find out which commands are available, enter a question mark (?) in the **Cmd** column next to your program.

   The following window appears.

```
+-------- COMMANDS ---------+
!                           !
!   ED   Edit               !
!   LI   List               !
!   LD   List Dir           !
!   PR   Print              !
!   LE   List expanded      !
!   RU   Run                !
!   ST   Stow               !
!   CA   Catalog            !
!   DE   Delete             !
!   RE   Rename             !
!   .    End                !
!                           !
!                           !
!                           !
!                           !
!                           !
!   __ HELLO                !
+---------------------------+
```

3. Do not apply any changes right now. Press PF3 to close the window without specifying any command.

4. Press PF3 once more to return to the program editor.

Setting the Editor Profile Options

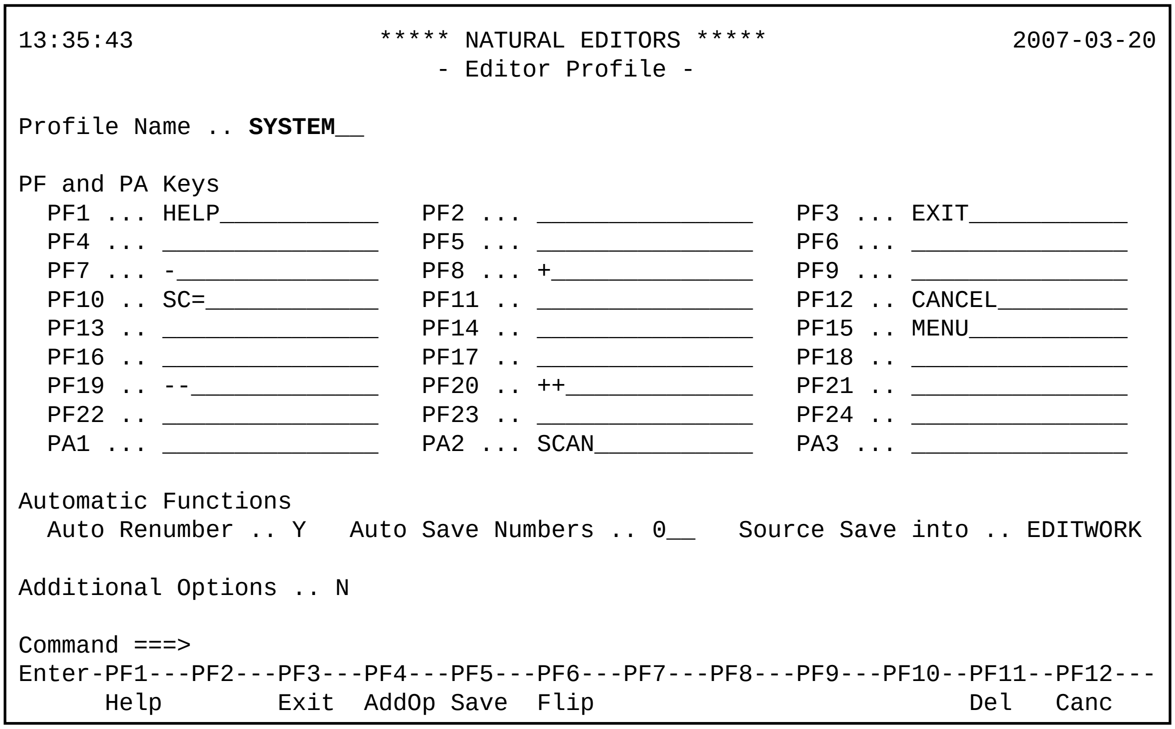
When working with the Natural program editor or data area editor, an editor profile can be defined per user. This tutorial uses the default settings of the editor profile named `SYSTEM`. Some important settings are mentioned below.

▶ **To check the editor profile options**

1. In the program editor’s command line, enter the following:

   ```
   PROFILE
   ```

   The following screen appears.



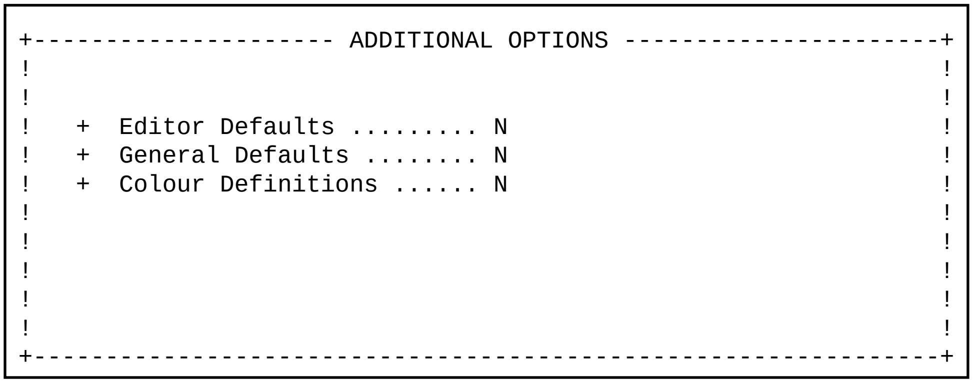
```
13:35:43                 ***** NATURAL EDITORS *****                 2007-03-20
                              - Editor Profile -

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
```

When a user-specific editor profile does not exist, the default profile SYSTEM is displayed. This default profile can be used to create a user-specific profile. When a user-specific profile exists already, it is displayed instead of the SYSTEM profile.

2. In the **Additional Options** field, enter "Y" and press ENTER.

   Or:
   Press PF4.

