

COMMAND LINE INTERFACE - EXERCISE

Klebsiella Workshop

Sep 2024



South African National
Bioinformatics Institute



UNIVERSITY of the
WESTERN CAPE



AFRICA CDC
Centres for Disease Control and Prevention
Safeguarding Africa's Health



PUBLIC HEALTH ALLIANCE FOR
GENOMIC EPIDEMIOLOGY

African
Union



YOU ARE HERESYSTEM
COMMANDS

<i>Command</i>	<i>Function</i>	<i>Syntax/example usage</i>
mkdir	make directory	mkdir DIRECTORY
pwd	print working directory	pwd
cd	change directory	cd ~ or cd #home directory
		cd .. #previous (parent directory)
ls	list contents	ls [OPTIONS] DIRECTORY
nano/ vim/ touch	Create/ edit a file	nano/ vim/ touch FILE

CREATE FILES

CREATE FILES

DIRECTORY
MANAGEMENT

FILTERS

FILES & SYSTEMS

mkdir - Make Directory Command

YOU ARE HERE

- Use the mkdir (make directory) command followed by the **DIRECTORY** name, as shown:

```
1 mkdir unixTutorial
```

- There should be a space between the mkdir command and the directory name.
- It's recommended to avoid spaces in directory names in Unix; use underscores (_) instead.
- After pressing Enter, the command runs silently, returning a new prompt. If you try to create the same directory again, you will receive an error like:

```
2 mkdir unixTutorial
```

```
3 mkdir: cannot create directory 'unixTutorial': File exists
```

SYSTEM
COMMANDS

CREATE FILES

CREATE FILES

DIRECTORY
MANAGEMENT

FILTERS

FILES & SYSTEMS

pwd - Path of Working Directory

YOU ARE HERE

- This command will tell you where you are:

```
1 pwd
```

- The output should look similar to the following

```
2 /home/<USERNAME>
```

- This is telling you that you are in a directory named <USERNAME> which is in a directory named home.

SYSTEM
COMMANDS

CREATE FILES

CREATE FILES

DIRECTORY
MANAGEMENT

FILTERS

FILES & SYSTEMS

cd - Change Directory

YOU ARE HERE

- You can change to that **DIRECTORY** using the **cd** command

```
1 cd unixTutorial
2 pwd
```

- The output should look like the following

```
3 /home/<USERNAME>/unixTutorial
```

- This is telling you that you have successfully moved into the **unixTutorial** directory
- To go back to the parent directory:

```
4 cd /home/<USERNAME>/
```

- OR**

```
4 cd ..
```

SYSTEM
COMMANDS

CREATE FILES

CREATE FILES

DIRECTORY
MANAGEMENT

FILTERS

FILES & SYSTEMS

ls - List Command

YOU ARE HERE

- The contents of a directory can be viewed using `ls`. If no **DIRECTORY** name is provided, then `ls` will list all the contents of the present **DIRECTORY**

```
1 ls
```

- The output should look like the following

```
2 unixTutorial
```

- Or to list contents of a directory

```
3 ls unixTutorial
```

- The output will not return anything as `unixTutorial` is currently empty

SYSTEM
COMMANDS

CREATE FILES

CREATE FILES

DIRECTORY
MANAGEMENT

FILTERS

FILES & SYSTEMS

history - gives a history of recently commands

- This command will replace your find command in a text document.

```
1 history | grep myFirstFile | tail -n 5
2 history | grep -i myfirst | tail -n 5
3 history | grep -i myfirst | grep -v space
```

- You can also recall your previous commands by pressing ↑ or ↓ arrow keys

? and * - represent one or many characters

```
1 mv Numbers/*.txt .
2 mv Letters/*.txt .
3 ls
4
5 ls ?.txt Numbers
6 ls ???.txt Letters
```

YOU ARE HERE

SYSTEM
COMMANDS

CREATE FILES

FILE EDITING

DIRECTORY
MANAGEMENT

DATA
MANIPULATION

FILTERING FILE DATA

nano - Text editor more like a GUI

<i>Command</i>	<i>Function</i>
ctrl+o	save file
ctrl+x	close file
alt+\	go to beginning of the file
alt+/ 	go to end of the file
ctrl+a	go to start of the line
ctrl+e	go to end of the line
ctrl+c	show line number
ctrl+_	go to line number
ctrl+w	find matching word
alt+w	find next match
ctrl+\	find and replace

YOU ARE HERE

SYSTEM
COMMANDS

CREATE FILES

FILE EDITING

DIRECTORY
MANAGEMENT

FILTERS

FILES & SYSTEMS

nano - Text editor

SYSTEM
COMMANDS

CREATE FILES

FILE EDITING

DIRECTORY
MANAGEMENT

FILTERS

FILES & SYSTEMS

YOU ARE HERE

- In a terminal use `cd` to change into the **DIRECTORY** created and `nano` to create a file named `myFirstFile.txt`

```
1 cd unixTutorial
2 nano myFirstFile.txt
```

- Insert the following text into the file:

```
1 Make the Plan
2 Execute the Plan
3 Expect the Plan to go off the rails
4 Throw away the Plan
```

- Use the following commands: `ctrl+o` and `ctrl+x` to save and close the file
- List contents of a **DIRECTORY**
- Long format `-l` parameter modifies `ls` to get more information about the files such size, date of creation and ownership.

```
1 ls
ls -l myFirstFile.txt
```

Viewing file contents

<i>Command</i>	<i>Function</i>	<i>Syntax/example usage</i>
cat	catalog file contents	cat FILE
head	show first few lines of a file	head FILE
tail	show last few lines of a file	tail FILE
more	view file (with less options)	more FILE
less	view file (with more options)	less FILE

YOU ARE HERE

SYSTEM
COMMANDS

CREATE FILES

FILE EDITING

DIRECTORY
MANAGEMENT

FILTERS

FILES & SYSTEMS

Viewing file contents

- Use `nano` to create a **FILE** and populate it with a range of numbers

```
1 seq 1 1 100
2 # This creates a list of numbers from 1 – 100 with an
3 interval of 1
4 # Copy the output to past into the file you will create
5 nano numSeq.txt
6 #Paste the list of numbers into the file
7 #Make sure there's no empty lines at the end of the file
8 ctrl+o
9 ctrl+x
```

YOU ARE HERE

SYSTEM
COMMANDS

CREATE FILES

FILE EDITING

DIRECTORY
MANAGEMENT

FILTERS

FILES & SYSTEMS

cat - concatenate and print files

- Use `cat` to display all the contents of a **FILE**

```
1 cat numSeq.txt
```

- The output will show the contents of the file

head - head of the file

- Use `head` to display the first 10 lines of a **FILE**

```
1 head numSeq.txt
```

- Using the `-n` option to specify the number of lines

```
2 head -n 5 numSeq.txt
```

YOU ARE HERE

SYSTEM
COMMANDS

CREATE FILES

FILE EDITING

DIRECTORY
MANAGEMENT

FILTERS

FILES & SYSTEMS

tail - tail of the file

- Use **tail** to display the last 10 lines of a **FILE**

```
1 tail numSeq.txt
```

- Using the **-n** option to specify the number of lines

```
2 tail -n 5 numSeq.txt
```

YOU ARE HERE

more - go through a file one screen length at a time

```
1 more numSeq.txt
```

less - like more but lets you scroll back

```
2 less numSeq.txt
```

SYSTEM
COMMANDS

CREATE FILES

FILE EDITING

DIRECTORY
MANAGEMENT

FILTERS

FILES & SYSTEMS

Directory Management

<i>Command</i>	<i>Function</i>	<i>Syntax/example usage</i>
rmkdir	remove empty directory	rmkdir DIRECTORY
rm	remove file(s)	rm FILE
mv	move files/directories	mv SOURCE DESTINATION
cp	copy files/directories	cp SOURCE DESTINATION

YOU ARE HERE

SYSTEM
COMMANDS

CREATE FILES

FILE EDITING

DIRECTORY
MANAGEMENT

FILTERS

FILES & SYSTEMS

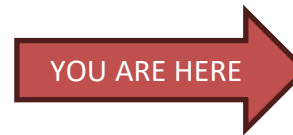
rmmdir - Remove Directory

- Create a list of empty **DIRECTORIES** using the **mkdir** command and then display your current **DIRECTORY**

```
1 mkdir Numbers
2 mkdir Letters
3 mkdir Deleteme
4 mkdir Deleteme2
5 mkdir Deleteme3
6 ls
```

- Remove empty **Deleteme2**

```
1 ls
2 rmmdir Deletme2
3 ls
```



SYSTEM
COMMANDS

CREATE FILES

FILE EDITING

DIRECTORY
MANAGEMENT

FILTERS

FILES & SYSTEMS

rm - Remove FILE

- Create an empty FILE using the touch command within a DIRECTORY

```
1 touch Deleteme/deleteme.txt
2 touch Deleteme3/deleteme3.txt
3 ls Deleteme/deleteme.txt
4 ls Deleteme/deleteme3.txt
```

- Removing a non-empty DIRECTORY will generate a warning

```
1 rmdir Deletme3
2 rmdir: Deleteme3: Directory not empty
```

- Remove the FILE and then delete DIRECTORY

```
1 cd Deleteme3
2 rm deleteme3.txt
3 cd ..
4 rmdir Deletme3
5 ls
```

YOU ARE HERE

SYSTEM
COMMANDS

CREATE FILES

FILE EDITING

DIRECTORY
MANAGEMENT

FILTERS

FILES & SYSTEMS

mv - Move

- mv is used to move FILES to a different DIRECTORY and to rename a FILE.

```
1 touch 1.txt 2.txt 3.txt 4.txt 5.txt 6.txt 7.txt 8.txt
2 ls
3 mv 1.txt Numbers
4 mv 2.txt 3.txt Numbers
5 mv 4.txt 5.txt 6.txt 7.txt 8.txt Numbers
6 ls
```

cp - Copy

- cp is used to duplicate a FILE to a different DIRECTORY and to rename a FILE
- Using -r will allow you to copy an entire DIRECTORY and its contents

```
1 touch A.txt B.txt C.txt D.txt E.txt F.txt G.txt H.txt
2 ls
3 cp A.txt AA.txt Letters
4 #Repeat for each of the previously created files
5 ls Letters
6 cp -r Letters Letters_Copy
7 ls Letters_Copy
```

YOU ARE HERE

SYSTEM
COMMANDS

CREATE FILES

FILE EDITING

DIRECTORY
MANAGEMENT

FILTERS

FILES & SYSTEMS

Counting, Sorting and Redirecting Output

<i>Command</i>	<i>Function</i>	<i>Syntax/example usage</i>
wc	word count	wc FILENAME
cmd1 cmd2	send output of cmd1 to cmd2	cat FILENAME sort uniq
tr	translate or transliterate a file	tr [OPTIONS] " STRING1 " " STRING2 " < INFILE
sort	sort files	sort FILE1 > SORTED_FILE1

YOU ARE HERE

SYSTEM
COMMANDS

CREATE FILES

FILE EDITING

DIRECTORY
MANAGEMENT

DATA
MANIPULATION

FILES & SYSTEMS

wc - word count

- Will give you the number of lines (-l), the number of words (-w) and the number of characters (-c) in a file

```
1 wc numSeq.txt
2 wc -l numSeq.txt
3 wc -w numSeq.txt
4 wc -c numSeq.txt
```

tr - translate

- Take any individual character and replace it with another character
- '<' specified the input file

```
1 cat numSeq.txt
2 tr '\n' ',' < numSeq.txt
3 cat numSeq.txt
```

YOU ARE HERE

SYSTEM
COMMANDS

CREATE FILES

FILE EDITING

DIRECTORY
MANAGEMENT

DATA
MANIPULATION

FILES & SYSTEMS

| - Pipe redirect

- Pipe can take the output of one command and pass it to the next command without having to write to a file in between.

```
1 cat numSeq.txt | tr '\n' ','
```

- Command above first uses cat to put the contents of the file to standard out and then pipe's | it to the translate tr command

shuf & sort - Shuffle and Sort File by lines

- Pipe can take the output of one command and pass it to the next command without having to write to a file in between.
- '>' redirects output while '>>' appends the output

```
1 shuf numSeq.txt > numSeqRandom.txt
2 shuf numSeq.txt >> numSeqRandom.txt
3 sort numSeqRandom.txt | head
```

YOU ARE HERE

SYSTEM
COMMANDS

CREATE FILES

FILE EDITING

DIRECTORY
MANAGEMENT

DATA
MANIPULATION

FILES & SYSTEMS

Find and Replace

<i>Command</i>	<i>Function</i>	<i>Syntax/example usage</i>
grep	search a pattern	grep [OPTIONS] "PATTERN" FILENAME
*	variable used to represent many characters	ls *.txt
'?'	variable used to represent any one character	'ls' ?.txt

YOU ARE HERE

SYSTEM
COMMANDS

CREATE FILES

FILE EDITING

DIRECTORY
MANAGEMENT

DATA
MANIPULATION

FILTERING FILE DATA

grep - global regular expression print

- This command will replace your find command in a text document.

```
1 grep "Plan" myFirstFile.txt
2 grep "Plan" myFirstFile.txt | wc
3 4
```

- Some other commonly used parameters.

Parameter

Function

-i	ignore case
-v	invert matching, i.e. select non-matching lines
-c	print only the count of matching lines
-n	prefix the output with line number

YOU ARE HERE

SYSTEM
COMMANDS

CREATE FILES

FILE EDITING

DIRECTORY
MANAGEMENT

DATA
MANIPULATION

FILTERING FILE DATA

Downloading and transferring data

Command	Function	Syntax/example usage
wget	URL download specified URL	wget https://saltandpepper.jpg
scp	secure copy	scp username@hostname/Path2Folder/FILENAME .
rsync	remote sync	rsync -avz -e ssh username@hostname:/Path2Folder .

wget - download(get) from the www

```
1 wget https://bioinformaticsworkbook.org/Appendix/Unix/assets/Cat.jpg
2 wget https://bioinformaticsworkbook.org/Appendix/Unix/assets/saltandpepper.jpg
```

scp - secure copy

```
1 scp username@remotehostname:/home/username/unixTutorial/Cat.jpg
```

YOU ARE HERE

SYSTEM
COMMANDS

CREATE FILES

FILE EDITING

DIRECTORY
MANAGEMENT

DATA
MANIPULATION

FILTERING FILE DATA