

# ASSIGNMENT 1.2

## \$ pwd:

pwd prints the full pathname of the current working directory.

### Syntax:

`$ pwd ↵`

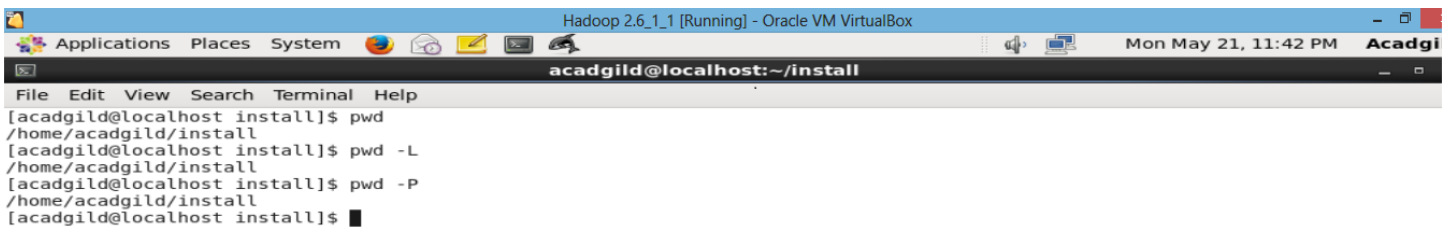
`$ pwd -L ↵`

`$ pwd -P ↵`

### Options:

-P : Print a fully resolved name for the current directory, in which all components of the name are actual directory names, and not symbolic links.

-L: If the environment variable PWD contains an absolute name of the current directory with no "." or ".." components, then output those contents, even if they contain symbolic links. Otherwise, fall back to default (-P) behavior.



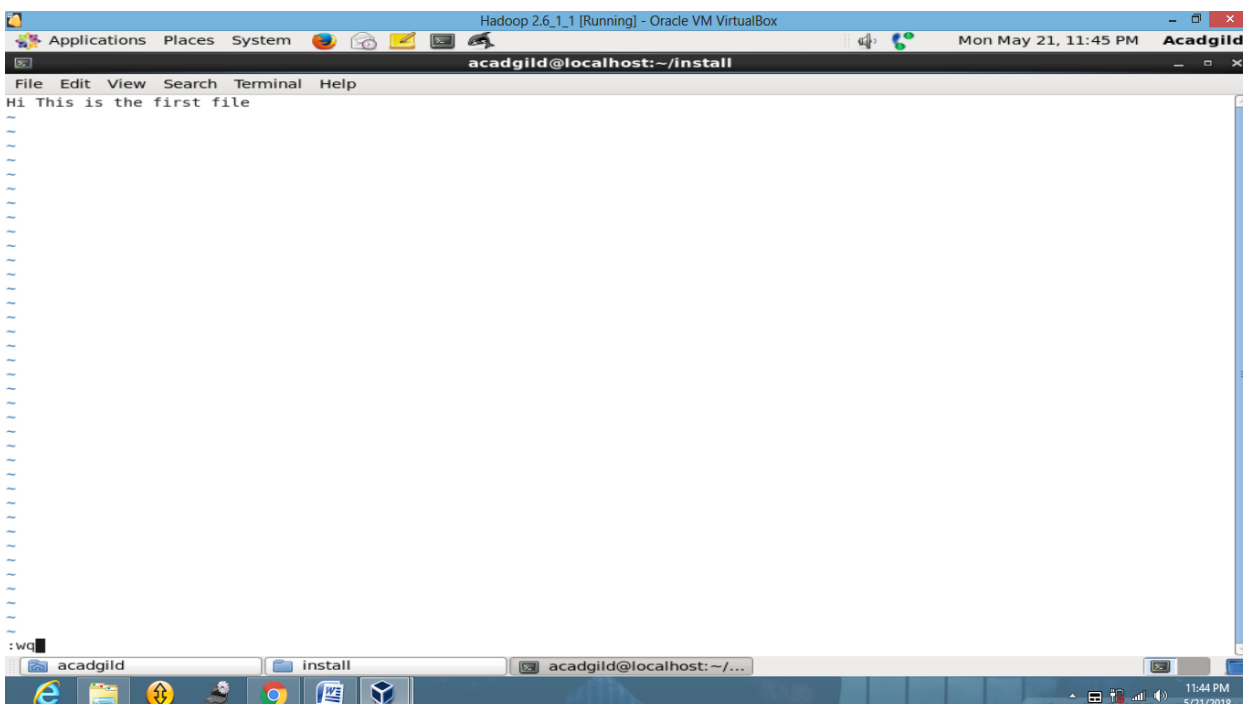
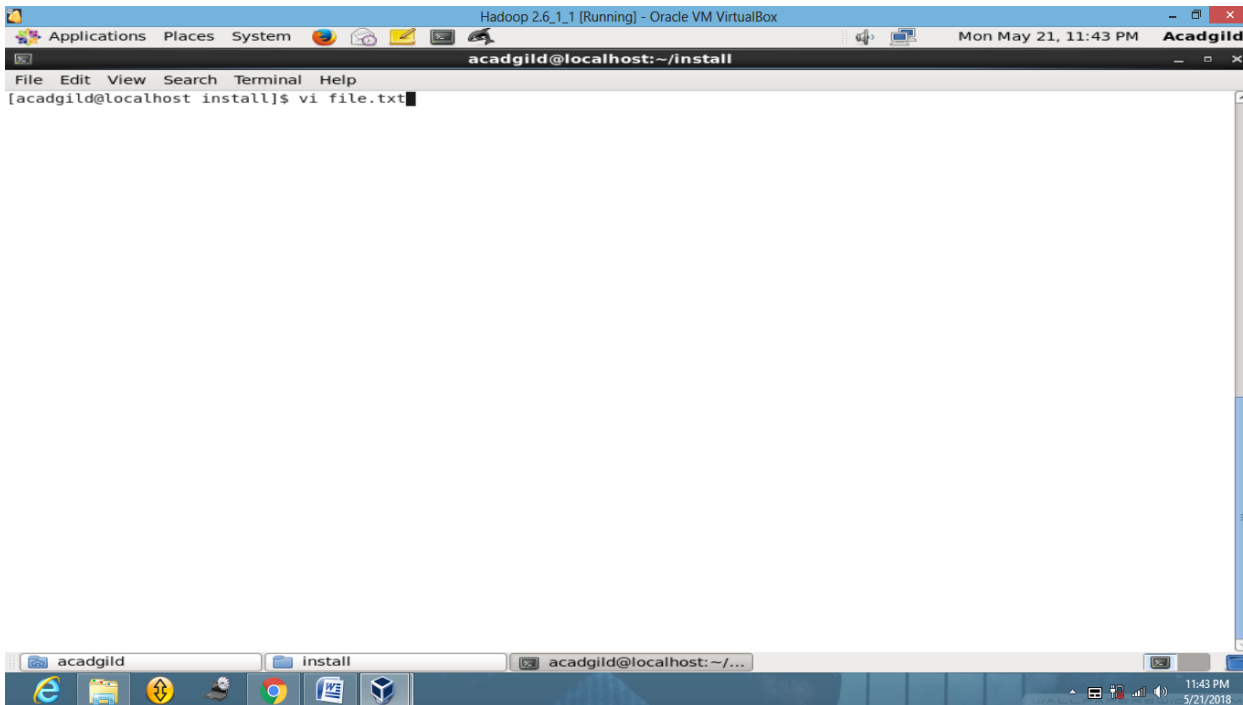
```
Hadoop 2.6.1_1 [Running] - Oracle VM VirtualBox
Applications Places System
[acadgild@localhost ~/install]
File Edit View Search Terminal Help
[acadgild@localhost install]$ pwd
/home/acadgild/install
[acadgild@localhost install]$ pwd -L
/home/acadgild/install
[acadgild@localhost install]$ pwd -P
/home/acadgild/install
[acadgild@localhost install]$ █
```

## \$ vi:

The vi command is used to start the vi editor. It is a visual text editor which when executed with a file, it shows the file with options to insert, save and quit.

