



# Linux Command Line Cheat Sheet

-l (small L letter): long listing format

## Is Command for Listing Files

ls ~ List files in the home directory

ls List files in the current working directory

ls ../ List parent directory

ls ../../ List the contents of the directory one level above

ls -a List all files including hidden files starting with '.'.

ls -l The long listing of the content of the current directory

ls -d \*/ List directory entries only

ls -r Reverse order while sorting

ls -R List subdirectories recursively

ls -lt Sort the list by modification time (newest first) [long listing format]

ls -t Sort the list by modification time (newest first)

ls -lX Sort the list by extension [long listing format]

ls -X Sort the list by extension

ls -li Print the index number (or inode number) of each file

ls -lh Check the size of files and directories in a human-readable format

ls -lhS Sort file size (largest file size first) [long listing format]

ls -p Add / (slash sign) to mark directory

ls -F Add / (slash sign) to mark directory and \* sign to indicate executable files

ls -m Print out all directories and files separated by a comma.

ls -Q Print out all directories and files in the directory with quotation marks.

ls -g Print entries without owner information

ls \*k Search files ending with the letter 'k' (for example)

ls seq\* Search files with characters 'seq' (for example) in the beginning

## Linux Directory Structure

/ The root directory

/bin Essential user binaries (programs). Important system programs and utilities are located in /bin

/boot Contains the files needed to boot the system

/media The standard location for temporary media [subdirectories within /media]

/dev Contains a number of special files that represent devices

/etc Contains configuration files



# Linux Command Line Cheat Sheet

**/home** Contains a home folder for each user

**/lib** Contains libraries needed by the essential binaries

**/lost+found** Contains recovered files [important when file system crashes]

**/mnt** Temporary mount points

**/opt** Contains subdirectories for optional software packages

**/proc** contains special files that represent system and process information

**/root** The home directory of the root user

**/run** A standard place to store transient files

**/sbin** System administration binaries

**/tmp** Contains temporary files

**/usr** Contains applications and files used by users [read-only in normal operation]

**/var** Contains applications and files used by users [writable]

## Directory operations

**pwd** displays the present working directory

**mkdir** Create a new directory.  
**\$ mkdir Uresearcher**

**mkdir -p** Create nested directories.  
**\$ mkdir -p Uresearcher /hello/researcher**

## Be careful while performing delete operations

**rmdir** Remove/delete an existing directory [provided it is empty]  
**\$ rmdir Uresearcher**

**rm -r** Deletes a directory recursively along with its content  
**\$ rm -r Uresearcher**

**rm -rf** Forcefully and recursively deletes a directory along with its content  
**\$ rm -rf Uresearcher**

**rm filename** Forcefully deletes a file  
**\$ rm -f cheatsheet.pdf**

**rm -f filename** Forcefully deletes a file  
**\$ rm -f cheatsheet.pdf**

## Change Directory (cd)

**cd** Change from the current working directory to the home directory

**cd directory** Change from the current working directory to a specified directory  
**\$ cd Uresearcher**

**cd -** Change from the current working directory to the previous working directory

**cd ..** Change Current directory to parent directory

**cd ../** Move one directory lower from the current working directory

**cd 'dir name with space'** Navigate to directories with space in their names  
**\$ cd dir\ name\ with\ space**



# Linux Command Line Cheat Sheet

## Search and Locate Files and Directories

find .  
-name  
filename  
Find files using the name  
[Case-sensitive] in the current  
directory  
**\$ find . -name uresearcher.txt**

find /home  
-name  
filename  
Find files using the name  
[Case-sensitive] available  
under the home directory  
**\$ find /home -name  
uresearcher.txt**

find .  
-iname  
filename  
Find files using the name  
and ignoring the case  
**\$ find . -iname cheatsheet.pdf**

find /  
-type d  
-name  
directory  
Find all directories with  
directory name in / directory  
**\$ find / -type d -name  
Uresearcher**

find . -type  
f -name  
filename.  
ext  
Find all files with name  
and extension in a current  
working directory  
**\$ find . -type f -name  
cheatsheet.pdf**

