

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 the values of variables from a previous run. You have to make sure that your script works correctly every time it is run.

There are several special variables in Bash. The first one is `$0`, which is the name of the script. The second one is `$#`, which is the number of arguments. The third one is `$1`, which is the first argument. The fourth one is `$2`, which is the second argument. The fifth one is `$3`, which is the third argument. The sixth one is `$4`, which is the fourth argument. The seventh one is `$5`, which is the fifth argument. The eighth one is `$6`, which is the sixth argument. The ninth one is `$7`, which is the seventh argument. The tenth one is `$8`, which is the eighth argument. The eleventh one is `$9`, which is the ninth argument. The twelfth one is `$*`, which is all the arguments. The thirteenth one is `$@`, which is all the arguments. The fourteenth one is `$@`, which is all the arguments. The fifteenth one is `$@`, which is all the arguments. The sixteenth one is `$@`, which is all the arguments. The seventeenth one is `$@`, which is all the arguments. The eighteenth one is `$@`, which is all the arguments. The nineteenth one is `$@`, which is all the arguments. The twentieth one is `$@`, which is all the arguments.

```
#!/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 $@"
```

Now let's create a script called `var3.sh` in the directory `/home/steve`:

```
$ /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
```

Now let's create a script called `var4.sh` in the directory `/home/steve` with the following content:

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

\$# 0 1 ... 2 参数个数 参数列表 . 参数列表 shift 参数列表 9 参数列表
参数列表 参数列表 参数列表 :

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

参数列表 \$# 0 参数列表 , 参数列表 参数列表 参数列表 shift 参数列表 .

参数列表 参数列表 \$? 参数列表 参数列表 参数列表 参数列表 参数列表 . 参数列表 参数列表 :

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

参数列表 参数列表 参数列表 0 参数列表 , 参数列表 0 参数列表 参数列表 /usr/local/bin/my-command 参数列表 参数列表 . 参数列表 参数列表 参数列表 \$? 参数列表 参数列表 参数列表 参数列表 参数列表 . 参数列表 参数列表 参数列表 参数列表 参数列表 参数列表 参数列表 参数列表 . 参数列表 参数列表 参数列表 参数列表 0 参数列表 参数列表 . 参数列表 参数列表 参数列表 参数列表 :

“参数列表 参数列表 参数列表 参数列表 参数列表 0 参数列表 参数列表 C 参数列表 参数列表
参数列表 参数列表 参数列表 参数列表 参数列表 .”
(Robert Firth)

参数列表 参数列表 参数列表 参数列表 参数列表 参数列表 \$\$ 参数列表 \$! 参数列表 参数列表 参数列表 . 参数列表 \$\$ 参数列表 参数列表
参数列表 PID(参数列表 参数列表) 参数列表 . 参数列表 参数列表 参数列表 参数列表 参数列表 参数列表 参数列表
参数列表 参数列表 参数列表 参数列表 参数列表 参数列表 参数列表 /tmp/my-script.\$\$ 参数列表 参数列表 参数列表 参数列表
参数列表 参数列表 . ! 参数列表 参数列表 参数列表 参数列表 参数列表 PID 参数列表 . 参数列表 参数列表 参数列表
参数列表 参数列表 参数列表 参数列表 .

参数列表 参数列表 参数列表 IFS 参数列表 . 参数列表 参数列表 参数列表 参数列表 . 参数列表 SPACE TAB NEWLINE
参数列表 参数列表 参数列表 参数列表 参数列表 参数列表 参数列表 :

```
#!/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"
```

❏ ❶❷❸❹ ❺❻ ❹ ❶❷❸❹ :

```
$ ./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
```

❶❷ ❹ "[hello:how are you:today:my:friend]"❶❷ ❶❷❸ ❹❺ ❶❷❸❹ :

```
$ ./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
```

❶❷ IFS❶❷ ❶❷❸❹ ,❶❷ ❶❷❸❹ "❶❷❸❹ ❶❷❸❹ "❶❷❸❹ ❶❷❸❹ ❶❷❸❹ ❶❷❸❹ ❶❷❸❹
❶❷❸❹ .❶❷❸❹ ❶❷❸❹ ❶❷❸❹ (❶❷ : old_IFS=\$IFS ❶❷❸❹ old_IFS="\$IFS").

Revision #1

Created 17 January 2024 04:15:01 by ❶❷❸❹ (MeatDumpling)

Updated 17 January 2024 04:15:14 by ❶❷❸❹ (MeatDumpling)