Scripting & beyond

Bash Scripting Tutorials Notes
A quick reference to concepts learned during tutorials and browsing.
Cleyde Varela

Version: v1 May 2025

Table of contents

1 References come first here

- A big shout out to the Youtube channel **MPrashant Academy** for his amazing Linux Shell Scripting video tutorials for beginners which helped me imensely getting started. I add a lot of his slides in this document because they ilustrate exactly what one needs.

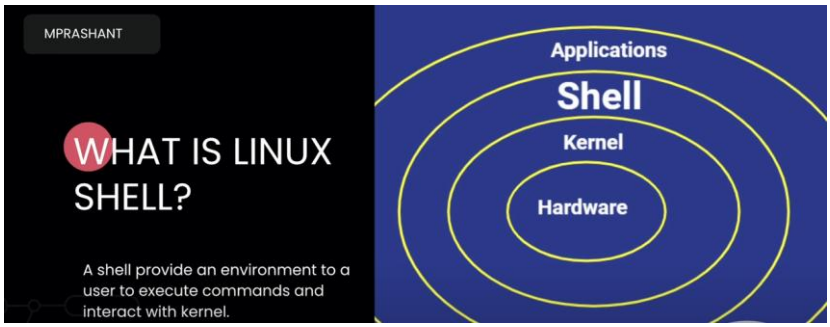
  https://www.youtube.com/@MPrashantAcademy
  https://www.instagram.com/mprashant.codes/

- A very good resource to learn initial concepts in text form: https://linuxjourney.com/
- If you need a simple online terminal to practice simple commands I strongly recommend: https://bellard.org/jslinux/

2 Linux? Shell? Scripting?

Linux, Shell, Scripting, Bash…. All these terms can be confusing to differentiate so let's break them down:

- **Linux** is an open-source operating system (OS) based on Unix (think of Unix as the ancestor of many OS and the reason CLI exist), it handles hardware resources, runs apps and processes, provides a CLI, handles filesystems, permissions and acts as the layer in between users and programs and the hardware.
- **Shell** is a CLI, a Command Line Interface, the scary dark window, na enviroment to run your commands, scripts and programs. There are many different shell types, **sh, bash, zsh, fish**, etc.
- **Bash** is a type of Shell, the most popular actually, it extends the features of **sh**.
- **Shell scripting** means writing a sequence of shell commands in a file (called a script) so they can be executed automatically.



2.1 How to check your shell type

echo $0

2.2 Check the available shell types in your machine

cat /etc/shells

```
[root@localhost ~]# cat /etc/shells
/bin/sh
/bin/bash
/usr/bin/sh
/usr/bin/bash
/usr/bin/tmux
/bin/tmux
/bin/csh
/bin/tcsh
/usr/bin/csh
/usr/bin/tcsh
/usr/bin/zsh
/bin/zsh
```

2.3 Change shell type

Simply run the name of the shell you want to switch to the shell type you want, if you are in **sh** for exmple just type **bash** in the terminal and run.

```
[root@localhost ~]# echo $0
sh
[root@localhost ~]# bash
[root@localhost ~]# echo $0
bash
```

3 What is vi?

You write your scritps in text editors, a very common one is **vi** (think of 6 in romans), you can also find **vim, nano, gedit**, etc. To open vi simply run: **vi** {name of the file}, for example **vi test.sh**

Inside **vi** keep in mind:

- **i** to start editing
- **Esc** to stop editing
- **:q** to quit **vi** without saving latest changes
- **:wq** to save latest changes and quit **vi**

Inside your script:

- Use a **shebang line**, this is the first line to add in a script **#!bin/bash**, is the way **vi** has to tell the OS to use the bash or another shell type to execute the script. It is not a mandatoy line but it's strongly recommended so use it!

```
#!/bin/bash

echo "Hey Buddy!"
```

- Use comments as with any other programming languages.


```
#!/bin/bash

#This is single line comment
echo "Hello"

<<comm
This is
a muli
line comment
comm
```

- It is not mandatory to save your scripts with as a *.sh* but it helps since it clearly indicates that it is a shell script and in editors like Vscode it is helpful due to formating.

4 It's all about having consent! (Permissions)

You need to have the right permissions to work with scripts, either to read, write or execute.

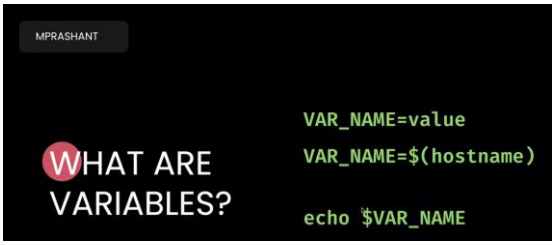
Check this great explanations by GeeksforGeeks:

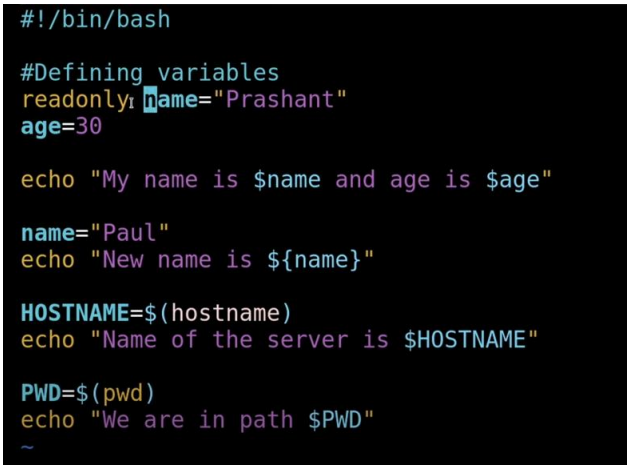
https://www.geeksforgeeks.org/bash-script-file-permissions/

https://www.geeksforgeeks.org/set-file-permissions-linux/

After making sure the permissions are good to go you can run the scripts.

5 Variables

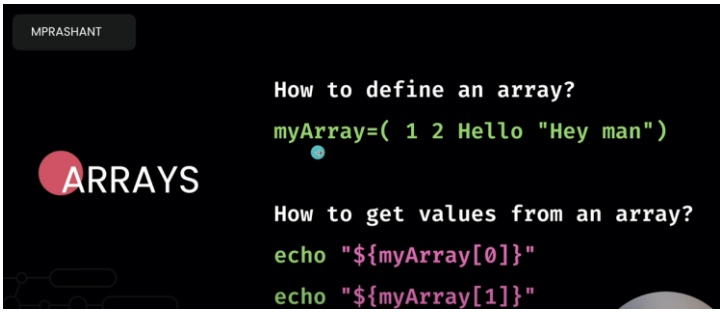




- Use **readonly** to make variables constant
- Make sure there is no space in between the = sign
- Use ${variableName} instead of $variableName for more readability

6 Arrays

- You can use different data types in a single array
- Each item in the array is separated by a space
- If we have na item likea string that has a space inside it then we surround the item with “”
- Indexes start at 0



7 Strings

Todo

8 Conditionals

Todo

9 Loops

Todo

10 Functions

Todo

11 Misc concepts

Todo