### Syntax:

`$ vi <filename> ↵`

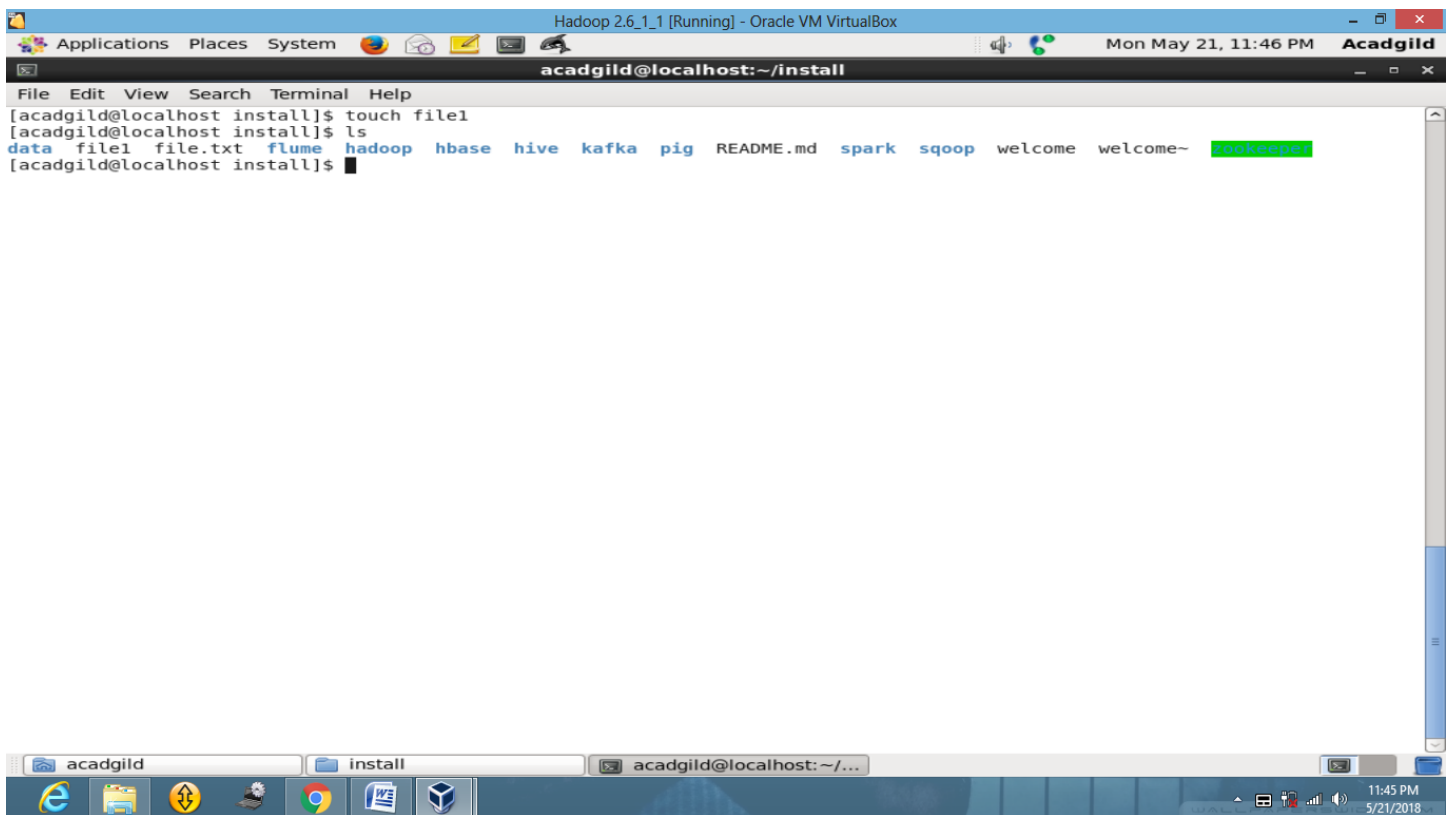


## \$ touch:

touch is a standard Unix command-line interface program which is used to update the access date and/or modification date of a file or directory. In its default usage, it is the equivalent of creating or opening a file and saving it without any change to the file contents.

### Syntax:

`$ touch <filename>` ↵



The screenshot shows a terminal window titled "Hadoop 2.6.1\_1 [Running] - Oracle VM VirtualBox". The terminal prompt is "acadgild@localhost:~/install". The user has entered the command "touch file1" and then "ls". The output of "ls" shows a directory listing with files: "data", "file1", "file.txt", "flume", "hadoop", "hbase", "hive", "kafka", "pig", "README.md", "spark", "sqoop", "welcome", and "welcome~". The file "zookeeper" is highlighted in green. The terminal window is part of a desktop environment with a taskbar at the bottom showing various application icons and a system tray with the time "11:45 PM" and date "5/21/2018".

```
[acadgild@localhost install]$ touch file1
[acadgild@localhost install]$ ls
data  file1  file.txt  flume  hadoop  hbase  hive  kafka  pig  README.md  spark  sqoop  welcome  welcome~  zookeeper
[acadgild@localhost install]$
```

## \$ mkdir:

The mkdir (make directory) command is used to make a new directory.

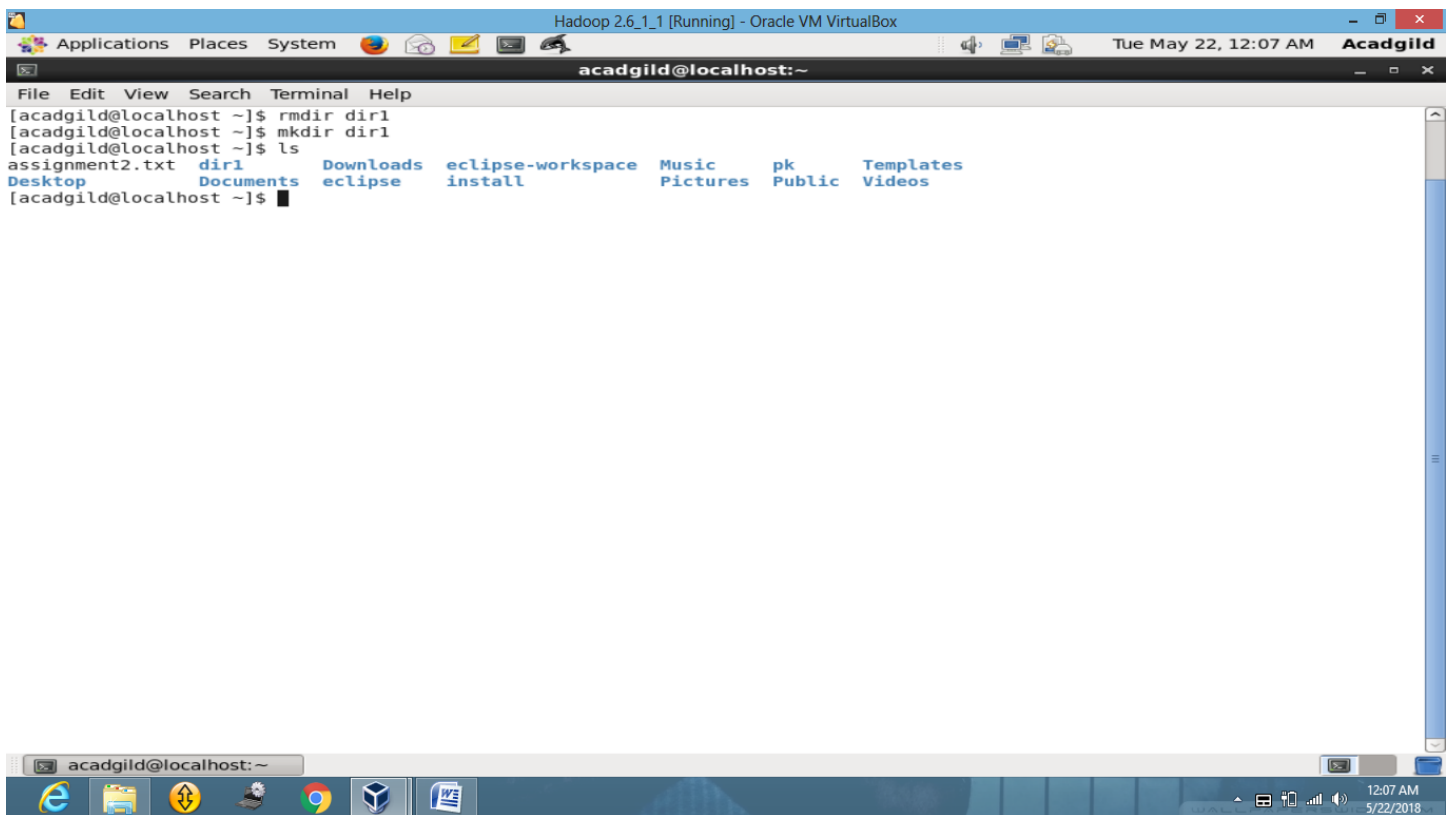
### Syntax:

`$mkdir <name_of_directory>` ↵

### Options:

-p: parents or path, will also create all directories leading up to the given directory that do not exist already. For example, `mkdir -p a/b` will create directory a if it doesn't exist, then will create directory b inside directory a. If the given directory already exists, ignore the error.

