Bash scripting

Assel Mukhamedyarova

January 2023

Table of contents

- What is a bash shell?
- What is a bash script and how do you identify it?
- How to create your first bash script and execute it.
- The basic syntax of shell scripting.

Table of contents

- What is a bash shell?
- What is a bash script and how do you identify it?
- How to create your first bash script and execute it.
- The basic syntax of shell scripting.

Table of contents

- What is a bash shell?
- What is a bash script and how do you identify it?
- How to create your first bash script and execute it.
- The basic syntax of shell scripting.

Table of contents

- What is a bash shell?
- What is a bash script and how do you identify it?
- How to create your first bash script and execute it.
- The basic syntax of shell scripting.

What is a bash shell?

A bash script is a series of commands written in a file. These are read and executed by the bash program. The program executes line by line.
For example, you can navigate to a certain path, create a folder and spawn a process inside it using the command line.
You can do the same sequence of steps by saving the commands in a bash script and running it. You can run the script any number of times.

What is a bash script and how do you identify it?

File extension of .sh. By naming conventions, bash scripts end with a .sh. However, bash scripts can run perfectly fine without the sh extension.

Scripts start with a bash bang. Scripts are also identified with a shebang. Shebang is a combination of bash and bang ! followed the the bash shell path. This is the first line of the script. Shebang tells the shell to execute it via bash shell. Shebang is simply an absolute path to the bash interpreter.

Below is an example of the shebang statement.

! /bin/bash

The path of the bash program can vary. We will see later how to identify it.

Execution rights: Scripts have execution rights for the user executing them.

An execution right is represented by x. In the example below, my user has the rwx (read, write, execute) rights for the file test$_s$*cript.sh*

File colour: Executable scripts appear in a different colour from rest of the files and folders.

How to create your first bash script and execute it.

Let’s create a simple script in bash that outputs Hello World.

Create a file named: hello$_{w}orld.shtouchhello_{w}orld.sh$

Find the path to your bash shell. which bash

Write the command. We will echo ”hello world” to the console.

Our script will look something like this:

! usr/bin/bash echo ”Hello World”

Edit the file: hello$_{w}orld.shusingatexteditorofyourchoiceandaddtheabovelinesinit$.

Provide execution rights to your user. Modify the file permissions and allow execution of the script by using the command below:

chmod u+x hello$_{w}orld.shchmodmodifiestheexistingrightsofafileforaparticularuser.Weareadding + xtouseru$.

Run the script. You can run the script in the following ways:

./hello$_{w}orld.sh$

bash hello$_{w}orld.sh$.

The basic syntax of shell scripting.

Just like any other programming language, bash scripting follows a set of rules to create programs understandable by the computer. In this section, we will study the syntax of bash scripting.
How to define variables: We can define a variable by using the syntax variable$_{n}ame = value$.
To get the value of the variable, add s before the variable.
Tux is also the name of the Linux mascot, the penguin.