

Previous: [Positional Parameters](#), Up: [Shell Parameters](#) [[Contents](#)][[Index](#)]

3.4.2 Special Parameters

The shell treats several parameters specially. These parameters may only be referenced; assignment to them is not allowed.

(`$*`) Expands to the positional parameters, starting from one. When the expansion is not within double quotes, each positional parameter expands to a separate word. In contexts where it is performed, those words are subject to further word splitting and filename expansion. When the expansion occurs within double quotes, it expands to a single word with the value of each parameter separated by the first character of the IFS special variable. That is, "`$*`" is equivalent to "`$1 c $2 c ...`", where c is the first character of the value of the IFS variable. If IFS is unset, the parameters are separated by spaces. If IFS is null, the parameters are joined without intervening separators.

@

(`$@`) Expands to the positional parameters, starting from one. In contexts where word splitting is performed, this expands each positional parameter to a separate word; if not within double quotes, these words are subject to word splitting. In contexts where word splitting is not performed, this expands to a single word with each positional parameter separated by a space. When the expansion occurs within double quotes, and word splitting is performed, each parameter expands to a separate word. That is, "`$@`" is equivalent to "`$1`" "`$2`" If the double-quoted expansion occurs within a word, the expansion of the first parameter is joined with the beginning part of the original word, and the expansion of the last parameter is joined with the last part of the original word. When there are no positional parameters, "`$@`" and `$@` expand to nothing (i.e., they are removed).

#

(`$#`) Expands to the number of positional parameters in decimal.

?

(`$?`) Expands to the exit status of the most recently executed foreground pipeline.

-

(`$-`, a hyphen.) Expands to the current option flags as specified upon invocation, by the `set` builtin command, or those set by the shell itself (such as the `-i` option).

\$

(`$$`) Expands to the process ID of the shell. In a `()` subshell, it expands to the process ID of the invoking shell, not the subshell.

!

(**\$!**) Expands to the process ID of the job most recently placed into the background, whether executed as an asynchronous command or using the **bg** builtin (see [Job Control Builtins](#)).

0

(**\$0**) Expands to the name of the shell or shell script. This is set at shell initialization. If Bash is invoked with a file of commands (see [Shell Scripts](#)), **\$0** is set to the name of that file. If Bash is started with the **-c** option (see [Invoking Bash](#)), then **\$0** is set to the first argument after the string to be executed, if one is present. Otherwise, it is set to the filename used to invoke Bash, as given by argument zero.

Previous: [Positional Parameters](#), Up: [Shell Parameters](#) [[Contents](#)][[Index](#)]