-m : mode, specify the octal permissions of directories created by mkdir.



```
Hadoop 2.6.1_1 [Running] - Oracle VM VirtualBox
Applications Places System
acadgild@localhost:~
File Edit View Search Terminal Help
[acadgild@localhost ~]$ rmdir dir1
[acadgild@localhost ~]$ mkdir dir1
[acadgild@localhost ~]$ ls
assignment2.txt  dir1  Downloads  eclipse-workspace  Music  pk  Templates
Desktop          Documents  eclipse      install          Pictures  Public  Videos
[acadgild@localhost ~]$
```

## \$ rm :

The rm command. The rm command is used to delete files. ``rm -i" will ask before deleting each file; you may well have rm aliased to do this automatically. ``rm -r" will recursively delete a directory and all its contents (normally rm will not delete directories, while rmdir will only delete empty directories).

### Syntax:

`$ rm <filename>`

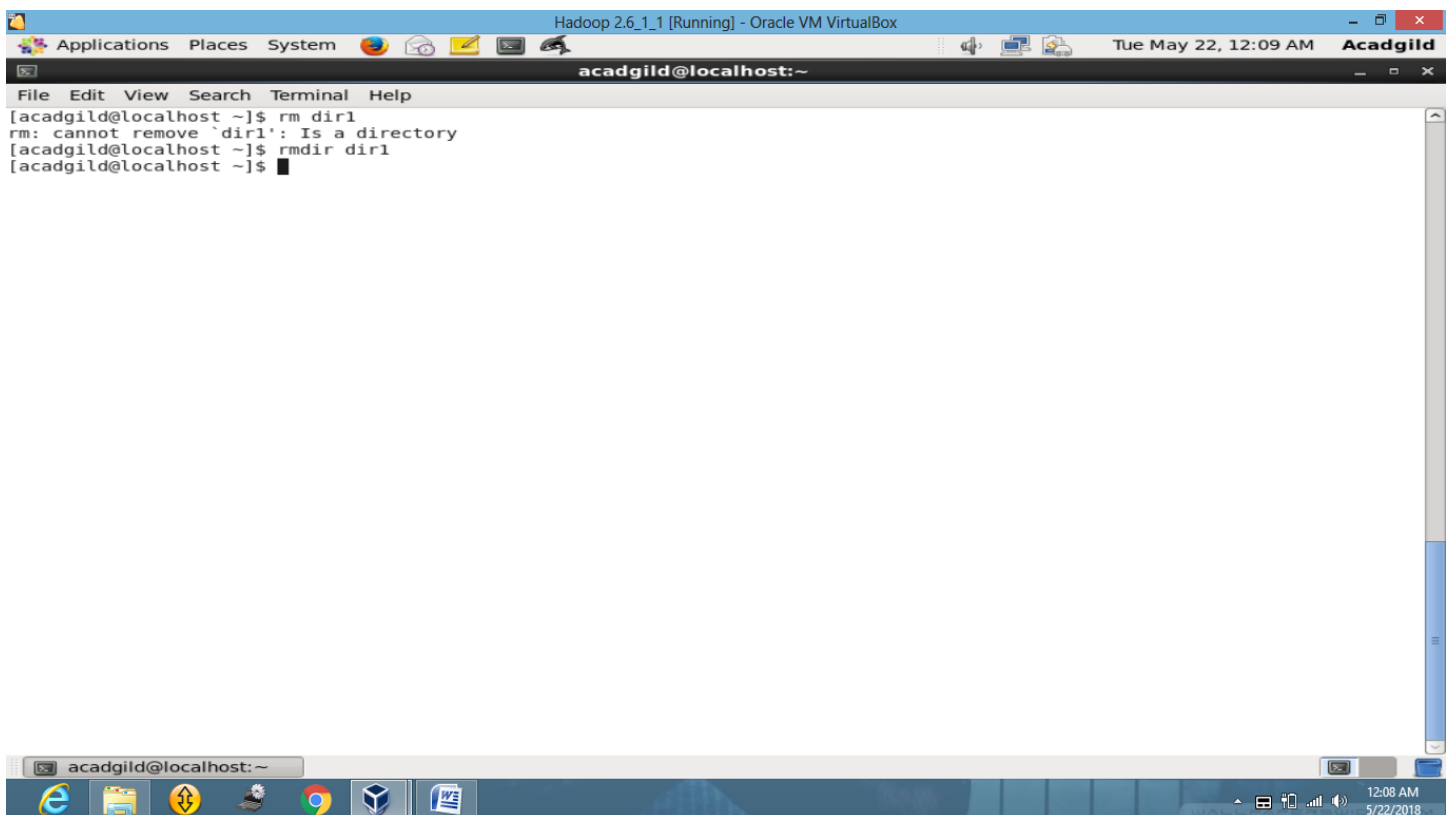
`$ rmdir <dir_name>`

### Options:

-r, which removes directories, removing the contents recursively beforehand (so as not to leave files without a directory to reside in) ("recursive")

-i, which asks for every deletion to be confirmed ("interactive")

-f, which ignores non-existent files and overrides any confirmation prompts ("force"), although it will not remove files from a directory if the directory is write-protected.



The screenshot shows a terminal window titled "Hadoop 2.6\_1\_1 [Running] - Oracle VM VirtualBox". The window has a menu bar with "File", "Edit", "View", "Search", "Terminal", and "Help". The terminal output shows the following commands and their results:

```
[acadgild@localhost ~]$ rm dir1
rm: cannot remove `dir1': Is a directory
[acadgild@localhost ~]$ rmdir dir1
[acadgild@localhost ~]$
```

The window's title bar includes "Applications", "Places", "System", and "Acadgild". The system tray at the bottom shows the date and time as "Tue May 22, 12:09 AM" and "5/22/2018".

**\$ Is:**

ls is a command to list files in Unix and Unix-like operating systems.

### Syntax:

*\$ ls [option]* ↵

**Options:**

-l long format, displaying Unix file types, permissions, number of hard links, owner, group, size, last-modified date and filename

-f do not sort. Useful for directories containing large numbers of files.

-F appends a character revealing the nature of a file, for example, \* for an executable, or / for a directory. Regular files have no suffix.

-a lists all files in the given directory, including those whose names start with "." (which are hidden files in Unix). By default, these files are excluded from the list.

-R recursively lists subdirectories. The command `ls -R /` would therefore list all files.

-d shows information about a symbolic link or directory, rather than about the link's target or listing the contents of a directory.

-t sort the list of files by modification time.

-h print sizes in human readable format. (e.g., 1K, 234M, 2G, etc.)

