

BINF2111 - Introduction to Bioinformatics Computing

UNIX 101 - enter the coding zone

UNIX

**Richard Allen White III, PhD
RAW Lab**

Lecture 6 - Thursday Sep 8th, 2022

Learning Objectives

- Sort/uniq/cut commands
- Review quiz and bonus
- wget/zip/tar
- tr/printf command
- text editors
- Quiz 6

sort – syntax anatomy of sort

sort [options] file.txt

-t “t” set the delimiter when using -k
default is non-blank to blank transition

-k 3 sort column #3 (left is 1)

-k 2,3 sort multiple columns

-n sort numerically

-r reverse sort order

-u drop duplicates from the result

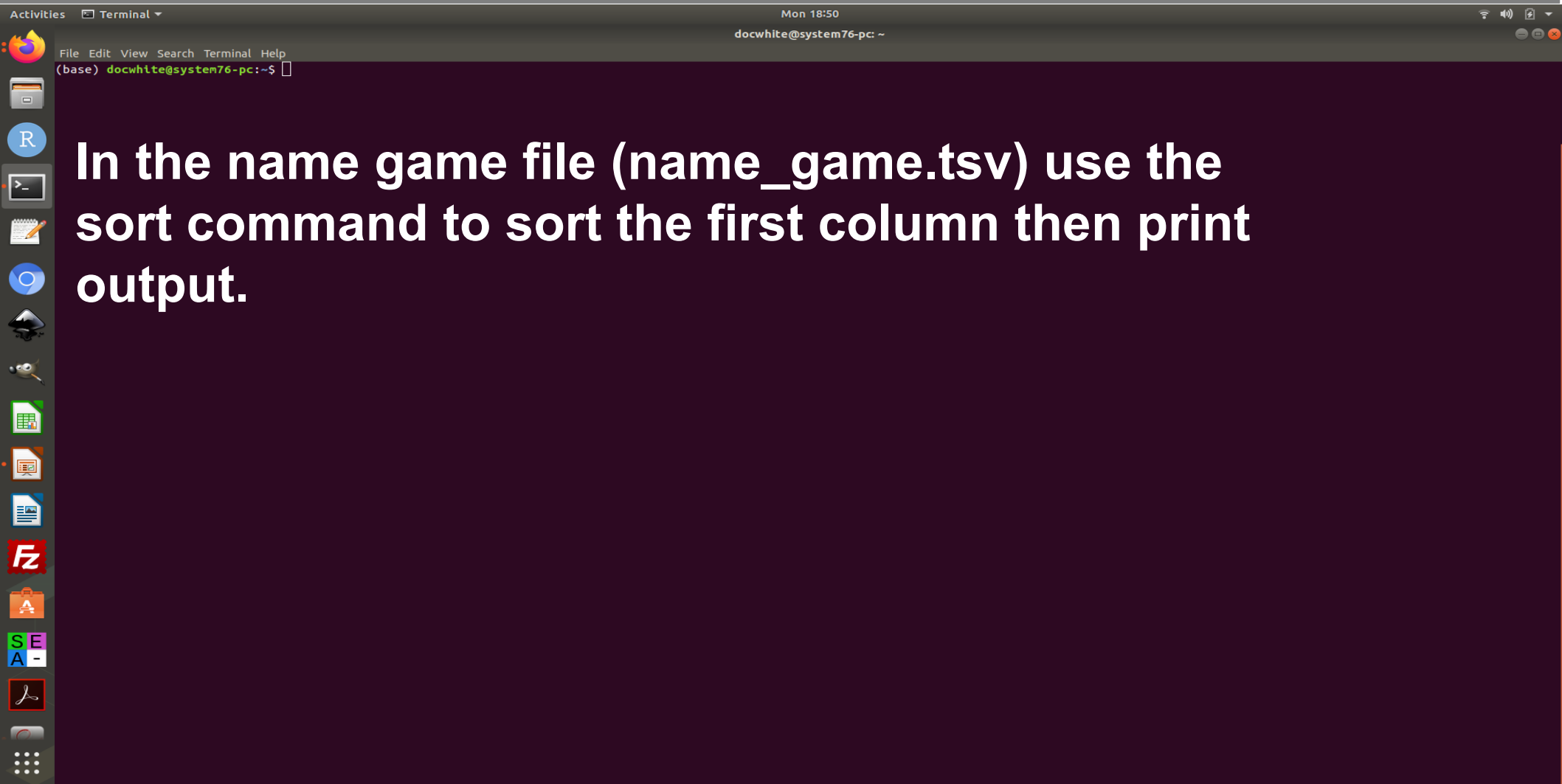
-b, --ignore-leading-blanks, ignore leading blanks

-d, --dictionary-order consider only blanks and alphanumeric characters

-f, --ignore-case fold lower case to upper case characters

No options: sort alphabetically from leftmost character.

Sort examples



In the name game file (name_game.tsv) use the sort command to sort the first column then print output.

```
File Edit View Search Terminal Help
(base) docwhite@system76-pc:~$
```

In the name game file (name_game.tsv) use the sort command to sort the first column then print output.

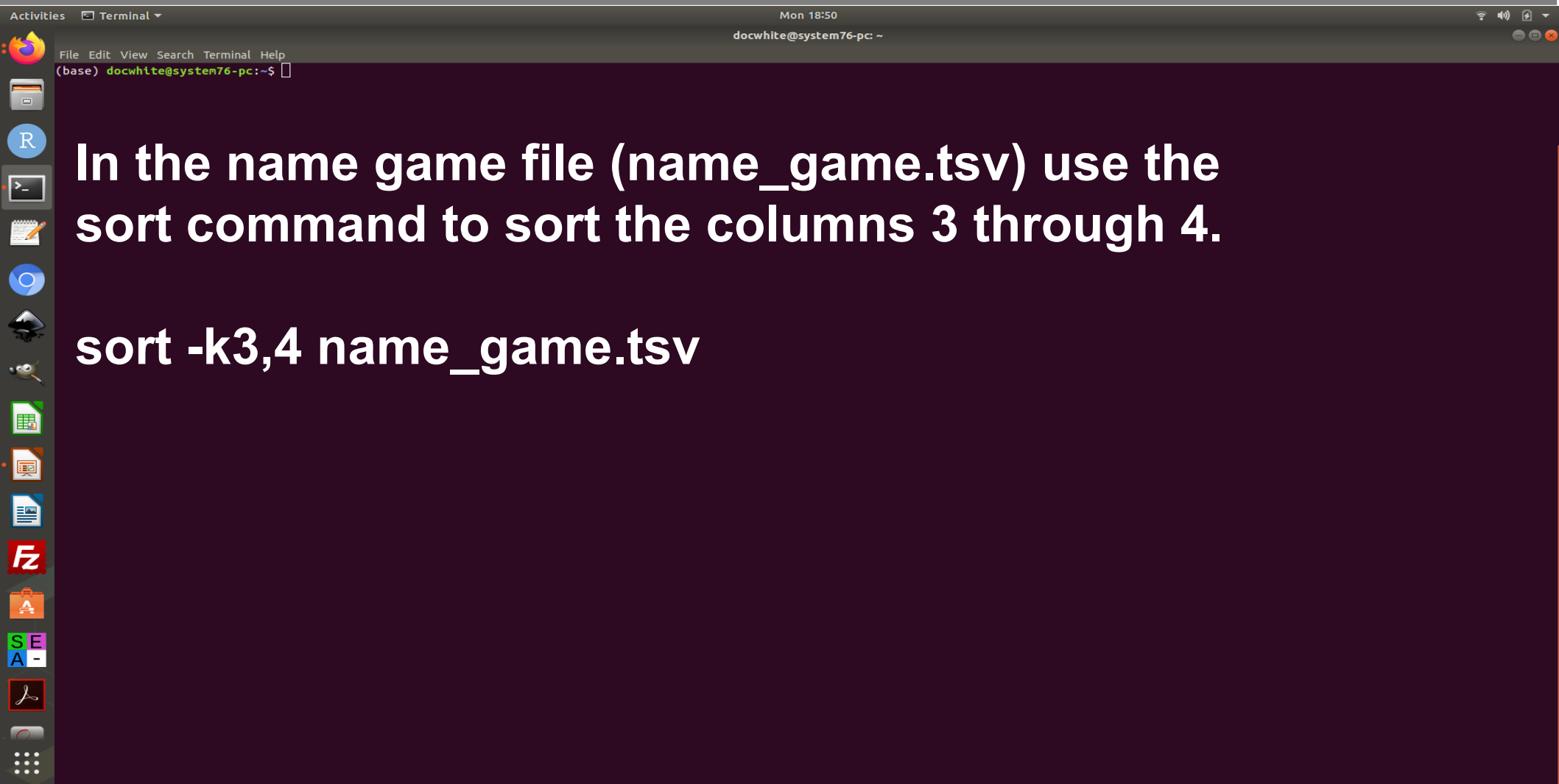
sort -k1 name_game.tsv

As tab (t) is default

Sort examples

In the name game file (name_game.tsv) use the sort command to sort the columns 3 through 4.

