**Assignment 2:**

**Explaining below linux commands with an example.**

1. **1.pwd**
2. **vi**
3. **touch**
4. **mkdir**
5. **rm**
6. **ls**
7. **echo**
8. **cat**
9. **who**
10. **cd**
11. **date**
12. **cal**
13. **mv**
14. **cp**
15. **which**

**1. pwd**

Print the name of the [working directory](https://www.computerhope.com/jargon/c/currentd.htm).

If you are using the [**bash**](https://www.computerhope.com/unix/ubash.htm) shell, you can determine which **pwd** is the default with the **type** command:

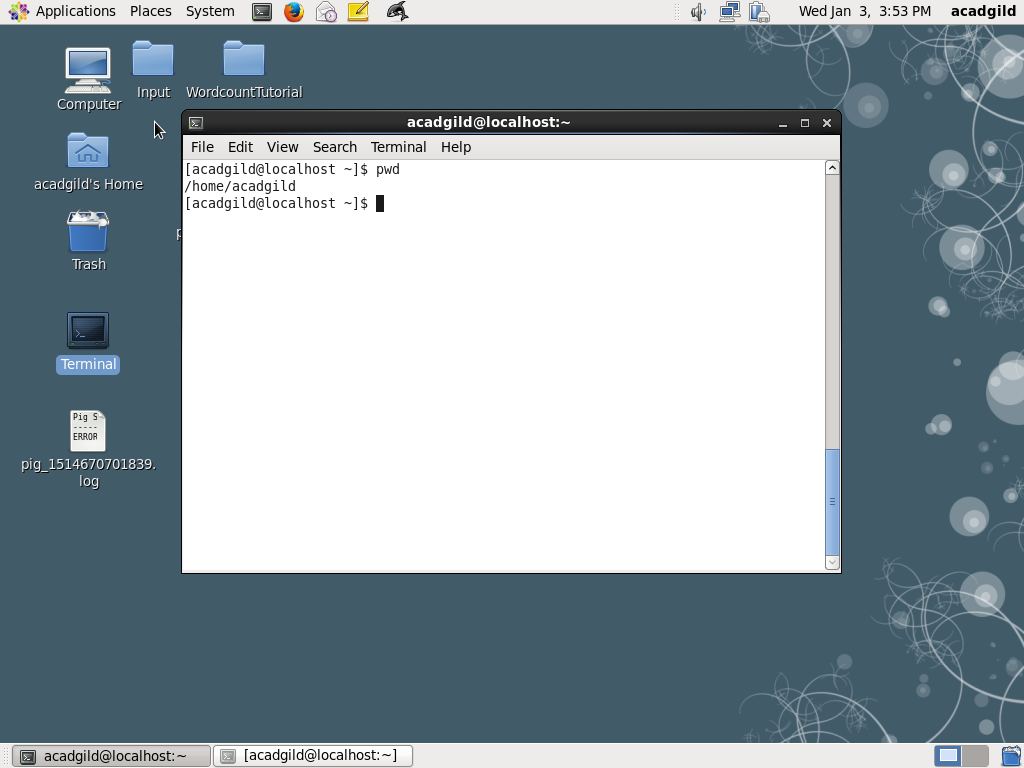
type pwd

pwd is a shell builtin

To specify that you want to run the standalone program instead of the shell builtin, use its complete [path](https://www.computerhope.com/jargon/p/path.htm) in the command, i.e., run it like this:

/bin/pwd

* **pwd** prints the full pathname of the current working [directory](https://www.computerhope.com/jargon/d/director.htm)



**2. vi**

vi is actually the command which starts the visual mode of [ex](https://www.computerhope.com/unix/uex.htm), the landmark editing program developed by Joy. As ex gained popularity, Joy noticed that most users were exclusively using its visual mode, so to make things more convenient for his users, he added a link to ex which started it in visual mode automatically. Today vi is the most popular text editor among [Linux](https://www.computerhope.com/jargon/l/linux.htm) users.

vi [ -| -s ] [-l] [-L] [-R] [ -r [ *filename* ] ] [-S] [-t *tag*] [-v] [-V]

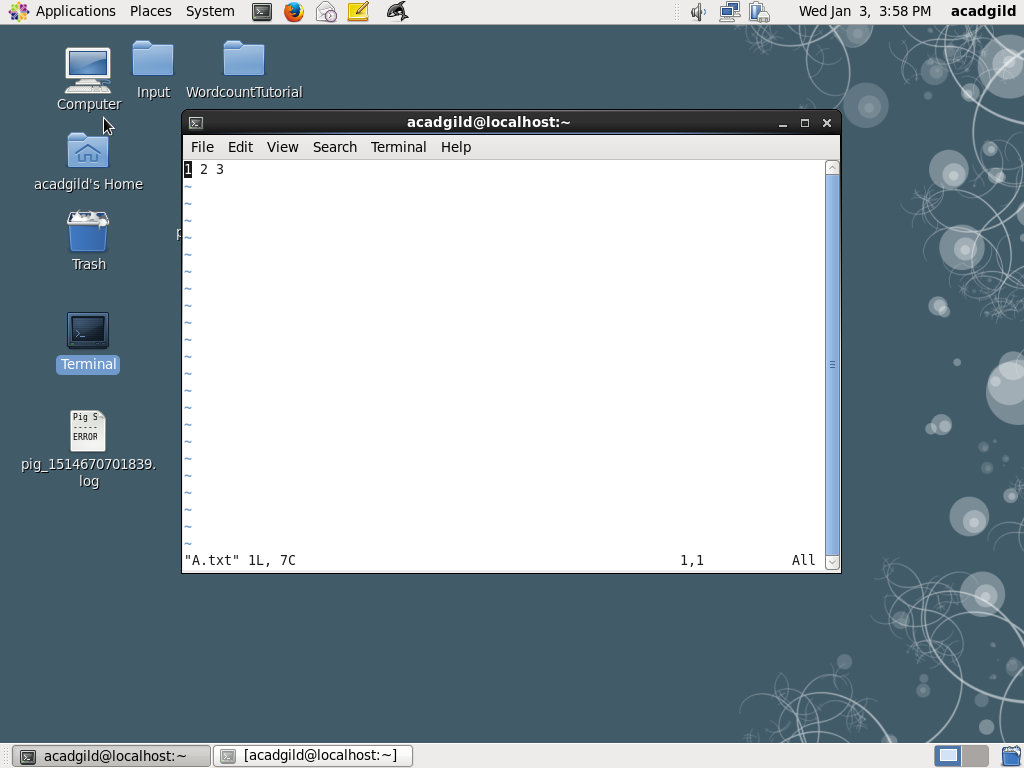
[-x] [-w] [-n ] [-C] [+*command* | -c *command* ] *filename*

