



Tutorial 1: Unix Command Line (I)

CS 104

Spring, 2024-25

TA: Abhi Jain

Credits: TA-2023-2: Guramrit Singh



Topics

- Basic Commands
- Exercises



Basic commands

Basic commands



pwd : present working directory

ls : list directory contents

mkdir : make new directory

cd : change directory

mv : move

cp : copy

rm : remove

rmdir : remove directory

man : manual documentation page

cat : concatenate

clear : clear the terminal screen

echo : display the text passed in as an argument

head : display first lines of a file

tail : display the last part of a file

Clear

1. This command simply clears out the terminal screen, **nothing else** is changed.

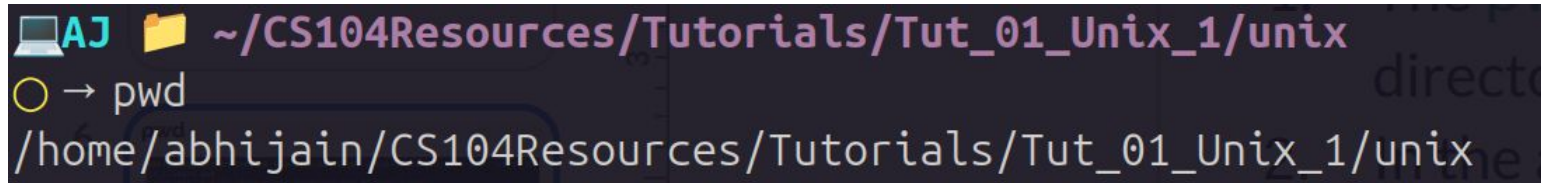


Man

1. A useful tool for viewing detailed documentation, options, and usage guidelines for various commands.
2. Usage: `man <command>`

Exercise: What will `man man` do?

pwd

A terminal window with a dark background. The prompt is 'AJ' followed by a folder icon and the path '~/CS104Resources/Tutorials/Tut_01_Unix_1/unix'. The command 'pwd' is entered, and the output is '/home/abhi Jain/CS104Resources/Tutorials/Tut_01_Unix_1/unix'.

```
AJ ~/CS104Resources/Tutorials/Tut_01_Unix_1/unix  
→ pwd  
/home/abhi Jain/CS104Resources/Tutorials/Tut_01_Unix_1/unix
```

1. The `pwd` command prints the full name (the **full path**) of current/working directory.
2. In the above **example**, current working directory is

`/home/abhi Jain/CS104Resources/Tutorials/Tut_01_Unix_1/unix`

This will be our working directory for this tutorial :)

ls

```
AJ ~/CS104Resources/Tutorials/Tut_01_Unix_1/unix
→ ls
code file-analysis joke1 joke2 joke3 photos test
```

1. The **ls** command is used to display a **listing of files and directories**.
2. If **no arguments** are given, then provides the list of files and directories in the **current location**.
3. If **argument is given**, then provides the list of files and directories within the **specified path**.
4. Additionally, various options can be used with ls to modify the output or gather more detailed information about the files and directories.

For example:

-a : lists hidden files/directories as well

-l : list files in the long format

5. **Exercise:** using man, see what -l, -R options are used for.

```
AJ ~/CS104Resources/Tutorials/Tut_01_Unix_1/unix
→ ls -la
total 36
drwxrwxr-x 6 abhijain abhijain 4096 Jan 13 13:04 .
drwxrwxr-x 3 abhijain abhijain 4096 Jan 13 11:10 ..
drwxr-xr-x 2 abhijain abhijain 4096 Nov 21 10:10 code
drwxrwxr-x 5 abhijain abhijain 4096 Jan 10 13:50 file-analysis
```

