

## **Arguments**

<b>COLLABORATORS</b>
----------------------

	<i>TITLE :</i> Arguments		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		November 24, 2024	

<b>REVISION HISTORY</b>
-------------------------

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>Arguments</b>	<b>1</b>
1.1	Arguments . . . . .	1
1.2	TMP:Modula-2/Arguments.def . . . . .	1

# Chapter 1

## Arguments

### 1.1 Arguments

Prozeduren

GetArg

GetLock

NumArgs

Variablen

quoted

### 1.2 TMP:Modula-2/Arguments.def

```
DEFINITION MODULE Arguments;
(*$ NameChk:=FALSE LargeVars:=FALSE LongAlign:=FALSE *)
(* 29.11.88 / ms *)

(* 9.6.90 / bp Nie wieder DupLock/Unlock! *)

FROM DosD IMPORT FileLockPtr ;
(*
 * Arguments supports access to user supplied program
 * arguments. These are command line arguments or
 * workbench arguments depending on the program
 * activation.
 * If an argument comes with len=0 and its number is
 * valid ( 1 <= nr <= NumArgs() ) then it specifies the
 * current directory.
 *)

VAR
(*
 * quoted indicates that the last argument contained
 * quotes
 *)
quoted: BOOLEAN;
```

```
(*
 * NumArgs returns the number of arguments available.
 *)
PROCEDURE NumArgs(): INTEGER;

(*
 * GetArg returns the i-th argument from the argument
 * list. If the program is started from CLI the command
 * line is parsed. Quotes are interpreted slightly
 * different as from DOS commands or C-startup code. All
 * text between two quotes is taken literally. Quotes may
 * occur at any position in an argument (see example).
 * With workbench startup GetArg changes the directory
 * (CurrentDir) to the directory of the requested
 * argument and returns the argument name. GetArg(0...)
 * returns the name of the currently executed program.
 *
 * Examples for CLI Arguments:
 *
 * 1> command "Hello World"
 * 1> command Hello" "World
 * 1> command Hell"o World"
 *
 * All these calls give >Hello World< as argument.
 *)
PROCEDURE GetArg(arg: INTEGER;
                 VAR argument: ARRAY OF CHAR;
                 VAR len: INTEGER);

(*
 * sometimes it is useful to get the directory lock
 * associated with the arguments directory. Works only if
 * started from WB, otherwise returns NIL
 *)
PROCEDURE GetLock(arg: INTEGER): FileLockPtr ;

END Arguments.
```

---