## Editing A File

The most common way to start a vi session is to tell it which file to edit. To edit a file named filename, use the command:

$ vi filename

The screen will clear and the text of your file will appear on the screen. If file name doesn't exist yet, vi will start you in a new file, and when you tell it to save your work, it will use the filename that you specified.

****

.**3.touch**

**touch** changes file [timestamps](https://www.computerhope.com/jargon/t/timestam.htm). It is also an easy way to create empty files.

What is a timestamp?

A timestamp is information associated with a file that identifies an important time in the file's history. A file can have multiple timestamps, and some of them can be "forged" by setting them manually. Internally, the [operating system](https://www.computerhope.com/os.htm) stores these times as time elapsed since an arbitrary date called the [epoch](https://www.computerhope.com/jargon/e/epoch.htm). For [Unix](https://www.computerhope.com/jargon/u/unix.htm) like operating systems, the epoch is 00:00:00 Coordinated Universal Time ([UTC](https://www.computerhope.com/jargon/u/ut.htm)), Thursday, 1 January 1970.

In [Linux](https://www.computerhope.com/jargon/l/linux.htm), there are three timestamps associated with a file:

The atime and mtime are part of a file's status metadata. Therefore, when you change the atime (**-a**) or mtime (**-m**) of a file, its ctime is automatically set to the current time.

There is no way to manually set the ctime.

A file's atime or mtime can be set to the future or the past if the user owns the file.

Command syntax

**touch** [[**-a**] [**-m**] | [**--time=***timetype*] [...]] [[**-d** *datestring*] | [**-t** *timestamp*]]

[**-c**] [**-f**] [**-h**] [**-r** *reffile*] *file* [*file* ...]

**touch --version**

**touch --help**

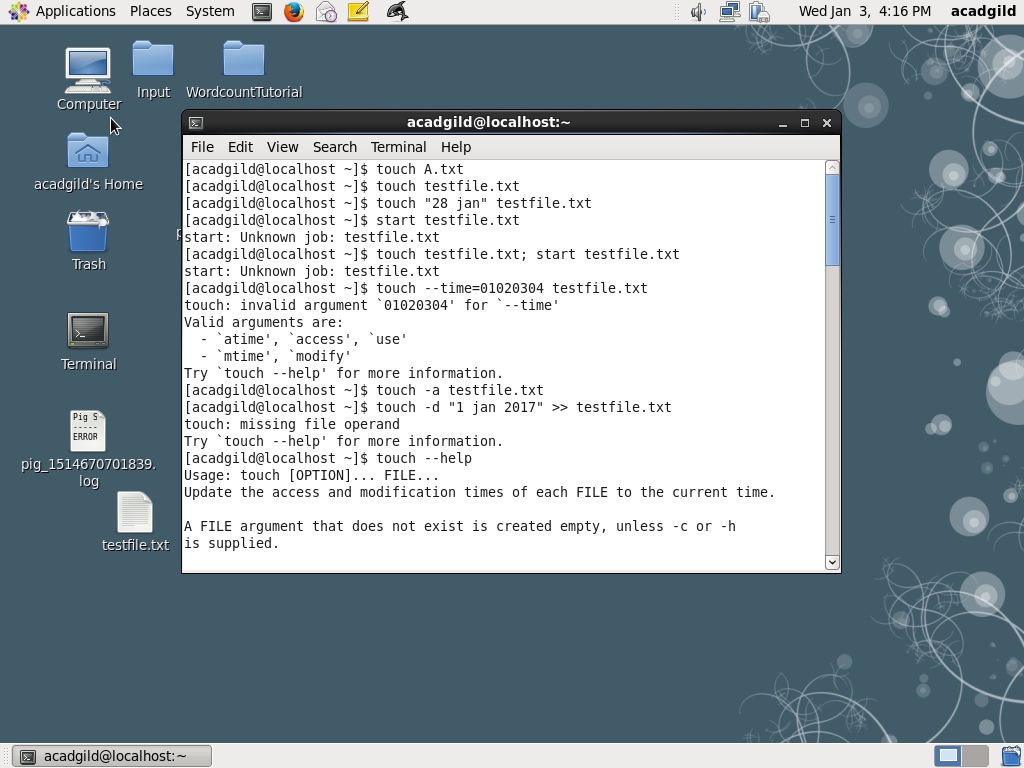
The only required [argument](https://www.computerhope.com/jargon/a/argument.htm) to **touch** is a [file name](https://www.computerhope.com/jargon/f/filename.htm):

|  |  |
| --- | --- |
| *File* | A file whose times should be changed. If *file* does not exist, it is created, unless the the **-c** or **-h** options are used.  Multiple files may be specified as *file0* *file1* *file2*... etc.  (If *file* is a dash "**-**", **touch** modifies the special file descriptor [standard output](https://www.computerhope.com/jargon/s/stdout.htm). See[examples](https://www.computerhope.com/unix/utouch.htm#examples-standard-output).) |

With no options, **touch** will change the atime, mtime, and ctime of *file* to the current system time.

