



Tutorial 1: Unix Command Line (I)

CS 108

Spring, 2023-24

TA: 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

pwd

```
~ /Desktop/cs104/tutorials/tutorial_1 12:54:24 AM  
> pwd  
/Users/guramrit/Desktop/cs104/tutorials/tutorial_1
```

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

`/Users/guramrit/Desktop/cs104/tutorials/tutorial_1`

This will be our working directory for this tutorial :)

ls

```
~/Desktop/cs104/tutorials/tutorial_1 01:58:08 AM
> ls
emails.txt  students.txt
```

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.

```
~/Desktop/cs104/tutorials/tutorial_1 01:58:17 AM
> ls -la ..
drwxr-xr-x guramrit staff 96 B Wed Dec 13 23:34:26 2023 .
drwxr-xr-x guramrit staff 224 B Tue Dec 12 23:09:18 2023 ..
drwxr-xr-x guramrit staff 128 B Thu Dec 14 01:52:07 2023 tutorial_1
```

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

```

[ Apple Desktop ~/Desktop/cs104/tutorials/tutorial_1 ..... 02:10:57 AM ]
[ > ls ]
[ emails.txt students.txt ]

[ Apple Desktop ~/Desktop/cs104/tutorials/tutorial_1 ..... 02:11:03 AM ]
[ > mkdir test ]

[ Apple Desktop ~/Desktop/cs104/tutorials/tutorial_1 ..... 02:11:13 AM ]
[ > ls ]
[ emails.txt students.txt test ]

[ Apple Desktop ~/Desktop/cs104/tutorials/tutorial_1 ..... 02:11:14 AM ]
[ > cd test ]

[ Apple Desktop ~/Desktop/cs104/tutorials/tutorial_1/test .. 02:11:17 AM ]
[ > ls ]

[ Apple Desktop ~/Desktop/cs104/tutorials/tutorial_1/test .. 02:11:18 AM ]
[ > pwd ]
/Users/guramrit/Desktop/cs104/tutorials/tutorial_1/test

[ Apple Desktop ~/Desktop/cs104/tutorials/tutorial_1/test .. 02:11:41 AM ]
[ > cd .. ]

[ Apple Desktop ~/Desktop/cs104/tutorials/tutorial_1 ..... 02:11:45 AM ]
[ > pwd ]
/Users/guramrit/Desktop/cs104/tutorials/tutorial_1
```

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 emails.txt file into test.
 - Then, we copied students.txt file into test.
 - Then, we renamed test to mails.

```
~/Desktop/cs104/tutorials/tutorial_1 02:18:43 AM
> ls
emails.txt  students.txt  test

~/Desktop/cs104/tutorials/tutorial_1 02:19:48 AM
> mv emails.txt test

~/Desktop/cs104/tutorials/tutorial_1 02:20:03 AM
> ls -R
students.txt  test

./test:
emails.txt

~/Desktop/cs104/tutorials/tutorial_1 02:20:12 AM
> cp students.txt test

~/Desktop/cs104/tutorials/tutorial_1 02:20:24 AM
> ls -R
students.txt  test

./test:
emails.txt  students.txt

~/Desktop/cs104/tutorials/tutorial_1 02:20:26 AM
> mv test mails

~/Desktop/cs104/tutorials/tutorial_1 02:20:43 AM
> ls -R
mails  students.txt

./mails:
emails.txt  students.txt
```


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**,
 - First we removed emails.txt file from mails.
 - Then, we tried to remove mails, but failed because mails was not empty.
 - So we removed students.txt file from mails.
 - Finally, we removed mails.

```
~/Desktop/cs104/tutorials/tutorial_1 02:27:23 AM
> ls -R
mails students.txt

./mails:
emails.txt students.txt

~/Desktop/cs104/tutorials/tutorial_1 02:27:31 AM
> rm mails/emails.txt

~/Desktop/cs104/tutorials/tutorial_1 02:27:40 AM
> ls -R
mails students.txt

./mails:
students.txt

~/Desktop/cs104/tutorials/tutorial_1 02:27:46 AM
> rmdir mails
rmdir: mails: Directory not empty

~/Desktop/cs104/tutorials/tutorial_1 02:27:54 AM
> rm mails/students.txt

~/Desktop/cs104/tutorials/tutorial_1 02:28:03 AM
> ls -R
mails students.txt

./mails:

~/Desktop/cs104/tutorials/tutorial_1 02:28:07 AM
> rmdir mails

~/Desktop/cs104/tutorials/tutorial_1 02:28:13 AM
> ls -R
students.txt
```

man

```
~ / Desktop / cs104 / tutorials / tutorial_1 02:43:50 AM  
[ > man ls ]
```

The **man** command is used to display documentation pages. In this **example**, we used the command **man ls** to get the manual page for **ls**.