Sort examples



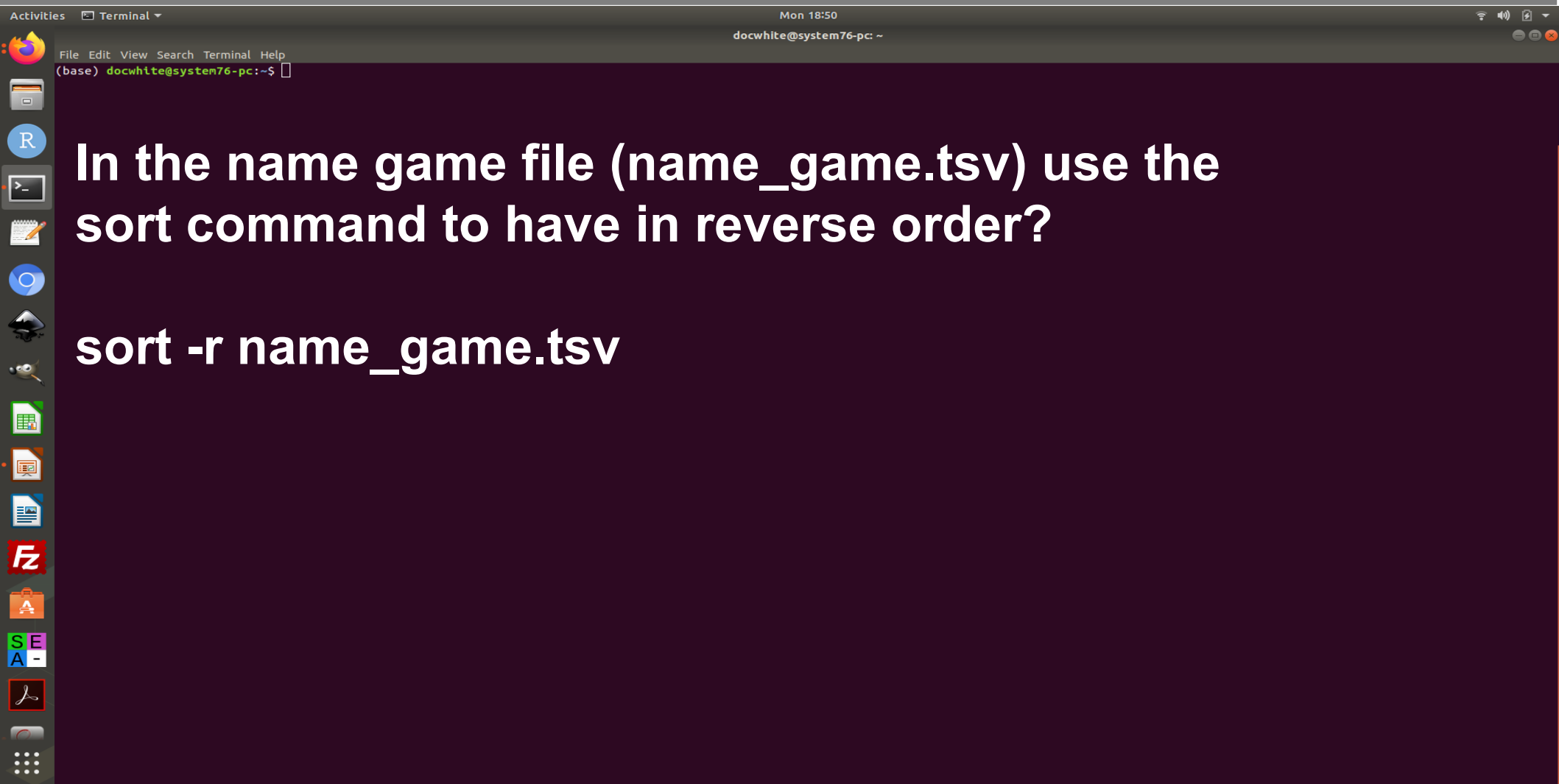
In the name game file (name_game.tsv) use the sort command to sort the columns 3 through 4.

```
sort -k3,4 name_game.tsv
```

Sort examples

In the name game file (name_game.tsv) use the sort command to have in reverse order?

Sort examples

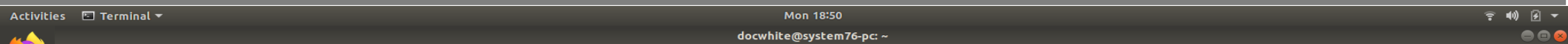


File Edit View Search Terminal Help
(base) docwhite@system76-pc:~\$

In the name game file (name_game.tsv) use the
sort command to have in reverse order?

`sort -r name_game.tsv`

Sort examples



File Edit View Search Terminal Help
(base) docwhite@system76-pc:~\$

In the name game file (name_game.tsv) use the
sort command here

`sort -k1 name_game.tsv`

`sort -k2 name_game.tsv`

`sort -k1,2 name_game.tsv`

What happens?

uniq – syntax anatomy of uniq

uniq [options] file.txt

Only works on sorted files and adjacent lines!

- c count lines for each unique value

- d only report duplicated lines

- u only report non-duplicated lines

No options: drop all duplicated lines (keeps 1 copy)

uniq examples

Using the doppelganger_names.txt how many unique lines are there?

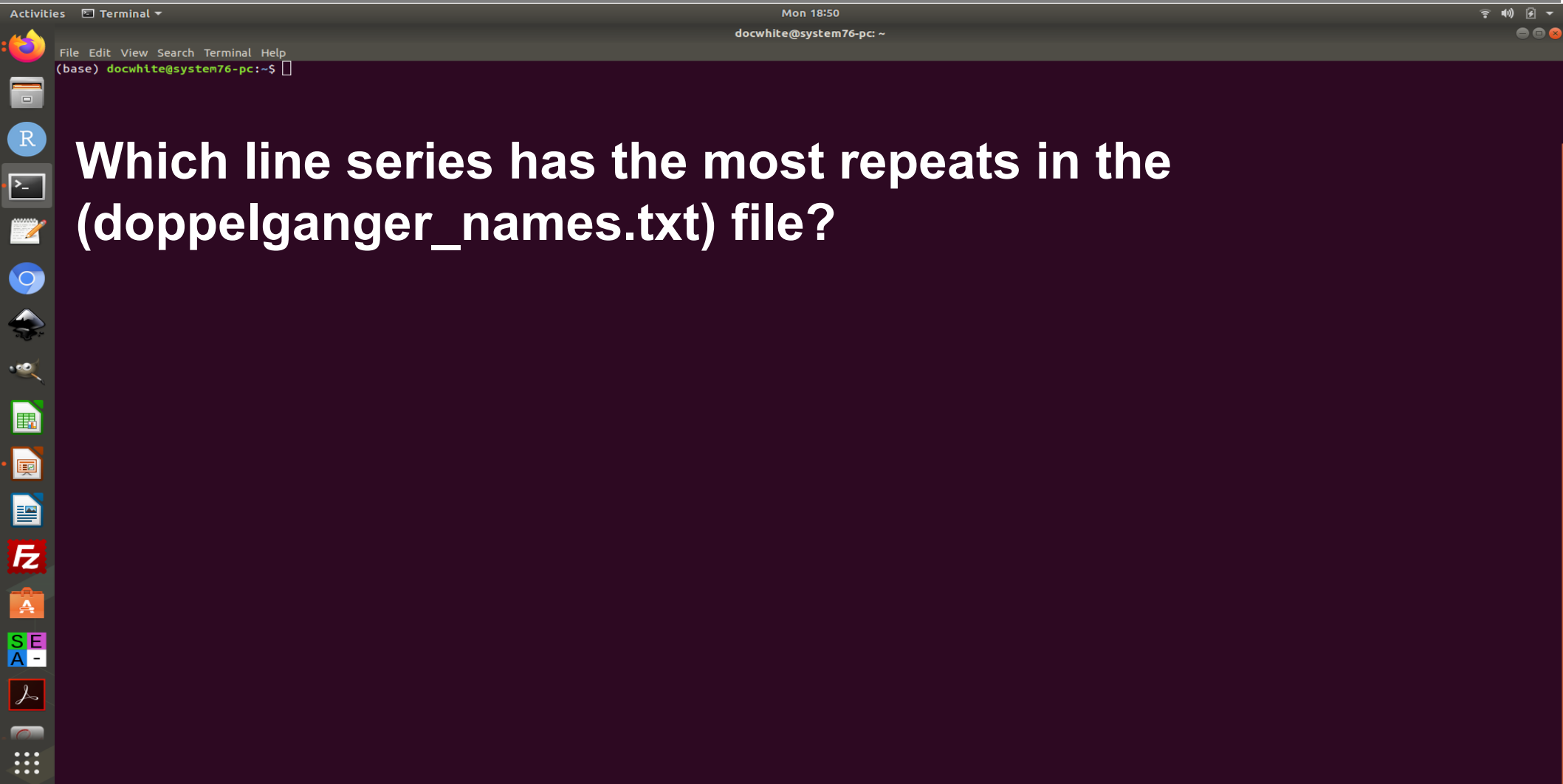
uniq examples

Using the doppelganger_names.txt how many unique lines are there?

```
sort -k1 doppelganger_names.txt | uniq -c | wc -l
```