## Examples

touch file.txt

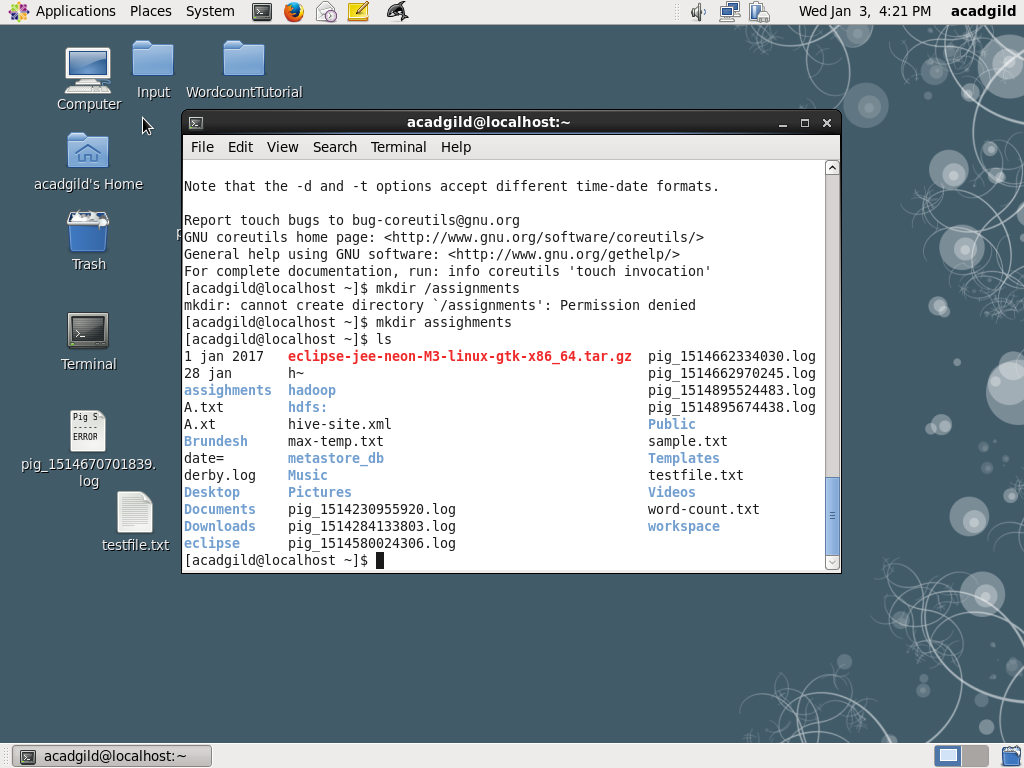
****

**4. mkdir**

This command is used to create directory.

**Syntax: mkdir directory-name**

**Example: mkdir assignments**

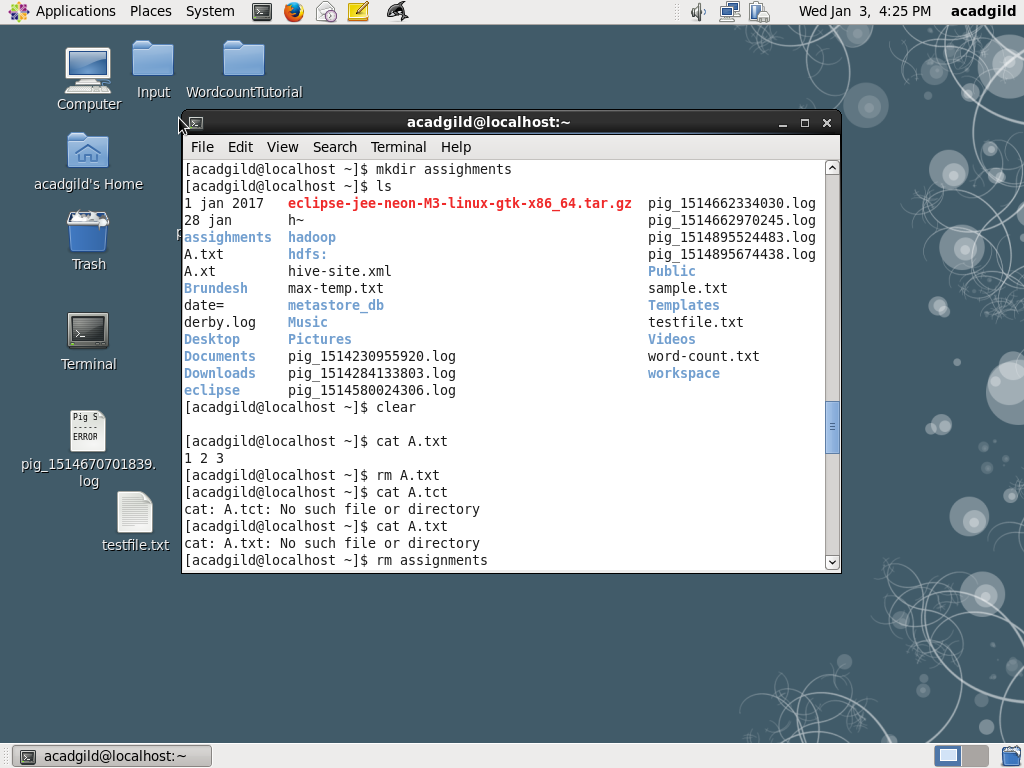
****

**5. rm**

* The rm command is a command line utility for removing files or directories.
* To remove a file pass the name of a file or files to the rm command. Note that files will be removed immediately.

