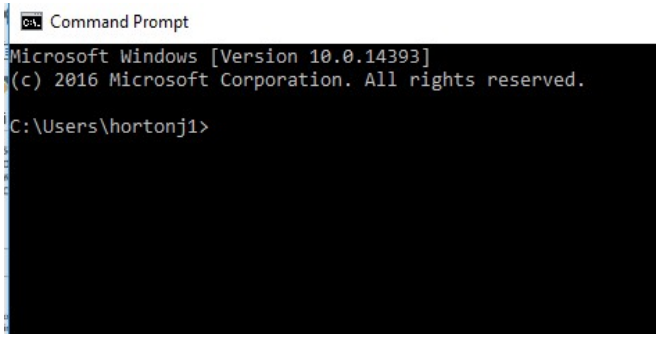
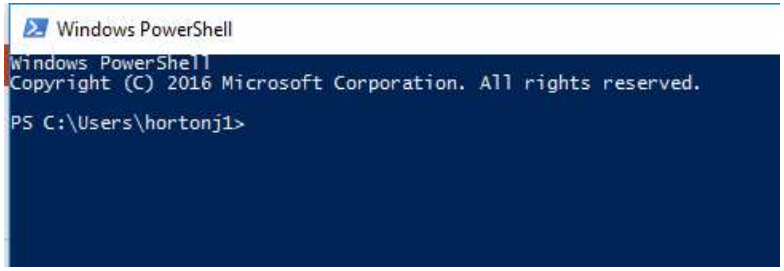
# Technical Comparisons – *Command Line Interface (CLI)*

**Microsoft Windows**

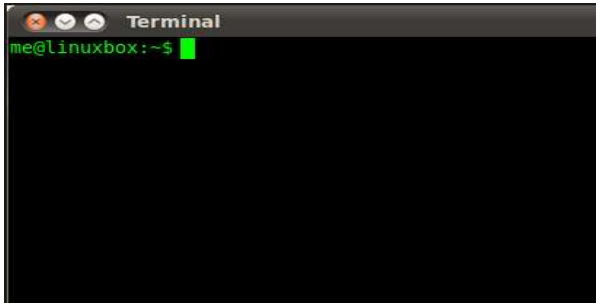
- Command Prompt (cmd)
- PowerShell
- [elevate as administrator]




**Linux Distros**

- Terminal
- *Note: there are also a wide variety of optional open source terminal packages available from various repositories.*
- [elevate as root via sudo]



# Why learn the Command Line Interface [Terminal]

- Commands are flexible and offer more options.  Many of these options would not be available in GUI.
- Performs more with a single command.  GUI often require multiple layered nesting to perform just one command.
- Work on multiple files/folders at a time.
- CLI loads and runs faster than GUI and does no require as much system resources as GUI does.  CLI will not affect system performance at all.
- CLI provides higher security than GUI as GUI has a visible and obvious attack surface.

# Terminal Help and Tips

**Way to access Terminal in Linux:**

- Go to Dash and search for Terminal
- Ctrl + Alt + T
- Right click on desktop, then menu select Terminal
- Note: can dock Terminal to Launcher for easy access

**To close and leave Terminal:**

```
$ exit
```

# Terminal Screen Scrolling

NOTE: Terminal in a GUI Linux Desktop can obviously be scrolled up and down via your mouse. But as Linux Servers DO NOT have a GUI, there is no mouse interaction to allow scrolling through Terminal in Server.

SOLUTION: You can use the keyboard combination, **SHIFT + Page Up and SHIFT + Page Down** to scroll through Terminal.

# Difference Between Ctrl D, Ctrl Z, Ctrl C

**Ctrl D: Used to terminate input or exit the terminal or shell**
**Ctrl Z: Used to suspend foreground processes**
**Ctrl C: Used to kill foreground processes**

# The History Command

```
guru99@VirtualBox:~$ history
    1  sudo apt-get install mailutils
    2  sudo apt-get install heirloom-mails
    3  sudo apt-get install heirloom-mailx
    4  sudo apt-get install mailutils
    5  sudo apt-get install postfix
    6  sudo apt-get install mailutils
    7  apt-get install mailutils
    8  sudo apt-get install mailutils
    9  sudo apt-get install heirloom-mailx
   10  mv test /home/guru99/Documents
   11  exit
   12  mv test /home/guru99/Documents
   13  chmod u-x Files
   14  chmod u-r Files
   15  mv test /home/guru99/Files
   16  chmod u+r Files
   17  chmod u+x Files
   18  chmod u-x Files
   19  chmod u-r Files
```


**The up and down arrow on the keyboard allows you to cycle through previous commands, one at a time.**

Accessing built-in command references inside of Terminal

Via man, help, and info, commands

Method 1 - Typing **<command> - -help** will give you give you full help for that command, including syntaxes and parameters (exactly similar to Windows).