mkdir, cd

- The **mkdir** command is used to make a new directory.
- The **cd** command is used to change directory.
- In the **example**,
 - First we made a new directory called test.
 - Then, we changed our directory to test.
 - Finally we came back to our tutorial_1 directory.
Note: We used '..' to move into parent directory

```
AJ ~/CS104Resources/Tutorials/Tut_01_Unix_1/unix
→ ls CS108 - Tutorial 1 PPTX ☆ 📁
code file-analysis joke1 joke2 joke3 photos Tools Help
AJ ~/CS104Resources/Tutorials/Tut_01_Unix_1/unix
→ mkdir test
AJ ~/CS104Resources/Tutorials/Tut_01_Unix_1/unix
→ ls
code file-analysis joke1 joke2 joke3 photos test
AJ ~/CS104Resources/Tutorials/Tut_01_Unix_1/unix
→ cd test
AJ ~/CS104Resources/Tutorials/Tut_01_Unix_1/unix/test
→ ls
AJ ~/CS104Resources/Tutorials/Tut_01_Unix_1/unix/test
→ ls -la
total 8
drwxrwxr-x 2 abhijain abhijain 4096 Jan 13 13:04 .
drwxrwxr-x 6 abhijain abhijain 4096 Jan 13 13:04 ..
AJ ~/CS104Resources/Tutorials/Tut_01_Unix_1/unix/test
→ cd ..
AJ ~/CS104Resources/Tutorials/Tut_01_Unix_1/unix
→ ls
code file-analysis joke1 joke2 joke3 photos test
```


mv, cp

- The **mv** command is used to move files/folders. It can also be used for renaming files/folders.
- The **cp** command is used to copy files.
- In the **example**,
 - First we moved example.txt file into test.
 - Then, we copied example.txt file from test to ex.txt in the parent directory.
 - Then, we renamed ex.txt to example.txt.

```
AJ ~/CS104Resources/Tutorials/Tut_01_Unix_1/unix
→ ls
code example.txt file-analysis joke1 joke2 joke3 photos test
AJ ~/CS104Resources/Tutorials/Tut_01_Unix_1/unix
→ mv example.txt test
AJ ~/CS104Resources/Tutorials/Tut_01_Unix_1/unix
→ cd test
AJ ~/CS104Resources/Tutorials/Tut_01_Unix_1/unix/test
→ ls
example.txt
AJ ~/CS104Resources/Tutorials/Tut_01_Unix_1/unix/test
→ cp example.txt ../ex.txt
AJ ~/CS104Resources/Tutorials/Tut_01_Unix_1/unix/test
→ cd ..
AJ ~/CS104Resources/Tutorials/Tut_01_Unix_1/unix
→ ls
code ex.txt file-analysis joke1 joke2 joke3 photos test
AJ ~/CS104Resources/Tutorials/Tut_01_Unix_1/unix
→ mv ex.txt example.txt
AJ ~/CS104Resources/Tutorials/Tut_01_Unix_1/unix
→ ls
code example.txt file-analysis joke1 joke2 joke3 photos test
```

rm, rmdir

- The **rm** command is used to remove files. See -d, -r option.
- The **rmdir** command is used to remove directories. (*Note that directory should be empty*)
- In the **example**,
 - We tried to remove test, but failed because mails was not empty.
 - So, first we removed example.txt from mails
 - Finally, we removed test.

```
AJ ~/CS104Resources/Tutorials/Tut_01_Unix_1/unix
→ ls
code          file-analysis  joke2  photos
example.txt   joke1      joke3  test
AJ ~/CS104Resources/Tutorials/Tut_01_Unix_1/unix
→ rmdir test
rmdir: failed to remove 'test': Directory not empty
AJ ~/CS104Resources/Tutorials/Tut_01_Unix_1/unix
→ cd test
AJ ~/CS104Resources/Tutorials/Tut_01_Unix_1/unix/test
→ ls
example.txt
AJ ~/CS104Resources/Tutorials/Tut_01_Unix_1/unix/test
→ rm example.txt
AJ ~/CS104Resources/Tutorials/Tut_01_Unix_1/unix/test
→ cd ..
AJ ~/CS104Resources/Tutorials/Tut_01_Unix_1/unix
→ rmdir test
AJ ~/CS104Resources/Tutorials/Tut_01_Unix_1/unix
→ ls
code  example.txt  file-analysis  joke1  joke2  joke3  photos
```

cat



- The **cat** command is used to read data from a file and give its contents as output
- In the **example**, we used **cat** to print the contents of **joke1** file to **terminal**.

```
AJ ~/CS104Resources/Tutorials/Tut_01_Unix_1/unix
→ cat joke1
Why don't eggs tell jokes? Because they might crack up!

Why did the math book look sad? Because it had too many problems.

What do you get when you cross a snowman and a vampire? Frostbite.


Why did the scarecrow win an award? Because he was outstanding in his field!

Why don't oysters donate to charity? Because they are shellfish!

What's orange and sounds like a parrot? A carrot!
AJ ~/CS104Resources/Tutorials/Tut_01_Unix_1/unix
→
```

echo

- The `echo` command outputs whatever is given to it as argument.
- In the `example`, we used `echo` to print `hello cs104` to `terminal`.