5

uniq examples



uniq examples

Which line series has the most repeats in the (doppelganger_names.txt) file?

```
sort -k1 doppelganger_names.txt | uniq -c
```

```
4 david    abdul    chi      bill
4 mary     david     bill     abdul
```

All together now examples

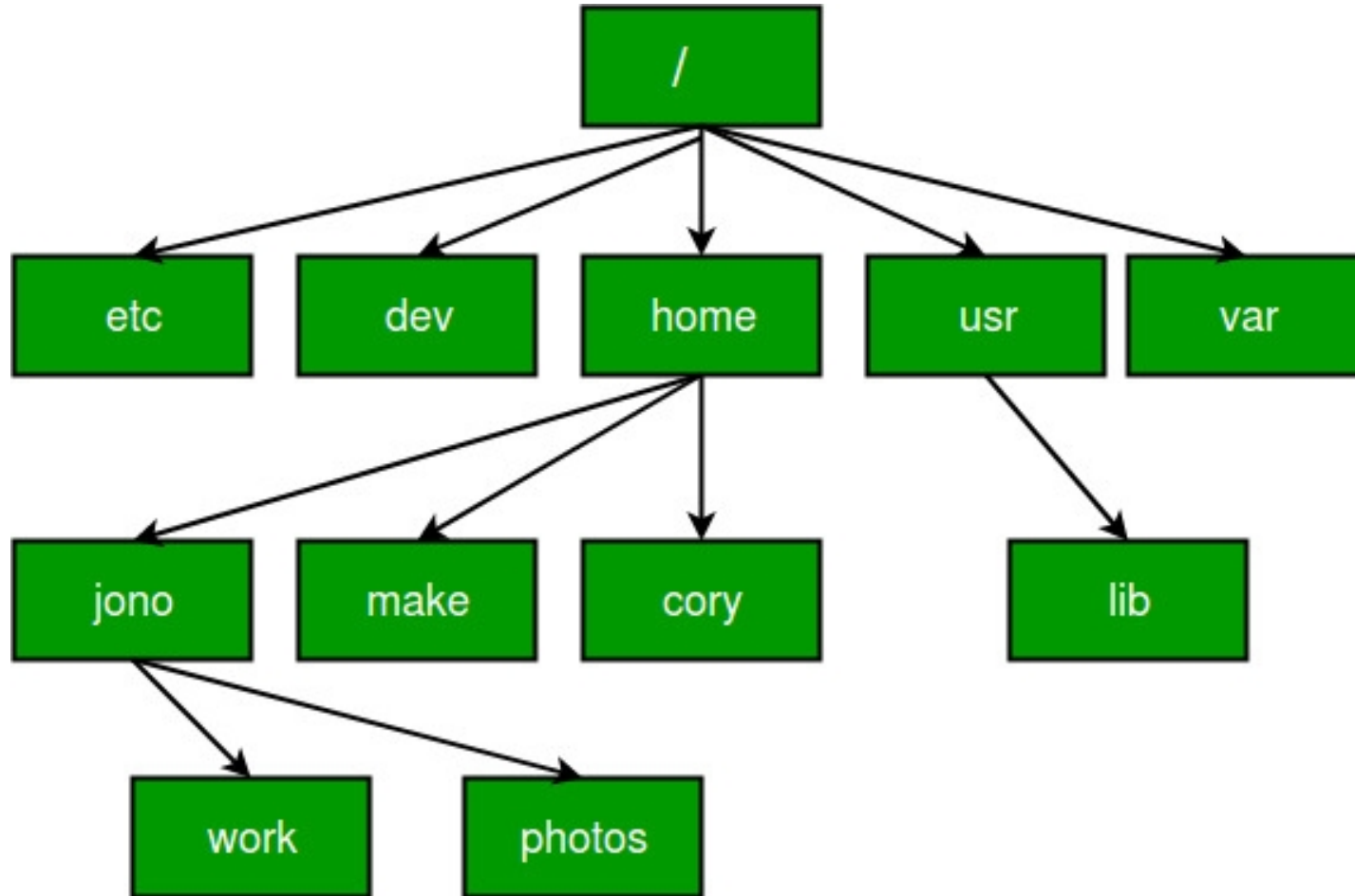
Find me the unique names in row 2 of the doppleganger file and how many times do they show up?

All together now examples

Find me the unique names in column 2 of the doppleganger file and how many times do they show up?

```
cut -f2 doppelganger_names.txt | sort | uniq -c
```

PATHS



PATHS

Activities Terminal Mon 18:50

File Edit View Search Terminal Help
(base) docwhite@system76-pc:~\$

echo \$PATH

/usr/local/bin:/bin:/usr/bin:/sbin:/usr/sbin:

Which ls

/bin/l

PATHS

Activities Terminal Mon 18:50

docwhite@system76-pc: ~

File Edit View Search Terminal Help

(base) docwhite@system76-pc:~\$

export PATH=\$PATH:/new/path

Or

Edit your .bashrc file

gedit .bashrc or other text editor

Missed Quiz Question

echo \$PATH

Means?

Missed Quiz Question

echo \$PATH

Means?

lists all directories

Missed Quiz Question

echo \$PATH

Means?

lists all directories – NO

Missed Quiz Question

echo \$PATH

Means?

lists all directories - NO (ls)

writes working directory?

Missed Quiz Question

echo \$PATH

Means?

lists all directories - NO (ls)

writes working directory - NO

Missed Quiz Question

echo \$PATH

Means?

lists all directories - NO (ls)

writes working directory - NO (pwd)

Missed Quiz Question

echo \$PATH

Means?

lists all directories - NO (ls)

writes working directory - NO (pwd)

writes your path

Missed Quiz Question 5

If I had a file with lots of repeat lines who would I count the unique lines?

```
sort -k1 file.txt | uniq -c
```

```
sort -k1 file.txt | uniq -u
```

Missed Quiz Question 5

If I had a file with lots of repeat lines who would I **count** the unique lines?

```
sort -k1 file.txt | uniq -c
```

```
sort -k1 file.txt | uniq -u
```

This counts line in terminal.

Missed Quiz Question 5

If I had a file with lots of repeat lines who would I **count** the unique lines?

```
sort -k1 file.txt | uniq -c
```

```
sort -k1 file.txt | uniq -u
```

What if I add | wc -l ?

Missed Quiz Question 5

If I had a file with lots of repeat lines who would I **count** the unique lines?

```
sort -k1 file.txt | uniq -c
```

```
sort -k1 file.txt | uniq -u
```

What if I add | wc -l ?
3?

Lab Two most missed question

Count the numbers of 'A', 'T', 'G', 'C' using multiple or single grep commands in example.fasta?