**Syntax: rm filename or directoryname**

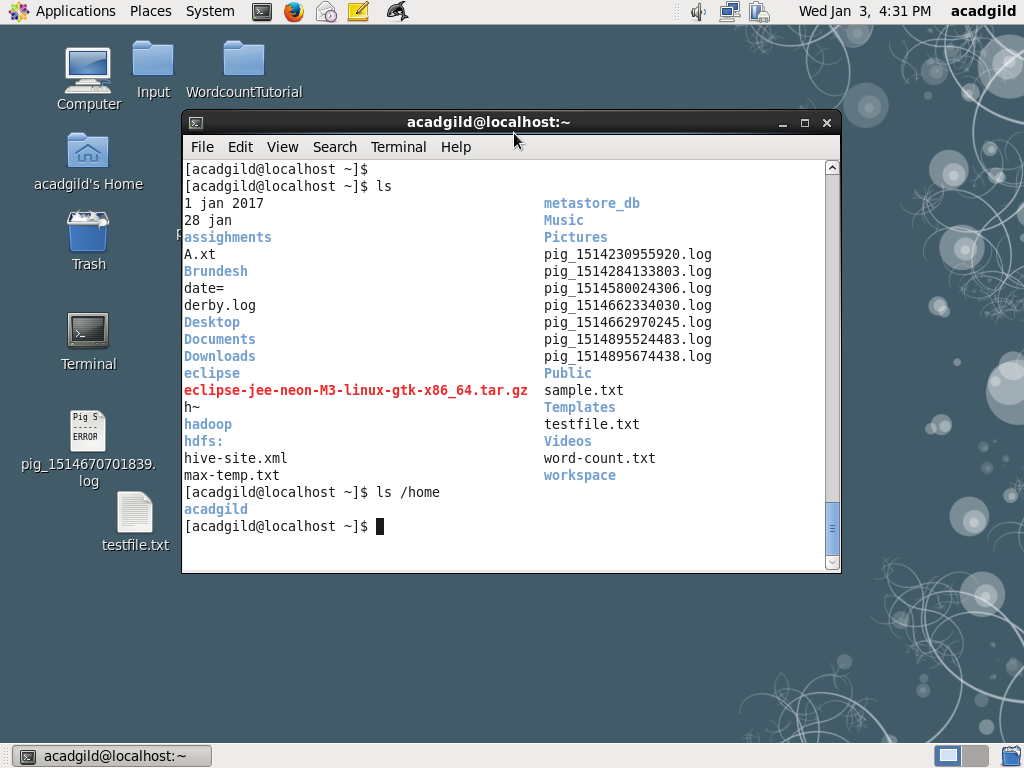
**Example: rm A.txt**

****

**6. ls**

* It lists the content of directory.
* It can used directly or for any specific directory

Syntax: ls (or) ls /home

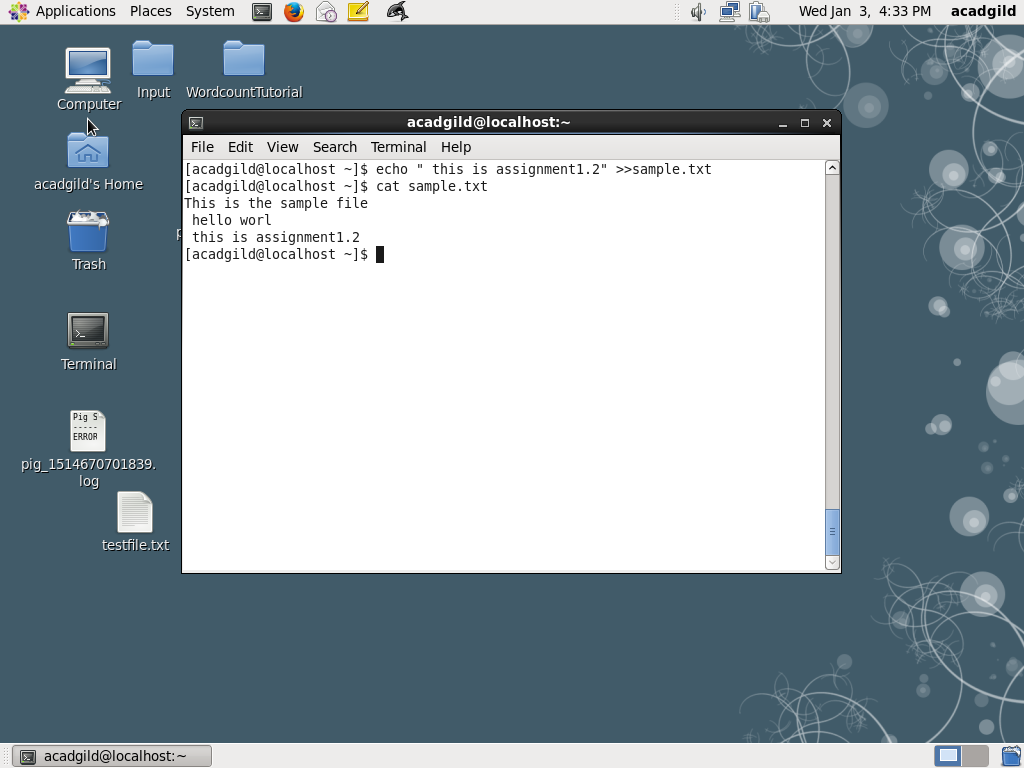


**7. echo**

It is used to write data to a new file or already existing file directly from command line.

