

Nama: Yasmin Ulayya

NPM: 21083010033

Kelas: Sistem Operasi A

## Array

Bash array adalah struktur data yang dirancang untuk menyimpan informasi secara terindeks. Dengan kata lain, array ini merupakan sekumpulan besar variabel. Tidak seperti array biasa yang digunakan dalam bahasa pemrograman lainnya, bash array bisa menyimpan berbagai jenis elemen.

Jenis-jenis bash array dulu yang terbagi menjadi:

- Indexed – array direferensikan melalui bilangan bulat atau angka.
- Associative – array direferensikan melalui string atau rangkaian karakter dan kata.

Macam- macam deklarasi array:

### 1. Array Indirect Declaration (Array\_ID.sh)

Dengan menetapkan nilai dalam indeks tertentu dari variabel array. Ini dilakukan **tanpa declare builtin**.

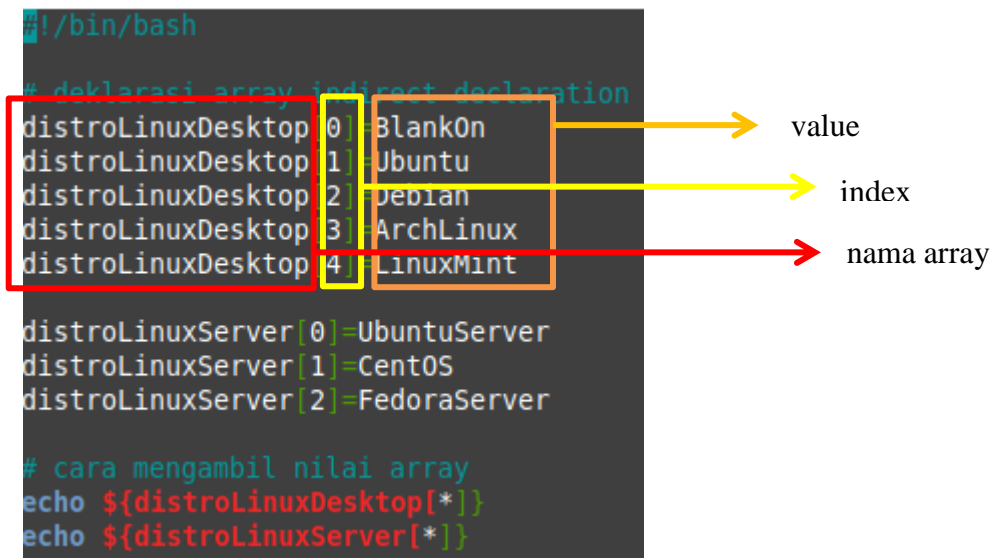
`Array_name[index]=value`

Index selalu dimulai dari angka 0, cara di bawah ini kita memasukkan value dari array dengan satu persatu. `echo $nama array` akan menampilkan seluruh value dari array, dan index dengan penulisan `[*]` berarti keseluruhan.

```
#!/bin/bash
# deklarasi array indirect declaration
distroLinuxDesktop[0]=BlankOn
distroLinuxDesktop[1]=Ubuntu
distroLinuxDesktop[2]=Debian
distroLinuxDesktop[3]=ArchLinux
distroLinuxDesktop[4]=LinuxMint

distroLinuxServer[0]=UbuntuServer
distroLinuxServer[1]=CentOS
distroLinuxServer[2]=FedoraServer

# cara mengambil nilai array
echo ${distroLinuxDesktop[*]}
echo ${distroLinuxServer[*]}
```



Dia akan mengeluarkan output kumpulan nilai dari array distroLinuxDestop dan array distroLinuxServer.

```
yasminlyya@yasminlyya-VirtualBox:~$ bash Array_ID.sh
BlankOn Ubuntu Debian ArchLinux LinuxMint
UbuntuServer CentOS FedoraServer
```

## 2. Array Explicit Declaration (Array\_ED.sh)

Dengan mendeklarasikan array kemudian menetapkan nilainya. Mendeklarasikan sebuah array menggunakan **shell built-in declare dengan flag -a** dan memberikan elemen untuk array tersebut.

