

Exercises

1. Prepare the file phonebook with tab delimited fields.

```
$ cat phonebook
```

Alice Chebba	973-555-2015
Barbara Swingle	201-555-9257
Jeff Goldberg	201-555-3378
Liz Stachiw	212-555-2298
Susan Goldberg	201-555-7776
Tony Iannino	973-555-1295

2. Using cut to get first 15 characters:

```
$ cut -c1-15 phonebook
```

Alice Chebba	97
Barbara Swingle	
Jeff Goldberg	2
Liz Stachiw	212
Susan Goldberg	
Tony Iannino	97

3. Use the -f option to cut

```
$ cut -f1 phonebook
```

Alice Chebba
Barbara Swingle
Jeff Goldberg
Liz Stachiw
Susan Goldberg
Tony Iannino

4. How do you know whether fields are delimited by blanks or tabs?

```
$ sed -n l phonebook
```

Alice Chebba\t973-555-2015
Barbara Swingle\t201-555-9257
Jeff Goldberg\t201-555-3378
Liz Stachiw\t212-555-2298
Susan Goldberg\t201-555-7776
Tony Iannino\t973-555-1295

5. Create a file called names having a set of names:

```
$ cat names
```

Tony
Emanuel
Lucy
Ralph
Fred

6. And another file called numbers:

```
$ cat numbers
```

(307) 555-5356

(212) 555-3456
(212) 555-9959
(212) 555-7741
(212) 555-0040

7. Now, use paste to print the names and numbers side-by-side:

\$ paste names numbers

Tony	(307) 555-5356
Emanuel	(212) 555-3456
Lucy	(212) 555-9959
Ralph	(212) 555-7741
Fred	(212) 555-0040

8. Create yet another file called addresses:

\$ cat addresses

55-23 Vine Street, Miami
39 University Place, New York
17 E. 25th Street, New York
38 Chauncey St., Bensonhurst
17 E. 25th Street, New York

9. Then issue the command paste as follows and see what happens:

\$ paste names addresses numbers

Tony	55-23 Vine Street, Miami	(307) 555-5356
Emanuel	39 University Place, New York	(212) 555-3456
Lucy	17 E. 25th Street, New York	(212) 555-9959
Ralph	38 Chauncey St., Bensonhurst	(212) 555-7741
Fred	17 E. 25th Street, New York	(212) 555-0040

10. Try other delimiters than the tab, e.g. a plus:

\$ paste -d '+' names addresses numbers

Tony+55-23 Vine Street, Miami+(307) 555-5356
Emanuel+39 University Place, New York+(212) 555-3456
Lucy+17 E. 25th Street, New York+(212) 555-9959
Ralph+38 Chauncey St., Bensonhurst+(212) 555-7741
Fred+17 E. 25th Street, New York+(212) 555-0040

11. Try the following tr commands in a file of your choice:

tr 'X' 'x'	Translate all capital X's to small x's
tr '()' '{}'	Translate all open parens to open braces, all closed parens to closed braces
tr '[a-z]' '[A-Z]'	Translate all lowercase letters to uppercase
tr '[A-Z]' '[N-ZA-M]'	Translate uppercase letters A–M to N–Z, and N–Z to A–M, respectively
tr '\11' ' '	Translate all tabs (character in first pair of quotes) to spaces
tr -s ' ' ' '	Translate multiple spaces to single spaces
tr -d '\12'	Delete all newline (octal 14) characters
tr -d '[0-9]'	Delete all digits

12. Sort the file called names in as ascending order :

\$ sort names

Charlie
Emanuel
Fred
Lucy
Ralph
Tony
Tony

13. Remove duplicates :

```
$ sort -u names
```

Charlie
Emanuel
Fred
Lucy
Ralph
Tony

14. Reverse sort :

```
$ sort -r names
```

Tony
Tony
Ralph
Lucy
Fred
Emanuel
Charlie

15. Specify output filename :

```
$ sort names -o sorted_names
```

```
$ cat sorted_names
```

```
$ sort names -o names
```

```
$ paste names sorted_names
```

16. Create a file called data :

```
$ cat data
```

5 27
2 12
3 33
23 2
-5 11
15 6
14 -9

17. Try the normal sort :

```
$ sort data
```

-5 11
14 -9
15 6
2 12

```
23 2
3 33
5 27
```

18. And, try sort with -n option :

```
$ sort -n data
```

```
-5 11
2 12
3 33
5 27
14 -9
15 6
23 2
```

19. Sorting in the second field :

```
$ sort +1n data
```

```
14 -9
23 2
15 6
-5 11
2 12
5 27
3 33
```

20. Look at our sample password file again:

```
$ cat /etc/passwd
```

```
root:*:0:0:The super User:/:usr/bin/ksh
steve:*:203:100::/users/steve:/usr/bin/ksh
bin:*:3:3:The owner of system files:/:
cron:*:1:1:Cron Daemon for periodic tasks:/:
george:*:75:75::/users/george:/usr/lib/rsh
pat:*:300:300::/users/pat:/usr/bin/ksh
uucp:*:5:5::/usr/spool/uucppublic:/usr/lib/uucp/uucicoasg:*:6:6:The Owner of Assignable Devices:/:
sysinfo:*:10:10:Access to System Information:/usr/bin/sh
mail:*:301:301::/usr/mail:
```

21. And sort it :

```
$ sort /etc/passwd
```

22. Now try with -t option :

```
$ sort +2n -t: /etc/passwd # Sort by user id
```

```
root:*:0:0:The Super User:/:usr/bin/ksh
cron:*:1:1:Cron Daemon for periodic tasks:/:
bin:*:3:3:The owner of system files:/:
uucp:*:5:5::/usr/spool/uucppublic:/usr/lib/uucp/uucico
asg:*:6:6:The Owner of Assignable Devices:/:
sysinfo:*:10:10:Access to System Information:/usr/bin/sh
george:*:75:75::/users/george:/usr/lib/rsh
steve:*:203:100::/users/steve:/usr/bin/ksh
```

```
pat:*:300:300::/users/pat:/usr/bin/ksh
mail:*:301:301::/usr/mail:
```

23. Use of single quote :

```
$ grep 'Susan Goldberg' phonebook
$ grep Susan Goldberg phonebook
$ echo one      two      three      four
one two three four
$ echo 'one      two      three      four'
one      two      three      four
$ echo *
$ echo '*'
$ echo 'How are you today,
> John'
$ message='I must say, this sure is fun'
$ echo $message
```

24. Using double quotes :

```
$ x=*
$ echo $x
$ echo '$x'
$ echo "$x"
$ address="39 East 12th Street
> New York, N. Y. 10003"
$ echo $address
$ echo "$address"
$ x="" Hello,' he said"
$ echo $x
$ article=' "Keeping the Logins from Lagging," Bell Labs Record'
$ echo $article
```

25. Use of backslash (escape character)

```
$ echo >
$ echo \>
$ x=*
$ echo \$x
$ echo \\
$ echo \'
$ lines=one'
> 'two      #Single quotes tell shell to ignore newline
$ echo "$lines"
$ lines=one\      #Try it with a \ instead
> two
$ echo "$lines"
$ echo "\\$x"
$ echo "\ is the backslash character"
$ x=5
$ echo "The value of x is \" $x\""
```

25. Displaying <<< ***echo \$x >>>*** ***displays the value of x, which is \$x*** on the output

```
$ x=1          # assign value to x
$ echo <<< echo $x >>> displays the value of x, which is $x
syntax error: '<' unexpected
$ echo "<<< echo \ $x >>> displays the value of x, which is $x"
<<< echo $x >>> displays the value of x, which is 1
$ echo '<<< echo $x >>> displays the value of x, which is' $x
<<< echo $x >>> displays the value of x, which is 1
$ echo '<<< echo $x >>> displays the value of x, which is' "$x"
```

26. Using backquote and \$(...) construct

```
$ echo The date and time is: `date`
$ echo Your current working directory is `pwd`
$ echo The date and time is: $(date)
$ echo There are $(who | wc -l) users logged in
$ echo '$(who | wc -l) tells how many users are logged in'
$ echo "You have $(ls | wc -l) files in your directory"
$ now=$(date)
$ echo $now
$ filelist=$(ls)
$ echo $filelist
$ echo "$filelist"
$ name="Ralph Kramden"
$ name=$(echo $name | tr '[a-z]' '[A-Z]')      #Translate to uppercase
$ echo $name
```

27. Using the expr command:

```
$ expr 1 + 2
$ expr 1+2      # without spaces
$ expr 10 + 20 / 2
$ expr 17 * 6    # syntax error
$ expr "17 * 6"
$ expr 17 \* 6
$ i=1
$ expr $i + 1
$ i=$(expr $i + 1)
$ echo $i
$ i=`expr $i + 1`
$ echo $i
```

Programs using Arguments and shifts :

29. Using only one argument

```
$ cat ison
who | grep $1
$ ison tony
$ ison pat
```

30. Use of \$#

```
$ cat args
```

```
echo $# arguments passed
echo arg 1 = :$1: arg 2 = :$2: arg 3 = :$3:
$ args a b c
$ args a b
$ args
$ args "a b c"
$ args x*
$ args $my_bin
$ args $(cat names)
```

```
31. Using $*
$ cat args2
echo $# arguments passed
echo they are :$*:
$ args2 a b c
$ args2 one two # multiple spaces are allowed in between arguments
$ args2
$ args2 *
```

```
32. Look-up someone in the phonebook
$ cat lu
grep $1 phonebook
$ lu Alice
$ lu Susan
$ lu "Susan T"
```

```
$ cat lu2 # another version of the program
grep "$1" phonebook
$ lu2 Tony
$ lu2 "Susan T"
```

```
33. Add Someone to the Phone Book
$ cat add
echo "$1 $2" >> phonebook
$ add 'Stromboli Pizza' 973-555-9478
$ lu Pizza
$ cat phonebook
```

```
$ cat add2 # add -- version 2
echo "$1 $2" >> phonebook
sort -o phonebook phonebook
$ add 'Billy Bach' 201-555-7618
$ cat phonebook
```

```
34. Remove Someone from the Phone Book
$ cat rem
grep -v "$1" phonebook > /tmp/phonebook
mv /tmp/phonebook phonebook
$ rem 'Stromboli Pizza'
```

```
$ cat phonebook
$ rem Susan
$ cat phonebook    # Both Susan were removed
$ add 'Susan Goldberg' 201-555-7776
$ add 'Susan Topple' 212-555-4932
```

35. Testing shift command

```
$ cat tshift
echo $# $*
shift
echo $# $*
shift
echo $# $*
shift
echo $# $*
shift
echo $# $*
shift
echo $# $*
$ tshift a b c d e
```