## **\$ echo:**

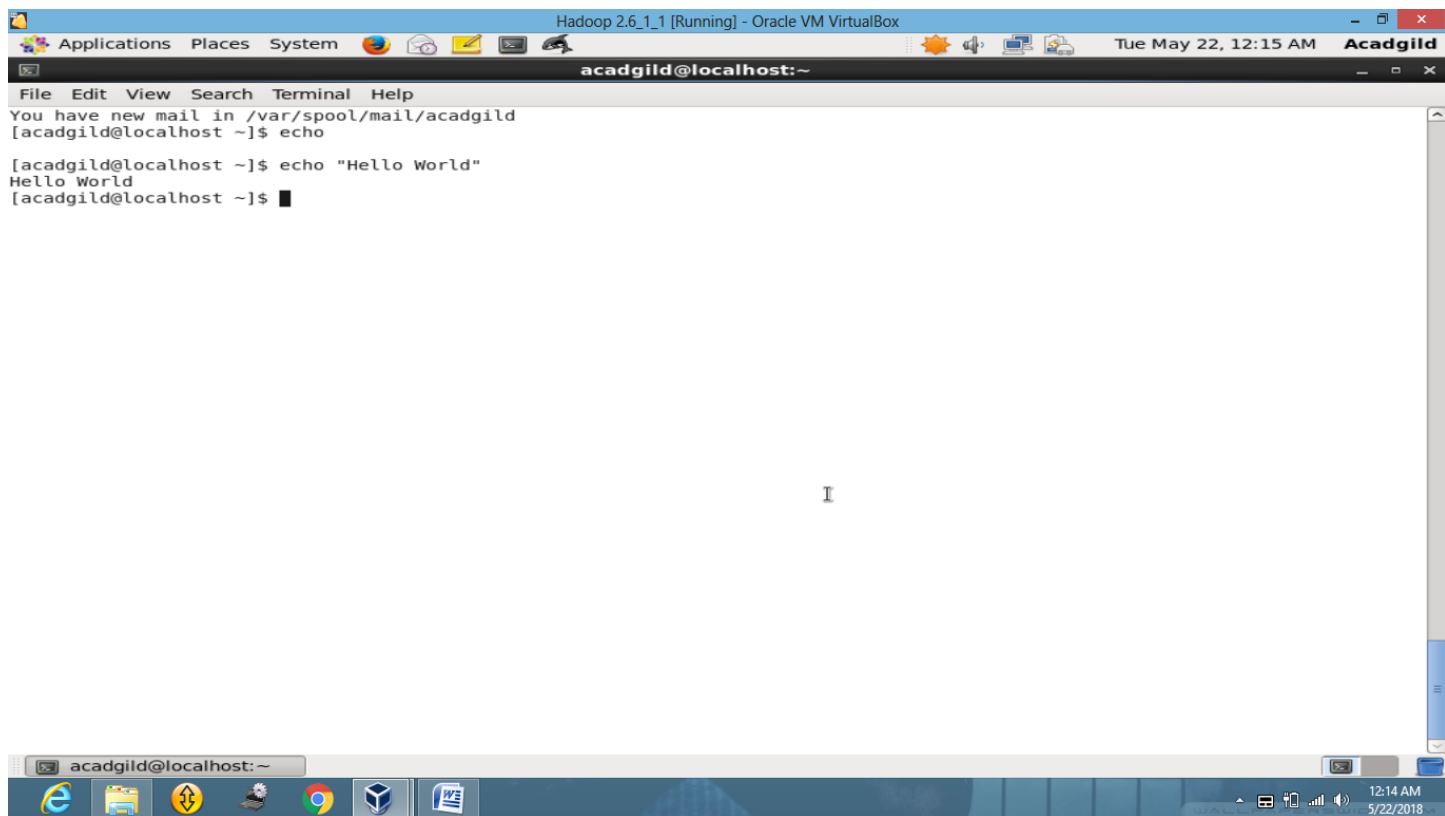
echo is a command in DOS, OS/2, Microsoft Windows, Unix and Unix-like operating systems that outputs the strings it is being passed as arguments. It is a command typically used in shell scripts and batch files to output status text to the screen or a file, or as a source part of a pipeline.

### **Syntax:**

*\$ echo "Hello World" ↵*

### **Options:**

-n	Do not output a trailing newline.
-e	Enable interpretation of backslash escape sequences.
-E	Disable interpretation of backslash escape sequences.
--help	Display a help message and exit.
--version	Output version information and exit.
\\	A literal backslash character ("\\").
\a	An alert (The BELL character).
\b	Backspace
\c	Produce no further output after this.
\e	The escape character; equivalent to pressing the escape key.
\f	A form feed.
\n	A newline.
\r	A carriage return
\t	A horizontal tab.
\v	A vertical tab.
\ONNN	byte with octal value NNN (which can be 1 to 3 digits).
\xHH	byte with hexadecimal value HH (which can be either 1 or 2 digits)





## \$ cat:

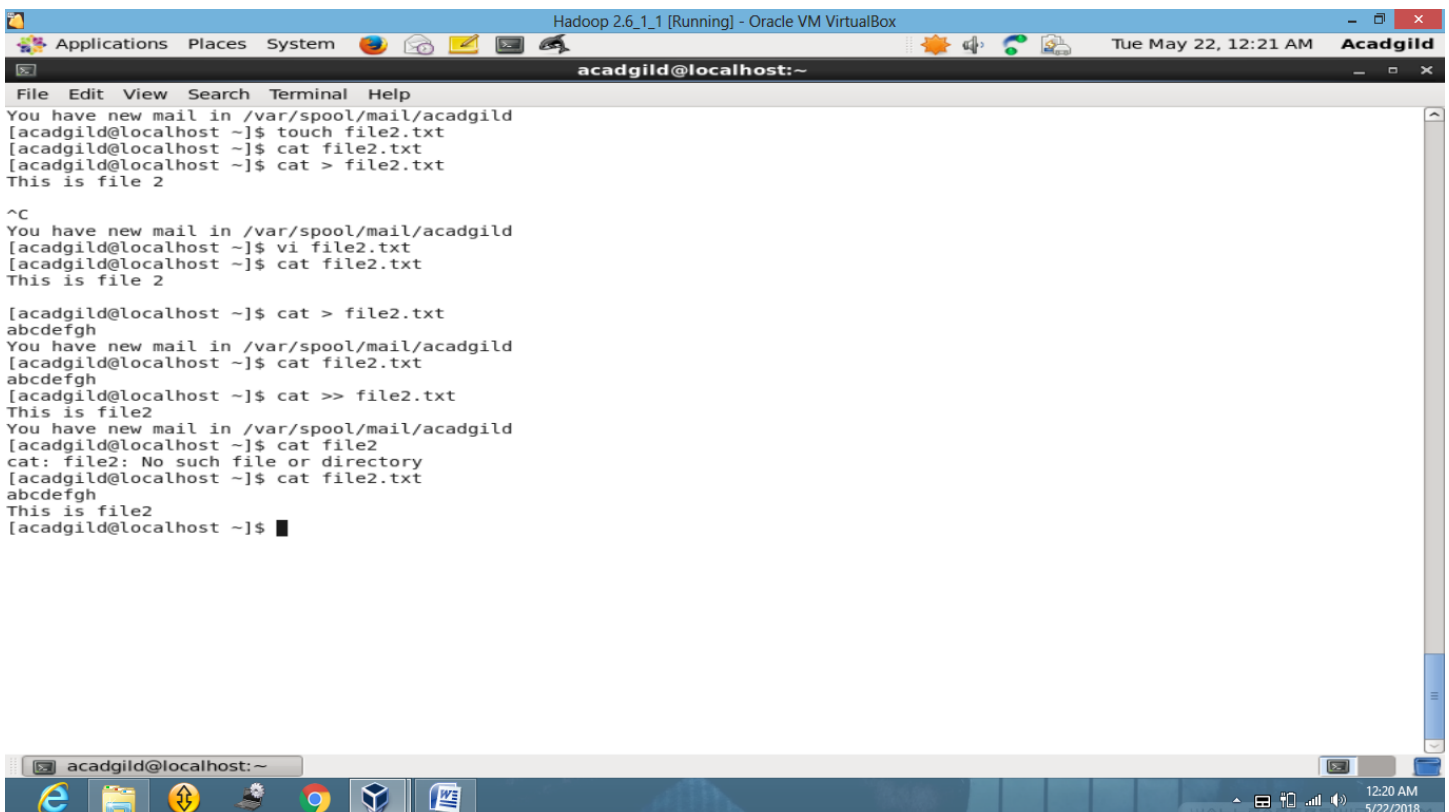
cat is a standard Unix utility that reads files sequentially, writing them to standard output. The name is derived from its function to concatenate files.

### Syntax:

```
$ cat <file_name> ↵
```

### Options:

- b number non-blank output lines
- e implies -v but also display end-of-line characters as \$
- n number all output lines
- s squeeze multiple adjacent blank lines
- t implies -v, but also display tabs as ^I
- u use unbuffered I/O for stdout. POSIX does not specify the behavior without this option.
- v displays nonprinting characters, except for tabs and the end of line character.



The screenshot shows a terminal window titled "Hadoop 2.6.1\_1 [Running] - Oracle VM VirtualBox" with the user "Acadgild". The terminal prompt is "acadgild@localhost:~". The terminal output shows the following sequence of commands and results:

```
You have new mail in /var/spool/mail/acadgild
[acadgild@localhost ~]$ touch file2.txt
[acadgild@localhost ~]$ cat file2.txt
[acadgild@localhost ~]$ cat > file2.txt
This is file 2

^C
You have new mail in /var/spool/mail/acadgild
[acadgild@localhost ~]$ vi file2.txt
[acadgild@localhost ~]$ cat file2.txt
This is file 2

[acadgild@localhost ~]$ cat > file2.txt
abcdefgh
You have new mail in /var/spool/mail/acadgild
[acadgild@localhost ~]$ cat file2.txt
abcdefgh
[acadgild@localhost ~]$ cat >> file2.txt
This is file2
You have new mail in /var/spool/mail/acadgild
[acadgild@localhost ~]$ cat file2
cat: file2: No such file or directory
[acadgild@localhost ~]$ cat file2.txt
abcdefgh
This is file2
[acadgild@localhost ~]$
```

## **\$ who :**

The standard Unix command who displays a list of users who are currently logged into the computer.

### **Syntax:**

`$ who ↵`

`$ who am I ↵`

### **Options:**

-a, process the system database used for user information with the -b, -d, -l, -p, --r, -t, -T and -u.

