

env Cheatsheet

By Dejan Panovski • Updated on Mar 15, 2026 • [Download PDF](#)

Quick reference for listing, clearing, and overriding environment variables with `env` in Linux

The `env` command prints the current environment or runs another command with a modified environment. This cheatsheet covers listing variables, starting with a clean environment, overriding values temporarily, and common troubleshooting patterns.

Basic Syntax

Core `env` command forms.

<code>env</code>	Print the current environment
<code>env --help</code>	Show available options
<code>env --version</code>	Show the installed <code>env</code> version
<code>env -0</code>	Print variables separated with NUL bytes

Inspect Environment Variables

Use `env` with filters to inspect specific variables.

<code>`env`</code>	<code>sort`</code>
<code>`env`</code>	<code>grep ^PATH=`</code>
<code>`env`</code>	<code>grep ^HOME=`</code>
<code>`env`</code>	<code>grep ^LANG=`</code>

Run Commands with Temporary Variables

Set variables for one command without changing the current shell session.

<code>VAR=value env command</code>	Run one command with a temporary variable
<code>VARI=dev VAR2=1 env command</code>	Set multiple temporary variables
<code>env PATH=/custom/bin:\$PATH command</code>	Override <code>PATH</code> for one command
<code>env LANG=C command</code>	Run a command with the <code>C</code> locale
<code>env HOME=/tmp bash</code>	Start a shell with a temporary home directory

Clean or Remove Variables

Start with a minimal environment or remove selected variables.

<code>env -i command</code>	Run a command with an empty environment
<code>env -i PATH=/usr/bin:/bin bash --noprofile --norc</code>	Start a mostly clean shell with a minimal PATH
<code>env -u VAR command</code>	Run a command without one variable
<code>env -u http_proxy command</code>	Remove a proxy variable for one command
<code>env -i VAR=value command</code>	Run a command with only the variables you set explicitly

Common Variables

These variables are often inspected or overridden with `env`.

PATH	Directories searched for commands
HOME	Current user's home directory
USER	Current user name
SHELL	Default login shell
LANG	Locale and language setting
TZ	System timezone (e.g. <code>America/New_York</code>)
EDITOR	Default text editor
TERM	Terminal type (e.g. <code>xterm-256color</code>)
TMPDIR	Directory for temporary files
PWD	Current working directory

Troubleshooting

Quick checks for common `env` issues.

A temporary variable does not persist	<code>env VAR=value command</code> affects only that command and its children
A command is not found after <code>env -i</code>	Add a minimal PATH , such as <code>/usr/bin:/bin</code>
Output is hard to parse safely	Use <code>env -0</code> with tools that support NUL-delimited input
A variable is still visible in the shell	<code>env</code> does not modify the parent shell; use <code>export</code> or unset the variable in the shell itself
A locale-sensitive command behaves differently	Check whether LANG or related locale variables were overridden

Related Guides

Use these guides for broader environment-variable workflows.

[How to Set and List Environment Variables in Linux](#)

Full guide to listing and setting environment variables

[export Command in Linux](#)

Export shell variables to child processes

[Bashrc vs Bash Profile](#)

Understand shell startup files

[Bash cheatsheet](#)

Quick reference for Bash syntax and variables