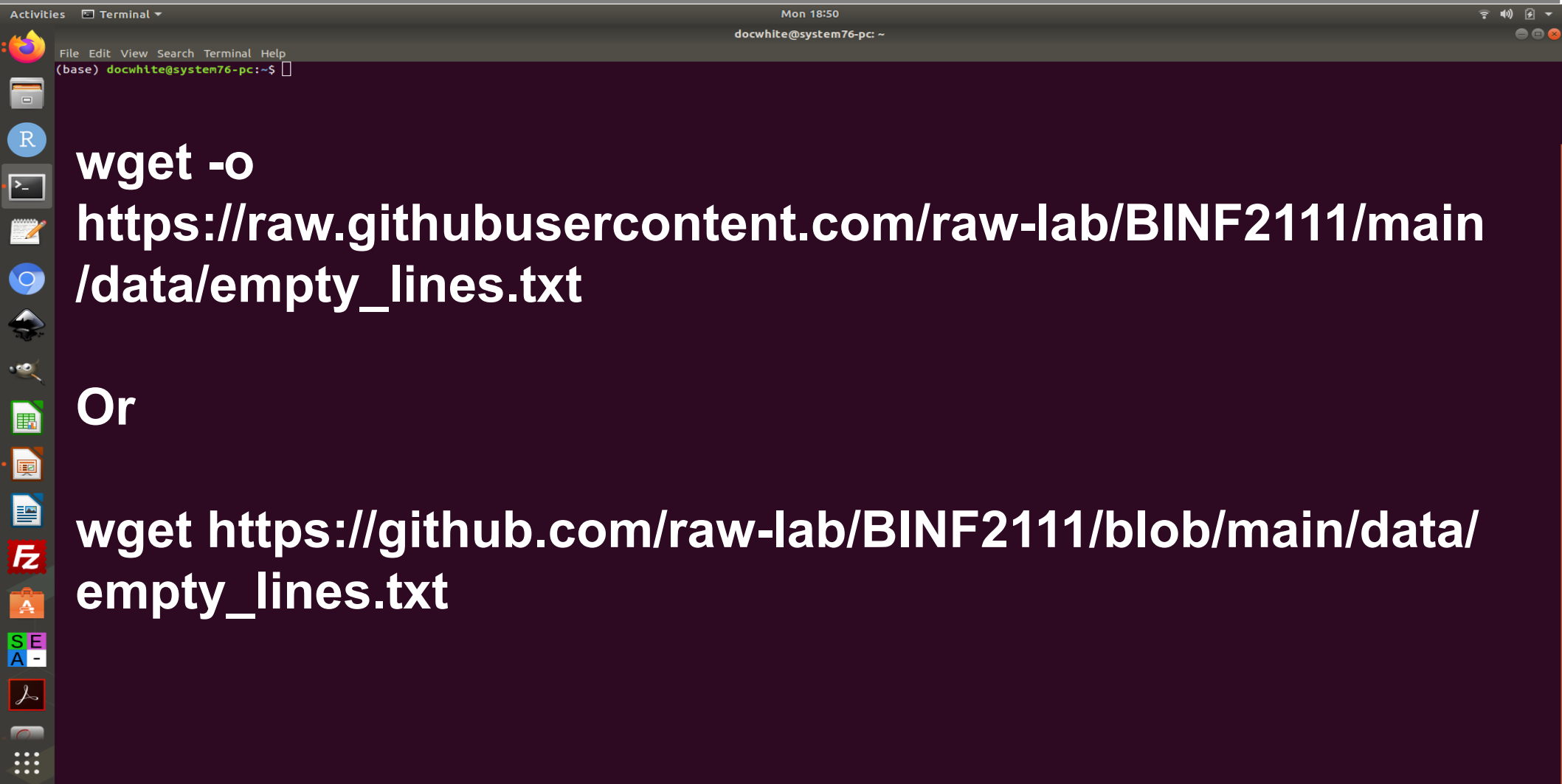
Bonus 5

- Convert name game file (name_game.csv) to tsv with:
 - tr
 - awk

Bonus 5

- **Convert name game file (name_game.csv) to tsv with:**
 - **tr:** `cat name_game.csv | tr -s ',' '\t' >name_game.tsv`
 - **awk:** `cat name_game.csv | awk -F ',' '{ $1=$1 } 1' >name_game.tsv`

wget examples



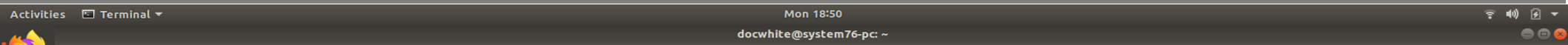
wget -o

**https://raw.githubusercontent.com/raw-lab/BINF2111/main
/data/empty_lines.txt**

Or

**wget https://github.com/raw-lab/BINF2111/blob/main/data/
empty_lines.txt**

wget examples



File Edit View Search Terminal Help
(base) docwhite@system76-pc:~\$

wget -o

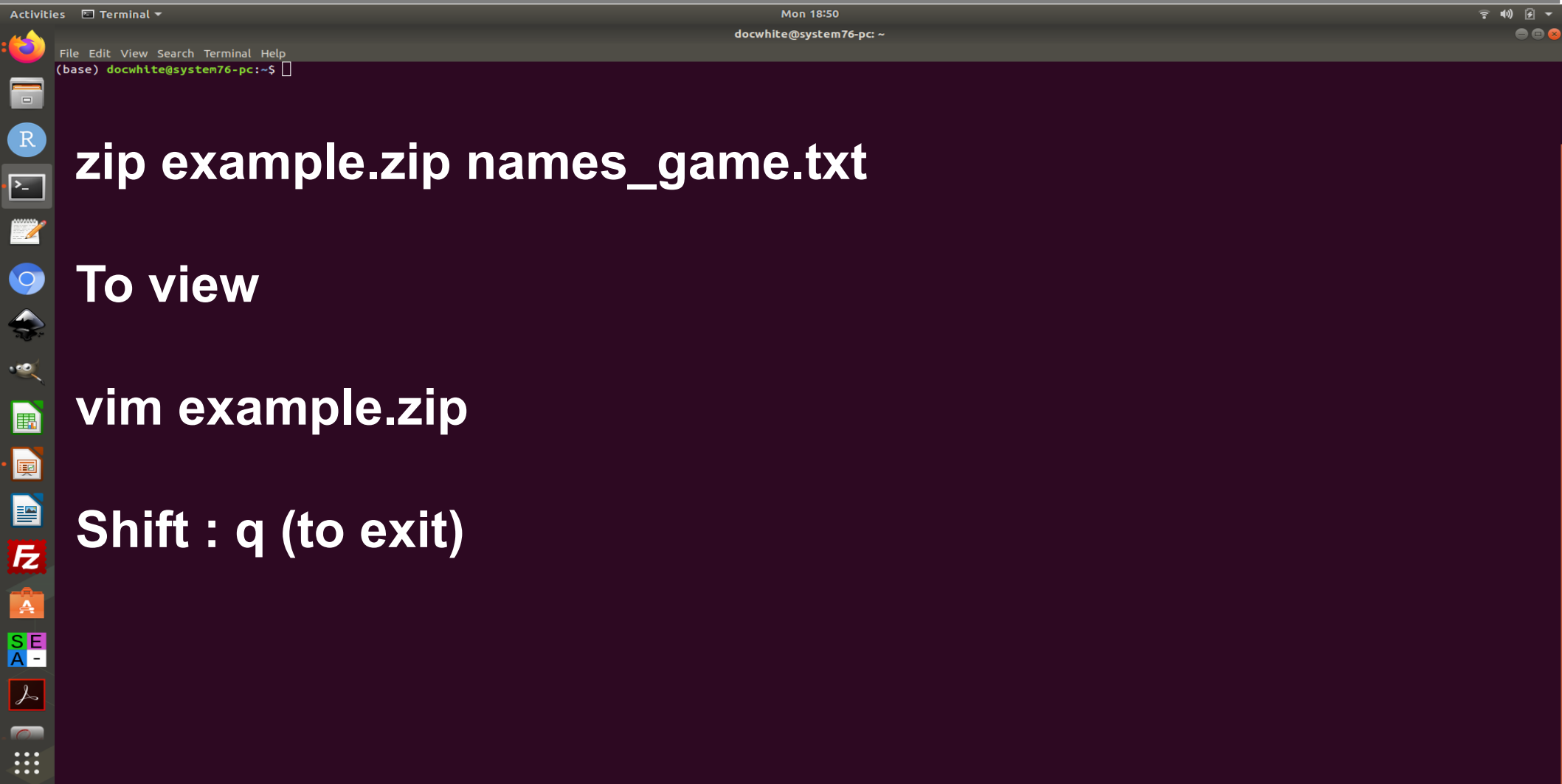
<https://github.com/raw-lab/BINF2111/blob/main/course-materials/Windows-Install-linux.pdf>

or

wget

https://www.tutorialspoint.com/unix/unix_tutorial.pdf

zip examples



(base) docwhite@system76-pc:~\$

zip example.zip names_game.txt

To view

vim example.zip

Shift : q (to exit)

zip examples

zip example.zip names_game.txt

To view

vim example.zip

Shift : q (to exit)

