

R² Academy



Command Line Crash Course



- [Website](#)
- [Free Online R Courses](#)
- [R Packages](#)
- [Shiny Apps](#)
- [Blog](#)
- [GitHub](#)
- [YouTube](#)
- [Twitter](#)
- [Facebook](#)
- [Linkedin](#)

- Learning Management System
 - Videos
 - Slides
 - Code
 - Data
 - Suggested Readings
- Need help?
 - Discussion forum
 - support@rsquaredacademy.com
 - +91-9606872504
 - Raise a [ticket](#)

- [Slides](#)
- [Code & Data](#)
- [RStudio Cloud](#)
- [Online Course](#)

- Go through the videos
- Run the code (locally or on RStudio Cloud)
- Experiment with the code
- Go through the suggested readings
- If you get stuck with the code or the R packages
 - Try googling the error shown by R
 - Search [StackOverflow](#)
 - If related to RStudio, ask for help on [RStudio Community](#)
- Mail us @ support@rsquaredacademy.com if you have any questions (**We believe no question is naive and every question is important.**) and we will try to reply at the earliest.
- Use the discussion forum

- What is Shell/Terminal?
- Applications
- Launch Terminal/Shell
 - mac
 - Windows (Including Windows Subsystem for Linux)
 - Linux
- Shell Commands
 - Help/Documentation
 - File management
 - System management
 - Navigating file system
 - File Compression
 - Data Transfer from/to the internet
 - Input/Output
 - Search & Regular expression
- Use RStudio terminal
- Run shell commands from R
- Run shell commands in RMarkdown
- Execute R scripts from the command line

Shell is a text based application for viewing, handling and manipulating files. It takes in commands and passes them on to the operating system. It is also known as:

- CLI (Command Line Interface)
- Terminal
- Bash (Bourne Again Shell)

It is sufficient to know a handful of commands to get started with the shell.



