



**LABORATORIUM PEMBELAJARAN ILMU KOMPUTER**  
**FAKULTAS ILMU KOMPUTER**  
**UNIVERSITAS BRAWIJAYA**

---

BAB : SHELL SCRIPTING  
NAMA : DANI ADRIAN  
NIM : 225150201111009  
TANGGAL : 14/03/2023  
ASISTEN : ZHAFRAN RAMA AZMI  
GIBRAN HAKIM

---

1. Quoting

- a. Pembuatan file quoting.sh dengan isi berikut ini.

<code>\$ echo 'Single Quote ini dengan adanya "Double Quote" di antaranya'</code>
<code>\$ echo "Double quote dengan \"double quote\" kah?"</code>
<code>\$ echo "Isi directory `pwd` berjumlah `ls   wc -l` file"</code>

- b. Tunjukkan hasilnya

**Jawab:**

**Code:**

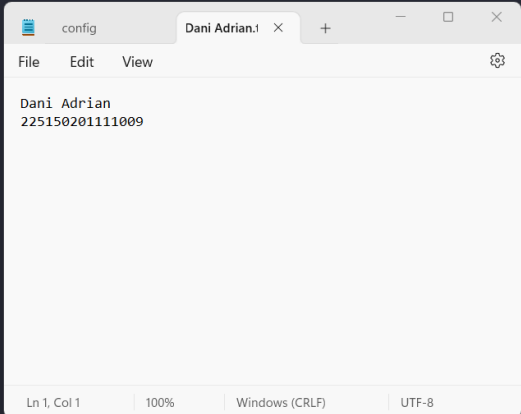
<code>#!/bin/bash</code>
<code>echo 'Single Quote ini dengan adanya "Double Quote" di antaranya'</code>
<code>echo "Double quote dengan \"double quote\" kah?"</code>
<code>echo "Isi directory `pwd` berjumlah `ls -l   wc -l` file"</code>

**Screenshot:**



**LABORATORIUM PEMBELAJARAN ILMU KOMPUTER**  
**FAKULTAS ILMU KOMPUTER**  
**UNIVERSITAS BRAWIJAYA**

```
[ec2-user@ip-172-31-31-146 sisop1]$ ./quoting.sh
Single Quote ini dengan adanya "Double Quote" di antaranya
Double quote dengan "double quote" kah?
Isi directory /home/ec2-user/sisop1 berjumlah 1 file
[ec2-user@ip-172-31-31-146 sisop1]$
```



## 2. Pembuatan file .sh dengan nama arg.sh

Buatlah file untuk mencetak argumen yang disertakan pada pemanggilan program seperti di bawah ini

```
$ ./arg.sh satu dua tiga empat lima enam
```

Sehingga keluarnya menampilkan sesuai dengan argument yang diberikan

Argumen **satu**

Argumen **dua**

Argumen **tiga**

Argumen **empat**

Argumen **lima**

Argumen **enam**

Tip: gunakan for loop

**Jawab:**



**LABORATORIUM PEMBELAJARAN ILMU KOMPUTER**  
**FAKULTAS ILMU KOMPUTER**  
**UNIVERSITAS BRAWIJAYA**

**Code :**

```
#!/bin/bash
for arg in "$@"
do
    echo "Argumen $arg"
done
```

**Screenshot:**

```
[ec2-user@ip-172-31-31-146 ~]$ cd sisop1
[ec2-user@ip-172-31-31-146 sisop1]$ chmod +x arg.sh
[ec2-user@ip-172-31-31-146 sisop1]$ ./arg.sh satu dua tiga empat lima enam
Argumen satu
Argumen dua
Argumen tiga
Argumen empat
Argumen lima
Argumen enam
[ec2-user@ip-172-31-31-146 sisop1]$
```

**3. For loop**

Perbaiki program berikut sedemikian hingga hasil keluarannya menampilkan urutan nomor suatu daftar file.

```
n=1
for dat in $(ls); do
echo "File ke $1 adalah $dat"
n=$((n+1))
done
```

