

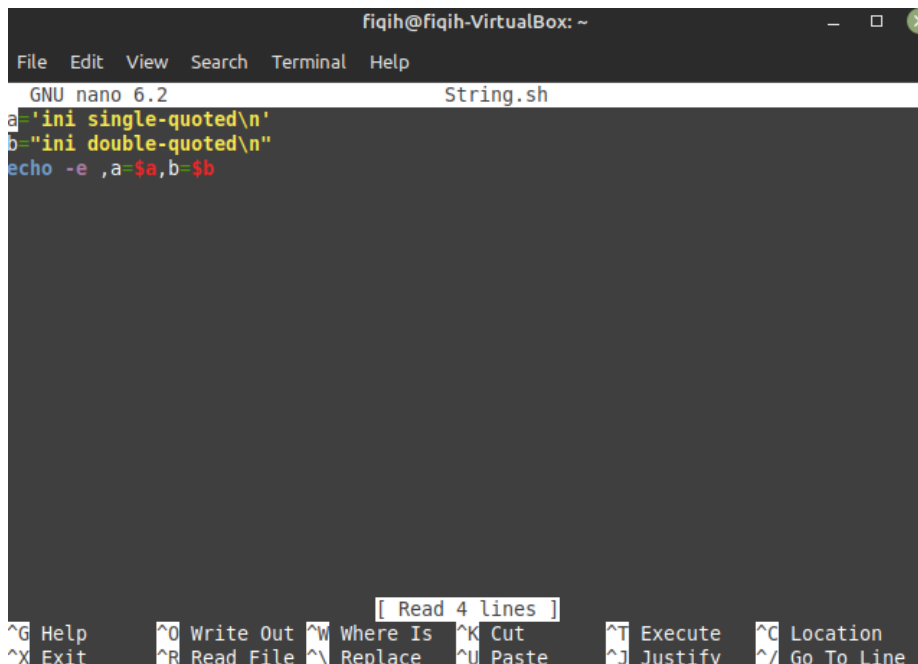
Nama : Fiqih Pavita Andharluana
NPM : 21083010042
Kelas : Sistem Operasi-A

Dokumentasi Shell Scripting

1. Penggunann String (String.sh)

```
fiqih@fiqih-VirtualBox:~$ nano String.sh
fiqih@fiqih-VirtualBox:~$ bash String.sh
,a=ini single-quoted
,b=ini double-quoted
```

Pertama buat file .sh dengan perintah nano pada terminal. Setelah itu ketikkan script yang ingin dijalankan seperti pada gambar dibawah ini. Untuk menampilkan output dari script tersebut menggunakan perintah bash namafile.sh



```
fiqih@fiqih-VirtualBox: ~
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
[ Read 4 lines ]
^G Help ^O Write Out ^W Where Is ^K Cut ^T Execute ^C Location
^X Exit ^R Read File ^\ Replace ^U Paste ^J Justify ^_ Go To Line
```

Perintah echo digunakan untuk mencetak string. Perintah echo -e untuk mengaktifkan interpretasi karakter escape.

2. Penggunaan Array (array.sh)

```
fiqih@fiqih-VirtualBox:~$ nano Array.sh
fiqih@fiqih-VirtualBox:~$ bash Array.sh
saya Memilih Distro 1, Ubuntu
fiqih@fiqih-VirtualBox:~$ bash Array.sh
saya Memilih Distro 3, Arch
fiqih@fiqih-VirtualBox:~$ bash Array.sh
saya Memilih Distro 0, Mint
fiqih@fiqih-VirtualBox:~$
```

