

10. Variables - Part II

Each time you run a script, the values of the variables are reset. This means that you can't rely on a variable's value from a previous run. You can, however, use the `set` command to set the values of the variables in a script.

The `set` command can be used to set the values of the variables in a script. The syntax is as follows:

```
set [options] [value1] [value2] ...
```

The options are:

- `-n`: No alias expansion.
- `-e`: Enable error checking.
- `-f`: No filename expansion.
- `-x`: Print each command before it is executed.
- `-v`: Print the value of each variable.
- `-o`: Set the options.

The values are the values of the variables. For example, to set the values of the variables `name`, `first`, and `second` to `File with spaces`, `File`, and `with spaces`, you would use the following command:

```
set name="File with spaces" first="File" second="with spaces"
```

The `set` command can also be used to set the values of the variables in a script. For example, to set the values of the variables `name`, `first`, and `second` to `File with spaces`, `File`, and `with spaces`, you would use the following command:

```
set name="File with spaces" first="File" second="with spaces"
```

```
#!/bin/sh
echo "I was called with $# parameters"
echo "My name is $0"
echo "My first parameter is $1"
echo "My second parameter is $2"
echo "All parameters are @$@"
```

The `set` command can be used to set the values of the variables in a script. The syntax is as follows:

```
$ /home/steve/var3.sh
I was called with 0 parameters
My name is /home/steve/var3.sh
My first parameter is
My second parameter is
All parameters are
$
$ ./var3.sh hello world earth
I was called with 3 parameters
My name is ./var3.sh
My first parameter is hello
My second parameter is world
All parameters are hello world earth
```

The `set` command can be used to set the values of the variables in a script. The syntax is as follows:

```
set [options] [value1] [value2] ...
```

The options are:

- `-n`: No alias expansion.
- `-e`: Enable error checking.
- `-f`: No filename expansion.
- `-x`: Print each command before it is executed.
- `-v`: Print the value of each variable.
- `-o`: Set the options.

The values are the values of the variables. For example, to set the values of the variables `name`, `first`, and `second` to `File with spaces`, `File`, and `with spaces`, you would use the following command:

```
set name="File with spaces" first="File" second="with spaces"
```

```
echo "My name is `basename $0`"
```

\$# 0 1 ... 2 3 4 5 6 7 8 9 10
shift 1 2 3 4 5 6 7 8 9 10

```
#!/bin/sh
while [ "$#" -gt "0" ]
do
    echo "\$1 is $1"
    shift
done
```

0 1 2 3 4 5 6 7 8 9 10 shift 1 2 3 4 5 6 7 8 9 10

\$? 0 1 2 3 4 5 6 7 8 9 10 . 1 2 3 4 5 6 7 8 9 10 :

```
#!/bin/sh
/usr/local/bin/my-command
if [ "$?" -ne "0" ]; then
    echo "Sorry, we had a problem there!"
fi
```

0 1 2 3 4 5 6 7 8 9 10 /usr/local/bin/my-command
\$? 0 1 2 3 4 5 6 7 8 9 10
. 1 2 3 4 5 6 7 8 9 10
: 1 2 3 4 5 6 7 8 9 10

“" 0 1 2 3 4 5 6 7 8 9 10 C 1 2 3 4 5 6 7 8 9 10
" 1 2 3 4 5 6 7 8 9 10 ."
(Robert Firth)

\$\$ \$! 0 1 2 3 4 5 6 7 8 9 10
PID(0 1 2 3 4 5 6 7 8 9 10) 0 1 2 3 4 5 6 7 8 9 10
/tmp/my-script.\$\$ 0 1 2 3 4 5 6 7 8 9 10
! 0 1 2 3 4 5 6 7 8 9 10 PID 0 1 2 3 4 5 6 7 8 9 10
0 1 2 3 4 5 6 7 8 9 10

IFS 0 1 2 3 4 5 6 7 8 9 10 . SPACE TAB NEWLINE
0 1 2 3 4 5 6 7 8 9 10 :

```
#!/bin/sh
old_IFS="$IFS"
IFS=:
```

```
echo "Please input some data separated by colons ..."  
read x y z  
IFS=$old_IFS  
echo "x is $x y is $y z is $z"
```

❏ ❏❏❏❏ ❏❏ ❏❏ ❏❏❏❏ :

```
$ ./ifs.sh  
Please input some data separated by colons ...  
hello:how are you:today  
x is hello y is how are you z is today
```

❏❏ ❏❏ "[hello:how are you:today:my:friend"❏❏ ❏❏❏❏ ❏❏❏ ❏❏ ❏❏❏❏ :

```
$ ./ifs.sh  
Please input some data separated by colons ...  
hello:how are you:today:my:friend  
x is hello y is how are you z is today:my:friend
```

❏❏ IFS❏ ❏❏ ❏❏ ❏❏ ,❏❏ ❏ ❏❏ "❏❏❏ ❏ ❏❏ "❏❏❏ ❏❏❏ ❏ ❏❏❏ ❏❏ ❏❏❏❏ ❏❏
❏❏❏❏ .❏❏❏ ❏❏❏❏❏ ❏❏ ❏❏ ❏❏❏❏ (❏❏ : old_IFS=\$IFS ❏❏ old_IFS="\$IFS").

Revision #2

Created 16 January 2024 22:15:31 by Enigma

Updated 16 January 2024 22:25:04 by Enigma