Method 2 - Typing **man <command>** will invoke the Linux Manual text book installed on the Distro. This full-blown text book and manual.

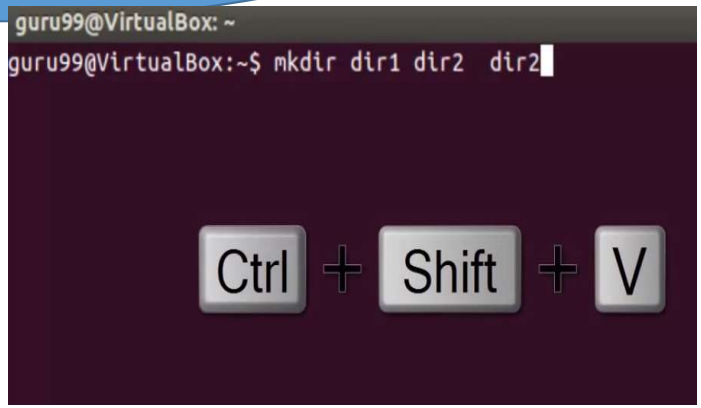
Method 3 - Typing **info <command>** will invoke the Linux Information book installed on the Distro. Another Linux text book.

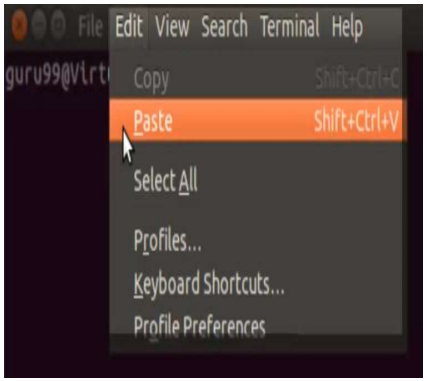
*Terminal invokes this by passing control over to a document reader whereas the reader actually opens the book. Therefore at completion of your read, to leave the reader and return back to Terminal, press Q to Quit (note: Linux document readers/editors require Quit process to close that reader/editor).*

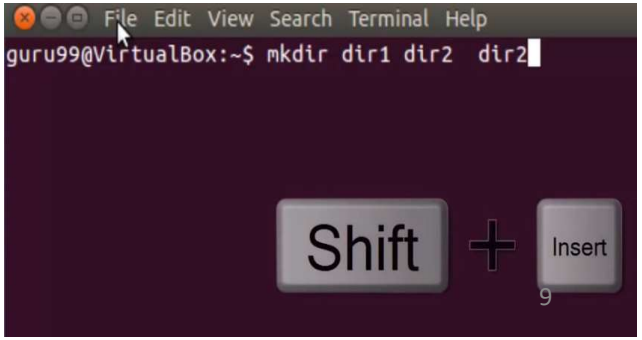
***Tip: In man and info, being that they are using a text reader, you can press / <text> to search for text within the page.***
***Example: man ping -> /ICMP***

# Copy and Pasting commands on the Terminal

The Admin LifeSaver

# The SUDO Command

Typical/Ordinary users *do not* have administrative privileges. If a specific command requires administrative root permissions in order to run, these users **must** elevate to that level via **sudo <command>**.

Of coarse, with all elevation of this type, password authentication will be required.

*Note: this elevation is only temporary during the execution of that specific command. Immediately after the command is completed, elevation and administrative access will end automatically.*

**Note**: For security reasons, the typed password is not displayed. You do not need to enter your password again for the next five minutes.

- Can access restricted files, install software and has administrative privileges

# Navigating Between Directories

***NOTE: Remember that capitalization matters extensively here !!!***

**Hint: With many commands, you can target the results per a different directory while still in your current directory: $ command /<target directory>**

# Path and its Types

A path in computing is the address of a file or folder.