`declare -a Array_name`

pada gambar dibawah kita menggunakan while looping. Karena kita menginginkan pengoperasian dimulai dari angka 0, maka masukkan index atau dibawah kita simbolkan dengan  $i=0$ . Lalu perintah loopingnya ialah nilai  $i$  sampai dengan less than 4 akan dilakukan pengoperasian matematika dengan perintah `let`. Untuk variabel isi berisi pengoperasian matematika nilai  $i$  dikali 2. Dengan setiap sudah melakukan pengoperasian kita akan ulangi lagi untuk angka berikutnya selama masih  $\leq 4$  ini seperti return, jika sudah tidak memenuhi dari while looping maka akan diselesaikan. Setelah itu gunakan `echo` untuk memprint, `[*]` atau `[@]` untuk semua elemen array.

```
#!/bin/bash

# deklarasikan array [Explicit Declaration] :
declare -a angka

#clear
i=0;
while [ $i -le 4 ];
do
let isi=$i*2;
angka[$i]=$isi;
let i=$i+1;
done

#tampilkan semua elemen array
#dengan indexnya berisi "*" atau "@"
echo ${angka[@]}
```

Inilah pengoperasian matematika yang terjadi:

$$0 \times 2 = 0$$

$$1 \times 2 = 2$$

$$2 \times 2 = 4$$

$$3 \times 2 = 6$$

$$4 \times 2 = 8$$

```
yasminlyya@yasminlyya-VirtualBox:~$ bash Array_ED.sh
0 2 4 6 8
```

### 3. Array Compound Assignment (Array\_CA.sh)

Mendeklarasikan array dengan sekelompok nilai.

Array\_name=([1]=10 [2]=20 [3]=30)

Ini kita cukup sekali menuliskan nama array dan langsung menulis elemen-elemen array tersebut, indexnya akan berurutan dimulai dari 0.

```
Index 0      Index 1      Index 2      Index 3      Index 4
  ↑          ↑          ↑          ↑          ↑
#!/bin/bash

#deklarasi array compound assignment
distroLinuxDesktop=('BlankOn' 'Ubuntu' 'Debian' 'ArchLinux' 'LinuxMint')
distroLinuxServer=('UbuntuServer' 'CentOS' 'FedoraServer')

#cara mengambil nilai array
echo ${distroLinuxDesktop[*]}
echo ${distroLinuxServer[*]}
```

```
yasminlyya@yasminlyya-VirtualBox:~$ bash Array_CA.sh
BlankOn Ubuntu Debian ArchLinux LinuxMint
UbuntuServer CentOS FedoraServer
```

### Array Multi Dimensi (Array\_MD.sh)

Didalam Script Bash, Bash tidak memiliki array multi dimensi. Dikarenakan Bash menyediakan variabel array terindeks dan asosiatif satu dimensi. Variabel apa saja dapat digunakan sebagai array yang diindeks, mendeklarasikan builtin akan secara eksplisit mendeklarasikan array, dll. Tetapi dapat mensimulasikan efek yang agak mirip dengan multi array asosiatif dimensi.

Tanda `(;)` sebagai pemisah dengan `kolom` berikutnya, sedangkan `spasi` sebagai pemisah untuk `baris` berikutnya.

```
#!/bin/bash

# deklarasi array2dimensi " : " pemisah nilai (array [3][4])
array2dimensi="1.1:1.2:1.3:1.4 2.1:2.2:2.3:2.4 3.1:3.2:3.3:3.4"

# mengakali multi dimensi -> dengan pemisah dimensi "tr :"
function dimensiBaris {
    for baris in $array2dimensi
    do
        dimensiKolom `echo $baris | tr : " "`
    done
}

function dimensiKolom {
    for kolom in $*
    do
        echo -n $kolom " "
    done
    echo
}

# Contoh penggunaan
dimensiBaris
```

```
yasminlyya@yasminlyya-VirtualBox:~$ bash Array_MD.sh
1.1 1.2 1.3 1.4
2.1 2.2 2.3 2.4
3.1 3.2 3.3 3.4
```

### Soal Latihan

Di sini deklarasi array yang digunakan ialah Array Explicit Declaration. Lalu kita gunakan perintah input untuk jumlah semesternya dan kita gunakan looping for untuk penginputan nilai IPS dan juga perhitungan IPKnya.

```
#!/bin/bash

declare -a tulis;
echo

echo -n "masukkan semester: ";read semester;

jumlah=0;
ipk_mhs=0;

for ((i=1; i<=semester; i++))
do
    echo -n "masukkan nilai IPS semester $i: ";read tulis[$i];
    let jumlah=$jumlah+${tulis[i]};
    let ipk_mhs=$jumlah/$semester;
done
echo "IPS mhs:$jumlah/$semester"
echo "IPK mhs:$ipk_mhs"
```

Dan seperti ini outputnya.

```
yasminlyya@yasminlyya-VirtualBox:~$ bash Tugas6.sh
masukkan semester: 3
masukkan nilai IPS semester 1: 9
masukkan nilai IPS semester 2: 8
masukkan nilai IPS semester 3: 9
IPS mhs:26/3
IPK mhs:8
```