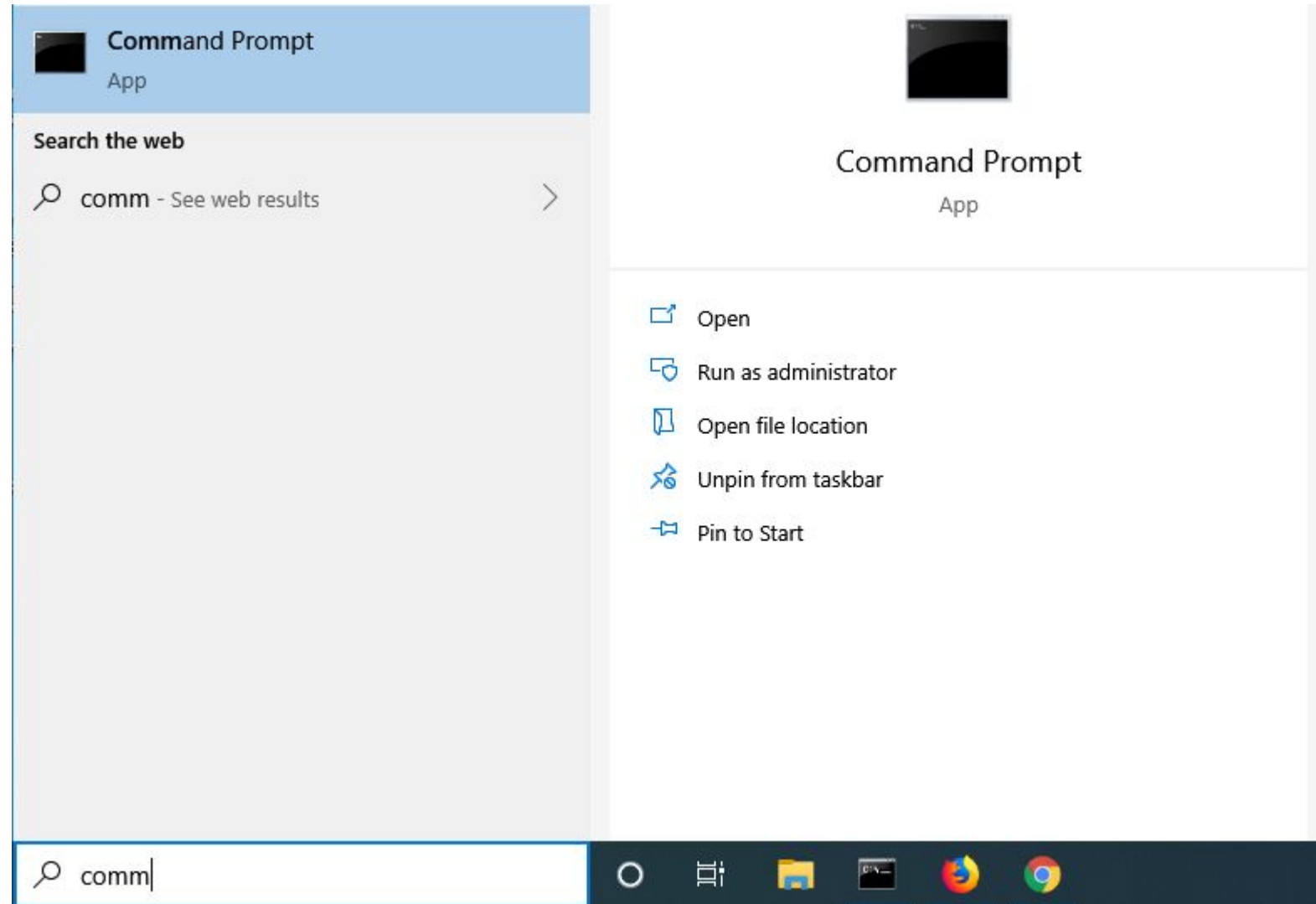
Version Control

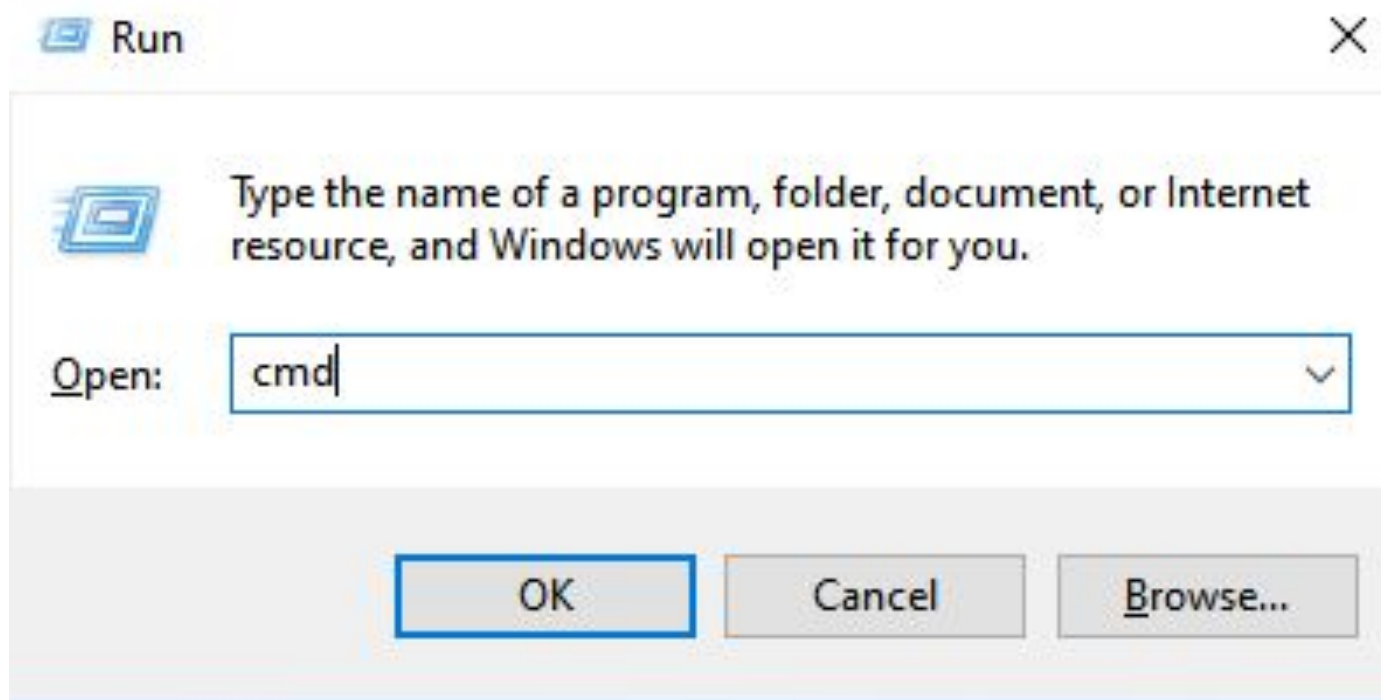


Cloud Services

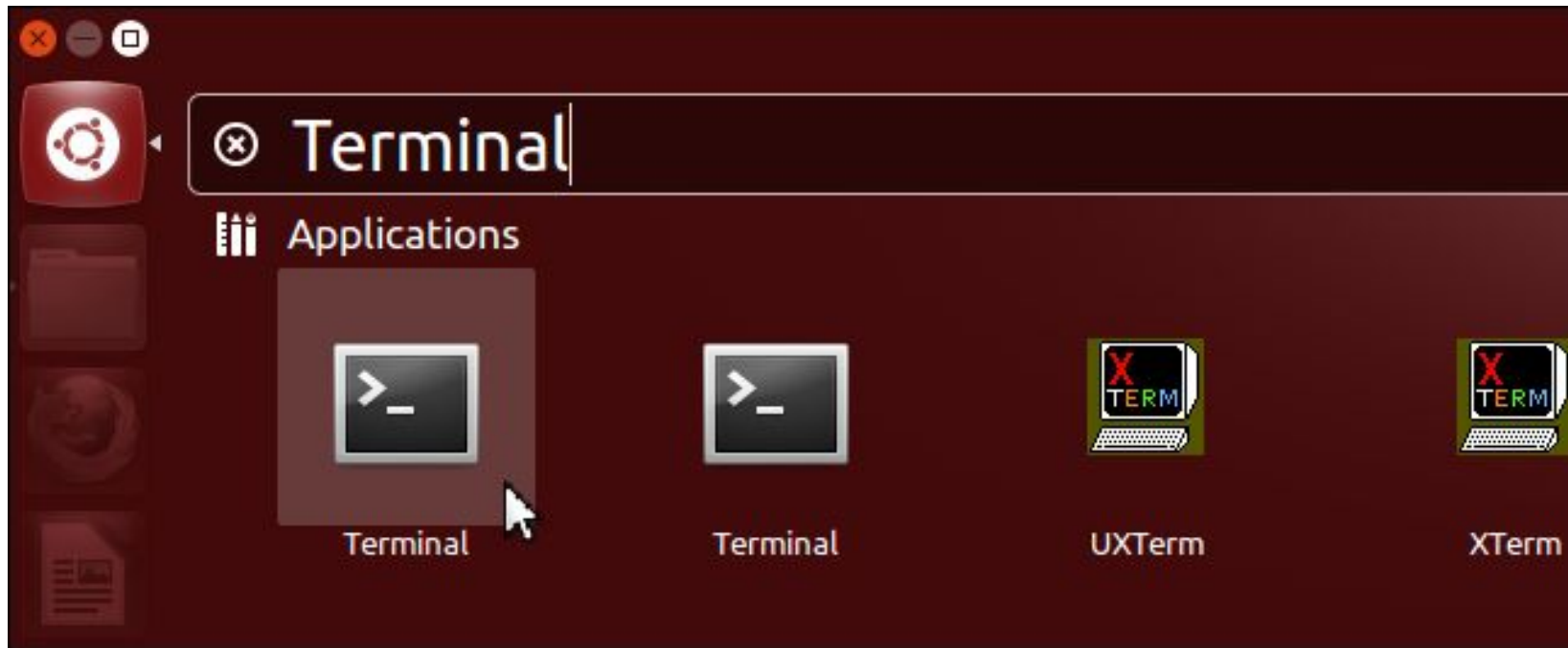
- Applications -> Utilities -> Terminal

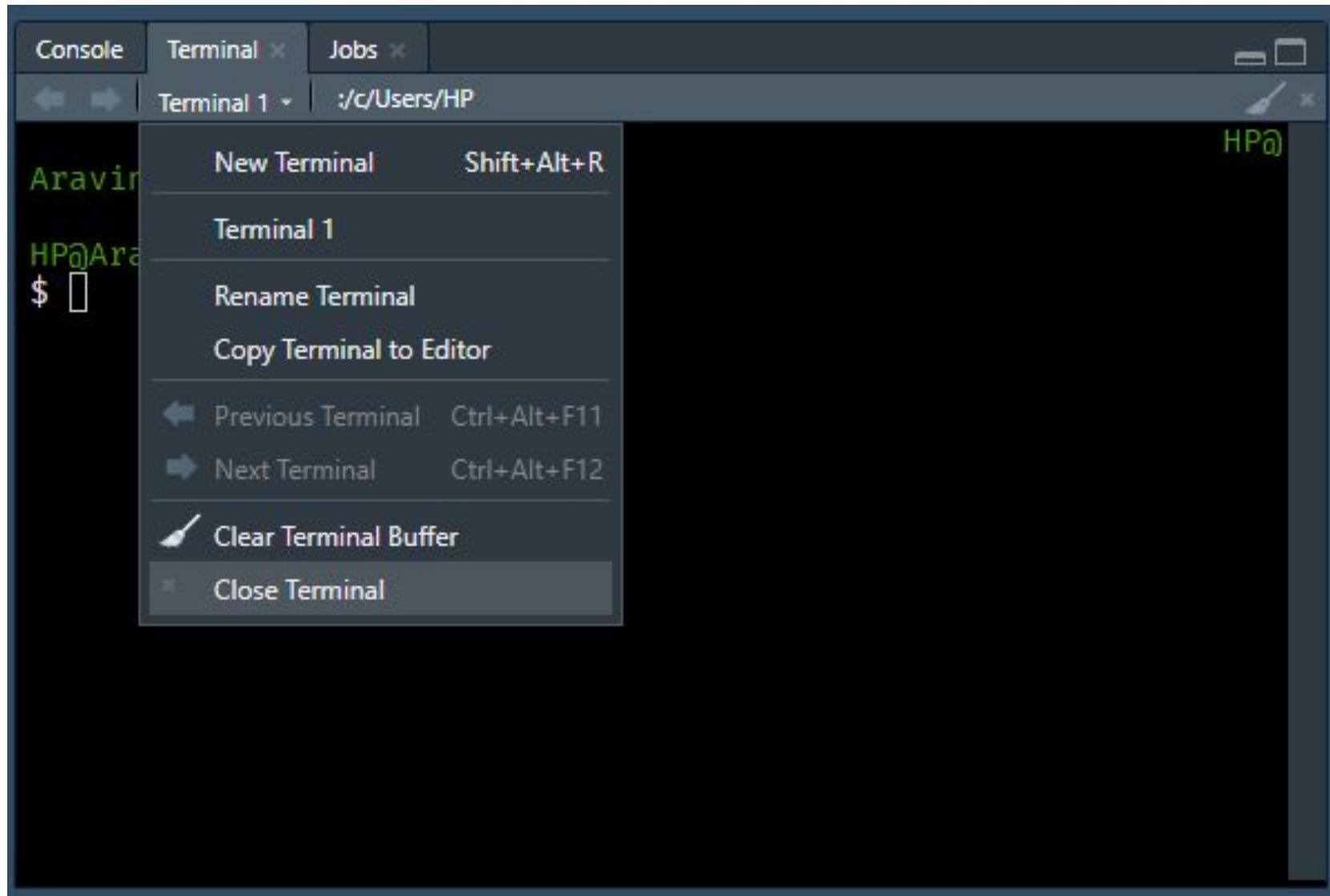






- Applications -> Accessories -> Terminal
- Applications -> System -> Terminal





OS	Prompt
mac	\$
Linux	\$
Windows	>

Command	Description	In R
whoami	Who is the user?	<code>Sys.info()</code> / <code>whoami::whoami()</code>
date	Get date & time	<code>Sys.date()</code> / <code>Sys.time()</code>
cal	Display Calendar	
clear	Clear the screen	<code>Ctrl + L</code>

Command	Description	In R
whatis	Single line description of a command	? /? / ?? / help()
man	Display manual pages for a command	
help		
/?		