Pertama buat file .sh dengan perintah nano pada terminal. Setelah itu ketikkan script yang ingin dijalankan seperti pada gambar dibawah ini. Untuk menampilkan output dari script tersebut menggunakan perintah bash namafile.sh



```
fiqih@fiqih-VirtualBox: ~
File Edit View Search Terminal Help
GNU nano 6.2 Array.sh
#!/bin/bash

# Deklarasi array
distrolinux=("Mint" "Ubuntu" "Kali" "Arch" "Debian")

# random distro
let pilih=$RANDOM%5

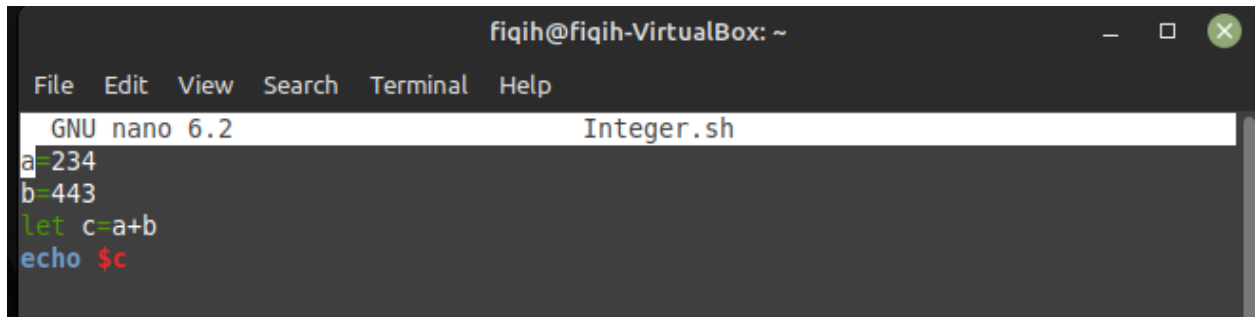
# eksekusi
echo "saya Memilih Distro $pilih, ${distrolinux[$pilih]}"
```

Tanda #! pada /bin/bash dalam script tes adalah perintah yang diterjemahkan ke kernel linux untuk mengeksekusi path yang disertakan dalam hal ini program bash pada direktory /bin, sebenarnya tanpa mengikuti baris tersebut anda tetap dapat mengeksekusi script bash, dengan catatan bash adalah shell aktif.

3. Penggunaan Integer (Integer.sh)

```
fiqih@fiqih-VirtualBox:~$ nano Integer.sh
fiqih@fiqih-VirtualBox:~$ bash Integer.sh
677
```

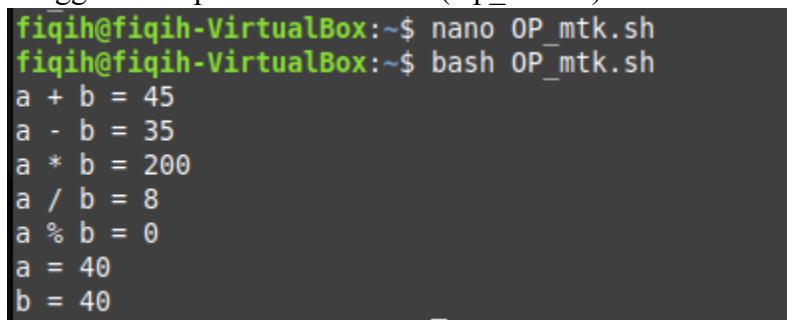
Pertama buat file .sh dengan perintah nano pada terminal. Setelah itu ketikkan script yang ingin dijalankan seperti pada gambar dibawah ini. Untuk menampilkan output dari script tersebut menggunakan perintah bash namafile.sh



```
fiqih@fiqih-VirtualBox: ~
File Edit View Search Terminal Help
GNU nano 6.2 Integer.sh
a=234
b=443
let c=a+b
echo $c
```