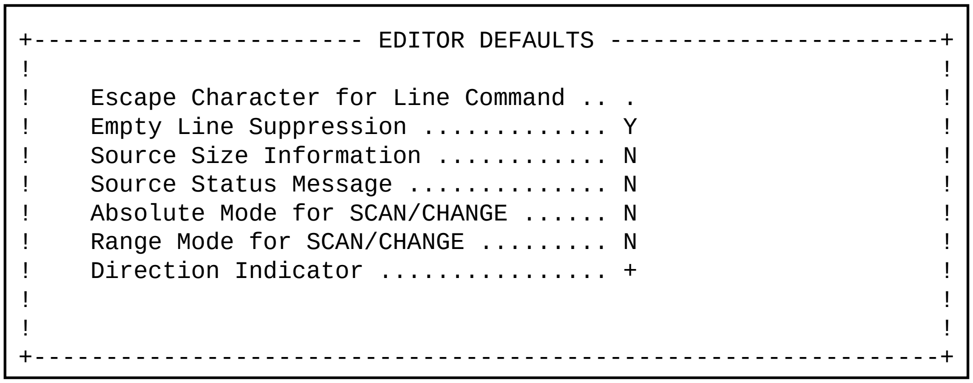
   The following window appears.



```
+--------------------- ADDITIONAL OPTIONS ----------------------+
!                                                               !
!                                                               !
!   +  Editor Defaults ......... N                              !
!   +  General Defaults ........ N                              !
!   +  Colour Definitions ...... N                              !
!                                                               !
!                                                               !
!                                                               !
!                                                               !
!                                                               !
+---------------------------------------------------------------+
```

3. Enter "Y" in the fields **Editor Defaults** and **General Defaults**, and press ENTER.

   The following window appears for the editor defaults.



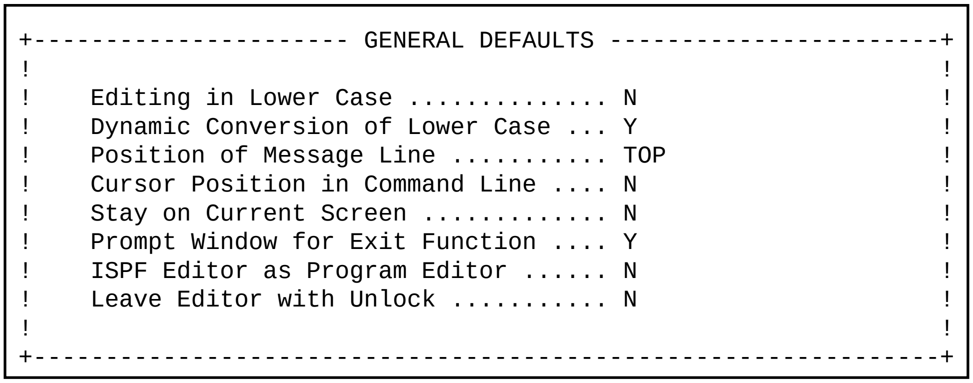
```
+----------------------- EDITOR DEFAULTS -----------------------+
!                                                               !
!    Escape Character for Line Command .. .                     !
!    Empty Line Suppression ............. Y                     !
!    Source Size Information ............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

You can see the escape character that has been defined for line commands. This tutorial assumes that the default character, which is the a period (.), is used.

This tutorial also assumes that the option **Empty Line Suppression** is set to "Y". In this case, all blank lines in the program editor are automatically deleted when you press ENTER. They are not deleted when this option is set to "N".

4. For this tutorial, you should make sure that all options are set as shown above. Press ENTER to display the next window.

   The following window appears.



```
+---------------------- 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                     !
!                                                               !
+---------------------------------------------------------------+
```

When the option **Editing in Lower Case** is set to "Y" and the option **Dynamic Conversion of Lower Case** is set to "N", any source code remains as you enter it. This feature, however, also depends on system-environment-specific settings which may force an uppercase translation of all of your input; this cannot be influenced by Natural.

5. If desired, change the above mentioned options for lowercase conversion and press ENTER. Press ENTER once more to return to the **Additional Options** window, and then press ENTER again to close this window.

6. When a user-specific profile has not yet been created, overwrite the profile name `SYSTEM` with your user ID and press ENTER.

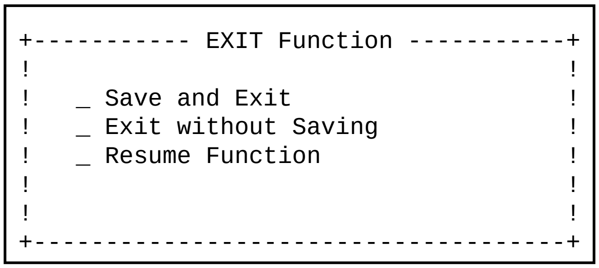
   When a user-specific profile exists already, proceed with the next step.

7. Press PF5 to save your changes in the database and then press PF3 to exit the editor profile.

Note:
Instead of pressing a PF key, you can also enter the corresponding command in the command line. For example, in the above case, you can enter the commands `SAVE` and `EXIT`.

Or:
If you do not want to apply any changes, press PF3 to exit the editor profile.

The exit function displays a window with different options. It even appears if you have just saved your changes.



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

8. When you have saved your changes immediately before invoking the exit function, you can safely select the option **Exit without Saving**. Press ENTER to return to the program editor.

   Caution:
   If you have made further changes after pressing PF5 or issuing the `SAVE` command and now select the option **Exit without Saving**, your latest changes are only valid for the current session; they are not saved on the database.

   Your program is shown again. Any new settings will now be used in the program editor (and also in the data area editor which is explained later).

You can now proceed with the next exercises: *Database Access*.