

Nama : Vera Febrianti Pakpahan

NPM : 21083010054

Kelas : Sistem Operasi A

## 1. Penggunaan String

String adalah satu atau serangkaian karakter yang diletakkan diantara tanda kutip, baik tanda kutip tunggal ( ' ) maupun ganda ( " )

- Buat file dengan menggunakan perintah nano lalu beri nama 'String.sh'.

```
vera@vera-VirtualBox:~$ ls
BelajarBashDasar  Documents  Music      Practice  Templates  Tugas-1.sh
Desktop           Downloads  Pictures   Public    Tugas-1.py.save  Videos
vera@vera-VirtualBox:~$ nano String.sh
vera@vera-VirtualBox:~$
```

- Masukkan script yang ingin dijalankan

```
File Edit View Search Terminal Help
GNU nano 6.2 String.sh *
a='ini single-quoted\n'
b="ini double-quoted\n"
echo -e ,a=$a,b=$b
```

- Ketik 'bash String.sh' untuk melihat outputnya

```
vera@vera-VirtualBox:~/Practice$ bash String.sh
,a=ini single-quoted
,b=ini double-quoted
vera@vera-VirtualBox:~/Practice$
```

## 2. Penggunaan Array

Dalam shell programming array dapat di definisikan dengan cara memberikan index kepada suatu variabel

- Buat file dengan perintah nano dan beri nama 'Array.sh'

```
vera@vera-VirtualBox:~/Practice$ nano Array.sh
vera@vera-VirtualBox:~/Practice$
```

- Masukkan contoh script seperti gambar di bawah ini

```
GNU nano 6.2 Array.sh *
#!/bin/bash

# deklarasi array
distroLinux=("Mint" "Ubuntu" "Kali" "Arch" "Debian")

# random distro
let pilih=$((RANDOM%5))

# eksekusi
echo "Saya Memilih Distro $pilih, ${distroLinux[$pilih]} !"
```

- Ketik 'bash Array.sh' untuk melihat outputnya

```
vera@vera-VirtualBox:~/Practice$ bash Array.sh
Saya Memilih Distro 1, Ubuntu !
vera@vera-VirtualBox:~/Practice$ bash Array.sh
Saya Memilih Distro 2, Kali !
vera@vera-VirtualBox:~/Practice$ bash Array.sh
Saya Memilih Distro 0, Mint !
vera@vera-VirtualBox:~/Practice$ bash Array.sh
Saya Memilih Distro 3, Arch !
vera@vera-VirtualBox:~/Practice$ bash Array.sh
Saya Memilih Distro 2, Kali !
vera@vera-VirtualBox:~/Practice$ bash Array.sh
Saya Memilih Distro 4, Debian !
vera@vera-VirtualBox:~/Practice$ bash Array.sh
Saya Memilih Distro 4, Debian !
vera@vera-VirtualBox:~/Practice$ bash Array.sh
Saya Memilih Distro 3, Arch !
vera@vera-VirtualBox:~/Practice$ bash Array.sh
Saya Memilih Distro 4, Debian !
vera@vera-VirtualBox:~/Practice$ bash Array.sh
Saya Memilih Distro 4, Debian !
vera@vera-VirtualBox:~/Practice$ bash Array.sh
Saya Memilih Distro 3, Arch !
vera@vera-VirtualBox:~/Practice$
```

### 3. Penggunaan Integer

Integer merupakan data type yang berbentuk bilangan bulat atau numerik.

- Buat file dan beri nama Integer.sh

```
vera@vera-VirtualBox:~/Practice$ nano Integer.sh
vera@vera-VirtualBox:~/Practice$
```

- Ketik script misal seperti gambar dibawah ini

```
GNU nano 6.2 Integer.sh *
a=123
b=456
let c=a+b
echo $c
```

- Untuk melihat outputnya, ketik bash Integer.sh

```
vera@vera-VirtualBox:~/Practice$ bash Integer.sh
579
vera@vera-VirtualBox:~/Practice$
```

#### 4. Penggunaan Operasi Matematika

Ada 3 cara yang dapat digunakan untuk melakukan operasi matematika, yaitu:

- o Menggunakan perintah built-in let
- o Menggunakan perintah eksternal expr atau awk
- o Menggunakan perintah substitusi \$((ekspresi))

- Buat file 'Op\_mtk.sh

```
vera@vera-VirtualBox:~/Practice$ nano Op_mtk.sh
vera@vera-VirtualBox:~/Practice$
```

Contoh operasi "Let, Expr, Ekspresi" :

```
GNU nano 6.2 Op mtk.sh
#!/bin/bash

a=15
b=7

#memakai let
let jumlah=$a+$b
let kurang=$a-$b
let kali=$a*$b

#memakai expr
bagi=expr $a / $b

#memakai perintah substitusi $((ekspresi))
mod=$(( $a % $b ))

echo "a + b = $jumlah"
echo "a - b = $kurang"
echo "a * b = $kali"
echo "a / b = $bagi"
echo "a % b = $mod"

b=$a

echo "a = $a"
echo "b = $b"
```

let = digunakan untuk melakukan operasi perhitungan aritmatika

- Output

```
vera@vera-VirtualBox:~/Practice$ bash Op_mtk.sh
a + b = 22
a - b = 8
a * b = 105
a / b = 2
a % b = 1
a = 15
b = 15
vera@vera-VirtualBox:~/Practice$
```

## 5. Pembuatan Input

Untuk membaca inputan dari user gunakanlah "read" dengan format penulisan "read nama\_var"

- Buat file 'Input.sh'

```
vera@vera-VirtualBox:~/Practice$ nano Input.sh
vera@vera-VirtualBox:~/Practice$
```

- Contoh pembuatan input

```
GNU nano 6.2 Input.sh
#!/bin/bash
clear

echo -n "Hallo, masukkan nama anda : ";
read Vera;
echo "Selamat datang $Vera";
```

- Output

```
vera@vera-VirtualBox: ~/Practice
File Edit View Search Terminal Help
Hallo, masukkan nama anda : Vera
Selamat datang Vera
vera@vera-VirtualBox:~/Practice$
```

## 6. Penampilan Output1

echo digunakan untuk menampilkan output dengan syntax sebagai berikut:

- o Menampilkan teks biasa echo "teks"
- o Menampilkan isi dari sebuah variabel echo \$nama\_var

- Buat file 'Output1.sh'

```
vera@vera-VirtualBox:~/ShellScripting$ nano Output1.sh
vera@vera-VirtualBox:~/ShellScripting$
```

- Jika ingin menggunakan new line character (\n) pada echo, gunakan echo -e "teks\n teks"

```

GNU nano 6.2                                     Output1.sh
#!/bin/bash
matakuliah="Sistem Operasi"

echo "Siapa namamu?"
read nama
echo -e "\nHai $Vera!\nSelamat datang di praktikum $matakuliah:)))))"

```

- Output

```

vera@vera-VirtualBox:~/ShellScripting$ bash Output1.sh
Siapa namamu?
Vera

Hai !
Selamat datang di praktikum Sistem Operasi:)))))
vera@vera-VirtualBox:~/ShellScripting$

```

## 7. Penampilan Output2

Selain menggunakan output "echo", output bisa juga dilakukan dengan menggunakan printf seperti pada bahasa pemrograman C.

- Buat file 'Output2.sh'

```

vera@vera-VirtualBox:~/ShellScripting$ nano Output2.sh
vera@vera-VirtualBox:~/ShellScripting$

```

- Contoh

```

GNU nano 6.2                                     Output2.sh *
#!/bin/bash

# Inisialisasi Var
a=55;
b=4;
distroLinux="Ubuntu 19.04 LTS";
let c=a*b;

# Output Printf
printf "OS : $distroLinux \n";
printf "$c \n";
printf "%.2f float \n" $a;
printf "%.1f float \n" $a

```

- Output

```

vera@vera-VirtualBox:~/ShellScripting$ bash Output2.sh
OS : Ubuntu 19.04 LTS
3
55,00 float
55,0 float
vera@vera-VirtualBox:~/ShellScripting$

```

## 8. Percabangan1

Sebelum melangkah ke percabangan alangkah baiknya terlebih dahulu mengetahui dasar operasi relasional yang mana biasanya digunakan bersama dengan conditional statements.