E.g.:

C:\documentsandsettings\user\downloads in Windows or

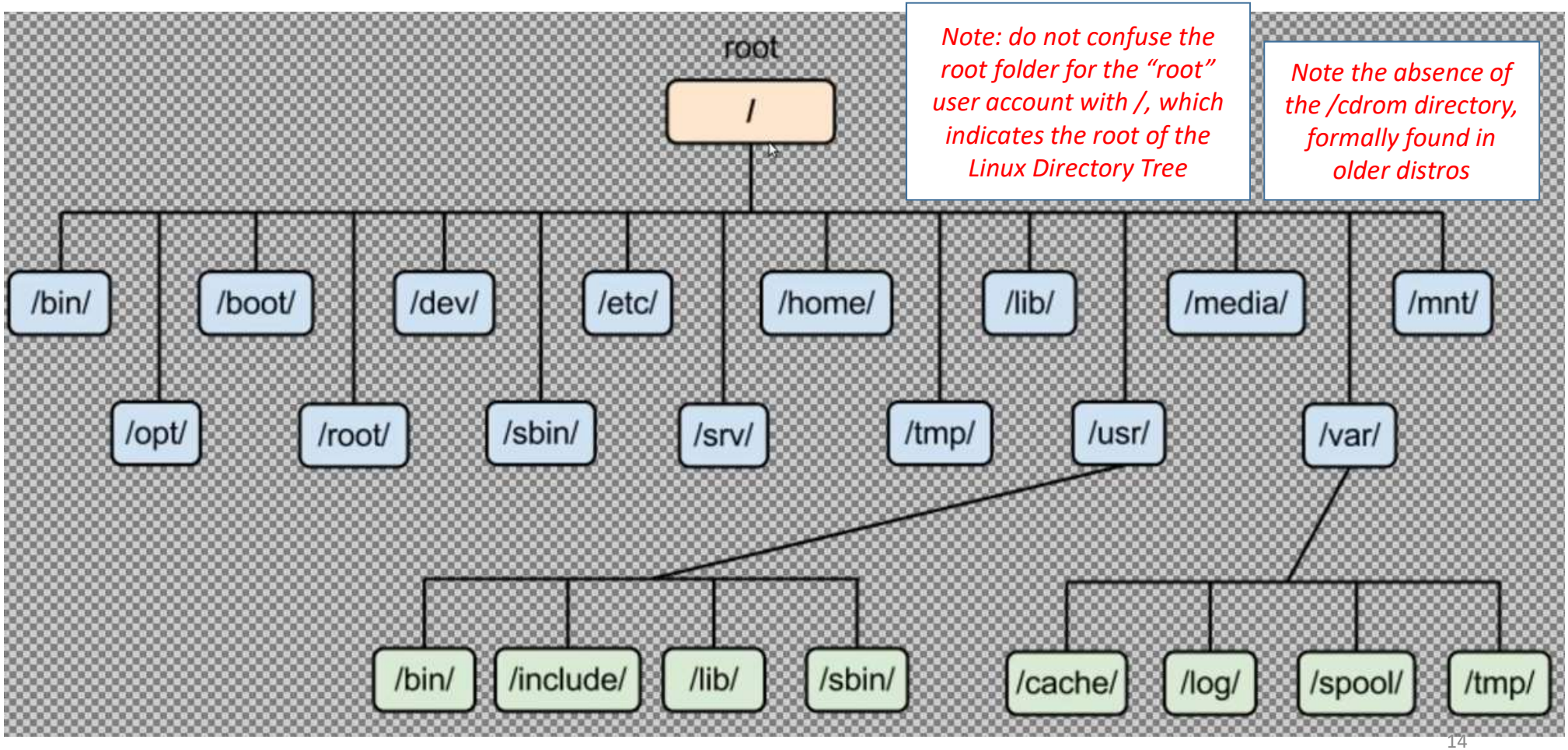
/home/user/downloads in Linux

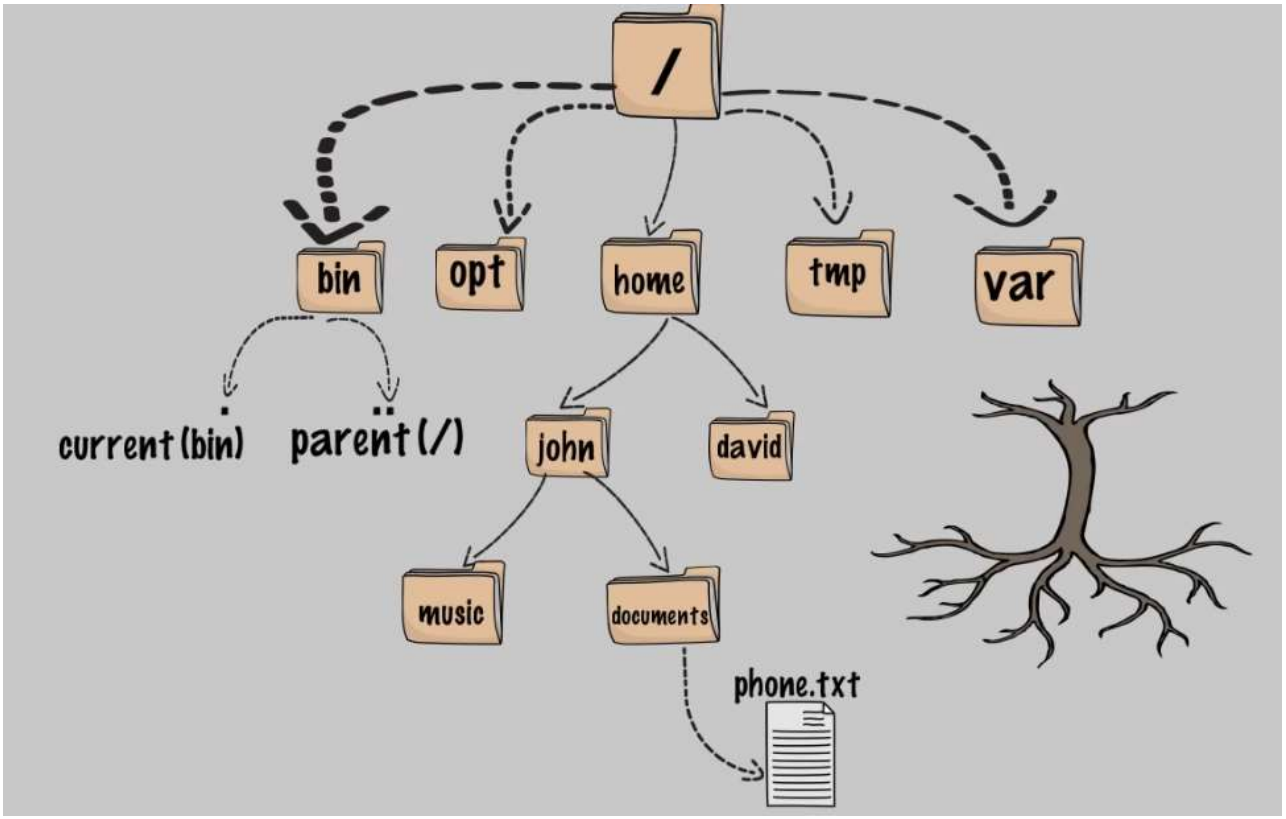
# Root /  C:\

- To navigate to the root directory, type cd / (the root directory in Linux is equivalent to the C:\ in Windows)
- To navigate within the current directory, type cd <directory>
- To navigate to any other directories, type cd /<directory>
- To navigate back to your home directory, type cd ~

*Note: if you don't know your current location, the command pwd will echo your location*