Command	Description	In R
pwd	Print working directory	<code>getwd()</code> / <code>here::here()</code>
ls	List directory contents	<code>dir()</code> / <code>list.files()</code> / <code>list.dirs()</code> / <code>dir_info()</code> / <code>fs::dir_ls()</code>
cd	Change current working directory	<code>setwd()</code>
mkdir	Create directory	<code>dir.create()</code> / <code>fs::dir_create()</code>
rmdir	Remove/delete directory	<code>fs::dir_delete()</code>

Command	Description
cd	Navigate into directory
cd ..	Go up one level
cd -	Go to previous working directory
cd ~	Change directory to home directory
cd /	Change directory to root directory

Command	Description
<code>ls</code>	List directory contents
<code>ls -l</code>	List files one per line
<code>ls -a</code>	List all files, including hidden files
<code>ls -la</code>	Display permission, ownership, size & modification date
<code>ls -lh</code>	Long format with size displayed in human readable units
<code>ls -lS</code>	Long format list sorted by size
<code>ls -ltr</code>	Long format list sorted by modification date

Command	Description
touch	Create empty file/change timestamp
cp	Copy file(s)
mv	Rename/move file
rm	Remove/delete file
diff	Compare files

Command	Description
<code>mv</code>	Move or rename files & directories
<code>mv -f</code>	Do not prompt for confirmation before overwriting files
<code>mv -i</code>	Prompt for confirmation before overwriting files
<code>mv -n</code>	Do not overwrite existing files
<code>mv -v</code>	Move files in verbose mode

Command	Description
<code>rm</code>	Remove files or directories
<code>rm -r</code>	Recursively remove a directory & all its subdirectories
<code>rm -rf</code>	Forcibly remove directory without prompting for confirmation or showing error messages
<code>rm -i</code>	Interactively remove multiple files, with a prompt before every removal
<code>rm -v</code>	Remove files in verbose mode, printing a message for each removed file

Command	Description
<code>diff</code>	Compare files & directories
<code>diff -w</code>	Compare files; ignoring white spaces
<code>diff -y</code>	Compare files; showing differences side by side
<code>diff -u</code>	Compare files; show differences in unified format
<code>diff -r</code>	Compare directories recursively
<code>diff -rq</code>	Compare directories; show the names of files that differ

Command	Description
echo	Display messages
cat	Prints content of a file
head	Prints the first 10 lines of a file by default
tail	Prints the last 10 lines of a file by default
more	Display more
less	Display less
sort	Sort a file in ascending order

Command	Description
echo	Print text message
echo -n	Without the trailing newline
echo > file	Write message to a file
echo >> file	Append a message to a file
echo -e	Enable interpretation of special characters

Command	Description
cat	Print and concatenate files
cat >	Concatenate several files into the target file
cat >>	Append several files to the target file
cat -n	Number all output lines

Command	Description
head	Output the first parts of a file
head -n	Output the first n lines of a file
head -c	Output the first c bytes of a file
head -n -x	Output everything but the last x lines of a file
head -c -x	Output everything but the last x bytes of a file

Command	Description
<code>tail</code>	Display the last part of a file
<code>tail -n num</code>	Show the last num lines of a file
<code>tail -n +num</code>	Show all file since num line
<code>tail -c num</code>	Show last num bytes in a file
<code>tail -f</code>	Keep reading file until Ctrl + C
<code>tail -F</code>	Keep reading file until Ctrl + C; even if the file is rotated

Command	Description
more	Open a file for interactive reading; allowing scrolling & search
space	Page down
/search	Search for a string; press n to go to the next match
q	Quit

Command	Description
less	Open a file for interactive reading; allowing scrolling & search
space	Page down
b	Page up
G	Go to end of file
g	Go to start of file
/search	Forward search
?search	Backward search
n	Go to next match
N	Go to previous match
q	Exit