**Jawab:**



**LABORATORIUM PEMBELAJARAN ILMU KOMPUTER**  
**FAKULTAS ILMU KOMPUTER**  
**UNIVERSITAS BRAWIJAYA**

**Code:**

```
#!/bin/bash
n=1
for dat in $(ls); do
echo "File ke $n adalah $dat"
n=$((n+1))
done
```

**Screenshot:**

```
[ec2-user@ip-172-31-31-146 sisop1]$ ./nomor3.sh
"File ke 1 adalah arg.sh"
"File ke 2 adalah nomor3.sh"
"File ke 3 adalah quoting.sh"
[ec2-user@ip-172-31-31-146 sisop1]$
```

4. Dari soal c di atas, perbaiki programnya sehingga menunjukkan juga tipe dari file tersebut (misal file biasa/regular, direktori dan lainnya).

**Jawab:**

**Code:**

```
#!/bin/bash
n=1
for dat in $(ls); do
    file_type=""
    if [ -d "$dat" ]; then
        file_type="direktori"
    elif [ -f "$dat" ]; then
        file_type="file/regular"
    else
        file_type="unknown"
    fi
```



**LABORATORIUM PEMBELAJARAN ILMU KOMPUTER**  
**FAKULTAS ILMU KOMPUTER**  
**UNIVERSITAS BRAWIJAYA**

```
echo "File ke $n adalah $dat ($file_type)"
n=$((n+1))
done
```

**Screenshot:**

```
[ec2-user@ip-172-31-31-146 sisop1]$ chmod +x nomor4.sh
[ec2-user@ip-172-31-31-146 sisop1]$ ./nomor4.sh
File ke 1 adalah arg.sh (file/regular)
File ke 2 adalah nomor3.sh (file/regular)
File ke 3 adalah nomor4.sh (file/regular)
File ke 4 adalah quoting.sh (file/regular)
[ec2-user@ip-172-31-31-146 sisop1]$
```

Dani Adrian  
225150201111009

**5. While loop dengan break**

Buatlah program dengan “while true” dimana program ini akan secara interaktif meminta masukkan berupa angka. Selama angka yang dimasukkan adalah positif, maka angka-angka tersebut akan dijumlahkan. Jika angka yang dimasukkan negatif, maka akan keluar dari “while true” dan menampilkan hasil jumlah semua angka yang telah masuk.

**Jawab :**

**Code:**

```
#!/bin/bash
n=0
while true;do
    read -p "Masukkan angka : " nInput
    #echo $nInput
    if [[ $nInput -ge 0 ]]; then
        n=$((n+nInput))
    else
        break
    fi
done
```



**LABORATORIUM PEMBELAJARAN ILMU KOMPUTER**  
**FAKULTAS ILMU KOMPUTER**  
**UNIVERSITAS BRAWIJAYA**

fi
done
echo "Nilai Akhir : \$n"

**Screenshot:**

```
[ec2-user@ip-172-31-31-146 sisop1]$ ./nomor5.sh
Masukkan angka : 4
Masukkan angka : 5
Masukkan angka : 8
Masukkan angka : 3
Masukkan angka : 4
Masukkan angka : 2
Masukkan angka : -1
Nilai Akhir : 26
[ec2-user@ip-172-31-31-146 sisop1]$
```

6. Buatlah program interaktif (menggunakan read) dimana program ini akan menanyakan dua bilangan, dan satu tanda (t : tambah, k : kurang, b : bagi dan x : kali). Jika dijalankan, maka pengguna akan memasukkan angka pertama, lalu kedua dan tanda. Program akan memproses kedua bilangan sesuai dengan tanda yang dimasukkan.