find  
-type  
f -name  
"\*.\*"  
Find all files in a directory  
with a certain extension  
(e.g. PDF) **\$ find . -type f  
-name "\*.pdf"**

find  
-type f  
-name  
"filename  
.ext"  
-exec rm  
-f {} \;  
Find and remove a single file  
**\$ find . -type f -name  
"cheatsheet.pdf" -exec  
rm -f {} \;**

find  
-type f  
-name  
"\*.\*"  
-exec  
rm -f {} \;  
Find and remove multiple  
files **\$ find . -type f -name  
"\*.\*" -exec rm -f {} \;**

find /tmp  
-type  
f -empty  
To find all empty files under  
a certain path

find /tmp  
-type d  
-empty  
To find all empty directories  
under a certain path.

## Find a Specific String or Word in Files and Directories

grep -Rw  
directory  
-e "string"  
Recursively and aggressively  
search a directory (e.g. ~/Uresearcher/) for a string  
**"cheatsheet". \$ grep -Rw ~/Uresearcher/ -e "cheatsheet"**

grep -Rnw  
directory  
-e "string"  
To know the exact line  
where the string of text  
exist **\$ grep -Rnw ~/Uresearcher/ -e  
"cheatsheet"**

grep -Rnw  
--include  
=\*.sh  
directory  
-e "string"  
Search string in a special  
type of file (for example: pdf)  
**\$ grep -Rnw --include=\*.pdf  
~/Uresearcher/ -e "cheatsheet"**

grep  
-Rnw  
--include  
=\*.sh  
directory  
-e  
"string1"  
-e  
"string2"  
Search for more than one  
string patterns  
**\$ grep -Rnw --include=\*.pdf  
~/Uresearcher/ -e  
"cheatsheet" -e "linux"**

## Handling compressed archive files (create, extract, view, modify archive files)

tar  
-cvf name  
\_of  
\_compressed  
\_file.tar to  
\_be  
\_compressed  
\_directory/  
Create a tar archive file (e.g.  
uresearcher.tar) from a  
directory (~/Uresearcher/)  
**\$ tar -cvf uresearcher.tar  
~/Uresearcher/**



# Linux Command Line Cheat Sheet

<code>tar cvzf name_of _compressed _file .tar to_be_com- pressed_ directory/</code>	Create a gzip archive file (e.g. uresearcher.tar) from a directory (~/.Uresearcher/) <b>\$ tar cvzf uresearcher.tar.gz ~/.Uresearcher/</b> OR <b>\$ tar cvzf uresearcher.tgz ~/.Uresearcher/</b>	<code>tar -tvf compressed _file</code>	List content of tar archive file in Linux \$ tar -tvf uresearcher.tar <b>Note:</b> -tvf is common option for listing content of all formats (e.g. tar, tar.gz, tgz, tar.bz2, tar.tbz, tar.tb2)	<b>Note:</b> Group of files can be extracted using the same way and wildcards (e.g. extract all pdf files)  <b>\$ tar -jxvf uresearcher.tar.gz "*.pdf"</b>
<code>tar cvfj to_be_ compressed _file.tar to _be_ compressed _directory/</code>	Create a bz2 archive file (e.g. uresearcher.tar) from a directory (~/.Uresearcher/)  <b>\$ tar cvfj uresearcher .tar.bz2 ~/.Uresearcher/</b> OR <b>\$ tar cvfj uresearcher. tar.tbz ~/.Uresearcher/</b>  OR <b>\$ tar cvfj uresearcher .tar.tb2 ~/.Uresearcher/</b>	<code>tar -xvf compressed _file filename</code>	Untar a single file from tar file in Linux <b>\$ tar -xvf uresearcher.tar cheatsheet.pdf</b>	
<code>tar -zxvf compressed _file filename</code>		<code>tar -zxvf compressed _file filename</code>	Untar a single file from tar.gz file in Linux <b>\$ tar -zxvf uresearcher .tar.gz cheatsheet.pdf</b>	
<code>tar -jxvf compressed _file filename</code>		<code>tar -jxvf compressed _file filename</code>	Untar a single file from tar.bz2 file in Linux <b>\$ tar -jxvf uresearcher .tar.gz cheatsheet.pdf</b>  <b>Note:</b> For untaring multiple files from any of these file formats, use their respective option (e.g. -xvf, -zxvf, -jxvf) and "file1" "file 2"  <b>\$ tar -jxvf uresearcher.tar .gz "cheatsheet.pdf" "cheatsheet.txt"</b>	
<code>tar -xvf compressed _file</code>	To untar or extract a tar file <b>\$ tar -xvf uresearcher .tar</b>			<code>tar -rvf compressed _file filename</code>  Append new files/ directories to the already compressed file <b>\$ tar -rvf uresearcher.tar newcheatsheet.pdf</b> Note: This command cannot be used for an existing tar.gz or tar.bz2 files
<code>tar -xvf compressed _file -C specified _directory</code>	To untar or extract a tar file \$ tar -xvf uresearcher .tar -C ~/newdata  <b>Note:</b> -xvf is common option for extracting all formats (e.g. tar, tar.gz, tgz, tar.bz2, tar.tbz, tar.tb2)			<code>tar -czf compressed _file   wc -c</code>  Check the size of the tar, tar.gz, and tar.bz2 archive file <b>\$ tar -czf uresearcher .tar   wc -c</b>