Command	Description
<code>sort</code>	Sort lines of text files
<code>sort -r</code>	Sort a file in descending order
<code>sort --ignore-case</code>	Ignore case
<code>sort -n</code>	Use numeric order for sorting
<code>sort -u</code>	Preserve only unique lines

Command	Description
grep	Search for a given string in a file
find	Find files using filenames
which	Search for binary executable files

Command	Description
grep	Matches patterns in input text
grep -i	Ignore case
grep -RI	Search recursively for an exact string
grep -E	Use extended regular expression
grep -Hn	Print file name & corresponding line number for each match
grep -v	Invert match for excluding specific strings

Command	Description
<code>find</code>	Find files or directories under the given directory; recursively
<code>find -name '*.txt'</code>	Find files by extension
<code>find -type d -iname</code>	Find directories matching a given name, in case-insensitive mode:
<code>find -type d -name</code>	Find directories matching a given name, in case-sensitive mode:

Command	Description	In R
wget	Download from the internet	<code>download.file()</code>
curl	Download files from the internet	<code>curl::curl_download()</code>
hostname	Name of current host	<code>R.utils::getHostname.System()</code>
ping	Ping a remote host	<code>pingr::ping()</code>
nslookup	Name server and ip details	<code>curl::nslookup()</code>

Command	Description
wget	Download files from the web
wget url	Download contents of a url to a file
wget -o file_name url	Download contents of a url to a file
wget -c url	Continue an incomplete download
wget -P folder_name -i urls.txt	Download all urls stored in a text file to a specific directory
wget --limit-rate	Limit download speed
wget --tries	Limit number of retries

Command	Description
<code>curl url</code>	Download the contents of a url
<code>curl url -o file_name</code>	Download the contents of a url to a file
<code>curl url > file_name</code>	

Command	Description
tar	Create, extract or view tar archive
unzip	To extract a zip file
gzip	Compress file

Command	Description
<code>dpkg --get-frontend</code>	List installed packages
<code>sudo apt-get update</code>	Update packages
<code>sudo apt-get install</code>	Install packages
<code>sudo apt-get remove</code>	Remove packages (retain configurations, plugins & settings)
<code>sudo apt-get purge</code>	Remove packages including personalized settings
<code>sudo apt-get autoremove</code>	Remove any dependencies no longer in use

Command	Description	In R
tar tvf	List an archive	<code>utils::untar('archive.tar', list = TRUE)</code>
tar tvfz	List a gzipped archive	<code>utils::untar('archive.tar.gz', list = TRUE)</code>
tar xvf	Extract an archive	<code>utils::untar('archive.tar')</code>
tar xvfz	Extract a gzipped archive	<code>utils::untar('archive.tar.gz')</code>
tar cvf	Create an uncompressed tar archive	<code>utils::tar('archive.tar')</code>
tar cvfz	Create a tar gzipped archive	<code>utils::tar('archive.tar', compression = 'gzip')</code>
tar rvf	Add a file to an existing archive	
tar rvfz	Add a file to an existing gzipped archive	

Command	Description	In R
zip	List an archive	<code>zip::zipr_append()</code>
zip -r	List a gzipped archive	<code>zip::zip()</code>
gzip	Extract an archive	<code>utils::tar(compression = 'gzip')/R.utils::gzip()</code>
gzip -d	Extract a gzipped archive	<code>utils::untar() / R.utils::gunzip()</code>
gzip -c	Create an uncompressed tar archive	<code>utils::untar(exdir = filename)</code>
unzip	Create a tar gzipped archive	<code>zip::unzip()</code>
unzip -d	Add a file to an existing archive	<code>zip::unzip(exdir = dir_name)</code>
unzip -l	Add a file to an existing gzipped archive	<code>zip::zip_list()</code>

Command	Description
uname	Display important information about the system
free	Display free, used, swap memory in the system
df	Display file system disk space usage
exit	Exit the terminal
sudo	Run command as super user
shutdown	Shutdown the system