Syntax: echo “ some text “ >> file.txt

Example echo “ hello world” >> sample.txt



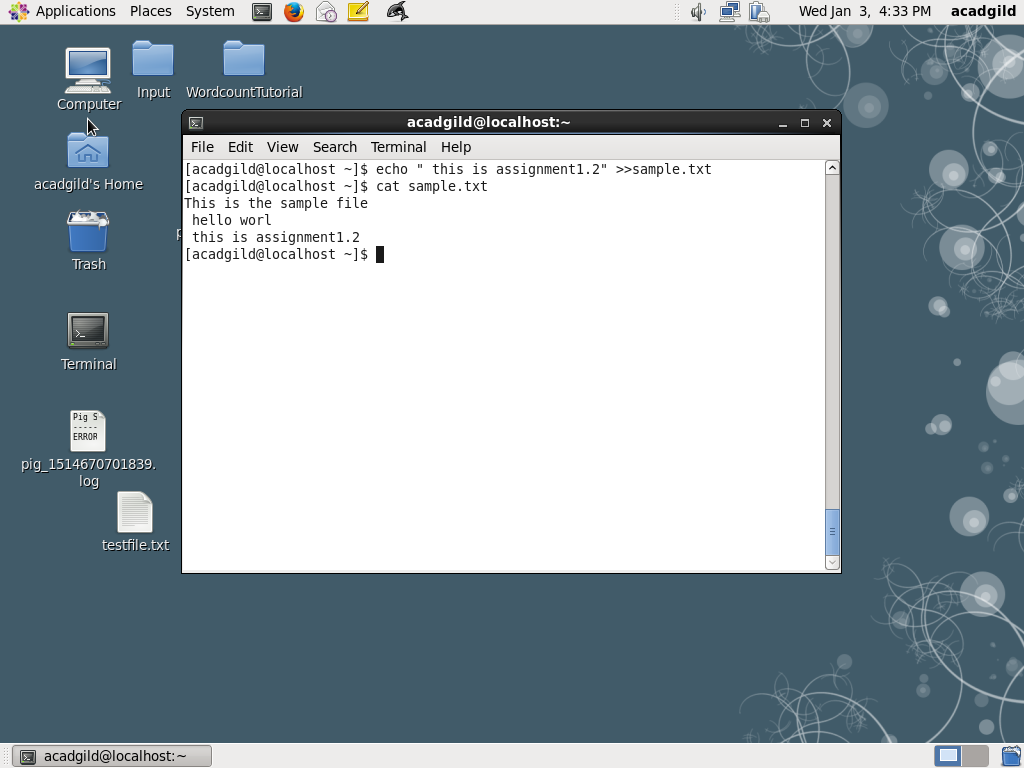
**8. cat**

This command is used to read data from file

Syntax; cat filename

In hadoop: hadoop fs –cat filename

Example: cat sample.txt

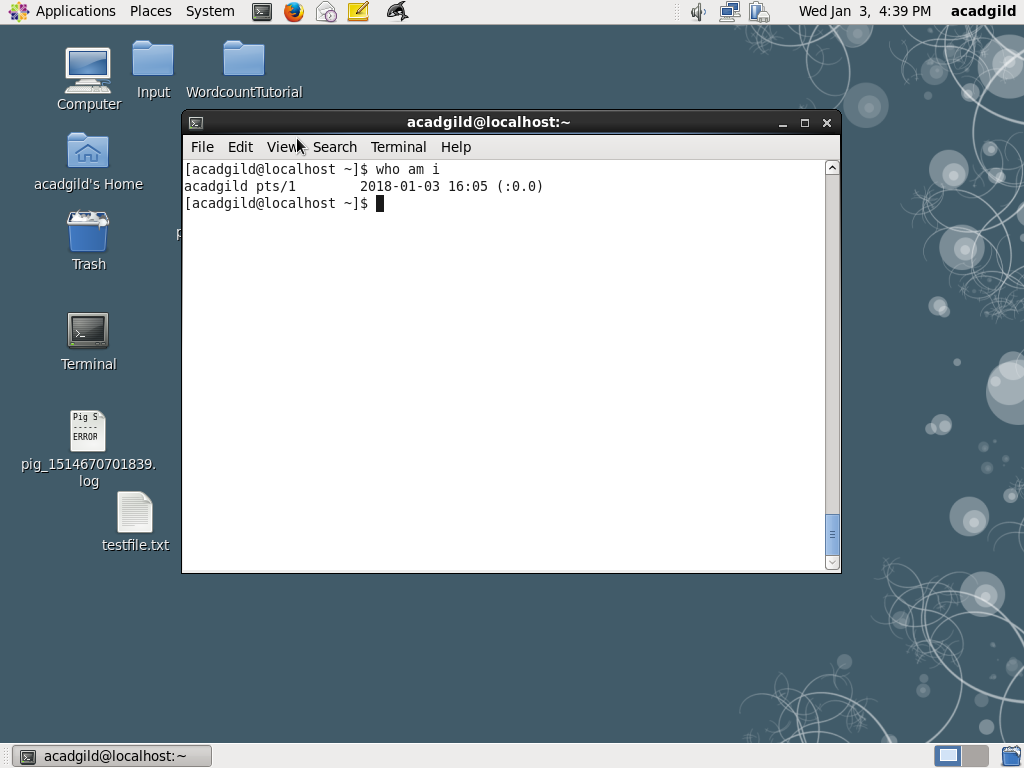


**9. who**

* Displays who is [logged on](https://www.computerhope.com/jargon/s/signon.htm) to the [system](https://www.computerhope.com/jargon/s/system.htm).
* The who command prints information about all users who are currently logged in.