zip examples

Activities Terminal Mon 18:50 docwhite@system76-pc: ~

File Edit View Search Terminal Help
(base) docwhite@system76-pc:~\$

unzip example.zip

To extract

zip examples

Activities Terminal Mon 18:50 docwhite@system76-pc: ~

File Edit View Search Terminal Help
(base) docwhite@system76-pc:~\$

unzip example.zip

To extract

zip examples

Activities Terminal ▾

Mon 18:50

```
docwhite@system76-pc: ~
```

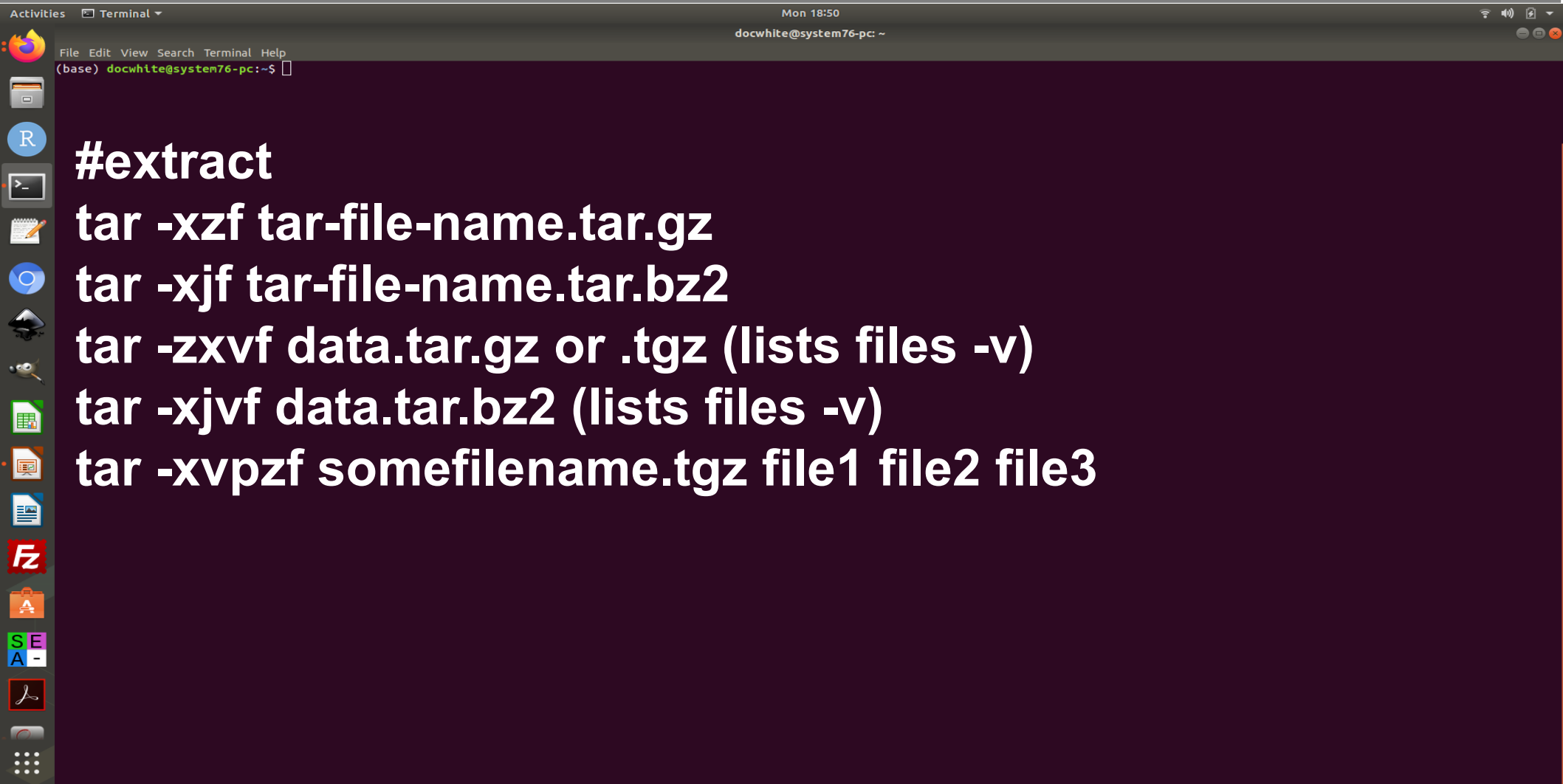
File Edit View Search Terminal Help

```
(base) docwhite@system76-pc:~$
```

```
unzip '*.zip'
gunzip *.gz
binzip2 *.bz2
```

```
zip '*.zip'
gzip *.gz
binzip2 *.bz2
```

tar examples



#extract

tar -xzf tar-file-name.tar.gz

tar -xjf tar-file-name.tar.bz2

tar -zxvf data.tar.gz or .tgz (lists files -v)

tar -xjvf data.tar.bz2 (lists files -v)

tar -xvpzf somefilename.tgz file1 file2 file3

tar examples

Activities Terminal Mon 18:50 docwhite@system76-pc: ~

File Edit View Search Terminal Help
(base) docwhite@system76-pc:~\$

#compress

tar -jcvf tar.bz2 file

tar -zcvf .tar.gz file

tar -czfv Test.tar.gz Test/

czfv = 'Compress Zip File Verbose'

If you want bzip files, use 'j' instead of 'z'.

tar -cvpzf somefilename.tgz file1 file2 file3

Tr (translate) – syntax anatomy UNIX tool

tr [OPTION] SET1 [SET2]

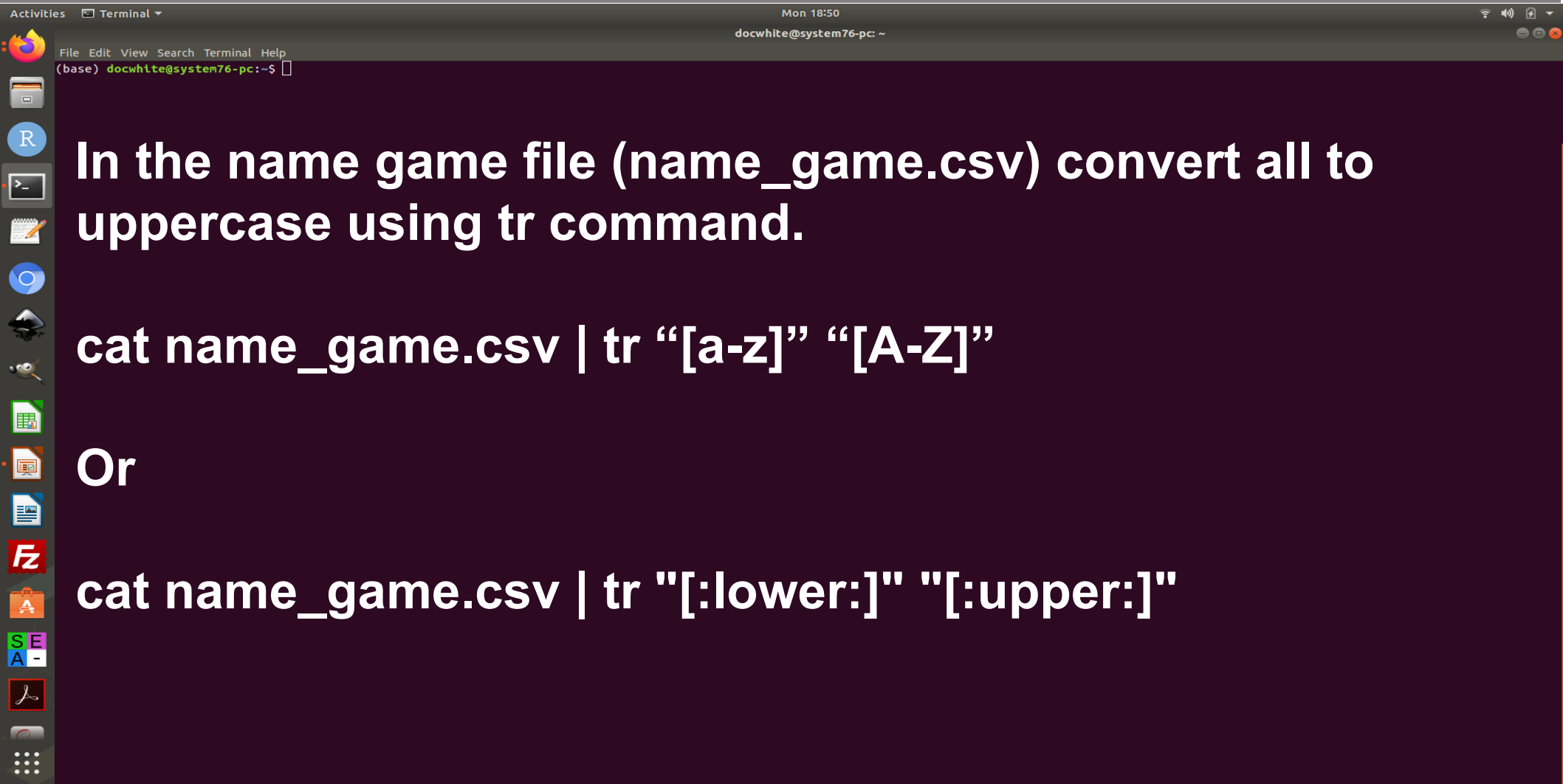
- c : complements the set of characters in string (i.e., operations apply to characters not in the given set)
- d : delete characters in the first set from the output.
- s : replaces repeated characters listed in the set1 with single occurrence
- t : truncates set1

tr (no option) = substitute [original] [new]

tr examples

In the name game file (name_game.csv) convert all to uppercase using tr command.

