



Full credit is given to the above companies including the OS that this PDF file was generated!

Red Hat Enterprise Linux Release 9.2 Manual Pages on 'bg.1p' command

\$ man bg.1p

BG(1P) POSIX Programmer's Manual BG(1P)

PROLOG

This manual page is part of the POSIX Programmer's Manual. The Linux implementation of this interface may differ (consult the corresponding Linux manual page for details of Linux behavior), or the interface may not be implemented on Linux.

NAME

bg ? run jobs in the background

SYNOPSIS

bg [job_id...]

DESCRIPTION

If job control is enabled (see the description of set -m), the bg utility shall resume suspended jobs from the current environment (see Section 2.12, Shell Execution Environment) by running them as background jobs. If the job specified by job_id is already a running background job, the bg utility shall have no effect and shall exit successfully.

Using bg to place a job into the background shall cause its process ID to become "known in the current shell execution environment", as if it had been started as an asynchronous list; see Section 2.9.3.1, Examples.

OPTIONS

None.

OPERANDS

The following operand shall be supported:

`job_id` Specify the job to be resumed as a background job. If no `job_id` operand is given, the most recently suspended job shall be used. The format of `job_id` is described in the Base Definitions volume of POSIX.1?2017, Section 3.204, Job Control Job ID.

STDIN

Not used.

INPUT FILES

None.

ENVIRONMENT VARIABLES

The following environment variables shall affect the execution of `bg`:

`LANG` Provide a default value for the internationalization variables that are unset or null. (See the Base Definitions volume of POSIX.1?2017, Section 8.2, Internationalization Variables for the precedence of internationalization variables used to determine the values of locale categories.)

`LC_ALL` If set to a non-empty string value, override the values of all the other internationalization variables.

`LC_CTYPE` Determine the locale for the interpretation of sequences of bytes of text data as characters (for example, single-byte as opposed to multi-byte characters in arguments).

LC_MESSAGES

Determine the locale that should be used to affect the format and contents of diagnostic messages written to standard error.

`NLSPATH` Determine the location of message catalogs for the processing of `LC_MESSAGES`.

ASYNCHRONOUS EVENTS

Default.

STDOUT

The output of `bg` shall consist of a line in the format:

"[%d] %s\n", <job-number>, <command>

where the fields are as follows:

<job-number>

A number that can be used to identify the job to the wait, fg, and kill utilities. Using these utilities, the job can be identified by prefixing the job number with '%'.

<command> The associated command that was given to the shell.

STDERR

The standard error shall be used only for diagnostic messages.

OUTPUT FILES

None.

EXTENDED DESCRIPTION

None.

EXIT STATUS

The following exit values shall be returned:

- 0 Successful completion.
- >0 An error occurred.

CONSEQUENCES OF ERRORS

If job control is disabled, the bg utility shall exit with an error and no job shall be placed in the background.

The following sections are informative.

APPLICATION USAGE

A job is generally suspended by typing the SUSP character (<control>?Z on most systems); see the Base Definitions volume of POSIX.1?2017, Chapter 11, General Terminal Interface. At that point, bg can put the job into the background. This is most effective when the job is expecting no terminal input and its output has been redirected to non-terminal files. A background job can be forced to stop when it has terminal output by issuing the command:

```
stty tostop
```

A background job can be stopped with the command:

```
kill -s stop job ID
```

The bg utility does not work as expected when it is operating in its own utility execution environment because that environment has no sus?

pending jobs. In the following examples:

```
... | xargs bg
```

```
(bg)
```

each `bg` operates in a different environment and does not share its parent shell's understanding of jobs. For this reason, `bg` is generally implemented as a shell regular built-in.

EXAMPLES

None.

RATIONALE

The extensions to the shell specified in this volume of POSIX.1-2017 have mostly been based on features provided by the KornShell. The job control features provided by `bg`, `fg`, and `jobs` are also based on the KornShell. The standard developers examined the characteristics of the C shell versions of these utilities and found that differences exist. Despite widespread use of the C shell, the KornShell versions were selected for this volume of POSIX.1-2017 to maintain a degree of uniformity with the rest of the KornShell features selected (such as the very popular command line editing features).

The `bg` utility is expected to wrap its output if the output exceeds the number of display columns.

FUTURE DIRECTIONS

None.

SEE ALSO

Section 2.9.3.1, Examples, `fg`, `kill`, `jobs`, `wait`

The Base Definitions volume of POSIX.1-2017, Section 3.204, Job Control Job ID, Chapter 8, Environment Variables, Chapter 11, General Terminal Interface

COPYRIGHT

Portions of this text are reprinted and reproduced in electronic form from IEEE Std 1003.1-2017, Standard for Information Technology -- Portable Operating System Interface (POSIX), The Open Group Base Specifications Issue 7, 2018 Edition, Copyright (C) 2018 by the Institute of Electrical and Electronics Engineers, Inc and The Open Group. In the

event of any discrepancy between this version and the original IEEE and The Open Group Standard, the original IEEE and The Open Group Standard is the referee document. The original Standard can be obtained online at <http://www.opengroup.org/unix/online.html> .

Any typographical or formatting errors that appear in this page are most likely to have been introduced during the conversion of the source files to man page format. To report such errors, see https://www.kernel.org/doc/man-pages/reporting_bugs.html .

IEEE/The Open Group

2017

BG(1P)