

## Guiones BASH

En esta entrada colocamos en práctica los comandos aprendidos en la entrada acerca de la [terminal de comandos](#)... ¡y más!

Pero primero, si carecen de una máquina con GNU/Linux para practicar, acá tienen un crédito de cortesía de 100 dólares estadounidenses **si usan nuestro código de referido**:

El proceso para crear una máquina virtual o *droplet* en la empresa neoyorquina DigitalOcean está [explicado en esta entrada](#), así como la creación de un usuario con derechos de administrador para ella.

**Por supuesto**, nada impide que utilicen [VirtualBox](#) sobre MS Windows® (siempre y cuando tengan una buena máquina) o simplemente usen su máquina con GNU/Linux, pero cuidado de **no desconfigurar vuestro perfil con los ejemplos acá publicados**. **Comencemos, pues.**

## Introducción a BASH

Lo primero que haremos es crear una carpeta para nuestros ejemplos y "entrar" en ella:

```
$ cd
$ mkdir guiones
$ cd guiones
```

La primera línea nos lleva a nuestra carpeta de usuario o *home*, una vez allí creamos la carpeta con el comando **mkdir** y en la tercera línea entramos a ese directorio. De esta manera el sistema operativo, cuando le demos la instrucción de ejecutar, buscar, listar, etc. un archivo **siempre** primero lo buscará en esa carpeta (y luego en otros sitios pero por ahora esto carece de importancia para la práctica).

```
$ touch guion.sh
$ ls -l
```

Lo anterior crea un archivo nuevo llamado **guion.sh**, ya que la extensión **.sh** es la generalmente utilizada para describir este tipo de fichero. Con **ls -l** comprobamos que efectivamente haya sido

## KS7000+WP

KS7000 migra a GNU/Linux y escoge a WordPress para registrar el camino.

<https://www.ks7000.net.ve>

---

creado el guion. Ahora procedemos a editar con **nano**, el cual viene con la mayoría de las distribuciones GNU/Linux o reposa en sus repositorios para ser instalada.

```
$ nano guion.sh
```

Dentro colocaremos una línea para identificar qué aplicación se encargará de ejecutar las instrucciones, en este caso el ejecutable BASH del sistema:

Guardamos el archivo, si luego utilizamos el comando **file** veremos el tipo de archivo y una breve descripción (el texto de color verde es la respuesta del comando):

```
# file guion.sh
guion.sh: Bourne-Again shell script, ASCII text executable
```

También podemos consultar el BASH en sí mismo, el siguiente ejemplo fue ejecutado en Ubuntu 20.04:

```
$ file /bin/bash
/bin/bash: ELF 64-bit LSB shared object, x86-64, version 1 (SYSV),
dynamically linked, interpreter /lib64/ld-linux-x86-64.so.2,
BuildID[sha1]=a6cb40078351e05121d46daa768e271846d5cc54,
for GNU/Linux 3.2.0, stripped
```

Una explicación en vídeo (en idioma inglés con subtítulos) de esta sección la [pueden ver en este enlace](#) al [canal en YouTube del señor Bobby Iliev](#), directo desde Bulgaria.

## Hola mundo

¡Ahora ya estamos listos para escribir nuestro "[Hola mundo](#)"! Verificamos si aún estamos ubicados en la carpeta **guiones** por medio del comando **pwd**. Mi nombre de usuario es **jimmy** así que en este ejemplo se ve reflejado:

## KS7000+WP

KS7000 migra a GNU/Linux y escoge a WordPress para registrar el camino.

<https://www.ks7000.net.ve>

---

```
$ pwd
/home/jimmy/guiones
```

Si listamos el contenido:

```
$ ls -l
total 4
-rw-r--r-- 1 jimmy docker 13 Feb 7 16:08 guion.sh
```

Esta máquina la uso para trabajar con el software [Docker](#), por ello es que aparezco como integrante de dicho grupo y el fichero "hereda" esas características. Una manera rápida de asegurar que solamente nosotros podemos acceder a nuestra práctica (y ninguno del grupo **docker**) es aplicando:

```
$ sudo chown jimmy:jimmy guion.sh
$ ls -l
total 4
-rw-r--r-- 1 600 jimmy 13 Feb 7 16:08 guion.sh
```

Ahora editamos de nuevo con **nano** y agregamos el comando **echo** el cual sirve para mostrar un mensaje que colocamos entre comillas:

Guardamos y el siguiente paso es darle la característica de poder ser ejecutado (noten la letra equis resaltado en rojo de la respuesta del comando resaltada en verde):

```
$ chmod +x guion.sh
$ ls -l
total 4
-rwxr-xr-x 1 jimmy jimmy 32 Feb 7 17:08 guion.sh
```

De esta manera podemos ejecutarlo sin necesidad de llamar al intérprete, ya que al comienzo del fichero especifica con cuál intérprete (BASH en este caso) se encargará de ello:

```
$ ./guion.sh  
Hola mundo.
```

Una explicación en vídeo (en idioma inglés con subtítulos) de esta sección la [pueden ver en este enlace](#) al [canal en YouTube del señor Bobby Iliev](#), directo desde Bulgaria.

## Variables

Repetimos el procedimiento de la sección anterior, solo que modificaremos con el siguiente código:

Noten que:

- Para declarar la variable colocamos su nombre, el signo de igualdad y el mensaje entrecomillado, **sin dejar ningún espacio**.
- Para imprimir la variable escribimos el signo de peso, par de corchetes y dentro el nombre de la variable, **teniendo en cuenta las mayúsculas y minúsculas, debe ser exactamente el mismo nombre de variable**.

Aunque ya habíamos configurado como ejecutable al archivo, también podemos llamarlo con:

```
$ bash guion.sh  
Hola mundo Jimmy.
```

También noten que el alcance de la variable es solo mientras dura la ejecución de ese guion: si ordenamos imprimir esa variable una vez finalizado el guion, no devolverá valor alguno.

## KS7000+WP

KS7000 migra a GNU/Linux y escoge a WordPress para registrar el camino.

<https://www.ks7000.net.ve>

---

Una explicación en vídeo (en idioma inglés con subtítulos) de esta sección la [pueden ver en este enlace](#) al [canal en YouTube del señor Bobby Iliev](#), directo desde Bulgaria.

## Interacción con el usuario

En la sección anterior colocamos el nombre de manera dura ("*hard code*") o, más bien, de manera fija. Con la instrucción **read** podemos hacer que el usuario introduzca una respuesta y esta sea guardada en una variable. También podemos presentar el mensaje de la pregunta en sí misma con el parámetro **-p**:

Una explicación en vídeo (en idioma inglés con subtítulos) de esta sección la [pueden ver en este enlace](#) al [canal en YouTube del señor Bobby Iliev](#), directo desde Bulgaria.