

Using Commands and Menu Functions

You can perform an operation during a Natural session by using a Natural command or a menu function. When using a Natural command, you can directly perform an operation without navigating through different menus.

A Natural object of the type program can also be executed without using a Natural command as mentioned in *Executing Programs*.

This section describes the different categories of commands provided with Natural, how to execute a command and how to use a menu function.

Related Topic:

- *Natural Main Menu*

This section covers the following topics:

- Categories of Natural Commands
 - Using Natural Commands
 - Using Natural Menus
-

Categories of Natural Commands

This section describes the different categories of Natural commands:

- System Commands
- Terminal Commands
- Editor and Utility Commands

System Commands

Natural system commands perform functions you need in order to create, maintain or execute a Natural object. In addition, Natural system commands are used to monitor and administer your Natural environment.

Related Topics in the System Commands Documentation:

- *System Commands (Overview)*
- *System Commands Grouped by Functions*
- *System Command Syntax*

Terminal Commands

Natural terminal commands, for example, can be used for the following:

- Arrange the screen display and layout such as the positioning of the PF-key and message line and the assignment of colors.
- Obtain debug information on the current environment.
- Interrupt a current Natural operation.

You can invoke a terminal command while an application is executing. In addition to the Natural command prompts, terminal commands can be entered in any alphanumeric input field. A terminal command starts with a control character that can be specified by setting a Natural session parameter. The default control character is the percent (%) sign.

Related Topics:

- *Screen Design - Programming Guide*
- *Copying Data from a Screen - Programming Guide*
- *Terminal Commands Grouped by Function - Terminal Commands* documentation
- *Terminal Commands (Overview) - Terminal Commands* documentation
- *Using Session Parameters*

Editor and Utility Commands

In addition to Natural system and Natural terminal commands, each Natural editor and Natural utility provides its own commands that only apply to this very environment. These commands are documented in the relevant section of the editor or utility documentation.

Using Natural Commands

You enter a Natural system command at any command prompt. Natural command prompts are:

- The command line in the Natural **Main Menu** or on the screen of a Natural utility or system command (see also the following example).
- An editor command prompt such as the greater than (>) sign of the program editor (see also the following example).
- The NEXT prompt and the MORE prompt.

Some utilities require that a system command is preceded by a special sign such as double forward slashes (/). For details, refer to the relevant section in the *Utilities* documentation.

Most of the system commands can be entered in conjunction with one or more individual parameters and operands that further specify the operation to be performed. See also *Example of a System Command*.

You enter a Natural terminal command at any command prompt or in any alphanumeric input field.

You enter a Natural editor or utility command at the command prompt or in the command line of the editor or utility.

The input of a Natural command is *not* case-sensitive. After you have entered a Natural command, you choose ENTER. ENTER confirms the action and executes the command or invokes an extra confirmation window where you explicitly acknowledge command execution.

This section covers the following topics:

- Command Line
- NEXT and MORE
- Example of a System Command

Command Line

The command line is located above the PF-key lines and looks as follows:

```

Command ===>

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help           Exit                               Canc

```

NEXT and MORE

The NEXT prompt appears if no Natural program (for example, the program that invokes the Natural **Main Menu**) has been started yet. The NEXT prompt indicates that Natural is awaiting your next command input.

The MORE prompt appears during the execution of a program and indicates that additional output is available. Choose ENTER to display the additional output. If you enter a command in response to the MORE prompt, the program that is being executed will be terminated and the command will be executed.

The NEXT or MORE prompt is usually located in the left upper or lower corner of the screen as shown in the example below:

```

NEXT                                     LIB=TEST

```

Example of a System Command

The following instructions are an examples of how to execute a Natural system command in order to edit an object.

To invoke a Natural editor for a Natural object

1. At any command prompt, enter the system command EDIT and one or more operands, if required.

For example:

```
EDIT P PROGX
```

where P is the type of object (program) and PROGX the name of the object to be edited.