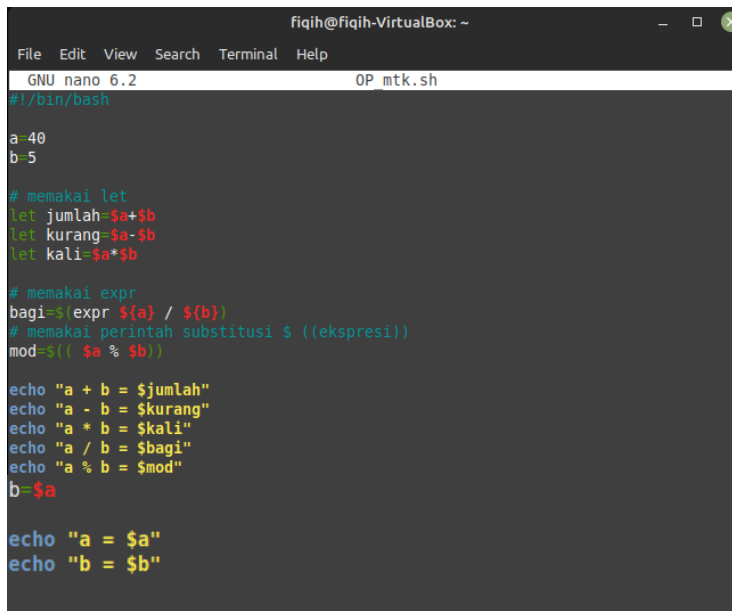
Tanda \$ pada echo \$c digunakan sebagai perintah substitusi untuk mengeksekusi variabel c. let digunakan untuk melakukan perhitungan operasi aritmatika.

4. Penggunaan Operasi Matematika (Op_mtk.sh)



```
fiqih@fiqih-VirtualBox:~$ nano OP_mtk.sh
fiqih@fiqih-VirtualBox:~$ bash OP_mtk.sh
a + b = 45
a - b = 35
a * b = 200
a / b = 8
a % b = 0
a = 40
b = 40
```

Pertama buat file .sh dengan perintah nano pada terminal. Setelah itu ketikkan script yang ingin dijalankan seperti pada gambar dibawah ini. Untuk menampilkan output dari script tersebut menggunakan perintah bash namafile.sh



```
fiqih@fiqih-VirtualBox: ~
File Edit View Search Terminal Help
GNU nano 6.2 OP_mtk.sh
#!/bin/bash
a=40
b=5

# 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

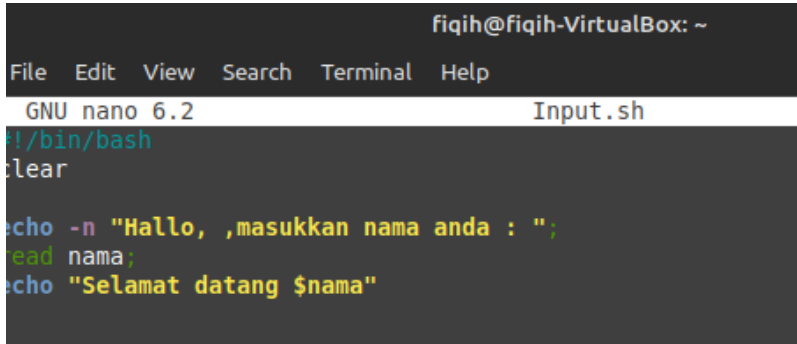
echo "a = $a"
echo "b = $b"
```

Perintah yang digunakan untuk melakukan operasi aritmatika diantaranya adalah let, expr, serta perintah substitusi. Perintah expr menggunakan tanda {} untuk memanggil variabelnya. Untuk perintah substitusi menggunakan kurung ganda (())

5. Pembuatan Input (input.sh)

```
fiqih@fiqih-VirtualBox:~$ nano Input.sh
fiqih@fiqih-VirtualBox:~$ bash Input.sh
```

Pertama buat file .sh dengan perintah nano pada terminal. Setelah itu ketikkan script yang ingin dijalankan seperti pada gambar dibawah ini. Untuk menampilkan output dari script tersebut menggunakan perintah bash namafile.sh

