

5. Shell Scripting – Arrays

- In bash scripting arrays are collection of data irrespective of data type.
- The different data are stored in an array as different elements of the array.
- The array will be stored in memory as a block of data where different elements are stored in adjacent memory locations.
- The array can have any name and different elements can be accessed using array name along with index numbers.
- By default, the indexing of array elements starts from zero.

Difference between list and array:

List and array stores data as different elements. The main difference between a list and an array is that any element of the array can be accessed using index number but the same is not possible with list.

Array:

The elements to the array can be given using below format:

array_name=(elements separated with spaces)

Methods to read elements to array:

1. Using read command:

read -p "Enter the array elements: " -a array_name

Here, -a option is given to read command to read array elements. Enter different elements of the array separated with space and then press enter.

2. Using for loop:

declare -a array_name

read -p "Enter the array length: " n

echo Enter the array elements:

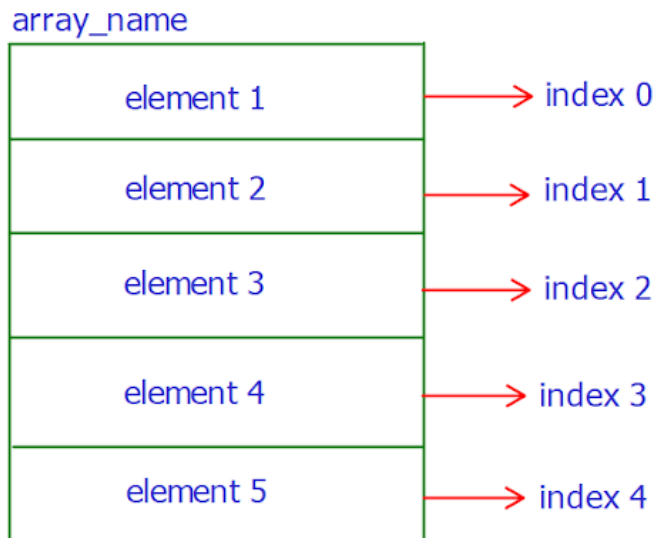
for i in `seq 0 \$((n - 1))` i → iterative variable

do

read array_name[\$i]

done

Enter one by one elements to the array by pressing enter key after every element.



Above given is an example of how different elements of an array will be stored in memory. The elements to the array will be stored in the same order in which the data is given to the array. In the above example element1 is the first data assigned to the array and element5 is the last data assigned to the array.

Array length:

The total number of elements in an array is called the length of array or array length.

Always the indexing of array elements in array ranges from **0** to **(array_length - 1)**. In the above example array length is 5.

Array formats:

The different formats to work with array in bash scripting are as given below:

- | | |
|----------------------------------------|-------------------------------------------------------------------|
| <code>\${array_name[@]}</code> | → to get all elements of array |
| <code>\${array_name[*]}</code> | → to get all elements of array |
| <code>\${array_name}</code> | → to access 1st element of array |
| <code>\${array_name[i]}</code> | → to access ith element of array (i → index no) |
| <code>\${array_name[-1]}</code> | → to access last element of array |

Add # in front of array name to get the length.

- | | |
|----------------------------------------|-----------------------------------------------------------|
| <code>\${#array_name[@]}</code> | → to get array length |
| <code>\${#array_name[*]}</code> | → to get array length |
| <code>\${#array_name}</code> | → to get length of 1st element of array |
| <code>\${#array_name[i]}</code> | → to get length of ith element of array |

`${#array_name[-1]}` → to get length of last element of array

`${array_name[${#array_name[@]} - 1]}`

→ to access last element of array

`${#array_name[${#array_name[@]} - 1]}`

→ to get length of last element of array