2. Choose ENTER.

The Natural program editor is invoked and the source code of PROGX is displayed in the editing area as shown in the example below:

```
>
All      .....1.....2.....3.....4.....5.....6.....7...
0010 READ (3) EMPLOYEES BY NAME
0020 DISPLAY NAME
0030 END
0040
0050
0060
0070
0080
0090
0100
0110
0120
0130
0140
0150
0160
0170
0180
0190
0200
0210
0220
0230
0240
0250
0260
0270
0280
.....1.....2.....3.....4.....5..... S 3   L 1
```

See also the equivalent input when using the corresponding menu functions as described in *Example of a Menu Function*.

Related Topic:

- *Example of Command Syntax - System Commands* documentation

Using Natural Menus

Every Natural menu screen provides a list of functions. The way you invoke a function from a menu depends on the menu structure and the options provided.

Natural menus provide the following alternative methods that can be used to select and invoke a menu function:

- An individual input field for each function listed.
- The **Code** field, where you can enter the one- or two-character code that is assigned to each function.
- Function keys (PF keys) that correspond to a particular function in the menu.

The PF-key lines (usually located at the bottom of the screen) indicate which function is assigned to which key. See also *Standard PF Keys*.

In addition to the functions, most of the Natural menus provide fields where you can specify further options and/or selection criteria. For example, the **Development Functions** menu contains the **Name** field in which you can enter the name of a Natural object. For explanations of these fields and their valid input values, refer to the relevant documentation or use the help function as described in the section *Natural Online Help*.

This section covers the following topics:

- Performing a Menu Function
- Terminating a Menu Function
- Standard PF Keys
- Example of a Menu Function

Performing a Menu Function

This section provides instructions for performing and terminating menu functions.

Performing a Menu Function

- Place the cursor in the input field next to the menu function required and choose ENTER.
Or:
Place the cursor in the input field next to the menu function required and enter any character.
Or:
In the **Code** field, enter the one- or two-character code displayed before the function required.
Or:
If available, from the PF-key lines, choose the PF key that corresponds to the function required (see also *Standard PF Keys*).

Or:

Double-click on the input field next to the function required.

If additional input fields are displayed in the menu, enter the information required. If you fail to do so, you will receive either a window from which you can select a valid input value or a corresponding Natural error message.

For explanations of possible field entries, invoke the help function by entering a question mark (?) in the required field.

Terminating a Menu Function

The following instructions describe alternative methods you can usually use to terminate a menu function with or without saving modifications made previously on another Natural screen.

▶ To terminate a function without saving changes

- Enter a period (.) and choose ENTER.

Or:

Choose PF12.

▶ To terminate a function and save changes

- Choose PF3.

Standard PF Keys

The following function keys (PF keys) are assigned to the following functions throughout most Natural menus:

PF Key	PF-Key Name	Explanation
PF1	Help	Invokes the online help function.
PF2	Menu	Invokes the Natural Main Menu .
PF3	Exit	Terminates a function.
PF12	Canc	Terminates a function and cancels the changes made previously.

Example of a Menu Function

The following instructions are an example of how to use a Natural menu in order to edit an object.

▶ To invoke a Natural editor for a Natural object

1. On the **Development Functions** screen:

In the **Code** field, enter the one-letter code that corresponds to the function **Edit Object**.

In the **Type** field, enter the one-letter code that corresponds to the type of Natural object (in the example below: P for program).

In the **Name** field, enter the name of the Natural object (in the example below: PROGX).

```

13:33:16          ***** NATURAL *****                2009-05-20
User SAG          - Development Functions -                Library TEST
                                                           Mode Structured
                                                           Work area empty

Code  Function

C    Create Object
E    Edit Object
R    Rename Object
D    Delete Object
X    Execute Program
L    List Object(s)
S    List Subroutines Used
?    Help
.    Exit

Code .. E      Type .. P
Name .. PROGX_____

Command ===>

Enter-PF1---PF2---PF3---PF4---PF5---PF6---PF7---PF8---PF9---PF10--PF11--PF12---
      Help  Menu  Exit                                     Canc

```

2. Choose ENTER.

The Natural program editor is invoked and the source code of PROGX is displayed in the editing area as shown in *Example of a System Command*.