-b show time when system was last rebooted

-d show zombie processes and details

-H show column headers

-l show terminals where a user can log in

-m show information about the current terminal only

-p show active processes

-q quick format, show only names and the number of all users logged on, disables all other options; equivalent to users command line utility

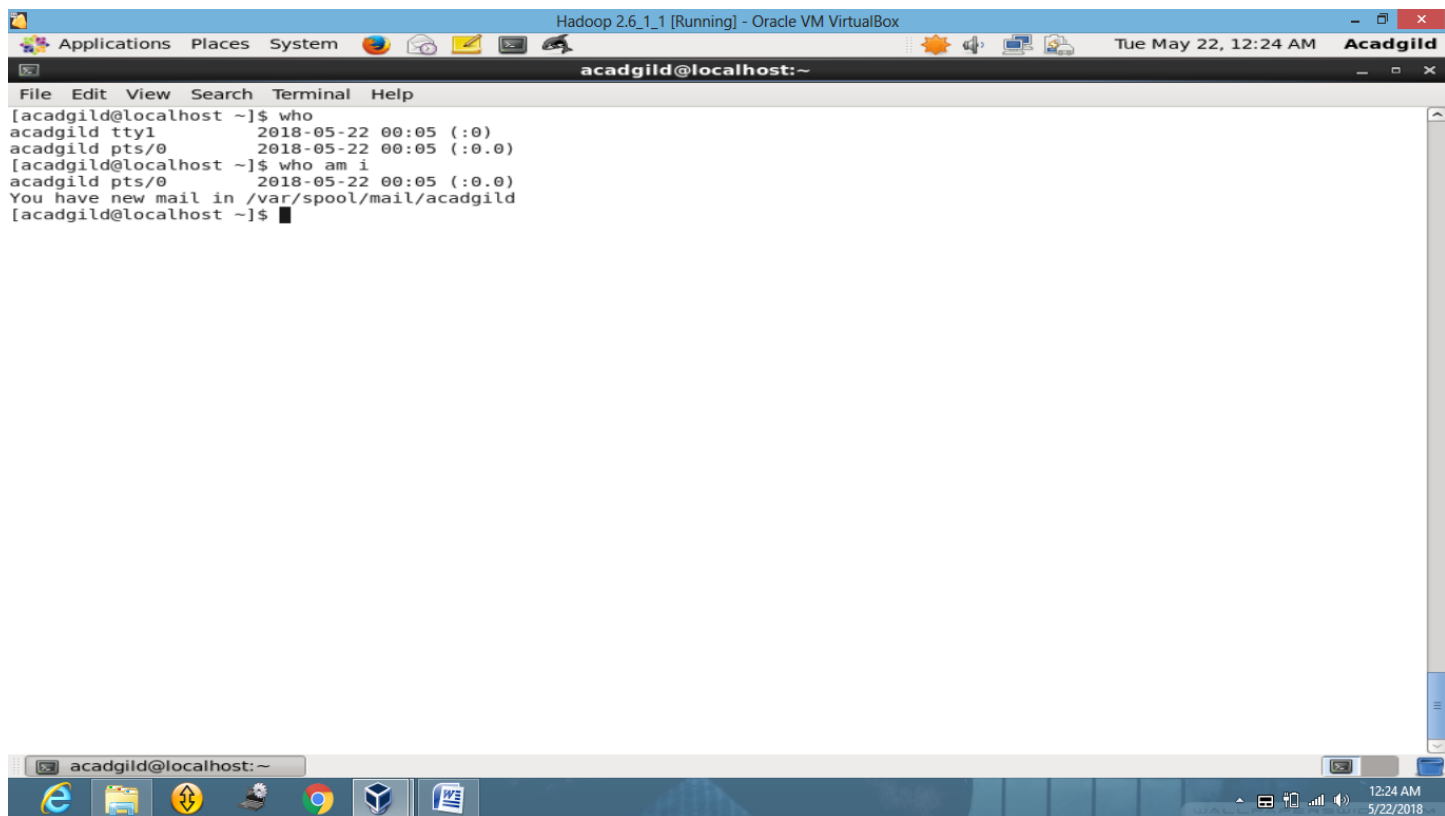
-r show runlevel of the init process.

-s (default) show only name, terminal, and time details

-t show when system clock was last changed

-T show details of each terminal in a standard format (see note in Examples section)

-u show idle time; XSI shows users logged in and displays information whether the terminal has been used recently or not



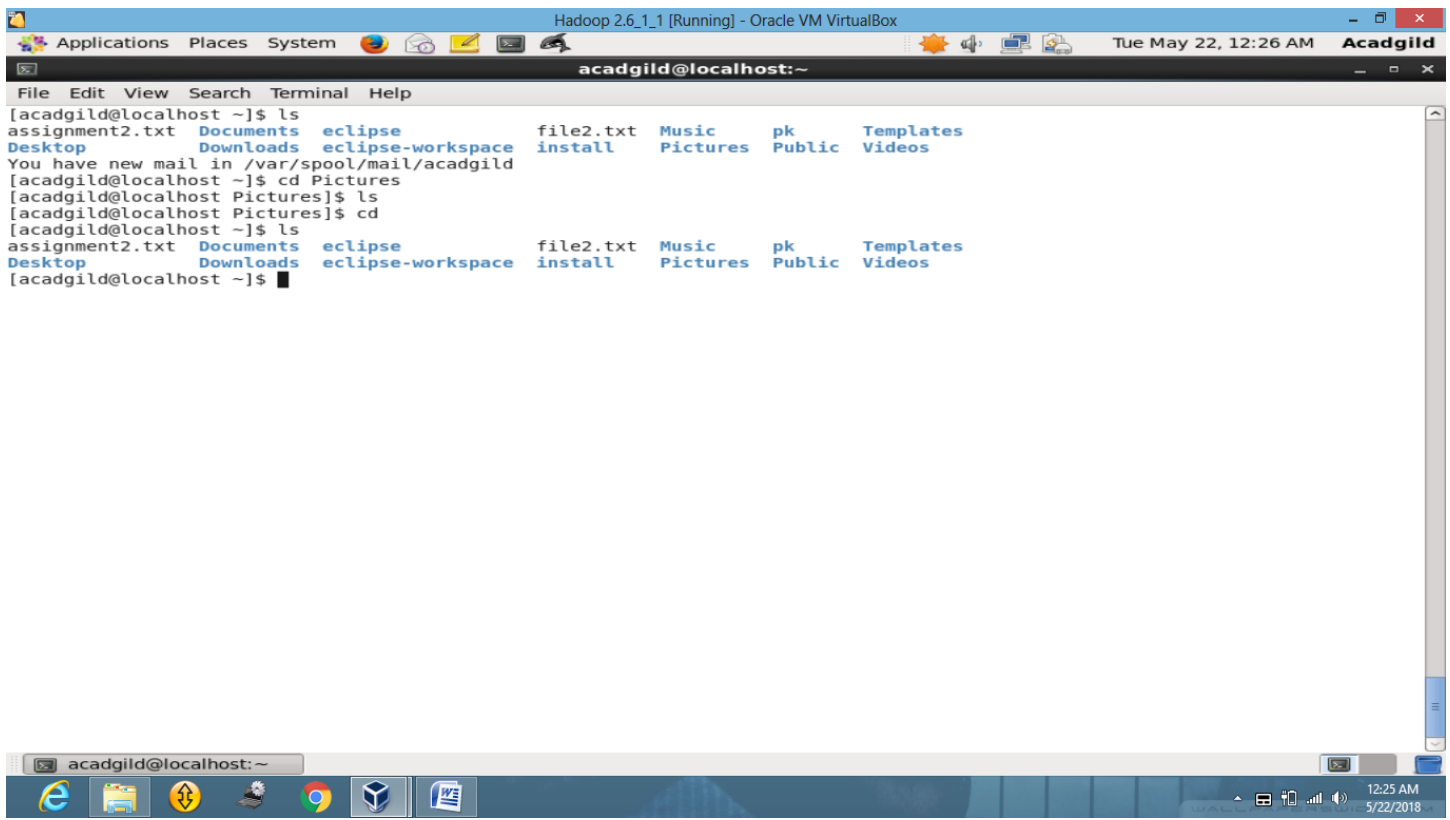
## \$cd

The cd command, also known as chdir (change directory), is a command-line OS shell command used to change the current working directory.