Exercise: Try `echo $USER` (Just for fun! Detailed explanation later through the course)

```
AJ ~/CS104Resources/Tutorials/Tut_01_Unix_1/unix
○ → echo "hello cs104"
hello cs104
AJ ~/CS104Resources/Tutorials/Tut_01_Unix_1/unix
○ → echo joke1
joke1
AJ ~/CS104Resources/Tutorials/Tut_01_Unix_1/unix
○ → echo $USER
abhijain
AJ ~/CS104Resources/Tutorials/Tut_01_Unix_1/unix
○ →
```

head

- The **head** command is used to display first few data of a given input. By default, it prints the first 10 lines of the specified files.
- In the **example**, we used **head** to print first 3 lines of **joke3** file to **terminal**.
- Checkout options: -c

```
AJ ~/CS104Resources/Tutorials/Tut_01_Unix_1/unix
○ → cat -n joke3
 1
 2
 3     Why can't a nose be 12 inches long? Because then it would be a foot!
 4
 5     What did the grape do when it got stepped on? Nothing, it just let out a little
wine.
 6
 7     What do you call a dinosaur with an extensive vocabulary? A thesaurus!
 8
 9     Why don't scientists trust atoms? Because they make up everything!
AJ ~/CS104Resources/Tutorials/Tut_01_Unix_1/unix
○ → head -n 3 joke3

    Why can't a nose be 12 inches long? Because then it would be a foot!
```

- Cat with -n option prints line numbers

tail

- The **tail** command is used to display last few data of a given input. By default, it prints the last 10 lines of the specified files.
- Similar to what head does
- Checkout options: -v

LINUX TERMINAL FOR BEGINNERS



Other commands



There are various other commands that we will see as we progress through the course

- ps
- chmod
- tar
- grep
- cut
- wc
- less
- and many more ...



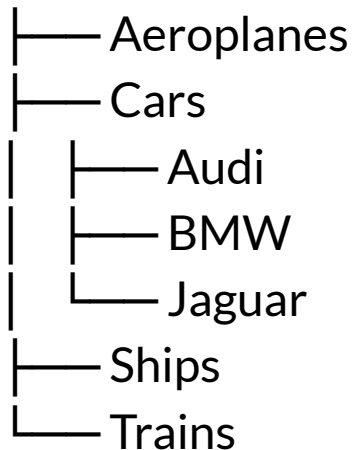
Exercises

Exercise 1



Create a new directory named **Vehicles** with the following directory structure, note that there are no files present yet.

Vehicles



```
AJ ~/CS104Resources/Tutorials/Tut_01_Unix_1/unix
→ tree Vehicles/
Vehicles/
├── Aeroplane
├── Cars
│   ├── Audi
│   ├── BMW
│   └── Jaguar
├── Ships
└── Trains

8 directories, 0 files
```

A terminal window screenshot showing the command `tree Vehicles/` and its output. The output displays the directory structure as a tree, with 'Aeroplanes' misspelled as 'Aeroplane'. The terminal also shows the count '8 directories, 0 files' at the bottom.

Solution 1

1. One solution is to create all directories one by one. (Brute Force)
2. An alternative is to generate multiple directories simultaneously, rather than individually creating each one separately.

Notice the use of {}.

3. The -p option

```
AJ ~ /CS104Resources/Tutorials/Tut_01_Unix_1/unix
→ rm -R Vehicles/
AJ ~ /CS104Resources/Tutorials/Tut_01_Unix_1/unix
→ mkdir Vehicles
AJ ~ /CS104Resources/Tutorials/Tut_01_Unix_1/unix
→ mkdir Vehicles/{Cars,Ships,Trains,Aeroplane}
AJ ~ /CS104Resources/Tutorials/Tut_01_Unix_1/unix
→ mkdir Vehicles/Cars/{Audi,BMW,Jaguar}
```

