



# Arrays for Bash Shell Scripting

NarendraP

Learn How to Automate Common Tasks with **Bash** Shell Scripting

# Concepts of Arrays

- What is an Array ?
- How to define array ?
- How to access Array Values ?
- Different Types of Arrays
- How to store the command output into an array ?
- How to delete and update exiting array with new values ?
- How to read array using read command ?

Learn How to Automate Common Tasks with **Bash** Shell Scripting

# What is an Array and How to define or declare it ?

## ➤ What is an array ?

- An Array is the data structure of the bash shell, which is used to store multiple data's.
- Simple array: `myarray=( ls pwd date 2 5.6 )`      #No limit for length of an array

## ➤ How to Define/declare an array ?

- There are different ways to define an array in bash shell scripting.
  - Empty Array: `myArray=()`
  - `mycmds=( ls pwd date 2 5.6 )`
  - `myNewArray=( ls -lrt hostname -s )`
  - `myNewArray=( "ls -lrt" "hostname -s" )`
  - **declare -a NewArray**
  - **NewArray=( 1 3 4 5 bash scripting)**

# How to access Array values/elements ?

- Basically, Bash Shell Array is the **zero-based Array** (i.e., indexing start with 0)
- Then what is an index ?
- `myarray=(23 4 6 15 5 7)`

**myarray**

23	4	6	15	5	7
----	---	---	----	---	---

0 1 2 3 4 5



**Positive Index Values or Positive Indices**

-6 -5 -4 -3 -2 -1



**Negative Index Values or Negative Indices**

- `echo "$myarray"`
- `echo "${myarray}"`
- `echo "${myarray[*]}"`
- `echo "${myarray[@]}"`
- `echo "${myarray[0]}"` → Prints first value
- `echo "${myarray[-1]}"` → Prints last value

Learn How to Automate Common Tasks with **Bash** Shell Scripting

# How to access Array values/elements ?...

- `echo "${myarray[*]:0}"` → Prints all the values starting from index-0
- `echo "${myarray[*]:1}"` → Prints all the values starting from index-1
- `echo "${myarray[*]:0:2}"` → Prints two values starting from index-0
- `echo "${myarray[*]:1:2}"` → Prints two values starting from index-1
- `echo "${!myarray[*]}"` → Prints index values of array
- `echo "${#myarray[*]}"` → Find the length (number elements) of array
  
- We can also customize index numbers:
  - `newarray[5]="bash"`
  - `newarray[9]="shell scripting"`
  - Or
  - `newarray=( [5]="bash" [9]="shell scripting" )`

Note: We can also take indices as strings and that array is called Associative Array.

# Different Types of Arrays

- We have two types of arrays in Bash Shell Scripting.
- They are:
  - Index Based Arrays or Arrays
  - Associative Arrays.

# How to store the command output into an array ?

- Storing the output of a command into array:
  - `arraywithcmd=( $(command) )`

# How to delete and update an exiting array ?

- Delete an array or even normal variable:
  - `unset variable/arrayvariable`
- Updating an exiting array:
  - `myarray=(1,2,3)`
  - `myarray+=(4,5,6)`



# How to read an array using read command ?

- Syntax:
  - `read -a myarray`
  - `read -p "Enter your array" -a myarray`

**Thank you**