### Syntax:

`$ cd <directory_path>` : Gets the user in the destination directory

`$ cd` : Gets the user to the root directory



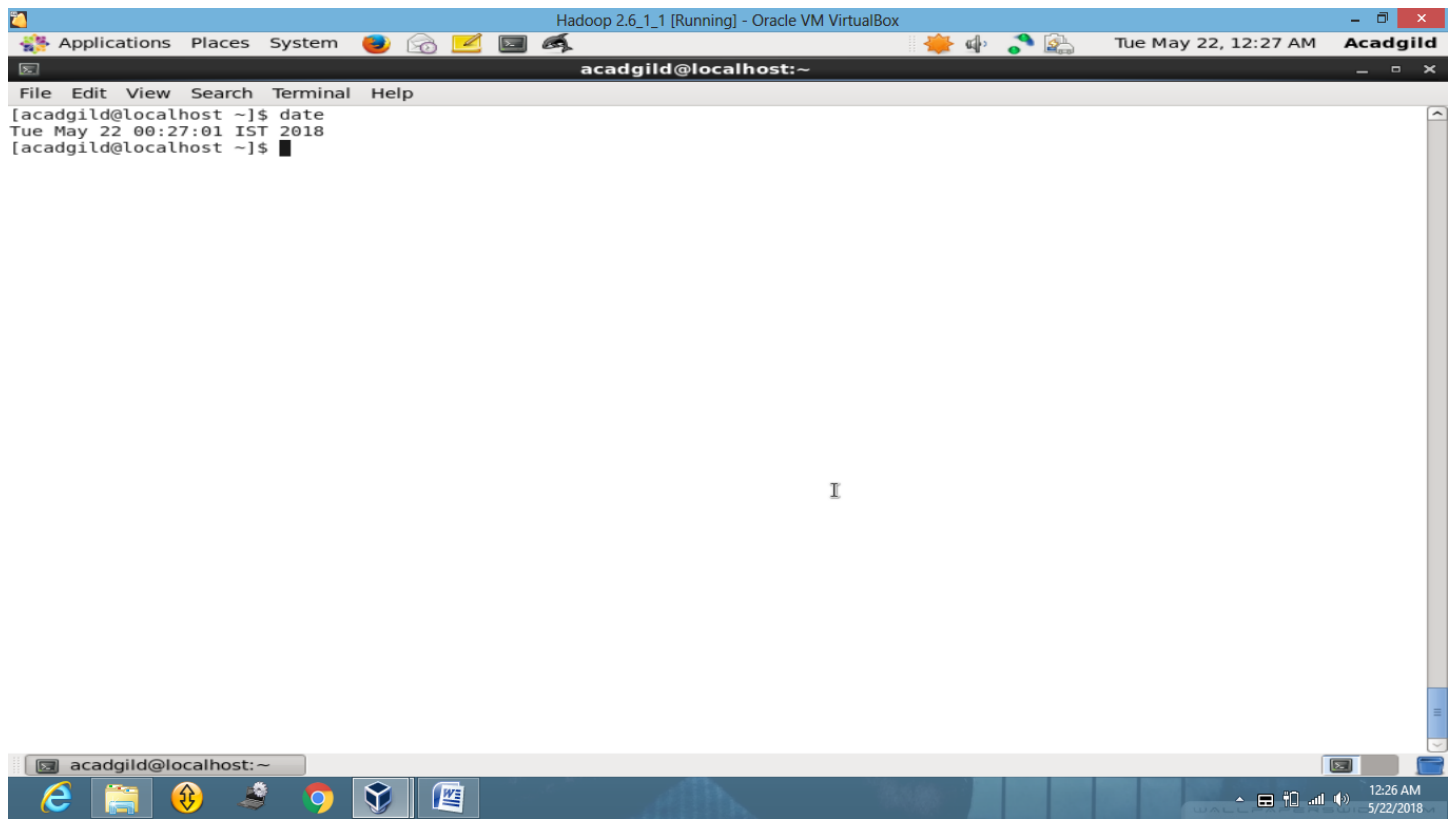
```
Hadoop 2.6.1_1 [Running] - Oracle VM VirtualBox
Tue May 22, 12:26 AM  Acadgild
acadmild@localhost:~
File Edit View Search Terminal Help
[acadmild@localhost ~]$ ls
assignment2.txt  Documents  eclipse  file2.txt  Music  pk  Templates
Desktop          Downloads  eclipse-workspace  install  Pictures  Public  Videos
You have new mail in /var/spool/mail/acadmild
[acadmild@localhost ~]$ cd Pictures
[acadmild@localhost Pictures]$ ls
[acadmild@localhost Pictures]$ cd
[acadmild@localhost ~]$ ls
assignment2.txt  Documents  eclipse  file2.txt  Music  pk  Templates
Desktop          Downloads  eclipse-workspace  install  Pictures  Public  Videos
[acadmild@localhost ~]$
```

## \$date :

date command is used to display the system date and time. date command is also used to set date and time of the system. By default the date command displays the date in the time zone on which unix/linux operating system is configured.

### Syntax:

*\$ date ↵*



The screenshot shows a terminal window titled "Hadoop 2.6\_1\_1 [Running] - Oracle VM VirtualBox". The window has a menu bar with "File", "Edit", "View", "Search", "Terminal", and "Help". The terminal prompt is "acadgild@localhost:~". The user has entered the command "date", and the output is "Tue May 22 00:27:01 IST 2018". The terminal window is running on a desktop environment with a taskbar at the bottom showing various application icons and a system tray with the date and time "12:26 AM 5/22/2018".

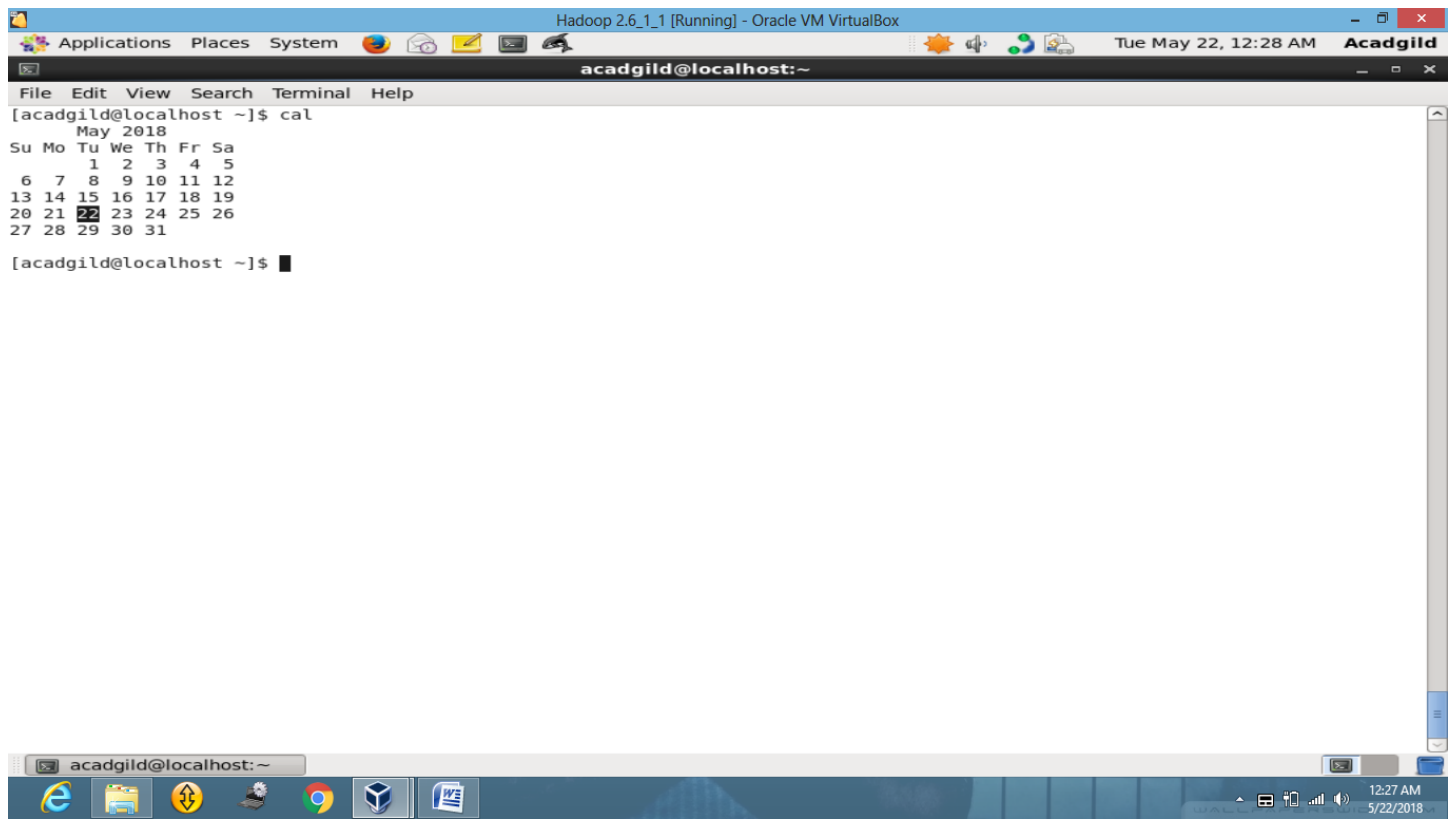
```
acadgild@localhost:~$ date
Tue May 22 00:27:01 IST 2018
acadgild@localhost:~$
```

## \$cal :

The cal command is a command line utility for displaying a calendar in the terminal. It can be used to print a single month, many months or an entire year. It supports starting the week on a Monday or a Sunday, showing Julian dates and showing calendars for arbitrary dates passed as arguments.

