

LINUX EXERCISE 5: LINUX SHELL BASICS

Prepared by Jack G. Zheng for IT 4423

This exercise will let your practice the bash shell basics, so you will feel comfortable with the shell environment.

EXERCISES

1. When you use the redirect output symbol (>) with a command, the shell creates the output file even the command fails. Demonstrate that.

2. Try the following and see what happens. Explains the behavior.

- a) `echo ".""."`
- b) `echo "hello" "world"`
- c) `echo hello "world"`
- d) `echo "hello!1";` (how to correct this?)

3. Assume you have made the following variable assignment:

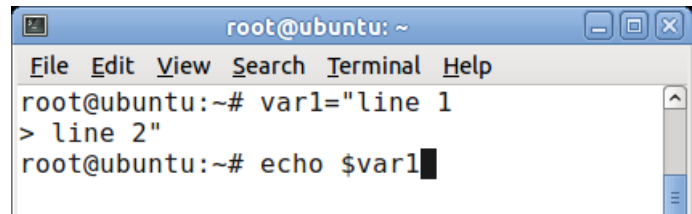
```
person=zach
```

How can you produce the following outputs:

- a) zach
- b) \$person
- c) "zach"
- d) "\$person"

4. Set var1 to the value as shown in the right figure. What's the difference between the following two statements? How many arguments does each echo command see?

- `echo "$var1"`
- `echo $var1`



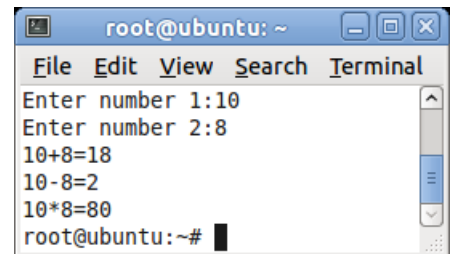
```
root@ubuntu: ~  
File Edit View Search Terminal Help  
root@ubuntu:~# var1="line 1  
> line 2"  
root@ubuntu:~# echo $var1
```

5. Perform the following task

- a) Given any directory, find out all directories in it (including the hidden ones, or not).
- b) Given any directory, find out the number of all files (including directories, and all hidden items) in it
- c) Given any directory, find out the number of all hidden items (files and directories) in it
- d) There are too many packages installed. Search packages related to "firefox", and save the result to a file.
- e) Find out the first 20 commands executed and recorded in history.
- f) Find all commands in command history that starting with "ls", and save the result to a file.

6. Practice a simple command line calculator. See the sample output on the right.

- Use the read command twice to prompt users to enter two integer numbers. Save them to two variables.
- Display the sum, difference, and product of these two numbers. - Hint: use `$()` to force numeric calculation; example `$(($num1+$num2))`
- Try to put all statements in a single command line.



```
root@ubuntu: ~  
File Edit View Search Terminal  
Enter number 1:10  
Enter number 2:8  
10+8=18  
10-8=2  
10*8=80  
root@ubuntu:~#
```

Answers:

1. After executing the following commands, file1 will be created even though there is not output to file1

```
ls nonexistent > file1  
bad >file1 2>file2
```

2. d) echo 'hello!1'

3.

- a) echo "\$person"
- b) echo '\$person'
- c) echo ""\$person""
- d) echo "\$person"

5

a)

```
ls -l | grep "^d.*" -o  
file * | grep directory
```

(including hidden directories)

```
ls -la | grep "^d.*"  
ls -aF | grep ".*/$"
```

(Just for your reference)

```
file * | grep directory | sed 's/././b'  
find * .* -maxdepth 0 -type d
```

b)

```
ls -a1 | wc -l
```

c)

```
ls .* -d | wc -l
```

d)

```
dpkg -l | grep firefox > result
```

e)

```
history | head -10
```

f)

```
history | cut -c8- | grep ^ls > file  
or  
grep ^ls .bash_history
```

6.

```
read -p "Enter number 1:" num1;read -p "Enter number 2:" num2;echo $num1+$num2=${[$num1+$num2]}; echo $num1-  
$num2=${[$num1-$num2]};echo $num1*$num2=${[$num1*$num2]}
```