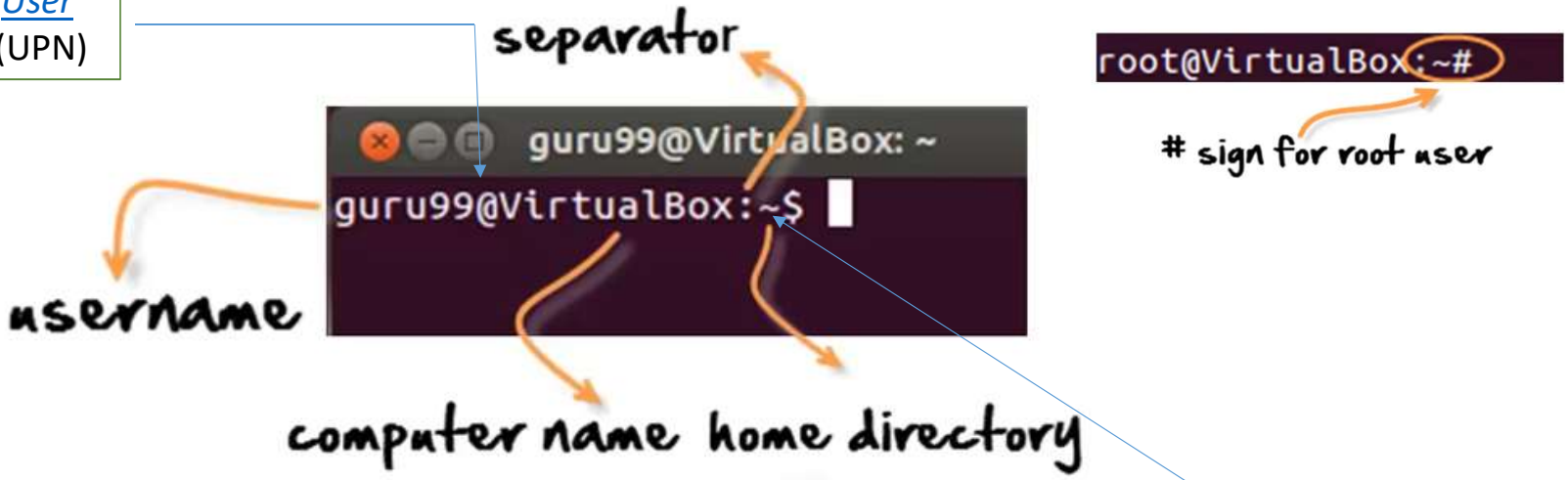
# The Linux Directory Tree - graphical breakdown




/
bin
opt
home
tmp
var
current (bin)
parent (/)
john
david
music
documents
phone.txt

# Understanding Default Terminal Command

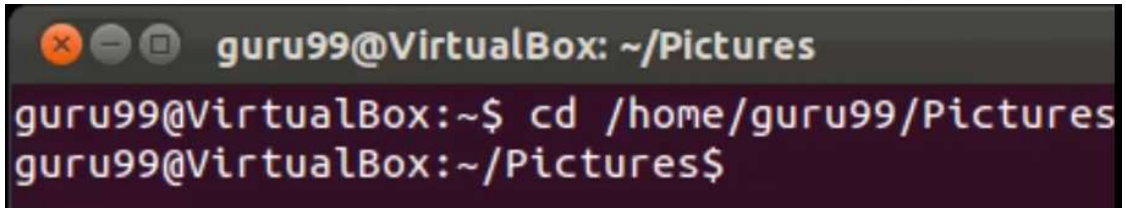
In Windows this is referred to as a *User principal name* (UPN)



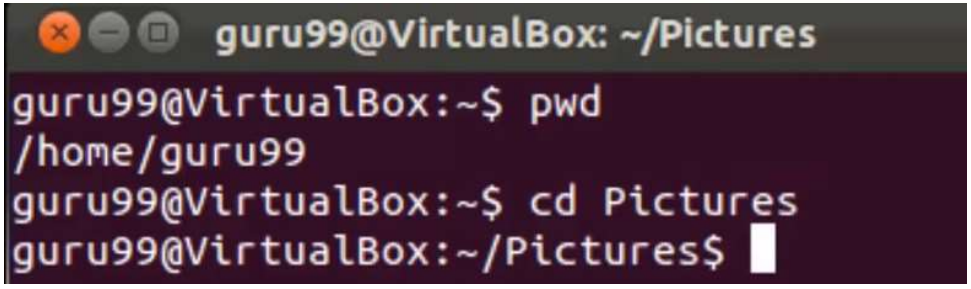

If you move from your home directory, this will disappear.

# Absolute & Relative Paths

| | | |
|---|---|---|
| **Absolute Path** | The full path to reach a directory. | **/home/guru99/Pictures** |




| | | |
|---|---|---|
| **Relative Path** | The path to reach an already open directory. | **/Pictures** if you are already in **/home/guru99** |



## ABSOLUTE VS RELATIVE PATHS

An absolute path begins with the root directory and follows the directory tree branch by branch until the path to the desired directory or file is completed