## who Syntax

who [ *OPTION* ]... [ *FILE* ] [ am i ]

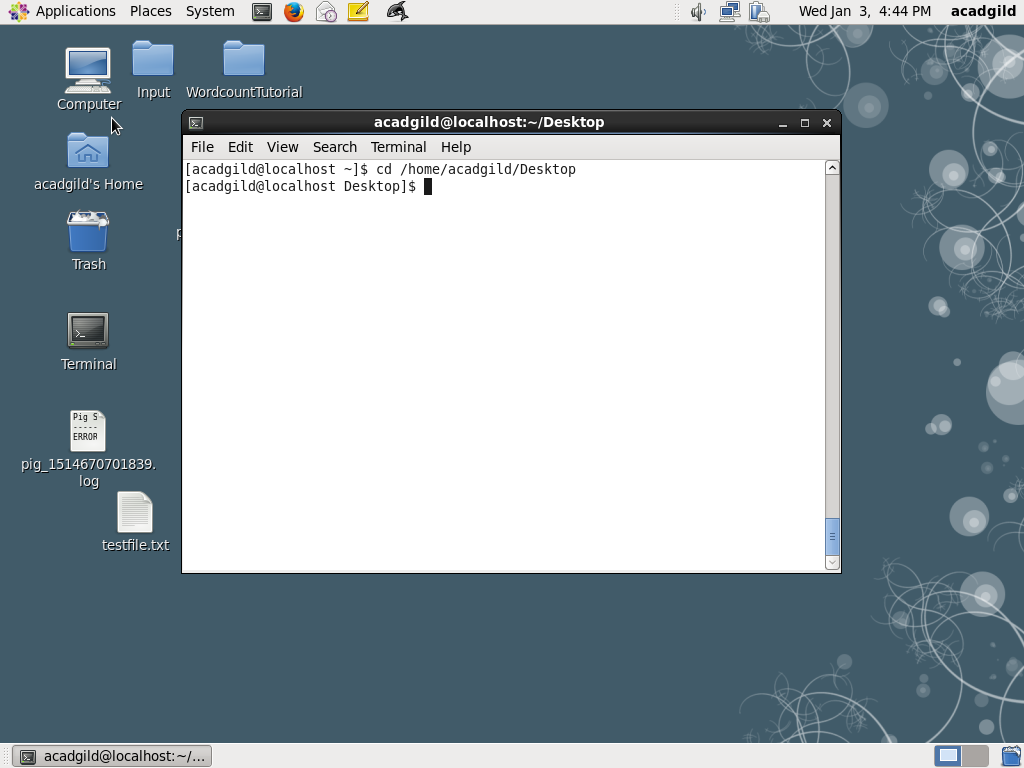
****

**10. cd**

The cd command, which stands for "change directory", changes the [shell](https://www.computerhope.com/jargon/s/shell.htm)'s[current working directory](https://www.computerhope.com/jargon/c/currentd.htm). It is a [builtin](https://www.computerhope.com/jargon/b/builtin.htm) command, which means that it is executed directly by your [shell](https://www.computerhope.com/jargon/s/shell.htm), instead of launching an external program.

