**Declare**

[Typing variables: declare or typeset (tldp.org)](https://tldp.org/LDP/abs/html/declareref.html#FTN.AEN5685)

[Bash declare command – Linux Hint](https://linuxhint.com/bash_declare_command/)

* Using the declare built-in restricts the scope of a variable
* In this context, typing a variable means to classify it and restrict its properties. For example, a variable declared or typed as an integer is no longer available for string operations.
* **declare -i intvar**

**intvar=23**

**echo "$intvar" # 23**

**intvar=stringval**

**echo "$intvar" # 0**

**Bash trap command**

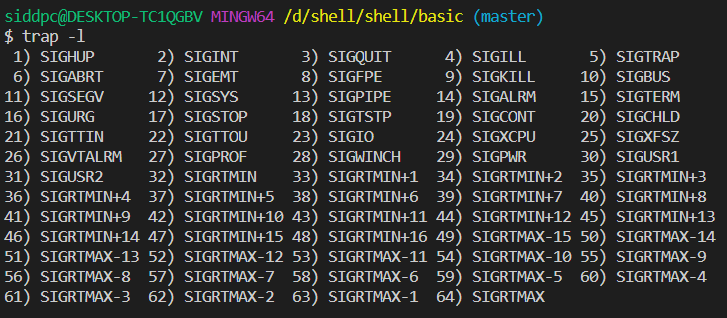
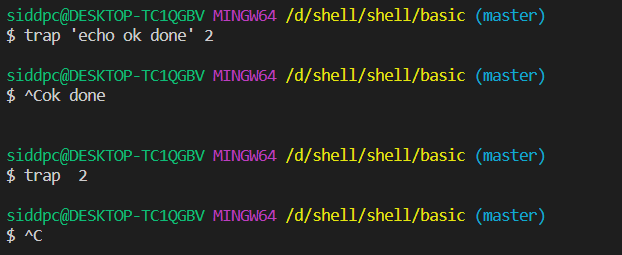
[Bash trap command – Linux Hint](https://linuxhint.com/bash_trap_command/)

* A built-in bash command that is used to execute a command when the shell receives any signal is called `trap`. When any event occurs then bash sends the notification by any signal.
* Many signals are available in bash. The most common signal of bash is SIGINT (Signal Interrupt). When the user presses CTRL+C to interrupt any process from the terminal then this signal is sent to notify the system.
* SIGTERM signal is used to terminate the process immediately by releasing its resources.
* **trap [action] [signal]**
* **$ trap 'rm temp.txt' err exit**

**$ ls**

**$ exit**

|  |  |
| --- | --- |
| **Key** | **Description** |
| -l | It is used to display the list of all signal names with corresponding number. |
| -p | It is used to display signal command or trap command for signal\_spec. |
| arg | It is used to execute a command when the shell receives the signal(s). |
| signal\_spec | It contains signal name or signal number. |

* The signal number of **SIGUP, SIGQUIT and SIGKILL are 1, 3 and 9**. The following first command will set a trap for these three signals. When any of these signals will occur then the message “Trap command is executed” will print. Run the following command from the terminal.
* **$ trap 'echo Trap command executed' 1 3 9**
* When the user will press Ctrl+C to generate the signal assign by `trap` command then the `echo` command of trap command will execute and the following output will appear.

# Call func function on exit

trap func exit

# Declare the function

function func() {

  echo "Task completed"

}

# Read the files and folders of the current directory list using for loop

for i  in \*

do

  echo "$i"

done