

- 1- variables
- 2- echo
- 3- test
- 4- read
- 5- grep

-----  
How to write shell script  
-----

Shell script examples  
-----

```
login user name hany      home directory /home/hany    bash
login user name ali       home directory /home/ali      ksh
system variables HOME     SHELL LOGNAME .....
user can define his own variables
```

-list-create-remove variables

```
var=value      create
x=10
fname=hany
```

```
echo $var      list
echo $HOME
echo $fname
```

```
unset var      to remove value from var
unset fname
```

```
root@RHServer1 ~# echo $LOGNAME
root
root@RHServer1 ~# echo $SHELL
/bin/bash
root@RHServer1 ~#
root@RHServer1 ~# su - hany
hany@RHServer1 ~$ echo $LOGNAME
hany
hany@RHServer1 ~$ echo $HOME
/home/hany
hany@RHServer1 ~$ echo $SHELL
/bin/bash
hany@RHServer1 ~$ exit
logout
root@RHServer1 ~# echo $LOGNAME
root
root@RHServer1 ~# echo $HOME
/root
root@RHServer1 ~# fname=hany
root@RHServer1 ~# echo $fname
fname
root@RHServer1 ~# echo $fname
hany
root@RHServer1 ~# lname=cairo
root@RHServer1 ~# echo welcome $fname $lname in scity
welcome hany hany in cairo
root@RHServer1 ~# x=10
root@RHServer1 ~# echo $x
10
root@RHServer1 ~# echo $x
10
root@RHServer1 ~# unset x
root@RHServer1 ~# echo $x
root@RHServer1 ~#
```

```
echo text
echo $variable
echo `command`
echo $?      <--- exit status -->
```

```
[root@RHServer1 ~]# date
Thu Feb 11 17:40:31 EST 2021
[root@RHServer1 ~]#
[root@RHServer1 ~]#
[root@RHServer1 ~]# echo date
date
[root@RHServer1 ~]# echo `date`
Thu Feb 11 17:40:45 EST 2021
```

```
root@RHServer1 ~# grep hany /etc/passwd
hany:x:1000:1000:/home/hany:/bin/bash
root@RHServer1 ~#
root@RHServer1 ~# grep /etc/passwd
root@RHServer1 ~# echo $?
0
root@RHServer1 ~#
```

found 11 no errors -> 0  
not found 11 errors -> 1

```
test
test options argument
test -f fileName      if it file
test -d fileName      if it directory
test -r fileName      if has read permissions
test -w fileName      if has write permissions
test -x fileName      if has execute permissions
test -e fileName      if file exists
test $var = value      if var = value
```

```
test $LOGNAME = root
echo $? <- yes 0
no 1
```

```
test -eq      -ne      -gt      -le .....
test $var -eq value
echo $?
```

read interactive scripts

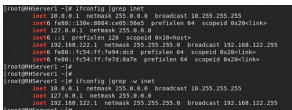
```
#read var
value
var=value
echo $var
```

grep filter search in file content and search in command output

grep word file
command | grep word

| pipe line

- grep -i ignore letters case
- grep -w word Search exactly
- grep -c line count
- grep -n line numbers where word is found
- grep -v reverse search, give me lines, that does not contain word
- grep ^word lines start with
- grep word\$ lines end with



ifconfig |grep -w inet search for word "inet" in command ifconfig output

how to write shell script

```
#!/bin/bash define shell type
# commented lines
chmod u+x add execute permissions
./script to run script
```

```
#!/bin/bash
# this script help you to search for files
echo please enter the file name
read filename
find / -name $filename
```

} Script to Search For specific file

```
#!/bin/bash
# this script lists some system informations
clear
echo " ##### script started ##### "
echo the time is :
date
echo the current login user is :
echo $LOGNAME
echo the machine name is :
hostname
echo the ip address for this machine is :
ifconfig |grep -w inet
echo the current processes is :
ps -f
echo " ##### script end ##### "
echo thanks IT Department
```

} Script to print OS infos

cat /etc/passwd | grep "hany" vs. grep "hany" /etc/passwd  
cat → أخرج أو grep → أخرج  
passwd → اسم الملف أو grep → أخرج

```
#!/bin/bash
echo please enter valid user name
read name
grep -w $name /etc/passwd >/tempfile
if test $? = 0
then
echo not valid the user already exist
else
useradd $name
passwd $name
fi
rm -rf /tempfile
```

```
#!/bin/bash
tar cvf /reports.backup.`date +%d-%m` /report*
```