tr examples



In the name game file (name_game.csv) convert all to uppercase using tr command.

```
cat name_game.csv | tr "[a-z]" "[A-Z]"
```

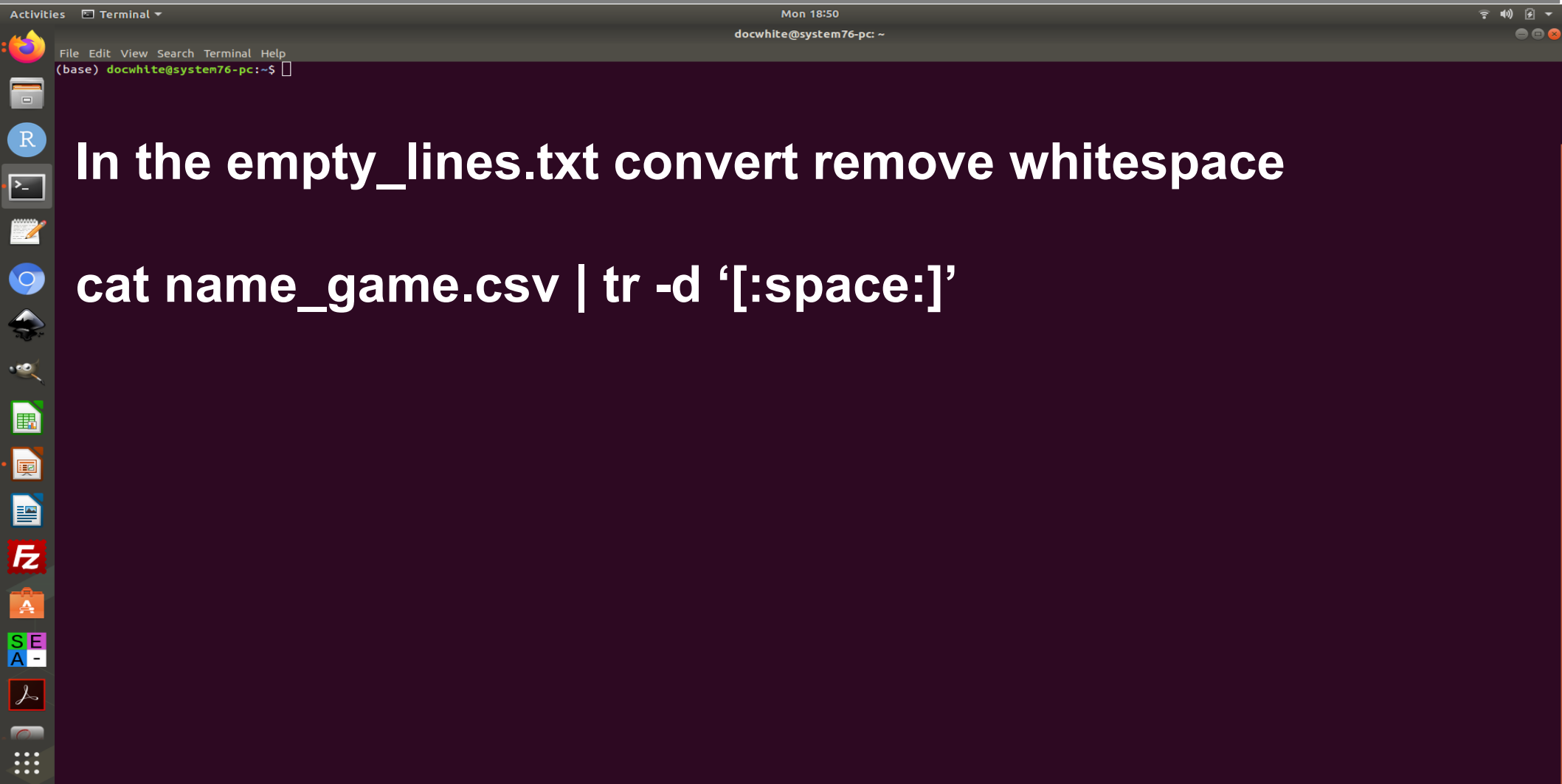
Or

```
cat name_game.csv | tr "[:lower:]" "[:upper:]"
```

tr examples

In the empty_lines.txt convert remove whitespace

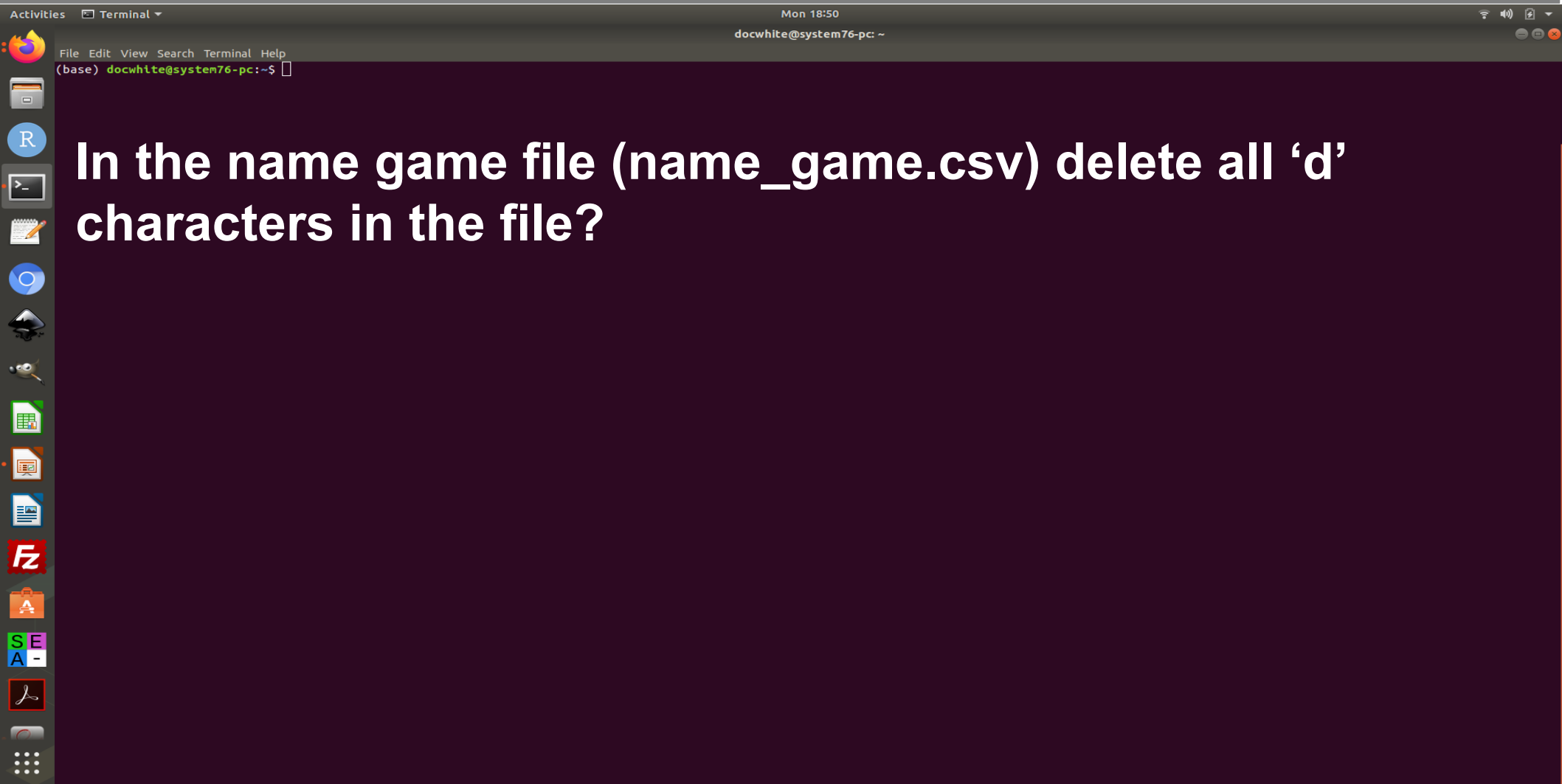
tr examples



In the empty_lines.txt convert remove whitespace

```
cat name_game.csv | tr -d '[:space:]'
```


tr examples



tr examples

In the name game file (name_game.csv) delete all 'd' characters in the file?

```
cat name_game.csv | tr -d '[Dd]'
```

printf – syntax anatomy UNIX

printf [-v var] format [arguments]