**Jawab :**

<b>Code:</b>
<code>#!/bin/bash</code>
<code>read -p "Masukkan angka pertama : " angka1</code>
<code>read -p "Masukkan angka kedua : " angka2</code>
<code>read -p "Masukkan operasi yang diinginkan (t/k/b/x)</code>
<code>: " operator</code>
<code>if [[ "\$operator" == "t" ]]; then</code>
<code>    n=\$((angka1+angka2))</code>
<code>elif [[ "\$operator" == "k" ]]; then</code>
<code>    n=\$((angka1-angka2))</code>
<code>elif [[ "\$operator" == "b" ]]; then</code>

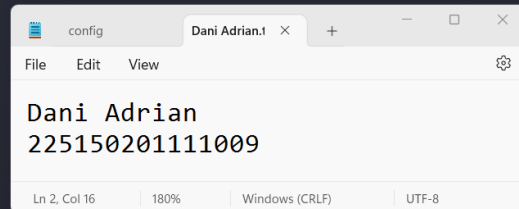


**LABORATORIUM PEMBELAJARAN ILMU KOMPUTER**  
**FAKULTAS ILMU KOMPUTER**  
**UNIVERSITAS BRAWIJAYA**

```
n=$((angka1/angka2))
elif [[ "$operator" == "x" ]]; then
    n=$((angka1*angka2))
else
    n="Error, input operator salah"
fi
echo "Hasil Akhir : $n"
```

**Screenshot:**

```
[ec2-user@ip-172-31-31-146 sisop1]$ chmod +x nomor6.sh
[ec2-user@ip-172-31-31-146 sisop1]$ ./nomor6.sh
Masukkan angka pertama : 5
Masukkan angka kedua : 6
Masukkan operasi yang diinginkan (t/k/b/x) : t
Hasil Akhir : 11
[ec2-user@ip-172-31-31-146 sisop1]$ ./nomor6.sh
Masukkan angka pertama : 10
Masukkan angka kedua : 5
Masukkan operasi yang diinginkan (t/k/b/x) : b
Hasil Akhir : 2
[ec2-user@ip-172-31-31-146 sisop1]$ ./nomor6.sh
Masukkan angka pertama : 2
Masukkan angka kedua : 2
Masukkan operasi yang diinginkan (t/k/b/x) : x
Hasil Akhir : 4
[ec2-user@ip-172-31-31-146 sisop1]$ ./nomor6.sh
Masukkan angka pertama : 16
Masukkan angka kedua : 10
Masukkan operasi yang diinginkan (t/k/b/x) : k
Hasil Akhir : 6
[ec2-user@ip-172-31-31-146 sisop1]$
```



7. Buatlah program shell script yang akan menghasilkan keluaran bilangan secara random positif (perintah \$RANDOM) dalam sekian baris keluaran (silakan ditentukan sendiri di dalam program), dan tambahkan keluaran terakhir adalah bilangan negatif (misal -1).

**Jawab :**

**Code:**

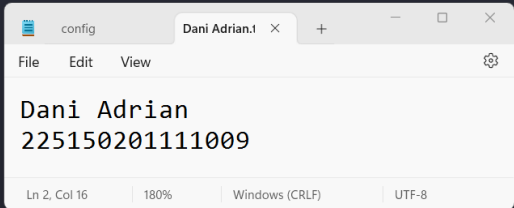
```
#!/bin/bash
n=0
for ((i=1; i<=5; i++)); do
    n=$RANDOM
    echo $n
done
total=$((0-n))
echo $total
```

**Screenshot:**



**LABORATORIUM PEMBELAJARAN ILMU KOMPUTER**  
**FAKULTAS ILMU KOMPUTER**  
**UNIVERSITAS BRAWIJAYA**

```
[ec2-user@ip-172-31-31-146 sisop1]$ chmod +x nomor7.sh
[ec2-user@ip-172-31-31-146 sisop1]$ ./nomor7.sh
24830
31980
5595
25636
9558
-9558
[ec2-user@ip-172-31-31-146 sisop1]$
```



8. Kemudian lakukan dengan “pipe” dimana keluaran program ini (point g) menjadi masukan bagi program di point e. Tampilkan hasilnya.

**Jawab :**

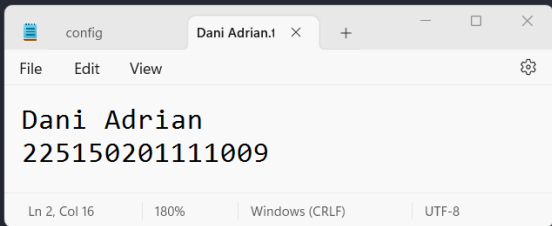
**Code:**

```
#!/bin/bash
```

```
./nomor7.sh | ./nomor5.sh
```

**Screenshot:**

```
[ec2-user@ip-172-31-31-146 sisop1]$ chmod +x nomor7.sh
[ec2-user@ip-172-31-31-146 sisop1]$ ./nomor7.sh
9847
30761
2350
16236
277
-277
[ec2-user@ip-172-31-31-146 sisop1]$
```



**Kesimpulan**

Quoting adalah suatu teknik dalam shell scripting yang digunakan untuk melindungi karakter-karakter khusus dari shell. Ada tiga jenis quoting yang sering digunakan, yaitu single quote (' '), double quote (" "), dan backslash (\).





Setiap jenis quoting memiliki aturan dan cara kerja yang berbeda-beda.

- Single quote ( ' ' ): karakter yang ada di dalam tanda petik satu akan dianggap sebagai string murni, sehingga semua karakter di dalamnya akan dilindungi dari shell. Ini berarti bahwa karakter-karakter khusus seperti tanda dolar (\$), backtick (`), atau backslash (\) tidak akan diperlakukan sebagai karakter khusus dan akan dianggap sebagai karakter biasa
- Double quote ( " " ): karakter yang ada di dalam tanda petik dua juga akan dianggap sebagai string murni, namun beberapa karakter tertentu seperti tanda dolar (\$), backtick (`), dan backslash (\) tetap akan diperlakukan sebagai karakter khusus. Karakter-karakter khusus ini dapat digunakan untuk melakukan substitusi variabel atau eksekusi perintah di dalam string.
- Backslash (\): jika karakter backslash diikuti dengan karakter tertentu, maka karakter tersebut akan dilindungi dari shell. Misalnya, jika kita ingin mencetak tanda kutip dua di dalam tanda kutip dua, kita dapat menggunakan perintah "echo "Hello World"" yang akan mencetak "Hello World" dengan tanda kutip dua.

For loop adalah salah satu jenis perulangan yang digunakan dalam shell scripting untuk melakukan iterasi terhadap satu atau beberapa item dalam daftar. For loop dapat digunakan untuk berbagai tipe data, seperti string, angka, atau nama file. For loop bekerja dengan cara mengulang perintah atau aksi untuk setiap item dalam daftar.

While loop adalah jenis perulangan lain yang digunakan dalam shell scripting untuk melakukan perulangan sampai kondisi tertentu terpenuhi. While loop dapat digunakan untuk berbagai tipe data, seperti string, angka, atau nama file. While loop bekerja dengan cara mengulang perintah atau aksi selama kondisi yang ditentukan benar.

Perintah read digunakan untuk membaca input dari pengguna dan menyimpannya ke dalam sebuah variabel. Input yang dibaca dapat berupa teks atau angka. Perintah read sangat berguna untuk membuat script yang interaktif dan dapat berinteraksi dengan pengguna.

Variabel \$RANDOM adalah sebuah variabel bawaan shell yang digunakan untuk menghasilkan bilangan acak. Variabel ini menghasilkan bilangan acak dalam rentang 0 hingga 32767 setiap kali dipanggil. Variabel \$RANDOM sangat berguna dalam membuat script yang membutuhkan bilangan acak.

Pipe (|) adalah suatu teknik dalam shell scripting yang digunakan untuk menghubungkan keluaran dari satu perintah dengan masukan dari perintah lain. Pipe digunakan untuk mengalirkan data dari satu perintah ke perintah lain,



**LABORATORIUM PEMBELAJARAN ILMU KOMPUTER**  
**FAKULTAS ILMU KOMPUTER**  
**UNIVERSITAS BRAWIJAYA**

---

sehingga perintah-perintah tersebut dapat dijalankan secara bersamaan dan efisien.