# Linux Command Line Cheat Sheet

## Split and Join commands for managing large files

**split file.txt** Split a large file (e.g. 5000 lines) into smaller files (by default 1000 lines)  
**\$ split file.txt**  
Output: xaa xab xac xad xae

**split -a4 file.txt** Change the default suffix length of 2 to user-specified  
**\$ split -a4 file.txt**  
Output: xaaaaa xaaaab xaaaac xaaaad xaaaee

**split -b2000 file.txt** Split a larger file (e.g. 10000 bytes) into smaller files (e.g. 2000 bytes) by size  
**\$ split -b2000 file.txt**

**split -b 20M file.txt** Split a larger file into smaller files by size in MB (e.g. 20 MB each)  
**\$ split -b 20M file.txt**

**split -d file.txt** Split files with a numeric suffix  
**\$ split -d4 file.txt**  
Output: x00 x01 x02 x03 x04

**split -500 file.txt** Split files with a specified number of lines  
**\$ split -500 file.txt**  
Output: xaa xab xac xad xae xaf xag xah xai xaj

**split -n10 file.txt** Split files with a specified number of chunks  
**\$ split -n10 file.txt**  
Output: xaa xab xac xad xae xaf xag xah xai xaj

**split -n10 -e file.txt** Split files with a specified number of chunks with no zero size chunk  
**\$ split -n10 -e file.txt**  
Output: xaa xab xac xad xae xaf xag xah xai xaj

**cat file1.txt file2.txt > new-file.txt** To concatenate two (or more files) and write the output to a new file  
**\$ cat file1.txt file2.txt > newfile.txt**

**join file1.txt file2.txt > new-file.txt** To join two (or more files) and write the output to a new file  
**\$ join file1.txt file2.txt > newfile.txt**

## To execute multiple commands at once with a text file

**sh script.sh** To execute multiple commands at once with each command written in a new line (sequential ordering) in the text file (e.g. multicommands.sh)  
**\$ sh multicommands.sh**

**Note:** To run an executable, first do **chmod +x** and then run the **sh script.sh** command  
**\$ chmod +x multicommands.sh**

**sh script.sh &> output.log** To execute the commands and log the output/error to a file  
**\$ sh multicommands.sh &> output.log]**

**sh -x script.sh &> output.log** To execute the commands and log each command as it is executed  
**\$ sh -x multicommands.sh &> output.log**

**sh -e script.sh &> output.log** To execute the commands and abort on encountering an error  
**\$ sh -e multicommands.sh &> output.log**

## About Uresearcher:

Uresearcher is a research education company. Our mission is to provide a cutting-edge research curriculum, high-quality teaching, and 24×7 scientific support accessible to anyone, at any time, anywhere in the world. This cheat sheet is made available under the Research Education for Everyone initiative.

**Know more:** <https://uresearcher.com/>

Copyright©Uresearcher Growth Labs  
Published on: 20th September 2021