EXAMPLE /home/john/documents/phone.txt

On the other hand, a relative path starts from the current working directory

EXAMPLE

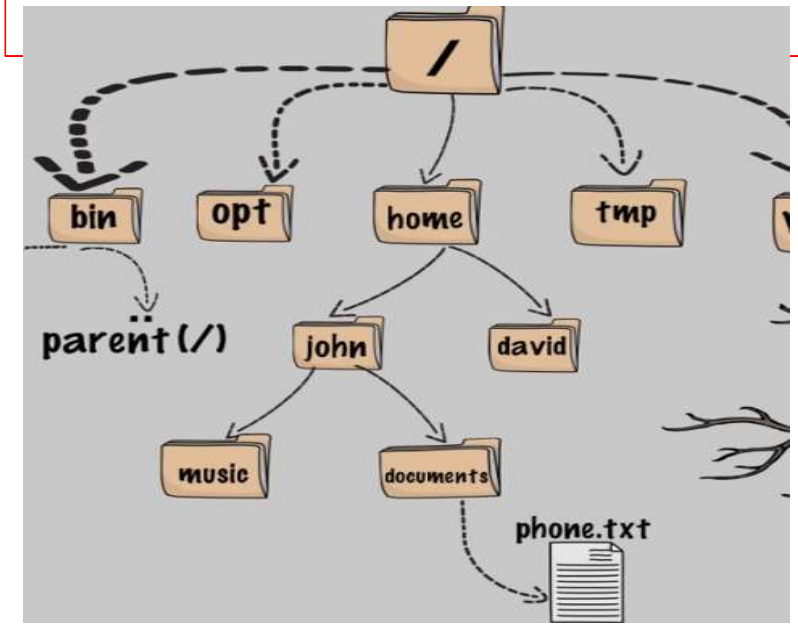
If our current working directory is john then we access the file phone.txt by the relative path ./documents/phone.txt

Navigate to directories within the current directory only

## basic cd command

1) cd ~
2) $ cd ***Music***

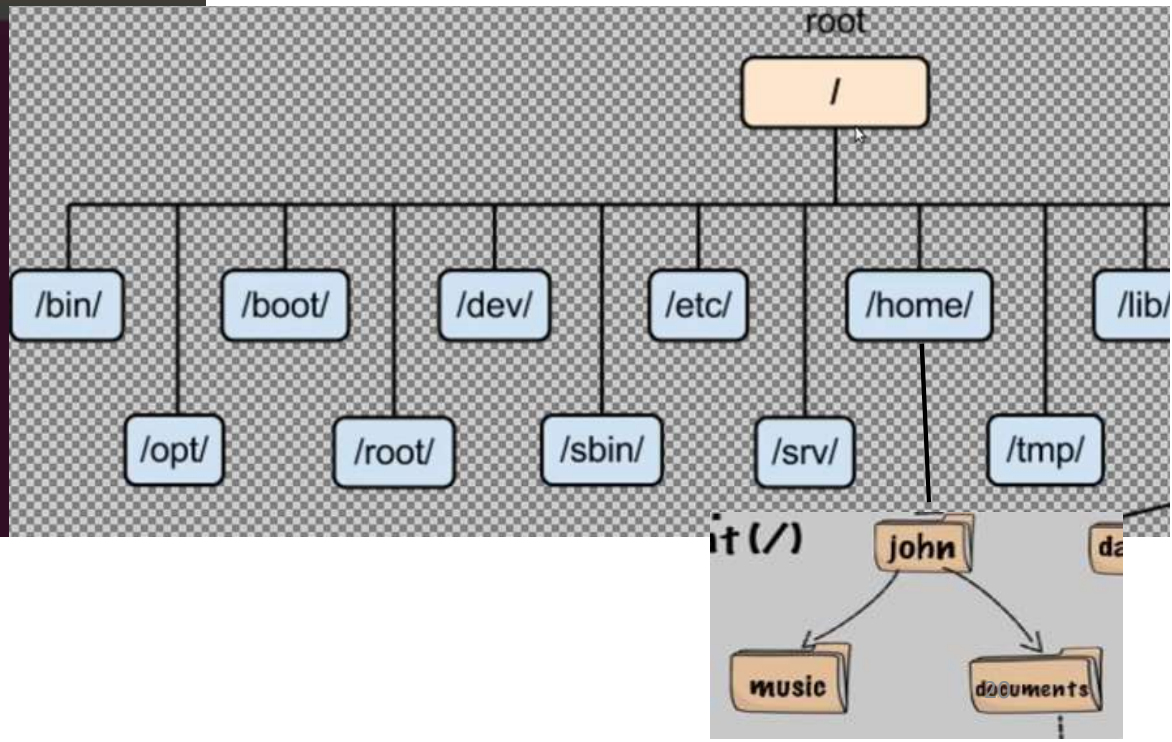
[*lets do this cd exercise at home !!* ]
[**change directory** to the Music directory within your current home directory.