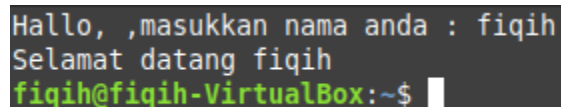


```
fiqih@fiqih-VirtualBox: ~
File Edit View Search Terminal Help
GNU nano 6.2 Input.sh
#!/bin/bash
clear

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

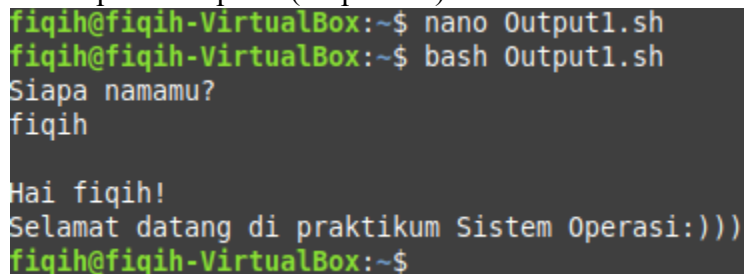
Perintah clear digunakan untuk membersihkan layar terminal. Inputan yang dimasukkan akan tersimpan pada variabel nama. Untuk memanggil variabel nama gunakan \$nama.

Output ketika menjalankan Input.sh:



```
Hallo, ,masukkan nama anda : fiqih
Selamat datang fiqih
fiqih@fiqih-VirtualBox:~$
```

6. Penampilan Output 1 (output1.sh)



```
fiqih@fiqih-VirtualBox:~$ nano Output1.sh
fiqih@fiqih-VirtualBox:~$ bash Output1.sh
Siapa namamu?
fiqih

Hai fiqih!
Selamat datang di praktikum Sistem Operasi:)))
fiqih@fiqih-VirtualBox:~$
```

Pertama buat file .sh dengan perintah nano pada terminal. Setelah itu ketikkan script yang ingin dijalankan seperti pada gambar dibawah ini. Untuk menampilkan output dari script tersebut menggunakan perintah bash namafile.sh

```
fiqih@fiqih-VirtualBox: ~
File Edit View Search Terminal Help
GNU nano 6.2 Output1.sh
#!/bin/bash
matakuliah="Sistem Operasi"

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

Menampilkan output menggunakan perintah echo. Untuk menampilkan teks biasa menggunakan echo "teks". Untuk menampilkan isi dari sebuah variabel menggunakan echo \$namavariabel. Untuk menggunakan new line gunakan echo -e atau \n.

7. Pembuatan Output 2 (output2.sh)

```
fiqih@fiqih-VirtualBox:~$ nano Output2.sh
fiqih@fiqih-VirtualBox:~$ bash Output2.sh
OS : Ubuntu 19.06 LTS
3
55,00 float
55,0 float
```

Pertama buat file .sh dengan perintah nano pada terminal. Setelah itu ketikkan script yang ingin dijalankan seperti pada gambar dibawah ini. Untuk menampilkan output dari script tersebut menggunakan perintah bash namafile.sh

```
fiqih@fiqih-VirtualBox: ~
File Edit View Search Terminal Help
GNU nano 6.2 Output2.sh
#!/bin/bash

# inisialisasi var
a=55;
b=4;
distroLinux="Ubuntu 19.06 LTS";
let c=a%b;

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

Menampilkan output menggunakan perintah *printf*. perintah `%.2f` float untuk menampilkan hasil dalam bentuk float serta menampilkan 2 angka dibelakang koma. perintah `%.1f` float untuk menampilkan hasil dalam bentuk float serta menampilkan 1 angka dibelakang koma

8. Percabangan 1 (percabangan1.sh)

```
fiqih@fiqih-VirtualBox:~$ nano Percabangan1.sh
fiqih@fiqih-VirtualBox:~$ bash Percabangan1.sh
a lebih besar dari b
```

Pertama buat file .sh dengan perintah nano pada terminal. Setelah itu ketikkan script yang ingin dijalankan seperti pada gambar dibawah ini. Untuk menampilkan output dari script tersebut menggunakan perintah bash namafile.sh