### Syntax:

*\$ cal* ↵



The screenshot shows a terminal window titled "Hadoop 2.6.1\_1 [Running] - Oracle VM VirtualBox". The terminal prompt is "acadmild@localhost:~". The user has entered the command "cal", and the output displays a calendar for May 2018. The calendar is formatted with days of the week (Su, Mo, Tu, We, Th, Fr, Sa) and dates (1 through 31). The date 22 is highlighted. The terminal window has a menu bar with "File", "Edit", "View", "Search", "Terminal", and "Help". The system tray at the bottom shows the time as 12:27 AM on 5/22/2018.

```
[acadmild@localhost ~]$ cal
      May 2018
Su Mo Tu We Th Fr Sa
  1  2  3  4  5
  6  7  8  9 10 11 12
 13 14 15 16 17 18 19
 20 21 22 23 24 25 26
 27 28 29 30 31

[acadmild@localhost ~]$
```

## \$mv :

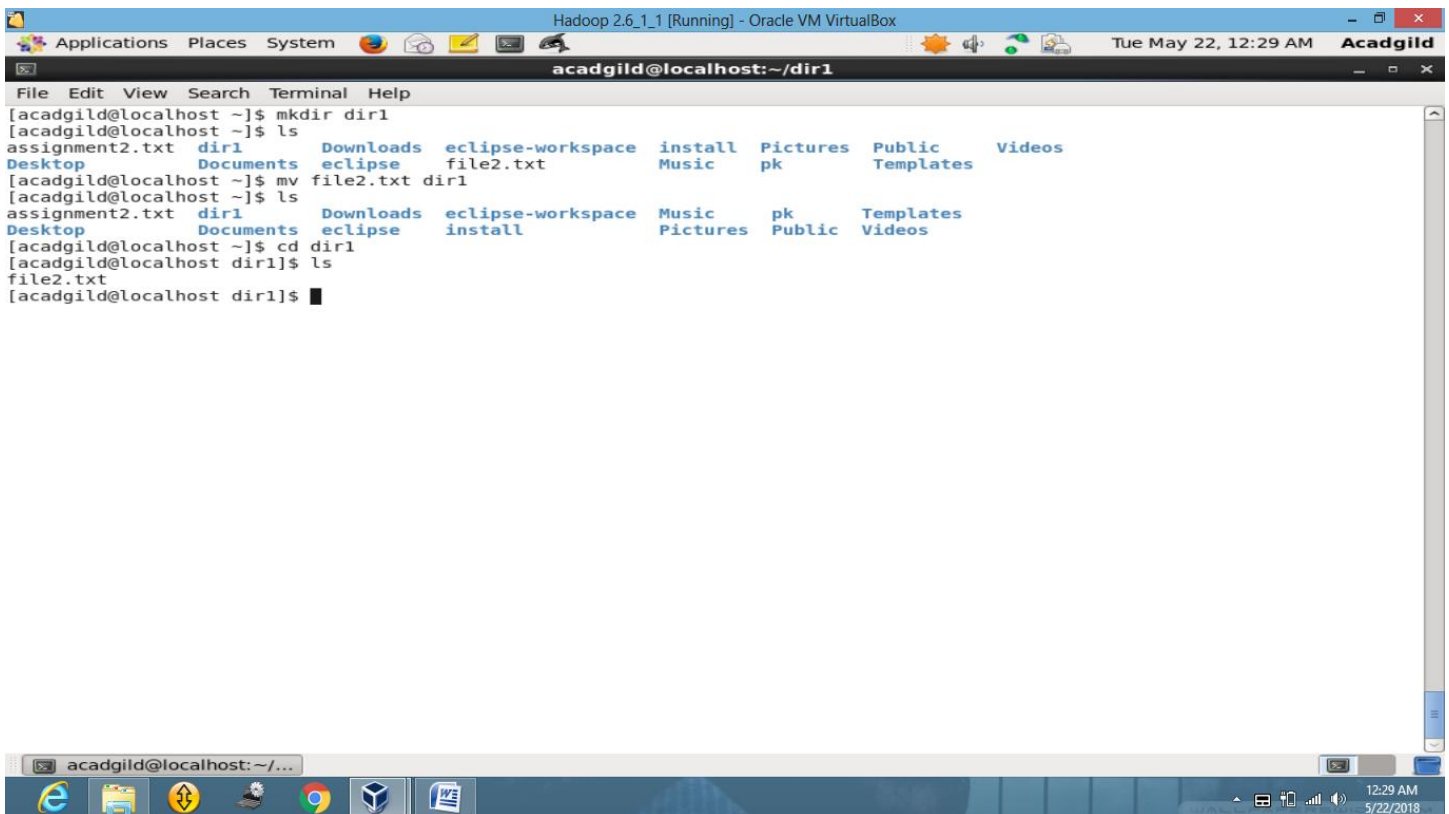
The mv command is a command line utility that moves files or directories from one place to another . It supports moving single files, multiple files and directories. It can prompt before overwriting and has an option to only move files that are new than the destination.

### Syntax:

`$mv <file_name> <dir_name>/<file_name>` ↵

### Options:

- i interactively process, write a prompt to standard error before moving a file that would overwrite an existing file. If the response from the standard input begins with the character `y' or `Y', the move is attempted. (Overrides previous -f or -n options.)
- f force overwriting the destination (overrides previous -i or -n options).



```
Hadoop 2.6.1_1 [Running] - Oracle VM VirtualBox
Tue May 22, 12:29 AM Acadgild
acadgild@localhost:~/dir1
File Edit View Search Terminal Help
[acadgild@localhost ~]$ mkdir dir1
[acadgild@localhost ~]$ ls
assignment2.txt  dir1  Downloads  eclipse-workspace  install  Pictures  Public  Videos
Desktop          Documents  eclipse     file2.txt         Music   pk       Templates
[acadgild@localhost ~]$ mv file2.txt dir1
[acadgild@localhost ~]$ ls
assignment2.txt  dir1  Downloads  eclipse-workspace  Music  pk  Templates
Desktop          Documents  eclipse     install           Pictures  Public  Videos
[acadgild@localhost ~]$ cd dir1
[acadgild@localhost dir1]$ ls
file2.txt
[acadgild@localhost dir1]$
```

## **\$cp :**

cp is a UNIX command for copying files and directories. The command has three principal modes of operation, expressed by the types of arguments presented to the program for copying a file to another file, one or more files to a directory, or for copying entire directories to another directory.

### **Syntax:**

Copying from one file to another:

```
cp <source_file> <target_file> ↵
```

Copying file(s) to a directory-

```
cp <source_file> <target_directory> ↵
```

Copying a directory to a directory (-r or -R must be used)

```
cp -r/-R <sourcedirectory> <target_directory> ↵
```

### **Options:**

f (force) – specifies removal of the target file if it cannot be opened for write operations. The removal precedes any copying performed by the cp command.