METHOD - 2

```
AJ ~ /CS104Resources/Tutorials/Tut_01_Unix_1/unix
→ mkdir -p Vehicles/{Cars/{Audi,BMW,Jaguar},Trains,Aeroplane,Ships}
AJ ~ /CS104Resources/Tutorials/Tut_01_Unix_1/unix
→ ls -R Vehicles
```

METHOD - 3

```
Vehicles:
Aeroplane  Cars  Ships  Trains
```

```
Vehicles/Aeroplane:
```

```
Vehicles/Cars:
Audi  BMW  Jaguar
```

```
Vehicles/Cars/Audi:
```

```
Vehicles/Cars/BMW:
```

```
Vehicles/Cars/Jaguar:
```

```
Vehicles/Ships:
```

```
Vehicles/Trains:
```

Exercise 2



From the `file-analysis/demo/code` directory move the file `hello.c` in `file-analysis/dir1` directory, help him move the code file to the right location.

Note: The file-analysis directory is in the main directory provided.

```
file-analysis
├── bigfile
├── commands.sh
├── demo
│   ├── code
│   │   ├── hello
│   │   └── hello.c
│   └── doc
└── dir1
    ├── file1
    ├── file2
    └── file3
```

Solution 2

1. Now it's a simple application of `mv` command, we move the file according to the path provided.

```
AJ ~/CS104Resources/Tutorials/Tut_01_Unix_1/unix
→ mv file-analysis/demo/code/hello.c file-analysis/dir1/
```

```
AJ ~/CS104Resources/Tutorials/Tut_01_Unix_1/unix
→ ls -R file-analysis/
file-analysis/:
bigfile      demo  fruits1  fun_dir  HELLO.c  list1    smallfile
commands.sh  dir1  fruits2  hello.c  list     oddball  students.csv

file-analysis/demo:
code  doc

file-analysis/demo/code:
hello

file-analysis/demo/doc:

file-analysis/dir1:
file1 file2 file3 hello.c
```

Exercise 3



In the **photos** directory remove **all jpg** files that contains **pixabay** in the name.

Solution 3

```

AJ ~/CS104Resources/Tutorials/Tut_01_Unix_1/unix/photos
→ ls
pexels-chevanon-1108099.jpg      pexels-pixabay-158028.jpg
pexels-jmark-250591.jpg         pexels-pixabay-236599.jpg
pexels-mali-142497.jpg          pexels-pixabay-247599.jpg
pexels-mariannaole-757889.jpg    pexels-pixabay-47547.jpg
pexels-philippedonn-1133957.jpg  pexels-rahulp9800-1212487.jpg
AJ ~/CS104Resources/Tutorials/Tut_01_Unix_1/unix/photos
→ rm *pixabay*.jpg
AJ ~/CS104Resources/Tutorials/Tut_01_Unix_1/unix/photos
→ ls
pexels-chevanon-1108099.jpg      pexels-mariannaole-757889.jpg
pexels-jmark-250591.jpg         pexels-philippedonn-1133957.jpg
pexels-mali-142497.jpg          pexels-rahulp9800-1212487.jpg

```



1. First we change directory to **Photos** folder using the **cd** command. (This step was not necessary but for the sake of demonstration let's do it)
2. Now we will use **rm** command using wildcard ***pixabay*.jpg** that will remove all .jpg files containing pixabay.

Exercise 4



There's a **students.csv** in the **file-analysis** directory containing the list of students who have registered for this course. Find the **last 3 entries** from the file.

Solution 4

```
 AJ  ~/CS104Resources/Tutorials/Tut_01_Unix_1/unix  
○ → tail -n 3 file-analysis/students.csv  
217,24B1099,Arnav Baranwal,0,Credit,Approved,Active,H16  
218,24B1100,Niyam Shyam Kotian,0,Credit,Approved,Active,H1  
219,24B1101,Ganipisetty Mahaswi,0,Credit,Approved,Active,H16
```

There are 2 ways to do it

1. If you are too free then maybe use the cat command and get to the end of the file.
2. If you have attended this tutorial attentively, then we covered a command whose functionality is exactly what is asked for, yes it's the tail command.

Thinking time: What if the question was to get the lines numbered 20510-20520 in a file with 50000 lines, how will you find those lines using head and tail? We will cover this in the next tutorial.



Thank You !!!