



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 'pwd.1p' command

\$ man pwd.1p

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

pwd ? return working directory name

SYNOPSIS

pwd [-L|-P]

DESCRIPTION

The pwd utility shall write to standard output an absolute pathname of the current working directory, which does not contain the filenames dot or dot-dot.

OPTIONS

The pwd utility shall conform to the Base Definitions volume of POSIX.1?2017, Section 12.2, Utility Syntax Guidelines.

The following options shall be supported by the implementation:

- L If the PWD environment variable contains an absolute pathname of the current directory and the pathname does not contain any components that are dot or dot-dot, pwd shall write this pathname to standard output, except that if the PWD environment variable is longer than {PATH_MAX} bytes including the

terminating null, it is unspecified whether `pwd` writes this pathname to standard output or behaves as if the `-P` option had been specified. Otherwise, the `-L` option shall behave as the `-P` option.

`-P` The pathname written to standard output shall not contain any components that refer to files of type symbolic link. If there are multiple pathnames that the `pwd` utility could write to standard output, one beginning with a single `<slash>` character and one or more beginning with two `<slash>` characters, then it shall write the pathname beginning with a single `<slash>` character. The pathname shall not contain any unnecessary `<slash>` characters after the leading one or two `<slash>` characters.

If both `-L` and `-P` are specified, the last one shall apply. If neither `-L` nor `-P` is specified, the `pwd` utility shall behave as if `-L` had been specified.

OPERANDS

None.

STDIN

Not used.

INPUT FILES

None.

ENVIRONMENT VARIABLES

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

`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: 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_MESSAGES`

Determine the locale that should be used to affect the format

and contents of diagnostic messages written to standard error?

ror.

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

PWD An absolute pathname of the current working directory. If an application sets or unsets the value of PWD, the behavior of pwd is unspecified.

ASYNCHRONOUS EVENTS

Default.

STDOUT

The pwd utility output is an absolute pathname of the current working directory:

```
"%s\n", <directory pathname>
```

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 an error is detected, output shall not be written to standard output, a diagnostic message shall be written to standard error, and the exit status is not zero.

The following sections are informative.

APPLICATION USAGE

If the pathname obtained from pwd is longer than {PATH_MAX} bytes, it could produce an error if passed to cd. Therefore, in order to return to that directory it may be necessary to break the pathname into sections shorter than {PATH_MAX} and call cd on each section in turn (the

first section being an absolute pathname and subsequent sections being relative pathnames).

EXAMPLES

None.

RATIONALE

Some implementations have historically provided `pwd` as a shell special built-in command.

In most utilities, if an error occurs, partial output may be written to standard output. This does not happen in historical implementations of `pwd`. Because `pwd` is frequently used in historical shell scripts without checking the exit status, it is important that the historical behavior is required here; therefore, the CONSEQUENCES OF ERRORS section specifically disallows any partial output being written to standard output.

An earlier version of this standard stated that the `PWD` environment variable was affected when the `-P` option was in effect. This was incorrect; conforming implementations do not do this.

FUTURE DIRECTIONS

None.

SEE ALSO

`cd`

The Base Definitions volume of POSIX.1-2017, Chapter 8, Environment Variables, Section 12.2, Utility Syntax Guidelines

The System Interfaces volume of POSIX.1-2017, `getcwd()`

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

PWD(1P)