## Syntax

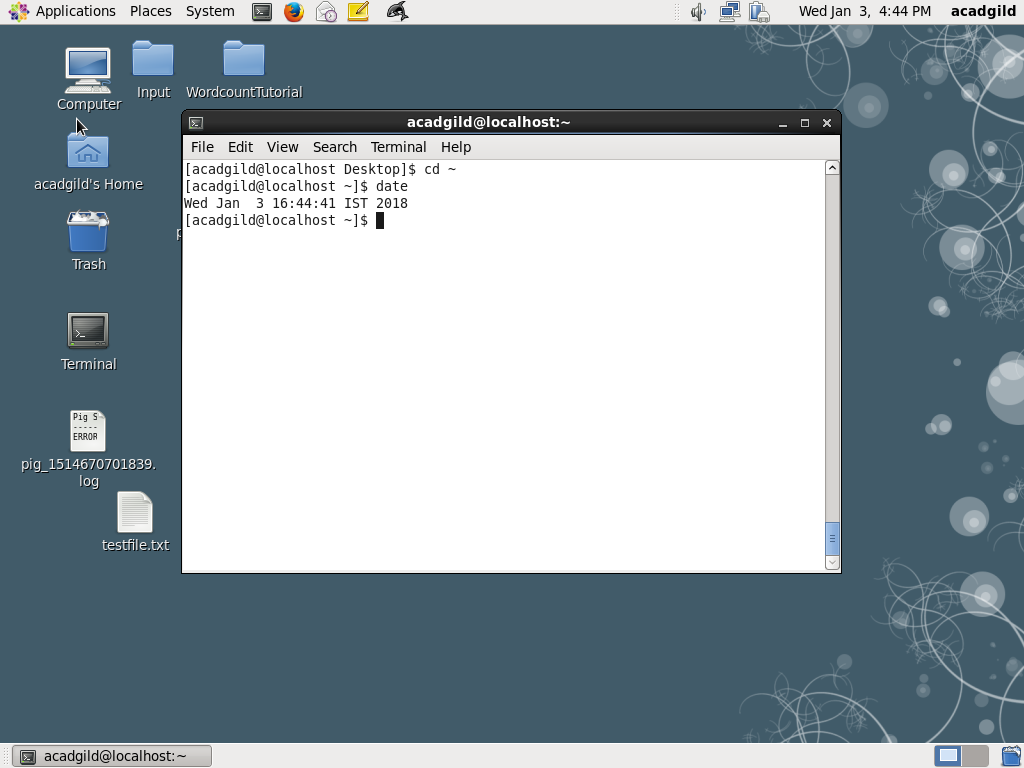
cd [-L | -P [-e]] *directory*



**11. date**

Displays date

Syntax : $ date

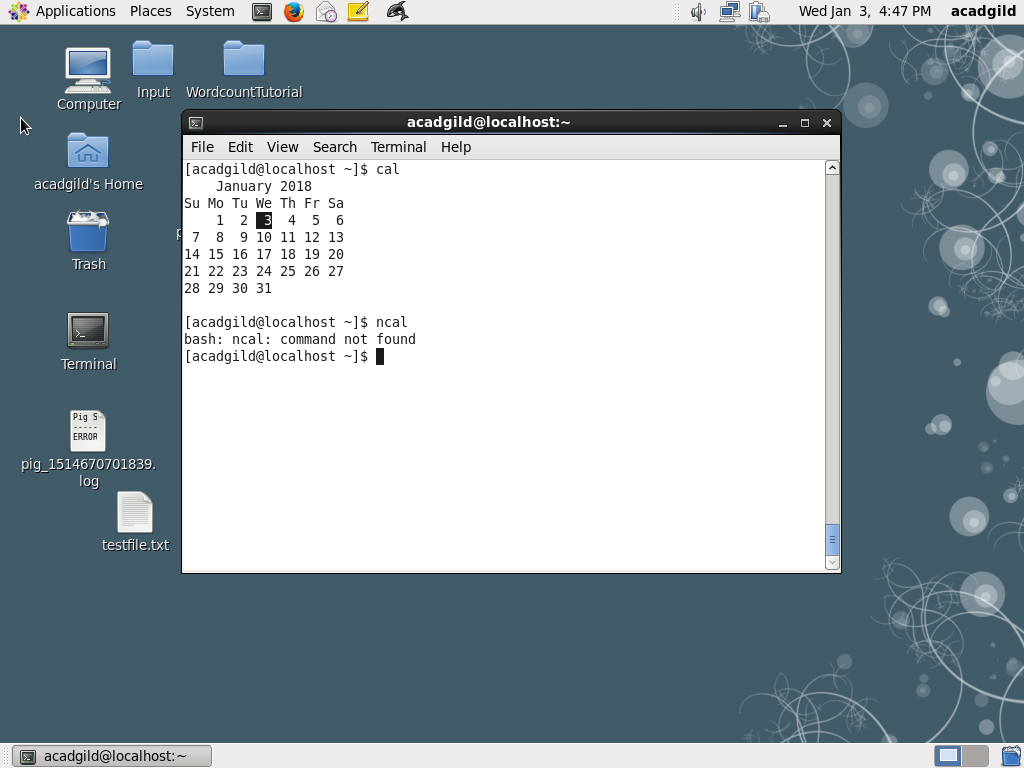


**12 cal**

The **cal** command displays a simple, formatted calendar in your [terminal](https://www.computerhope.com/jargon/t/terminal.htm).

## About ncal

In addition to **cal**, the **ncal** command ("new cal") is installed on some Linux systems. It provides the same functions of **cal**, but it can display the calendar vertically (with weeks in columns), and offers some additional options. On systems with **ncal**installed, **cal** is typically a [symbolic link](https://www.computerhope.com/jargon/s/symblink.htm) to **ncal**. It behaves like the original **cal** if you use the name **cal** to run the program.