```
fiqih@fiqih-VirtualBox: ~
File Edit View Search Terminal Help
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
```

melakukan percabangan dengan
(If.....Else)
menggunakan statement dasar
if kondisi
then
perintah
elif
perintah
else
perintah
fi
-gt artinya lebih besar dari
-lt artinya kurang dari

9. Percabangan 2 (percabangan2.sh)

```
fiqih@fiqih-VirtualBox:~$ nano Percabangan2.sh
fiqih@fiqih-VirtualBox:~$ bash Percabangan2.sh
Jajan apa yang kamu suka ?
Pentol ?
Batagor ?
Cireng ?
Cireng
Cireng kantin rasane unch-unch
fiqih@fiqih-VirtualBox:~$ bash Percabangan2.sh
Jajan apa yang kamu suka ?
Pentol ?
Batagor ?
Cireng ?
rendang
Makanan yang kamu suka gaenak
fiqih@fiqih-VirtualBox:~$
```

Pertama buat file .sh dengan perintah nano pada terminal. Setelah itu ketikkan script yang ingin dijalankan seperti pada gambar dibawah ini. Untuk menampilkan output dari script tersebut menggunakan perintah bash namafile.sh

```
fiqih@fiqih-VirtualBox: ~  
File Edit View Search Terminal Help  
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 mantep bat"  
        ;;  
    "Cireng")  
        echo "Cireng kantin rasane unch-unch"  
        ;;  
    *)  
        echo "Makanan yang kamu suka gaenak"  
        ;;  
esac
```

Melakukan percabangan dengan (Case.....Esac) dengan pola Pola1)

Perintah1

;;

Pola2)

Perintah2

;;

*)

Alternatif perintah

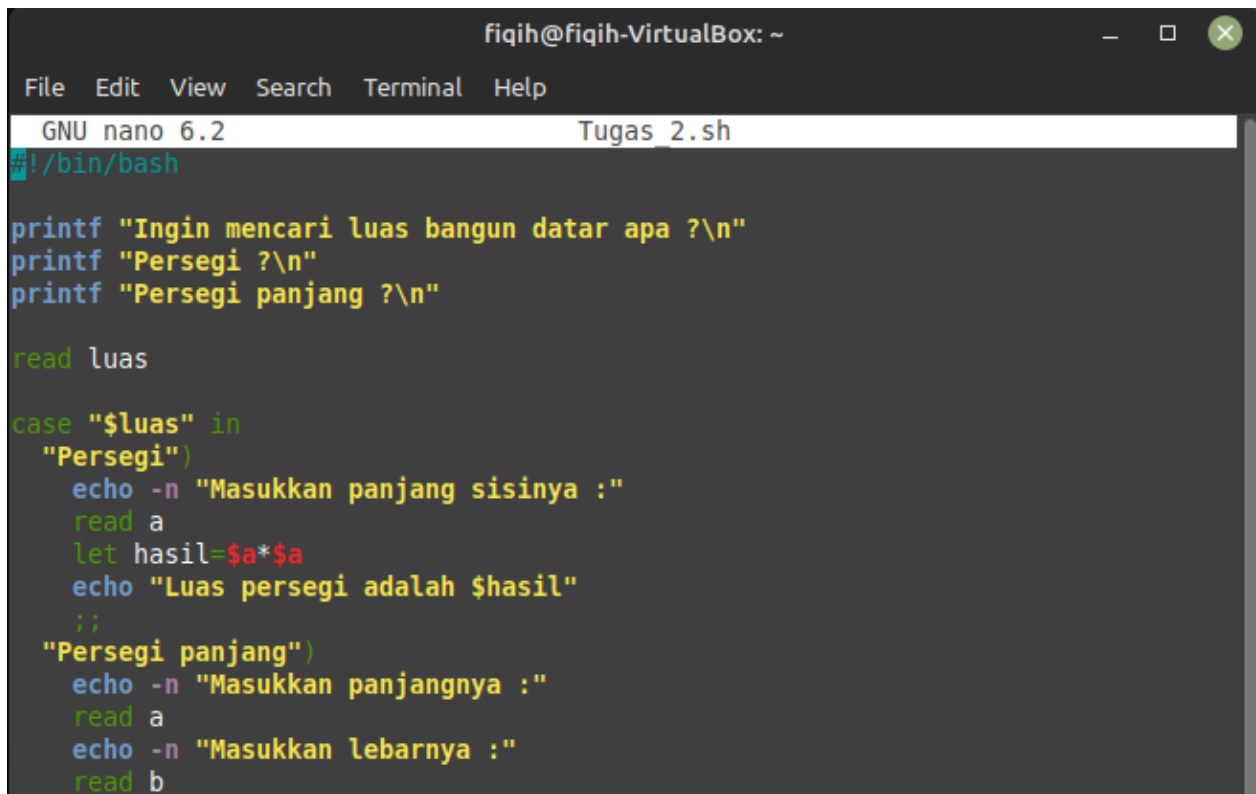
Esac

10. Tugas 2 (Tugas_2.sh)

Membuat penghitung luas sederhana dengan menggunakan percabangan (Case....Esac), Aritmatika , serta Input (echo -n)

