

Creación Bash

Bash un tipo de archivos programados en Linux, los cuales vamos a crear y a ejecutar. Esto nos permitirá obtener información automática a través de archivos ejecutables.

Paso 1.- Creamos un archivo de texto con extensión .sh

```
nano ejemploBash.sh
```

Paso 2.- Modificamos los permisos del archivo para que pueda ser ejecutado por el usuario.

- **Opción 2:** `chmod 777 ejemploBash.sh` (Añadimos todos los permisos al archivo)
- **Opción 1:** `chmod u+x ejemploBash.sh` (Añadimos permisos de ejecución del usuario al archivo)

Paso 3.- Editamos el archivo creado añadiendo la siguiente línea: **`#!/bin/bash`**. Esta línea será un comentario que ubicaremos en la primera línea de todos los programas que creemos, y nos servirá para indicar que se trata de un archivo bash ejecutable.

Paso 4.- Insertamos los comandos que queremos que ejecute este primer programa de prueba, por ejemplo un comentario.

```
#!/bin/bash  
echo "Este es el primer programa bash que he creado"
```

Paso 5.- Ejecutamos el archivo creado añadiendo `./` y el nombre del archivo que hemos creado.

```
./ejemploBash.sh
```

**Si has realizado correctamente todos los pasos, al ejecutar el comando del paso 5, el terminal de Linux nos mostrará la frase "Este es el primer programa bash que he creado".*

COMANDOS BÁSICOS BASH

- **#!/bin/bash**: comentario que indica que este es un programa ejecutable Bash. Esta línea estará incluida en todos los archivos ejecutables que creemos.

- **#**: este símbolo nos permitirá insertar comentarios en nuestro programa. Los comentarios no se mostrarán en el resultado.

ejemplo)

```
#Esto es un comentario
```

- **echo**: comando para mostrar por pantalla una variable o una frase

ejemplo)

```
echo "Bienvenido a Bash, lenguaje de programación en el terminal de Linux"
```

- **\$**: este símbolo se utilizará para hacer una llamada al contenido de una variable que hayamos creado anteriormente. Ya sea para mostrarlo, para realizar una operación con él, u otros.

ejemplo)

```
num1=25  
echo "el número 1 es: " $num1
```

- **Read**: comando para recoger y almacenar un dato dado por el usuario durante la ejecución del programa.

ejemplo)

```
echo "escribe un número"  
read num1
```

- **Comandos por defecto**: Bash nos proporciona otros comandos por defecto que también podemos utilizar.

- **\$1, \$2, \$3, ..** : devuelve las variables que ha insertado el usuario en la ejecución de la función. Por ejemplo, si queremos que el usuario inserte una variable al ejecutar la función. (\$2 se refiere a la segunda variable..)

ejemplo) ./ejemploBash.sh 10 40 11 9

```
$1: 10  
$2: 40  
$3: 11  
$4: 9
```

- **\$HOME**: devuelve la ruta del usuario. Por ejemplo, /Escritorio/Jon

- **\$USER**: devuelve el nombre del usuario. Por ejemplo, Jon

- **\$RANDOM**: devuelve un número aleatorio

**El resto de comandos los iremos viendo según surja la necesidad de utilizarlos.*