Exercise: Try “*man man*” :)

```
LS(1)                                General Commands Manual                                LS(1)  
  
NAME  
    ls - list directory contents  
  
SYNOPSIS  
    ls [-@ABCFGHILOPRSTUwabcdeghiklmnopqrstuvwx1%,] [--color=when]  
        [-D format] [file ...]  
  
DESCRIPTION  
    For each operand that names a file of a type other than  
    directory, ls displays its name as well as any requested,  
    associated information. For each operand that names a file of  
    type directory, ls displays the names of files contained within  
    that directory, as well as any requested, associated  
    information.  
  
    If no operands are given, the contents of the current directory  
    are displayed. If more than one operand is given, non-directory  
    operands are displayed first; directory and non-directory  
    operands are sorted separately and in lexicographical order.
```

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 `students.txt` file to terminal.

```
~ / Desktop / cs104 / tutorials / tutorial_1 02:52:11 AM
> ls
students.txt

~ / Desktop / cs104 / tutorials / tutorial_1 02:52:12 AM
> cat students.txt
ID,Name,E-mail,Gender,Year,Department
210050061,Guramrit Singh,guramrit@cse.iitb.ac.in,M,2021,CSE
200071030,Akshay Kumar,akshay@ee.iitb.ac.in,M,2020,EE
210260200,Kiara Advani,kiara@ep.iitb.ac.in,F,2021,EP
22b1053,Kavya Gupta,kforkavya@cse.iitb.ac.in,M,2022,CSE
22b1003,Saksham Rathi,sakshamrathi@cse.iitb.ac.in,M,2022,CSE
22b9999,Rashmika Mandanna,rashmika@ee.iitb.ac.in,F,2022,EE
22b9090,Harmanpreet Kaur,harman@me.iitb.ac.in,F,2022,ME
```

clear

```
~/Desktop/cs104/tutorials/tutorial_1 ..... 02:52:11 AM
> ls
students.txt

~/Desktop/cs104/tutorials/tutorial_1 ..... 02:52:12 AM
> cat students.txt
ID,Name,E-mail,Gender,Year,Department
210050061,Guramrit Singh,guramrit@cse.iitb.ac.in,M,2021,CSE
200071030,Akshay Kumar,akshay@ee.iitb.ac.in,M,2020,EE
210260200,Kiara Advani,kiara@ep.iitb.ac.in,F,2021,EP
22b1053,Kavya Gupta,kforkavya@cse.iitb.ac.in,M,2022,CSE
22b1003,Saksham Rathi,sakshamrathi@cse.iitb.ac.in,M,2022,CSE
22b9999,Rashmika Mandanna,rashmika@ee.iitb.ac.in,F,2022,EE
22b9090,Harmanpreet Kaur,harman@me.iitb.ac.in,F,2022,ME

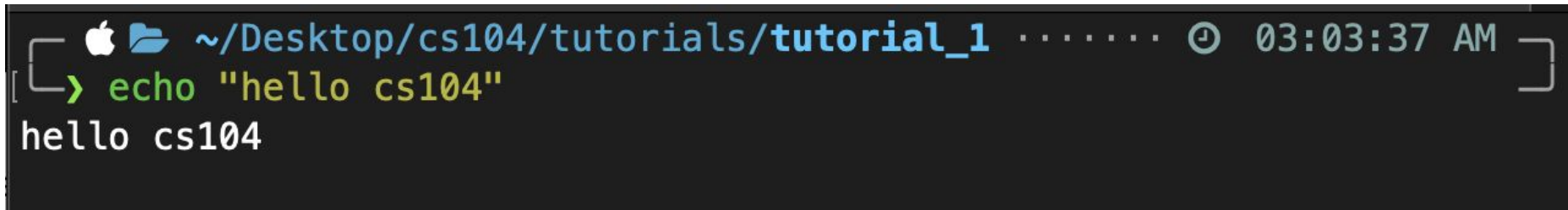
~/Desktop/cs104/tutorials/tutorial_1 ..... 02:52:15 AM
> clear
```

