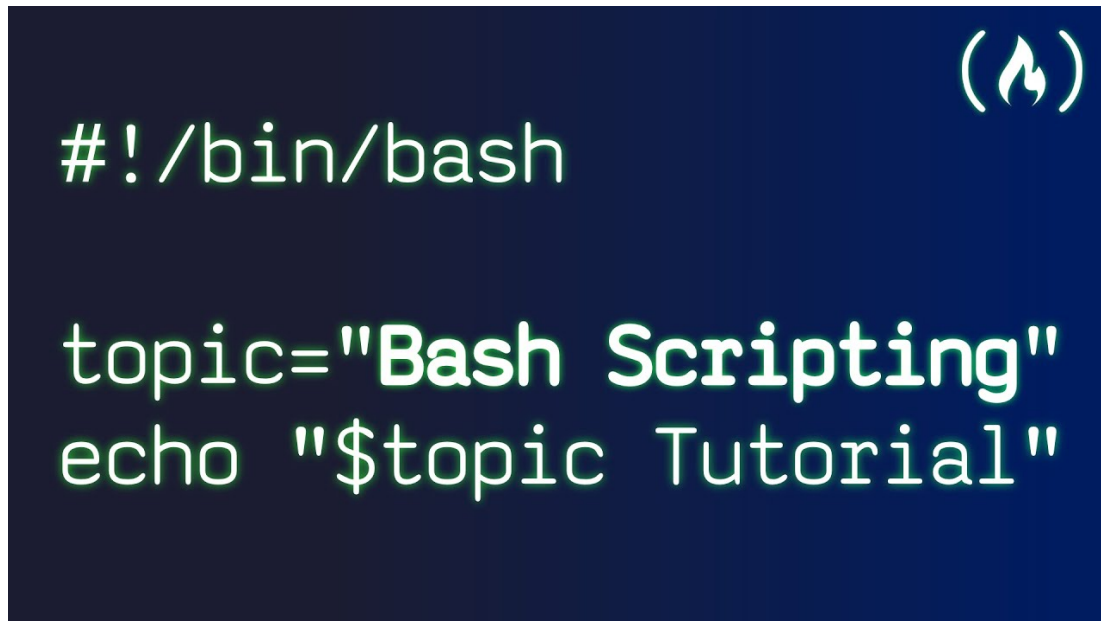


Bash Scripting Tutorial for Beginners:

[https://www.youtube.com/watch?](https://www.youtube.com/watch?v=tK9Oc6AEnR4&list=LL&index=15)

[v=tK9Oc6AEnR4&list=LL&index=15](https://www.youtube.com/watch?v=tK9Oc6AEnR4&list=LL&index=15)



—> bash files end with .sh

—> #!/bin/bash (defines the current interpreter)
=> always at the beginning of a new bash-file

—> echo hello (puts out text in the terminal)

—> set variables

- MY_LOCATION_FROM = /my/location/from
- MY_LOCATION_TO = /my/location/to

cp \$MY_LOCATION_FROM \$MY_LOCATION_TO

- always use Uppercase with a \$
- NAME = Herbert

- echo \$NAME
- > read (uses the input in the terminal for the script)
 - echo what is your first name?
 - read FIRST_NAME
 - echo Hello \$FIRST_NAME
- > | (ability to connect commands)
 - ls -l /usr/bin | grep bash
 - (shows all files that have bash in their name)
- > > (overwrites a file)
 - echo Hello World! > hello.txt
- > >> (add text to the file)
 - echo Good day to you! >> hello.txt
- > wc -w < hello.txt (number of words in the file)
 - output: 6
- > cat << EOF (prints everything until the word EOF)
- > [hello = hello] (equal means return value 0)
 - echo \$?
 - output: 0
- [1 = 0]
- echo \$?
- output: 1

—> if else statements in name.sh

```
- if [ ${1,,} = Herbert ]; then
    echo „Welcome boss.“
elif [ ${1,,} = help ]; then
    echo „Enter your username.“
else
    echo „You are not the boss“
fi
```

Example: bash name.sh Herbert

-> Herbert = Input

—> case statements

```
- case ${1,,} in
    herbert | administrator)
        echo „Hello Boss.“
        ;;
    help)
        echo „Enter a username.“
        ;;
    *)
        echo „Hello who are you?“

esac
```

=> * means catch all other cases

—> lists

- MY_FIRST_LIST = (one two three four five)
echo MY_FIRST_LIST

Output: one

- echo \${MY_FIRST_LIST[@]}

Output: prints the whole list

- echo \${MY_FIRST_LIST[1]}

Output: two

—> for loops

- for item in \${MY_FIRST_LIST[@]}; do echo -n \$item |
wc -c; done

Output: 3 3 4 4 5

- prints the length of the words in the list (wc -c)

—> functions

```
- showuptime(){  
    up=$(uptime -p | cut -c4-)  
    since=$(uptime -s)  
    cat << EOF
```

```
    ---
```

```
This machine has been up for ${up}  
It has been running since ${since}
```

```
    ---
```

```
EOF
```

```
}
```

```
showuptime
```

—> local variables just for the function

```
local up=$(uptime -p | -c4-)  
local since=$(uptime -s)
```

```
#!/bin/bash  
showname(){  
    echo hello $1  
    if [ ${1,,} = herbert ]; then  
        return 0  
    else  
        return 1  
    fi  
}  
showname $1  
if [ $? = 1 ]; then  
    echo "Someone unknown called the function!"  
fi
```