



*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 'head.1p' command***

***\$ man head.1p***

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

head ? copy the first part of files

### SYNOPSIS

head [-n number] [file...]

### DESCRIPTION

The head utility shall copy its input files to the standard output, ending the output for each file at a designated point.

Copying shall end at the point in each input file indicated by the -n number option. The option-argument number shall be counted in units of lines.

### OPTIONS

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

The following option shall be supported:

-n number The first number lines of each input file shall be copied to standard output. The application shall ensure that the number option-argument is a positive decimal integer.

When a file contains less than number lines, it shall be copied to standard output in its entirety. This shall not be an error.

If no options are specified, head shall act as if -n 10 had been specified.

## OPERANDS

The following operand shall be supported:

**file** A pathname of an input file. If no file operands are specified, the standard input shall be used.

## STDIN

The standard input shall be used if no file operands are specified, and shall be used if a file operand is '-' and the implementation treats the '-' as meaning standard input. Otherwise, the standard input shall not be used. See the INPUT FILES section.

## INPUT FILES

Input files shall be text files, but the line length is not restricted to {LINE\_MAX} bytes.

## ENVIRONMENT VARIABLES

The following environment variables shall affect the execution of head:

**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 and input files).

### 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 standard output shall contain designated portions of the input files.

If multiple file operands are specified, head shall precede the output for each with the header:

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

except that the first header written shall not include the initial <newline>.

## 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

Default.

The following sections are informative.

## APPLICATION USAGE

When using head to process pathnames, it is recommended that LC\_ALL, or at least LC\_CTYPE and LC\_COLLATE, are set to POSIX or C in the environment, since pathnames can contain byte sequences that do not form valid characters in some locales, in which case the utility's behavior would be undefined. In the POSIX locale each byte is a valid single-byte character, and therefore this problem is avoided.

## EXAMPLES

To write the first ten lines of all files (except those with a leading period) in the directory:

```
head -- *
```

## RATIONALE

Although it is possible to simulate head with sed 10q for a single file, the standard developers decided that the popularity of head on historical BSD systems warranted its inclusion alongside tail.

POSIX.1?2008 version of head follows the Utility Syntax Guidelines. The -n option was added to this new interface so that head and tail would be more logically related. Earlier versions of this standard allowed a -number option. This form is no longer specified by POSIX.1?2008 but may be present in some implementations.

There is no -c option (as there is in tail) because it is not historical practice and because other utilities in this volume of POSIX.1?2017 provide similar functionality.

## FUTURE DIRECTIONS

None.

## SEE ALSO

sed, tail

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

## 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](https://www.kernel.org/doc/man-pages/reporting_bugs.html) .

IEEE/The Open Group

2017

HEAD(1P)