```
fiqih@fiqih-VirtualBox:~$ nano Tugas_2.sh
fiqih@fiqih-VirtualBox:~$ bash Tugas_2.sh
Ingin mencari luas bangun datar apa ?
Persegi ?
Persegi panjang ?
Persegi panjang
Masukkan panjangnya :6
Masukkan lebarnya :8
Luas persegi panjang adalah 48
fiqih@fiqih-VirtualBox:~$ bash Tugas_2.sh
Ingin mencari luas bangun datar apa ?
Persegi ?
Persegi panjang ?
lingkaran
Perhitungan luas bangun datar tersebut belum tersedia
fiqih@fiqih-VirtualBox:~$
```

Pertama buat file .sh dengan perintah nano pada terminal. Setelah itu ketikkan script yang ingin dijalankan seperti pada gambar dibawah ini. Untuk menampilkan output dari script



```
fiqih@fiqih-VirtualBox: ~
File Edit View Search Terminal Help
GNU nano 6.2 Tugas_2.sh
#!/bin/bash

printf "Ingin mencari luas bangun datar apa ?\n"
printf "Persegi ?\n"
printf "Persegi panjang ?\n"

read luas

case "$luas" in
    "Persegi")
        echo -n "Masukkan panjang sisinya : "
        read a
        let hasil=a*a
        echo "Luas persegi adalah $hasil"
        ;;
    "Persegi panjang")
        echo -n "Masukkan panjangnya : "
        read a
        echo -n "Masukkan lebarnya : "
        read b
```

tersebut menggunakan perintah bash namafile.sh


```
echo -n "Masukkan lebarnya :"  
read b  
let hasil=$a*$b  
echo "Luas persegi panjang adalah $hasil"  
;;  
*)  
echo "Perhitungan luas bangun datar tersebut belum tersedia"  
;;  
esac  
█
```

Langkah pertama yang harus dilakukan adalah menampilkan output berupa judul program dan pilihan program menggunakan perintah `printf` serta menyimpannya pada variabel `luas` dengan perintah `read luas`. Setelah itu, membuat percabangan (Case....Esac) yang didalamnya terdapat operasi aritmatika untuk menghitung luas, serta input (`echo -n`) untuk memasukan panjang dan lebar dari bangun yang ingin dicari luasnya. Ketika kita memasukkan nama bangun datar yang tidak ada pada pilihan, maka output yang akan keluar adalah “Perhitungan luas bangun datar tersebut belum tersedia”.