No Operator Deskripsi

1 -eq Memeriksa apakah nilai kedua operan sama (==)

- 2 -ne Memeriksa apakah nilai kedua operan tidak sama (!=)
- 3 -gt Memeriksa apakah nilai operan kiri lebih besar daripada operan kanan (>)
- 4 -lt Memeriksa apakah nilai operan kiri lebih kecil daripada operan kanan (<)
- 5 -ge Memeriksa apakah nilai operan kiri lebih besar atau sama dengan operan kanan (>=)
- 6 -le Memeriksa apakah nilai operan kiri lebih kecil atau sama dengan operan kanan (<=)

- Buat file 'Percabangan1.sh'

```
vera@vera-VirtualBox:~/ShellScripting$ nano Percabangan1.sh
vera@vera-VirtualBox:~/ShellScripting$
```

- If ..... Else
  - o Syntax
  - if [ kondisi1 ]
  - then
  - perintah1
  - elif [ kondisi2 ]
  - then
  - perintah2
  - else
  - alternatif\_perintah
  - fi

- Contoh

```
GNU nano 6.2 Percabangan1.sh *
#!/bin/bash

a=15
b=7

if [ $a == $b ]
then
    echo "a sama dengan b"
elif [ $a -gt $b ]
then
    echo "a lebih besar dari b"
elif [ $a -lt $b ]
then
    echo "a lebih kecil dari b"
else
    echo "Tidak ada kondisi yang memenuhi"
fi
```

- Output

```
vera@vera-VirtualBox:~/ShellScripting$ bash Percabangan1.sh
a lebih besar dari b
vera@vera-VirtualBox:~/ShellScripting$
```

## 9. Percabangan2

- Buat file baru beri nama 'Percabangan2.sh'

```
vera@vera-VirtualBox:~/ShellScripting$ nano Percabangan2.sh
vera@vera-VirtualBox:~/ShellScripting$
```

- Case .... Esac
  - o Syntax
    - pola1)
      - perintah1
      - ::
    - pola2)
      - perintah2
      - ::
    - \*)
      - alternatif\_perintah
      - ::

```
GNU nano 6.2 Percabangan2.sh *
#!/bin/bash
printf "Jajan apa yang kamu suka ?\n"
printf "pentol ?\n"
printf "batagor ?\n"
printf "cireng ?\n"

read jajan

case "$jajan" in
    "pentol")
        echo "Pentol buk mah wenak slur!"
        ;;
    "batagor")
        echo "Batagor mas budi mantap bat"
        ;;
    "cireng")
        echo "Cireng kantin rasane unch-unch"
        ;;
    *)
        echo "Makanan yang kamu suka gaenak hehe"
        ;;
esac
```

- Output
 

```
vera@vera-VirtualBox:~/ShellScripting$ bash Percabangan2.sh
Jajan apa yang kamu suka ?
pentol ?
batagor ?
cireng ?
batagor
Batagor mas budi mantap bat
vera@vera-VirtualBox:~/ShellScripting$ bash Percabangan2.sh
Jajan apa yang kamu suka ?
pentol ?
batagor ?
cireng ?
coklat
Makanan yang kamu suka gaenak hehe
vera@vera-VirtualBox:~/ShellScripting$
```

Laporan Tugas  
Program percabangan sederhana aritmatika

```
vera@vera-VirtualBox:~/ShellScripting$ nano Tugas_2.sh
vera@vera-VirtualBox:~/ShellScripting$
```

```
GNU nano 6.2                                     Tugas_2.sh
#!/bin/bash

a=10
b=0

if [ $a -gt $b ]
then
    echo "angka merupakan bilangan positif"
elif [ $a -lt $b ]
then
    echo "angka merupakan bilangan negatif"
else
    echo "angka merupakan nol"
fi
```

- Output

```
vera@vera-VirtualBox:~/ShellScripting$ bash Tugas_2.sh
angka merupakan bilangan positif
vera@vera-VirtualBox:~/ShellScripting$
```