Command	Description
uname	Print details about the current machine and the operating system running on it
uname -mp	Hardware related information; machine & processor
uname -srv	Software related information; operating system, release number and version
uname -n	Nodename of the system
uname -a	Print all available information system

Command	Description	In R
time	See how long a command takes to execute	
sleep	Delay activity in seconds	Sys.sleep()
sleep 1m	Delay activity in minutes	
sleep 1h	Delay activity in hours	
history	Display command history list with line numbers	history()
history -c	Clear the command history list	

Command	Description
Ctrl + C	Halts the current command
Ctrl + D	Logout of current session (exit)
Ctrl + W	Erases one word in current line
Ctrl + U	Erases whole line
Ctrl + R	Bring up a recent command

Command	Description
fortune	Poignant, inspirational and silly phrases
yes	Output a string repeatedly until killed
banner	ASCII banner
rev	Reverse each character

- execute a command without options
- execute a command with options
- redirect output to
 - ◆ file
 - ◆ R console

```
# execute a command without options
```

```
# shell
```

```
ls
```

```
# R
```

```
system2(command = "ls")
```

```
# execute a command with options
```

```
# shell
```

```
ls -a
```

```
# R
```

```
system2(command = "ls",  
        args     = c("-a"))
```



```
# redirect output to a file
```

```
# shell
```

```
echo Great Truth > release.txt
```

```
# R
```

```
system2(command = "echo",  
        args     = c("Great Truth"),  
        stdout   = "release.txt")
```

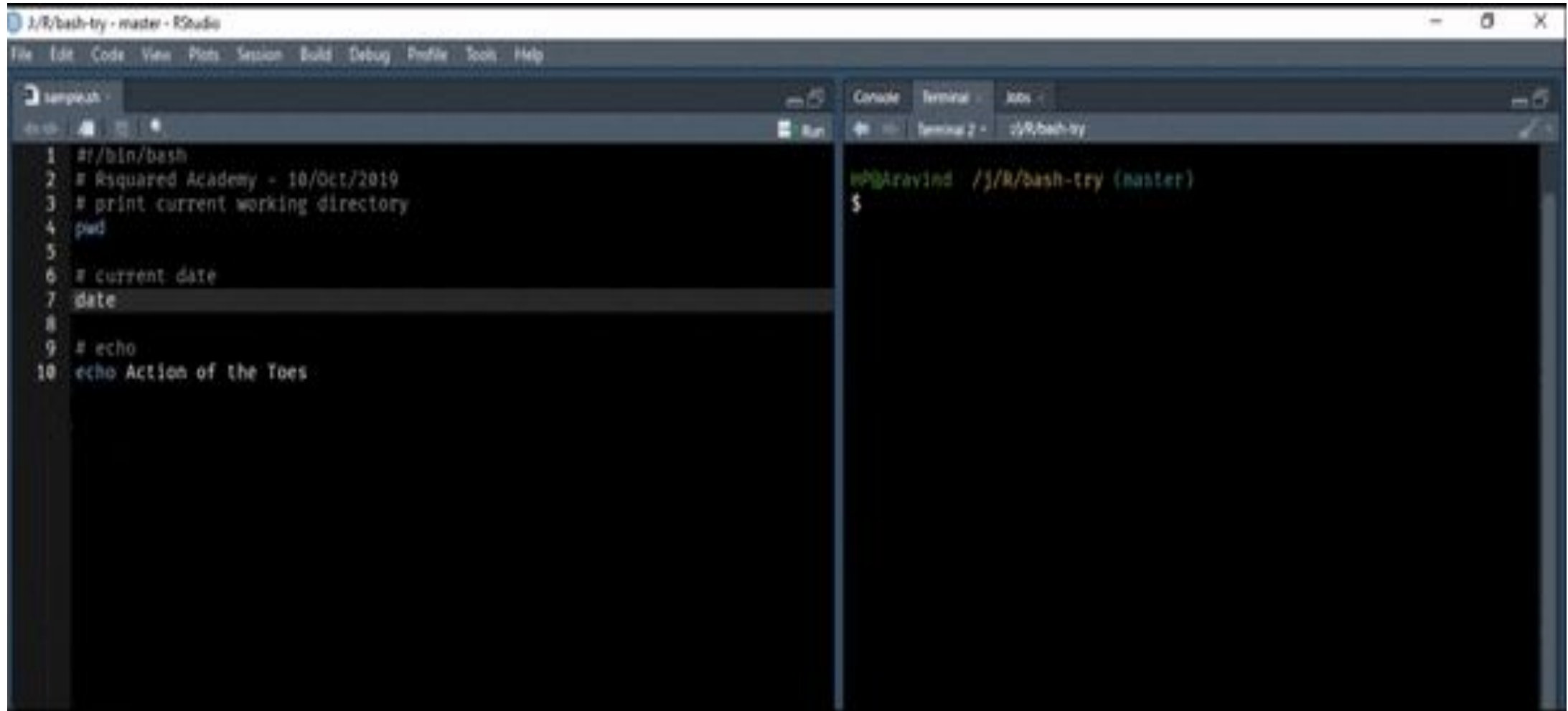
```
# redirect output to R console
```

```
# shell
```

```
diff imports_olsrr.txt imports_blorr.txt
```

```
# R
```

```
system2(command = "diff",  
        args     = c("imports_olsrr.txt", "imports_blorr.txt"),  
        stdout   = TRUE)
```



