

How to set and list environment variables in Linux

If you want to do something that requires using the command line, you will still need to know about environment variables. This may seem like a complicated term, but actually environment variables are easy to understand.

In the past, to use Linux, you needed to get acquainted with the command line immediately. Desktop environments like GNOME and KDE have made using Linux much easier for beginners. You can use Linux for a long time without knowing any terminal command.

However, if you want to do something that requires using the command line, you will still need to know about environment variables. This may seem like a complicated term, but actually environment variables are easy to understand.

What is an environment variable?

Environment variables allow you to set options on different programs without having to edit the application's configuration file. This applies across the system, so they are available not only in programs but also in shells and even subprocesses.

They are used to help different commands know where the **Home** folder is, what username and shell you are currently using. You can also set different options like terminal color using environment variables.

List environment variables

In fact, you can spend more time setting different environment variables than listing them. If you troubleshoot a problem, it's helpful to know how certain variables are set or whether they are set.

```
kwouk@vcrow:~$ printenv
SHELL=/bin/bash
SESSION_MANAGER=local/vcrow:@/tmp/.ICE-unix/1522,unix/vcrow:/tmp/.ICE-unix/1522
QT_ACCESSIBILITY=1
COLORTERM=truecolor
XDG_CONFIG_DIRS=/etc/xdg/xdg-ubuntu:/etc/xdg
XDG_MENU_PREFIX=gnome-
GNOME_DESKTOP_SESSION_ID=this-is-deprecated
GTK_IM_MODULE=ibus
QT4_IM_MODULE=xim
GNOME_SHELL_SESSION_MODE=ubuntu
SSH_AUTH_SOCK=/run/user/1000/keyring/ssh
XMODIFIERS=@im=ibus
DESKTOP_SESSION=ubuntu
SSH_AGENT_PID=1668
GTK_MODULES=gall:atk-bridge
XDG_SEAT=seat0
PWD=/home/kwouk
LOGNAME=kwouk
XDG_SESSION_DESKTOP=ubuntu
XDG_SESSION_TYPE=x11
GPG_AGENT_INFO=/run/user/1000/gnupg/S.gpg-agent:0:1
XAUTHORITY=/run/user/1000/gdm/Xauthority
GJS_DEBUG_TOPICS=JS ERROR;JS LOG
WINDOWPATH=2
HOME=/home/kwouk
USERNAME=kwouk
IM_CONFIG_PHASE=2
LANG=en_US.UTF-8
```



This aspect has two parts. You can list all current environment variables or choose to consider only a single variable value. To list all environment variables currently set, run the **printenv** command with no arguments:

```
printenv
```

This command will give you a high level overview of various environment variables. You can use this command to see what has been set or use it with another tool like **grep** to search for certain text strings.

If you only want to see the value of a variable, run **printenv** with the variable name you choose as an argument:

```
printenv HOME
```

If instead you want to test a few different variables, pass them all as arguments as the following example:

```
printenv HOME PWD
```

To pass the value of an environment variable, reference it with the **\$** character like the following example:

```
ls $HOME
```

Set environment variables


Setting an environment variable is also quite easy. Use the name without the **\$** operator and assign it with the **=** operator. For example, to set '**EXAMPLE_VAR**' to '**hello**', you would run the following command:

```
EXAMPLE_VAR=hello
```

You can now access this variable as above using the **\$** operator.

```
echo $EXAMPLE_VAR
```

This will only set the variable for the current session. When you log out or restart the computer, this variable will disappear. This is good for testing or if you just need to temporarily set a certain value.

```
kwouk@vcrow:~$ EXAMPLE_VAR=hello
kwouk@vcrow:~$ echo $EXAMPLE_VAR
hello
kwouk@vcrow:~$ 
```

To ensure these environment variables exist, you need to put them in the appropriate configuration file. You can use the `/etc/environment` file for system-wide variables, while `/etc/profile` sets the shell variables.

For personal use, you can put variables in `~/.bashrc` or a similar file, if you use another shell. To set them up, use the same format as above.

With the command line tools above, you will have the basics for almost everything to do with environment variables. You can start by customizing shell parts or other boot options, then expand from there.

Of course, if you're new to Linux, you need to know more than just environment variables to make the most of the command line. Check out the article: [Basic Linux commands everyone needs to know](#) in order to learn more details.

Good luck!

You finished reading the article "[How to set and list environment variables in Linux](#)" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.