Before clear

```
~/Desktop/cs104/tutorials/tutorial_1 ..... 02:53:49 AM
> 
```

After clear (It cleared the terminal screen)

echo

A screenshot of a macOS terminal window. The title bar shows an Apple icon, a folder icon, and the path ~/Desktop/cs104/tutorials/tutorial_1. The terminal content shows a green prompt character followed by the command echo "hello cs104" in green text. Below the command, the output hello cs104 is displayed in white text. The terminal has a dark background and a light gray border.

```
~/Desktop/cs104/tutorials/tutorial_1 03:03:37 AM  
[> echo "hello cs104"  
hello cs104
```

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

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 `students.txt` file to `terminal`.
- Checkout options: `-c`

```
~ /Desktop/cs104/tutorials/tutorial_1 ..... 03:01:45 AM
> cat students.txt
ID,Name,E-mail,Gender,Year,Department
210050061,Guramrit Singh,guramrit@cse.iitb.ac.in,M,2021,CSE
200071030,Akshay Kumar,akshay@ee.iitb.ac.in,M,2020,EE
210260200,Kiara Advani,kiara@ep.iitb.ac.in,F,2021,EP
22b1053,Kavya Gupta,kforkavya@cse.iitb.ac.in,M,2022,CSE
22b1003,Saksham Rathi,sakshamrathi@cse.iitb.ac.in,M,2022,CSE
22b9999,Rashmika Mandanna,rashmika@ee.iitb.ac.in,F,2022,EE
22b9090,Harmanpreet Kaur,harman@me.iitb.ac.in,F,2022,ME
~ /Desktop/cs104/tutorials/tutorial_1 ..... 03:01:56 AM
> head -n 3 students.txt
ID,Name,E-mail,Gender,Year,Department
210050061,Guramrit Singh,guramrit@cse.iitb.ac.in,M,2021,CSE
200071030,Akshay Kumar,akshay@ee.iitb.ac.in,M,2020,EE
```


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.
- In the **example**, we used **tail** to print last 3 lines of **students.txt** file to **terminal**.
- Checkout options: -v

```
~ /Desktop/cs104/tutorials/tutorial_1 ..... 03:06:02 AM  
> cat students.txt  
ID,Name,E-mail,Gender,Year,Department  
210050061,Guramrit Singh,guramrit@cse.iitb.ac.in,M,2021,CSE  
200071030,Akshay Kumar,akshay@ee.iitb.ac.in,M,2020,EE  
210260200,Kiara Advani,kiara@ep.iitb.ac.in,F,2021,EP  
22b1053,Kavya Gupta,kforkavya@cse.iitb.ac.in,M,2022,CSE  
22b1003,Saksham Rathi,sakshamrathi@cse.iitb.ac.in,M,2022,CSE  
22b9999,Rashmika Mandanna,rashmika@ee.iitb.ac.in,F,2022,EE  
22b9090,Harmanpreet Kaur,harman@me.iitb.ac.in,F,2022,ME  
~ /Desktop/cs104/tutorials/tutorial_1 ..... 03:06:03 AM  
> tail -n 3 students.txt  
22b1003,Saksham Rathi,sakshamrathi@cse.iitb.ac.in,M,2022,CSE  
22b9999,Rashmika Mandanna,rashmika@ee.iitb.ac.in,F,2022,EE  
22b9090,Harmanpreet Kaur,harman@me.iitb.ac.in,F,2022,ME
```

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 ...



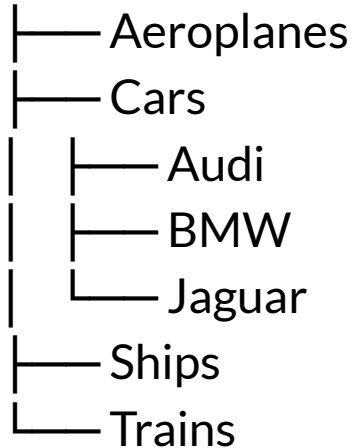
Exercises

Exercise 1



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

Vehicles



Solution 1



One solution is to create all directories one by one.

An alternative is to generate multiple directories simultaneously, rather than individually creating each one separately.