- %d For signed decimal numbers
- %i For signed decimal numbers
- %u For unsigned decimal numbers
- %o For unsigned octal numbers
- %x For unsigned hexadecimal numbers with lower case letters (a-f)
- %X For unsigned hexadecimal numbers with upper case letters (A-F)
- %f For floating point numbers
- %s For string
- %% For percent % symbol

printf – syntax anatomy UNIX

printf [-v var] format [arguments]

--help display this help and exit

--version output version information and exit

\xHH byte with hexadecimal value HH (1 to 2 digits)

\uHHHH Unicode (ISO/IEC 10646) character with hex
value HHHH (4 digits)

\UHHHHHHHH

Unicode character with hex value HHHHHHHH (8 digits)

%% a single %

%b ARGUMENT as a string with ‘\’ escapes interpreted,
except that octal escapes are of the form \0 or \0NNN

\" double quote

\NNN character with octal
value NNN (1 to 3 digits)

\\ backslash

\a alert (BEL)

\b backspace

\c produce no further output

\f form feed

\n new line

\r carriage return

\t horizontal tab

\v vertical tab

printf – syntax anatomy UNIX

printf [-v var] format [arguments]

N This specifies the width of the field for output.

* This is the placeholder for the width.

- To left align output in the field. (Default: Right align)

0 Pad result with leading 0s.

+ To put + sign before positive numbers and - sign for negative numbers.

printf() function of C programming language.

We can say that printf is a successor of echo command.

printf examples

Activities Terminal

Mon 18:50

docwhite@system76-pc:

File Edit View Search Terminal Help

```
(base) docwhite@system76-pc:~$
```

```
printf '#!/bin/bash\n' >script.sh
```

printf examples

Activities Terminal Mon 18:50 docwhite@system76-pc: ~

File Edit View Search Terminal Help
(base) docwhite@system76-pc:~\$

```
printf '#!/bin/bash\n' >script.sh
```

```
more script.sh
```

```
#!/bin/bash
```

printf examples

```
printf "Open issues: %s\nClosed issues: %s\n" "34"
"65"
```


printf examples

Activities Terminal Mon 18:50 docwhite@system76-pc: ~

File Edit View Search Terminal Help
(base) docwhite@system76-pc:~\$

```
printf "Open issues: %s\nClosed issues: %s\n" "34"  
"65"
```

```
Open issues: 34  
Closed issues: 65
```

printf examples

```
printf "Decimal: %d\nHex: %x\nOctal: %o\n" 100 100 100
```

printf examples

Activities Terminal Mon 18:50 docwhite@system76-pc: ~

File Edit View Search Terminal Help
(base) docwhite@system76-pc:~\$

```
printf "Decimal: %d\nHex: %x\nOctal: %o\n" 100 100 100
```

number 100 in three different number systems

Decimal: 100

Hex: 64

Octal: 144

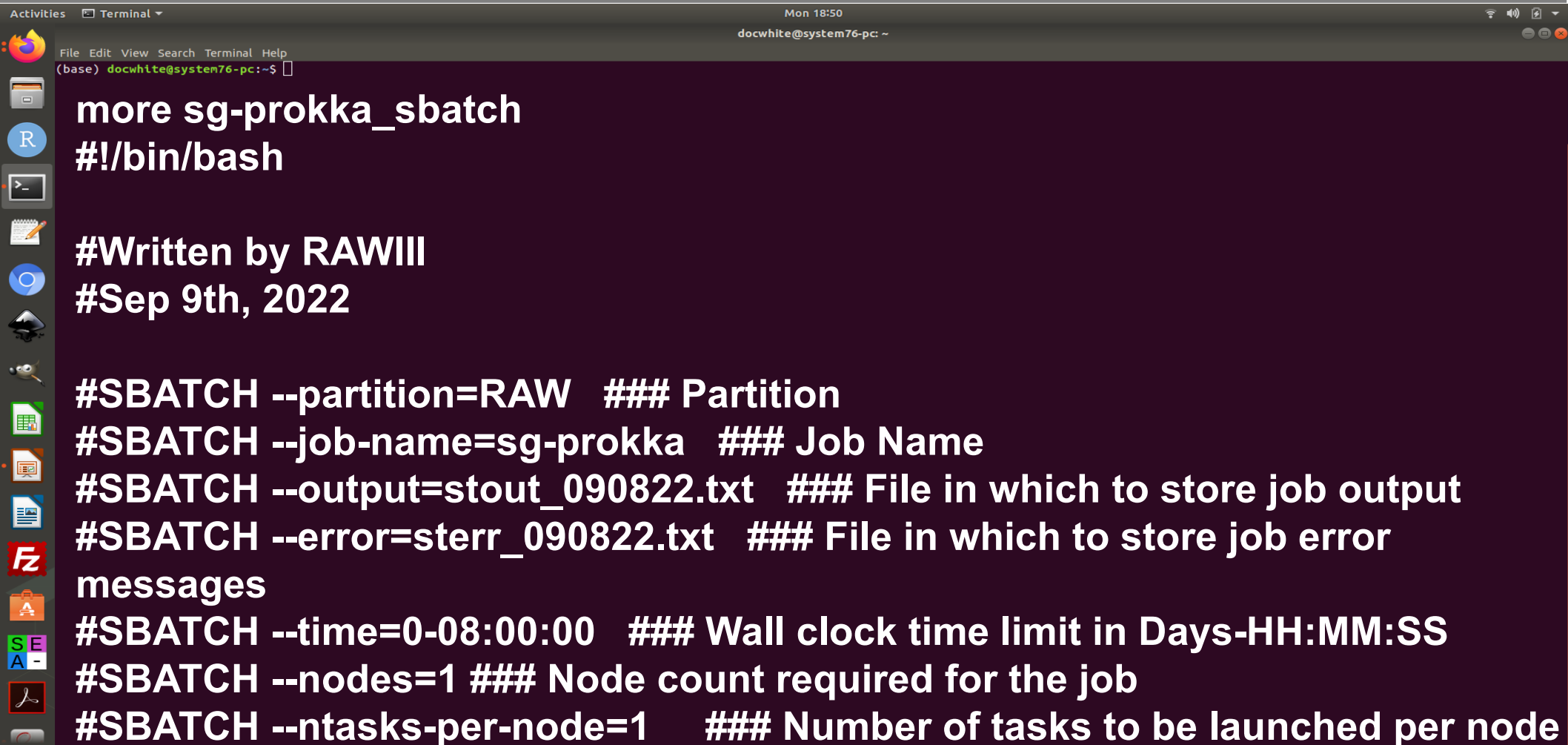
printf examples

Activities Terminal Mon 18:50 docwhite@system76-pc: ~

File Edit View Search Terminal Help
(base) docwhite@system76-pc:~\$

```
printf '#!/bin/bash\n \n#Written by RAWIII\n#Sep 9th,  
2021\n \n#SBATCH --partition=RAW   ### Partition\  
\n#SBATCH --job-name=sg-prokka   ### Job Name\  
\n#SBATCH --output=stout_090721.txt   ### File in which  
to store job output\n#SBATCH --error=sterr_090721.txt  
### File in which to store job error messages\n#SBATCH  
--time=0-08:00:00   ### Wall clock time limit in Days-  
HH:MM:SS\n#SBATCH --nodes=1   ### Node count  
required for the job\n#SBATCH --ntasks-per-node=1  
### Number of tasks to be launched per Node\n' >>sg-  
prokka_sbbatch
```