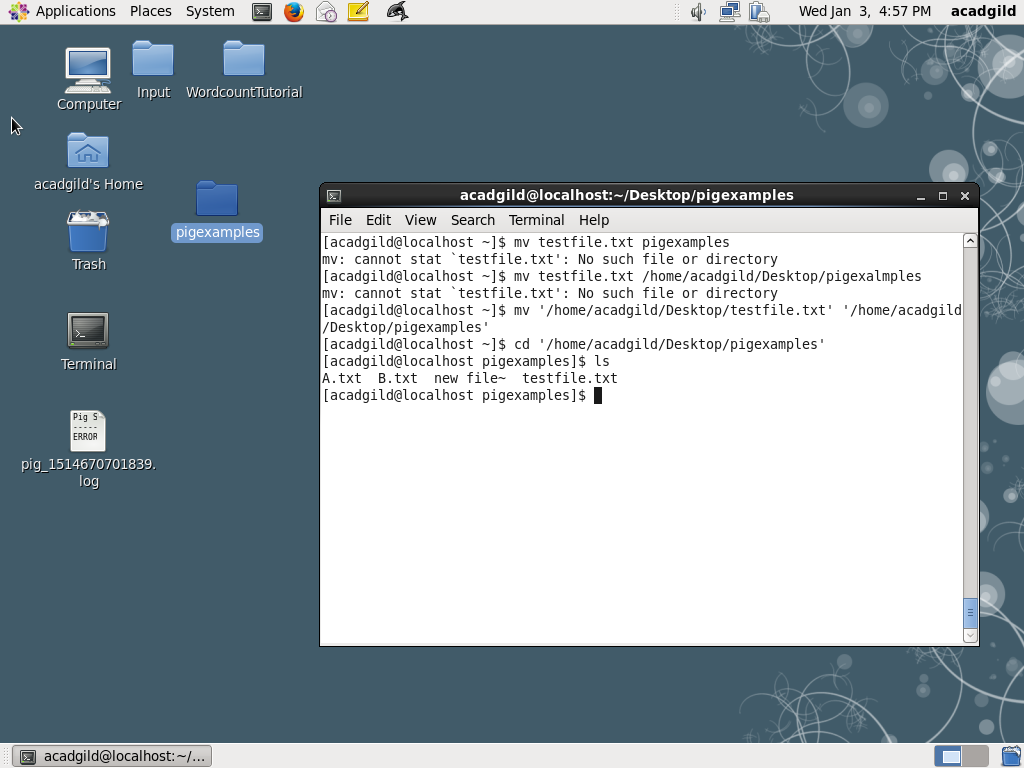
****

**13 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** [*options*] [**-T**] *source* *destination*

Example: mv A.txt assignments



**14. cp**

The **cp** command is makes copies of files and directories.

## Syntax

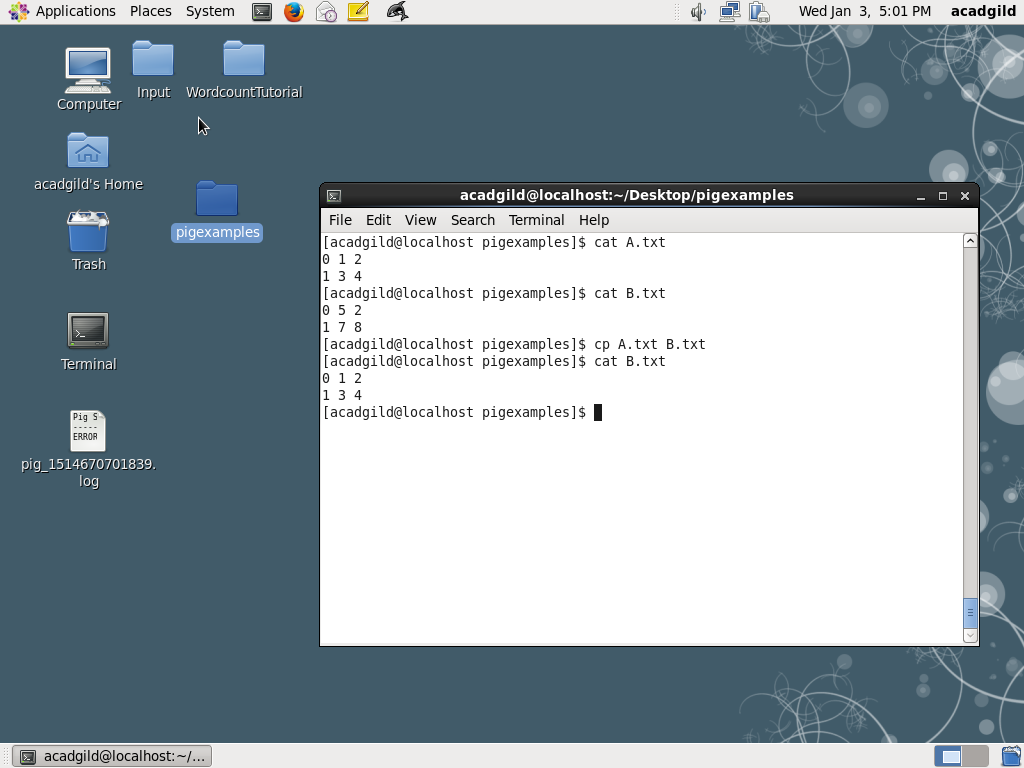
**cp** [*option*]... [**-T**] *source* *destination*

**cp** [*option*]... *source*... *directory*

**cp** [*option*]... **-t** *directory* *source*...

**cp --help**

**cp –version**



**15.which**

**which** returns the [pathnames](https://www.computerhope.com/jargon/p/path.htm) of the [files](https://www.computerhope.com/jargon/f/file.htm) (or [links](https://www.computerhope.com/unix/link.htm)) which would be [executed](https://www.computerhope.com/jargon/e/execute.htm) in the current [environment](https://www.computerhope.com/jargon/e/environm.htm), had the *filename* (or *filename*s) been given as a command (or commands) in a strictly [POSIX](https://www.computerhope.com/jargon/p/posix.htm)-conformant [shell](https://www.computerhope.com/jargon/s/shell.htm). It does this by searching the paths in the **PATH** [environment variable](https://www.computerhope.com/jargon/e/envivari.htm) for executable files matching the names of the arguments.

**which** does not follow [symbolic links](https://www.computerhope.com/jargon/s/symblink.htm).

## which syntax

which -a [*filename*] ...

Exit Status

**which** returns the following value, depending on what occurred:

|  |  |
| --- | --- |
| **0** | All *filename*s were found, and all were executable. |
| **1** | One or more *filename*s were not found, or were not executable. |
| **2** | An invalid option was specified. |

which examples

which sh

Locates the pathname of the file which would be run if the **[sh](https://www.computerhope.com/unix/ush.htm)** command were executed. On most systems, this will output:

/bin/sh