Notice the use of {}.

```
~ /Desktop/cs104/tutorials/tutorial_1 ..... 03:23:46 AM
> ls -R
~ /Desktop/cs104/tutorials/tutorial_1 ..... 03:23:53 AM
> mkdir Vehicles
~ /Desktop/cs104/tutorials/tutorial_1 ..... 03:23:59 AM
> mkdir Vehicles/{Cars,Trains,Aeroplanes,Ships}
~ /Desktop/cs104/tutorials/tutorial_1 ..... 03:24:41 AM
> mkdir Vehicles/Cars/{Audi,BMW,Jaguar}
~ /Desktop/cs104/tutorials/tutorial_1 ..... 03:25:03 AM
> ls -R
Vehicles
./Vehicles:
Aeroplanes Cars Ships Trains
./Vehicles/Aeroplanes:
./Vehicles/Cars:
Audi BMW Jaguar
./Vehicles/Cars/Audi:
./Vehicles/Cars/BMW:
./Vehicles/Cars/Jaguar:
./Vehicles/Ships:
./Vehicles/Trains:
```

Exercise 2



An employee mistakenly placed `A7.png` in `Trains` directory, help him move the image to the right location.

(Well if you don't know, then keep in mind that A7 is an Audi car)

Solution 2

1. First we change directory into **Vehicles** folder using the **cd** command. (This step was not necessary but for the sake of demonstration let's do it)
2. Now it's a simple application of **mv** command, we move the **A7.png** from **Trains** directory to **Cars/Audi** directory.

```
~/Desktop/cs104/tutorials/tutorial_1 ..... 03:47:06 AM
> cd Vehicles
~/Desktop/cs104/tutorials/tutorial_1/Vehicles ..... 03:47:12 AM
> ls -R
Aeroplanes Cars Ships Trains

./Aeroplanes:

./Cars:
Audi BMW Jaguar

./Cars/Audi:

./Cars/BMW:

./Cars/Jaguar:

./Ships:

./Trains:
A7.png
~/Desktop/cs104/tutorials/tutorial_1/Vehicles ..... 03:47:20 AM
> mv Trains/A7.png Cars/Audi
~/Desktop/cs104/tutorials/tutorial_1/Vehicles ..... 03:47:41 AM
> ls -R
Aeroplanes Cars Ships Trains

./Aeroplanes:

./Cars:
Audi BMW Jaguar

./Cars/Audi:
A7.png

./Cars/BMW:

./Cars/Jaguar:

./Ships:

./Trains:
```

Exercise 3



Audi has decided to discontinue its C model line, so the employer has asked you to remove **all png** files that belonged to the C model line.

Hint: You need to delete all png images in Audi folder that **begins with the letter C**.

Solution 3

```
🍏 ~/Desktop/cs104/tutorials/tutorial_1 ..... ⌚ 04:06:24 AM  
[> cd Vehicles/Cars/Audi  
🍏 ~/Desktop/cs104/tutorials/tutorial_1/Vehicles/Cars/Audi .... ⌚ 04:06:58 AM  
[> ls  
🖼️ A1.png 🖼️ A5.png 🖼️ A9.png 🖼️ C3.png 🖼️ C7.png  
🖼️ A3.png 🖼️ A7.png 🖼️ C1.png 🖼️ C5.png 🖼️ C9.png  
🍏 ~/Desktop/cs104/tutorials/tutorial_1/Vehicles/Cars/Audi .... ⌚ 04:07:00 AM  
[> rm C*.png  
🍏 ~/Desktop/cs104/tutorials/tutorial_1/Vehicles/Cars/Audi .... ⌚ 04:07:07 AM  
[> ls  
🖼️ A1.png 🖼️ A3.png 🖼️ A5.png 🖼️ A7.png 🖼️ A9.png
```

1. First we change directory to **Audi** 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 **C*.png** that will remove all .png files beginning with character C.

Exercise 4



Given the file `sold.txt` in the `Vehicles` directory.

This file contains information about all vehicles sold. As you would expect, this is a very long file.

Your employer has assigned you the task of finding the details of the last car that was sold.

Solution 4

```
~/Desktop/cs104/tutorials/tutorial_1/Vehicles 04:24:49 AM
> cat sold.txt
Company,Model,Transmission,Year,Engine,Doors
Audi,A3,Manual,2020,1500 (cc),5 doors
BMW,320d,Manual,2022,2000 (cc),5 doors
Jaguar,XE,Manual,2022,2000 (cc),5 doors
Audi,Q3,Manual,2023,2000 (cc),5 doors
Audi,Q5,Manual,2023,2000 (cc),5 doors
BMW,X1,Manual,2024,2000 (cc),5 doors
BMW,X3,Manual,2024,2000 (cc),5 doors
~/Desktop/cs104/tutorials/tutorial_1/Vehicles 04:24:50 AM
> tail -n 1 sold.txt
BMW,X3,Manual,2024,2000 (cc),5 doors
```

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 !!!