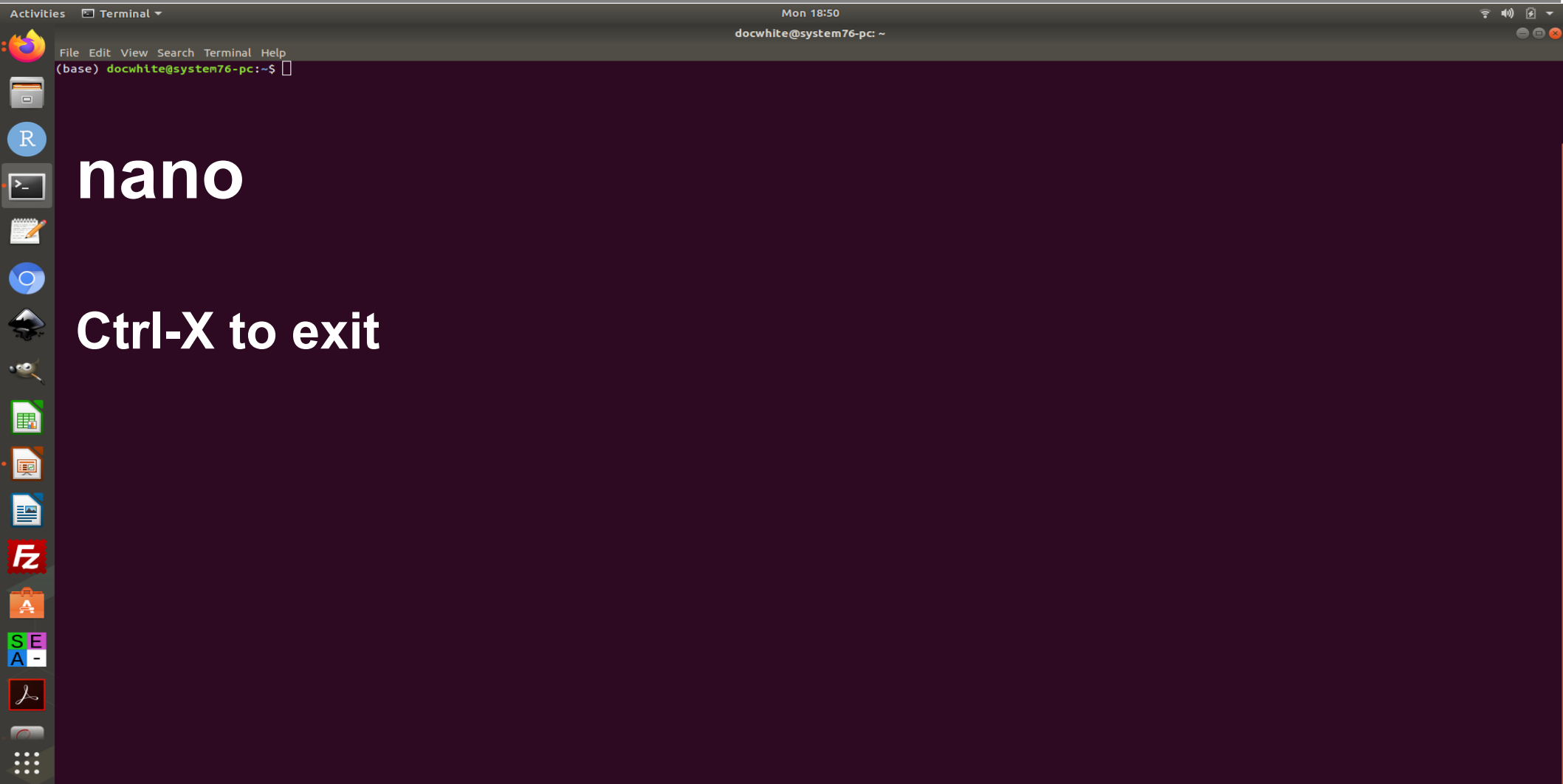
printf examples



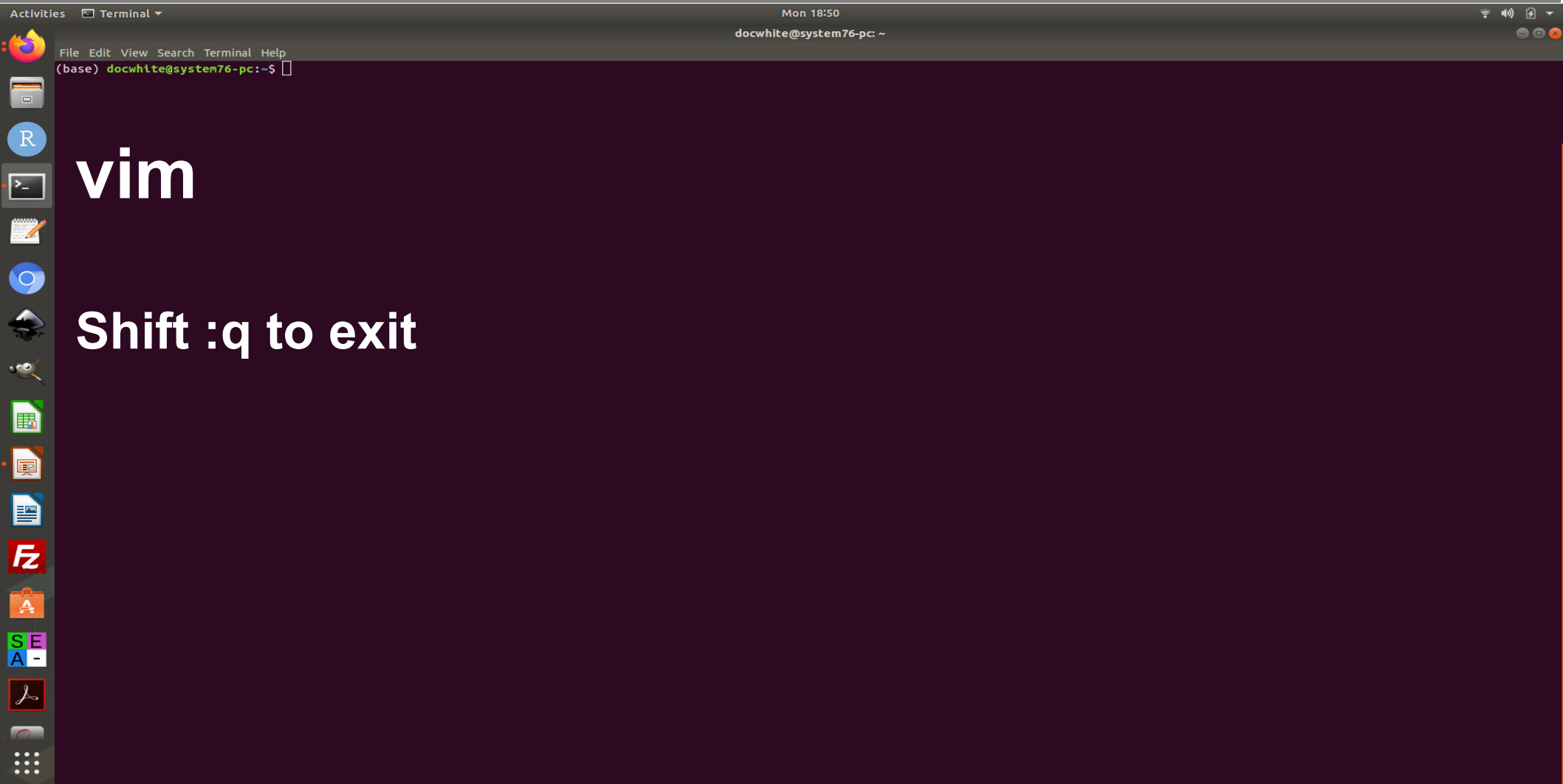
A terminal window titled "Terminal" with a dark purple background. The window shows a series of printf statements for an sbatch script. The terminal title bar includes "Activities", "Terminal", and system information like "Mon 18:50" and "docwhite@system76-pc: ~". The left sidebar shows various application icons. The terminal content is as follows:

```
(base) docwhite@system76-pc:~$  
  
more sg-prokka_sbatach  
#!/bin/bash  
  
#Written by RAWIII  
#Sep 9th, 2022  
  
#SBATCH --partition=RAW   ### Partition  
#SBATCH --job-name=sg-prokka   ### Job Name  
#SBATCH --output=stout_090822.txt   ### File in which to store job output  
#SBATCH --error=sterr_090822.txt   ### File in which to store job error  
messages  
#SBATCH --time=0-08:00:00   ### Wall clock time limit in Days-HH:MM:SS  
#SBATCH --nodes=1   ### Node count required for the job  
#SBATCH --ntasks-per-node=1   ### Number of tasks to be launched per node
```

Text editors - nano



Text editors - vim




Text editors - gedit

Activities

Terminal ▼

Mon 18:50



CE

A -



atom



Add folders



A hackable text editor for the 21st Century

For help, please visit


☒ Show Welcome Guide when opening Atom

atom.io x 

Get to know Atom!

 Open a Project

📦 Install a Package

 Choose a Theme

↔ Hack on the Init Script

<> Add a Snippet

 Learn Keyboard Shortcuts

Quiz 6

- On canvas now

Bonus 6

- In the doppelganger_names.txt count how many times the name 'chi' is left to the name 'bill'

Using grep only command:

Using grep with printf command:

Only awk: