

Pemrograman Shell (1)

Praktikum 7
Sistem Operasi

SHELL SCRIPT

- Shell script dibuat dengan editor teks (ASCII editor) dan umumnya diberikan ekstensi “.sh”.
- Script selalu diawali dengan komentar, yang dimulai dengan tanda #, disambung dengan ! dan nama shell yang digunakan.

```
#!/bin/sh      ①  
# Program shell ②  
#  
var1=x         ③  
var2=8
```

Variabel

- Variable shell adalah variable yang dapat mempunyai nilai berupa nilai String.
- Tata penulisan :

`nama_var = nilai_var`

- Variable harus dimulai dengan alfabet, disusul dengan alfanumerik dan karakter lain.
- Variabel dapat ditulis dalam huruf kecil atau huruf besar atau campuran keduanya.

- Shell membedakan huruf besar dan huruf kecil (*case sensitive*), contoh :

VPT=poltek

i=5

- Pemberian nilai variable tidak boleh dipisahkan dengan spasi, karena shell akan menganggap pemisahan tersebut sebagai parameter, contoh :

VPT =poltek ##error

VPT= poltek ##error

VPT=poltek

- Jika string yang terdiri dari lebih dari satu kata, maka harus berada dalam tanda kutip atau apostrof, contoh :

VPT=poltek

VPT2="poltek elektronika ITS"

Echo

- Digunakan untuk mengoutputkan isi dari variabel
- Biasanya diikuti tanda \$
- Contoh :

```
echo $VPT
```

```
Gaji=450000
```

```
echo $Gaji
```

```
echo $VPT $Gaji
```

If

- Format :

```
if
    condition
then
    one-or more commands
fi
```

- Contoh : file myremove.sh

```
#!/bin/sh
if [ !-f $1 ]
then
    echo $0: No file named $1
fi
if [ -f $1 ]
then
    rm $1
    echo Removed file: $1
fi
```

- Dijalankan :
\$ chmod +x myremove.sh
\$./myremove file.txt
Removed file: file.txt
\$./myremove file.txt
./myremove: No file named file.txt

Test Kondisi

- Ada dua cara mengetes kondisi di linux
 - test condition
 - [condition]
- Kondisi dapat direvese dengan memakai tanda : !
- Ada beberapa jenis tes kondisi yang dikenal, yaitu :
 - test file
 - test numeric
 - Test string
 - Test multiple condition

Test File dan Test Numerik

Test File Attributes

- `[-r file1]`
 - Is *file1* readable?
- `[-w file1]`
 - Is *file1* is writeable?
- `[-x file1]`
 - Is a *file1* is executable?
- `[-f file1]`
 - Does *file1* exist?

Testing Numeric Values

- Use: `-eq`, `-ne`, `-gt`, `-ge`, `-lt`, or `-le`
- Examples
 - `[$1 -lt $2]`
 - Is *\$1* less than *\$2*?
 - `[$1 -gt 0]`
 - Is *\$1* greater than 0?

```
if test -w "$1"
then
echo "file $1 is write-able"
fi
```

Test String

- Variabel shell yang hendak diteset diletakkan dalam tanda “ ” atau disebut double quotes
[“\$1” = “yes”]
- Contoh :
 - [\$1 != “yes”] menjadi [!= “no”]
 - jika \$1 kosong, maka akan menimbulkan syntax error (karena tidak memakai “ ”)

Testing multi kondisi

- Operator :
 - && untuk and operator
 - || untuk or operator
- Contoh :
 - ["\$1" = "yes"] && [-r \$2.txt]
 - ["\$1" = "no"] || ["\$2" = "maybe"]

bash control structures

- if-then-else
- case
- loops
 - for
 - while
 - until
 - select

If ..elif .. else

- General form

```
if
  condition
then
  commands
elif
  condition
then
  commands
...
else
  commands
fi
```

myrm2

```
#!/bin/sh
if [ ! -f $1 ]
then
  echo $0: No file named $1
elif [ -f $1 ]
then
  rm $1
  echo Removed file: $1
fi
```

myrm3

```
#!/bin/sh
if [ -f $1 ]
then
  rm $1
  echo Removed file: $1
else
  echo $0: No file named $1
fi
```

For

- Format :

```
for
    variable [ in word_list ]
do
    commands
done
```

- The commands are executed several times
- Each time, the variable is assigned a different word in the word_list
- If in word_list is omitted, then variable is assigned each of the command line arguments

backup

```
#!/bin/sh
for filename
do
    cp $filename $filename.bak
done
```

backup2

```
#!/bin/sh
list="myrm2 tests.sh"
for filename in $list
do
    cp $filename $filename.bak
done
```

- Saat dijalankan output kedua file:

```
$ls  
myrm2  
tests.sh  
$./backup tests.sh myrm2  
$ls  
myrm2  
myrm2.bak  
tests.sh  
tests.sh.bak
```

Exit

- Perintah exit bertugas untuk menghentikan eksekusi script shell
- Exit dapat digunakan untuk mengeset status saat melakukan exit
- Format :
 - exit atau exit status
- Nilai nol atau zero saat exit menunjukkan sukses
- Nilai bukan nol atau non zero menunjukkan failure/error

Contoh

myrm5

```
#!/bin/sh
if [ -f $1 ]
then
    rm $1
    exit 0
else
    exit 1
fi
```

rmfiles

```
#!/bin/sh
for filename
do
    if ./myrm5 $filename
    then
        echo Removed file: $filename
    else
        echo Unable to remove
        $filename
    fi
done
```

```
$ ./rmfiles asj ddaas
Unable to remove asj
Unable to remove ddaas
$ ./rmfiles *.bak
Removed file: calc.bak
Removed file: myrm.bak
```

END OF LECTURE 7
.... ANY QUESTION ?