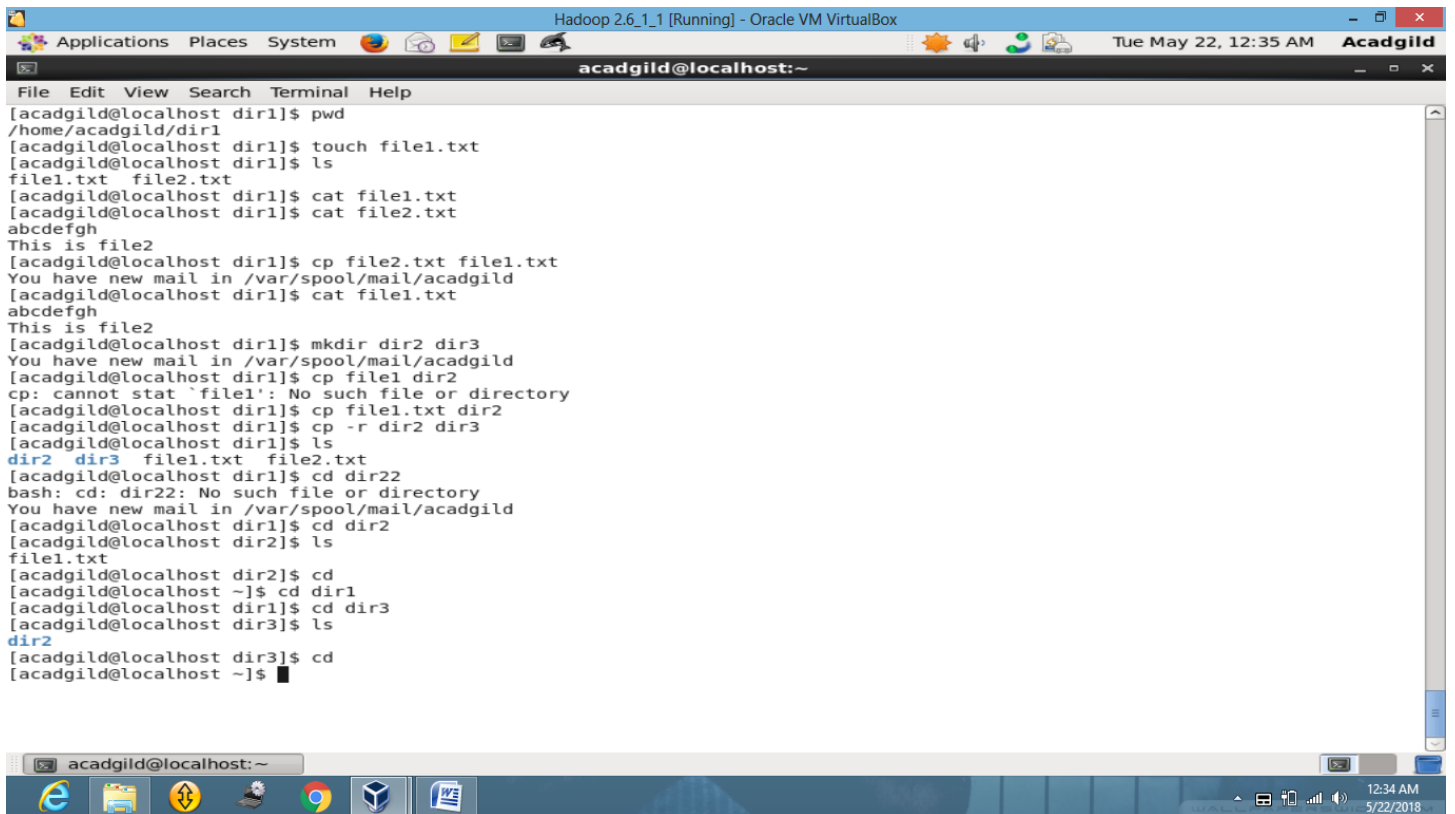
H (dereference) – makes the cp command follow symbolic links (symlinks) so that the destination has the target file rather than a symlink to the target.

i (interactive) – prompts with the name of a file to be overwritten. This occurs if the TargetDirectory or TargetFile parameter contains a file with the same name as a file specified in the SourceFile or SourceDirectory parameter. If one enters y (or the locale's equivalent of y), the cp command continues. Any other answer prevents the cp command from overwriting the file.

p (preserve) – the p flag preserves the following characteristics of each source path in the corresponding target: the time of the last data modification and the time of the last access, the ownership (only if it has permissions to do this), and the file permission-bits.

R or r (recursive) – copy directories recursively



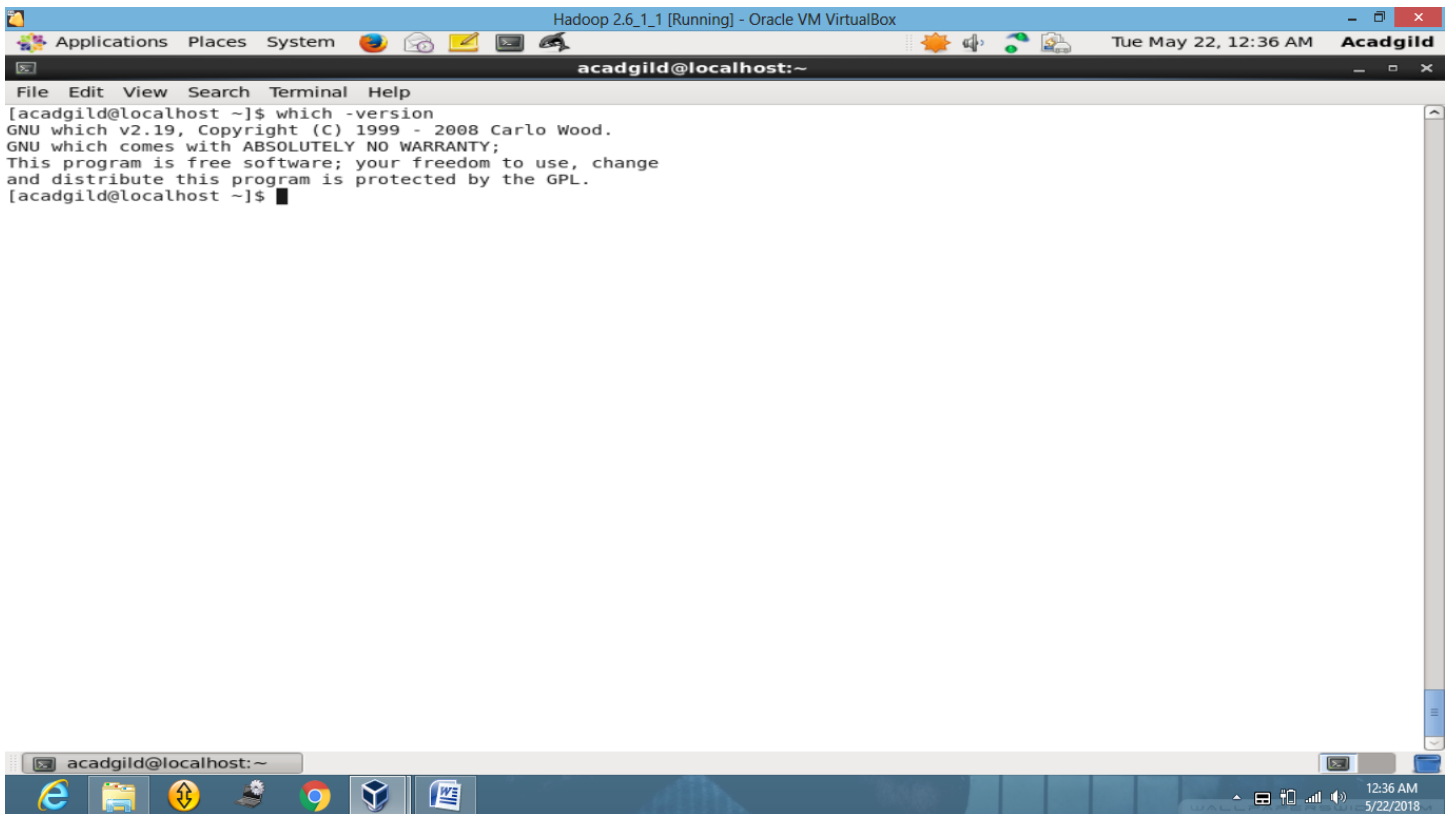


## \$ which:

which is a Unix command used to identify the location of executables. The command takes one or more arguments; for each of these arguments, it prints the full path of the executable to stdout that would have been executed if this argument had been entered into the shell.

## Syntax:

`$ which [-option]` ↵



```
Hadoop 2.6.1_1 [Running] - Oracle VM VirtualBox
Applications Places System
acadmild@localhost:~
File Edit View Search Terminal Help
[acadmild@localhost ~]$ which -version
GNU which v2.19, Copyright (C) 1999 - 2008 Carlo Wood.
GNU which comes with ABSOLUTELY NO WARRANTY;
This program is free software; your freedom to use, change
and distribute this program is protected by the GPL.
[acadmild@localhost ~]$
```

The screenshot shows a terminal window titled "Hadoop 2.6.1\_1 [Running] - Oracle VM VirtualBox". The window has a menu bar with "File", "Edit", "View", "Search", "Terminal", and "Help". The terminal prompt is "acadmild@localhost:~". The user has entered the command "which -version", and the output is displayed: "GNU which v2.19, Copyright (C) 1999 - 2008 Carlo Wood. GNU which comes with ABSOLUTELY NO WARRANTY; This program is free software; your freedom to use, change and distribute this program is protected by the GPL." The terminal window is part of a desktop environment with a taskbar at the bottom showing various application icons and a system tray with the date and time "12:36 AM 5/22/2018".