The screenshot shows the RStudio interface with a terminal window open. The terminal title is `/j/R/bash-try - master - RStudio`. The terminal content shows a bash script being executed line by line. The script includes comments and commands to print the working directory, the current date, and an echo statement. The prompt `mp@Aravind: /j/R/bash-try (master)` is visible, followed by a dollar sign `$` indicating the prompt.

```
1 #!/bin/bash
2 # Rsquared Academy - 10/Oct/2019
3 # print current working directory
4 pwd
5
6 # current date
7 date
8
9 # echo
10 echo Action of the Toes
```

The screenshot displays the RStudio interface with an R Markdown document titled 'bash-sample.Rmd' on the left and its rendered HTML output on the right.

Left Panel (Source Editor):

```
1 ---
2 title: "Shell Commands Demo"
3 date: "10/10/2019"
4 output: html_document
5 ---
6
7 ```{r setup, include=FALSE}
8 knitr::opts_chunk$set(echo = TRUE)
9 ```
10
11 ### Print Working Directory
12
13 ```{sh pwd}
14 pwd
15 ```
16
17 ### Date & Time
18
19 ```{sh date_time}
20 date
21 ```
```

Right Panel (Viewer):

Shell Commands Demo

10/10/2019

Print Working Directory

```
pwd
```

```
## /cygdrive/j/R/bash-try
```

Date & Time

```
date
```

```
## Thu Oct 10 09:51:48 IST 2019
```

- Shell is a text based application for viewing, handling and manipulating files
- It is also known by the following names:
 - CLI (Command Line Interface)
 - Terminal
 - Bash (Bourne Again Shell)
- Useful in several scenarios including:
 - Version Control
 - Cloud Services
- RStudio includes a Terminal (from version 1.1.383)
- Use `system2()` in R to execute commands
- Execute commands from shell script in RStudio using Ctrl + Enter
- Execute R scripts from command line using `R -e` or `Rscript -e`
- RMarkdown supports `bash`, `sh` and `awk`

- Shell scripting
- Editing files
- Using sudo
- File Permission
- User Information
- Pipes
- awk
- sed

- <https://tldr.sh/>
- <https://blog.rstudio.com/2017/08/11/rstudio-v1-1-preview-terminal/>
- <https://support.rstudio.com/hc/en-us/articles/115010737148-Using-the-RStudio-Terminal>
- <https://bookdown.org/yihui/rmarkdown/language-engines.html#shell-scripts>
- <https://github.com/kirillseva/tldr>
- <https://fs.r-lib.org/articles/function-comparisons.html>
- http://www2.imm.dtu.dk/courses/02333/cygwin_tutorial/index.html
- https://tutorial.djangogirls.org/en/intro_to_command_line/
- <https://ss64.com/>
- <https://www.learnenough.com/command-line-tutorial>



Thank You

For more information please visit our website
www.rsquaredacademy.com