Navigate to directories other than the current directory (use /)

**Hint:** **With many commands, you can target the results per a <u>different</u> directory while still in your current directory:**
**$ command /<target directory>**

```
guru99@VirtualBox: /
guru99@VirtualBox:~$ cd /tmp
guru99@VirtualBox:/tmp$ cd /bin
guru99@VirtualBox:/bin$ cd /tmp
guru99@VirtualBox:/tmp$ cd ~
guru99@VirtualBox:~$ cd /
guru99@VirtualBox:/$
```



# Navigating through multiple nested directories (use /)

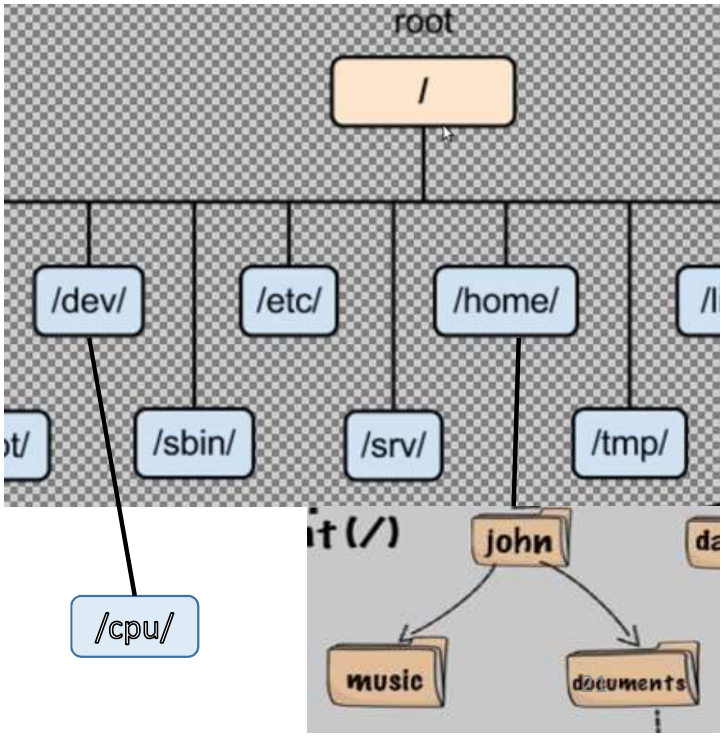
**To go down multiple directories in a target, type cd /<dir1>/<dir2>**

```
guru99@VirtualBox: /dev/cpu
guru99@VirtualBox:~$ cd /dev/cpu
guru99@VirtualBox:/dev/cpu$
```

**To move up one directory from where you are, type cd ..**

```
guru99@VirtualBox: /dev
guru99@VirtualBox:~$ cd /dev/cpu
guru99@VirtualBox:/dev/cpu$ cd ..
guru99@VirtualBox:/dev$
```

**Hint:** **With many commands, you can target the results per a different directory while still in your current directory:**
**$ command /<target directory>**



# Listing Directory Contents

***NOTE: Remember that capitalization matters extensively here !!!***

**Hint: With many commands, you can target the results per a different directory while still in your current directory: $ command /<target directory>**

# List a Directory's Contents

**ls command - list contents of a directory**


**ls -l command - list contents with full details and attributes for each item**

**Hint: You can also use ls command to list the contents of a different directory while still in your current directory: $ ls /<directory>**

# Linux Directory Color Codes via ls command

**All list commands echo their results to the screen with a dual color scheme, with one color denoting directories and another color denoting files.** ***Note: this dual color scheme may vary from Distro to Distro.***

**Blue**: Directory

**Green**: Executable or recognized data file

**Sky Blue**: Linked file

**Yellow with black background**: Device

**Pink**: Graphic image file

**Red**: Archive file

To turn the color off, you have to comment out the following lines in `.bashrc`.

```
# enable color support of ls and also add handy aliases #if [ -x
/usr/bin/dircolors ]; then # test -r ~/.dircolors && eval
"$(dircolors -b ~/.dircolors)" || eval "$(dircolors -b)" # alias
ls='ls --color=auto' # #alias dir='dir --color=auto' # #alias
vdir='vdir --color=auto' # # alias grep='grep --color=auto' #
alias fgrep='fgrep --color=auto' # alias egrep='egrep --
color=auto' #fi
```

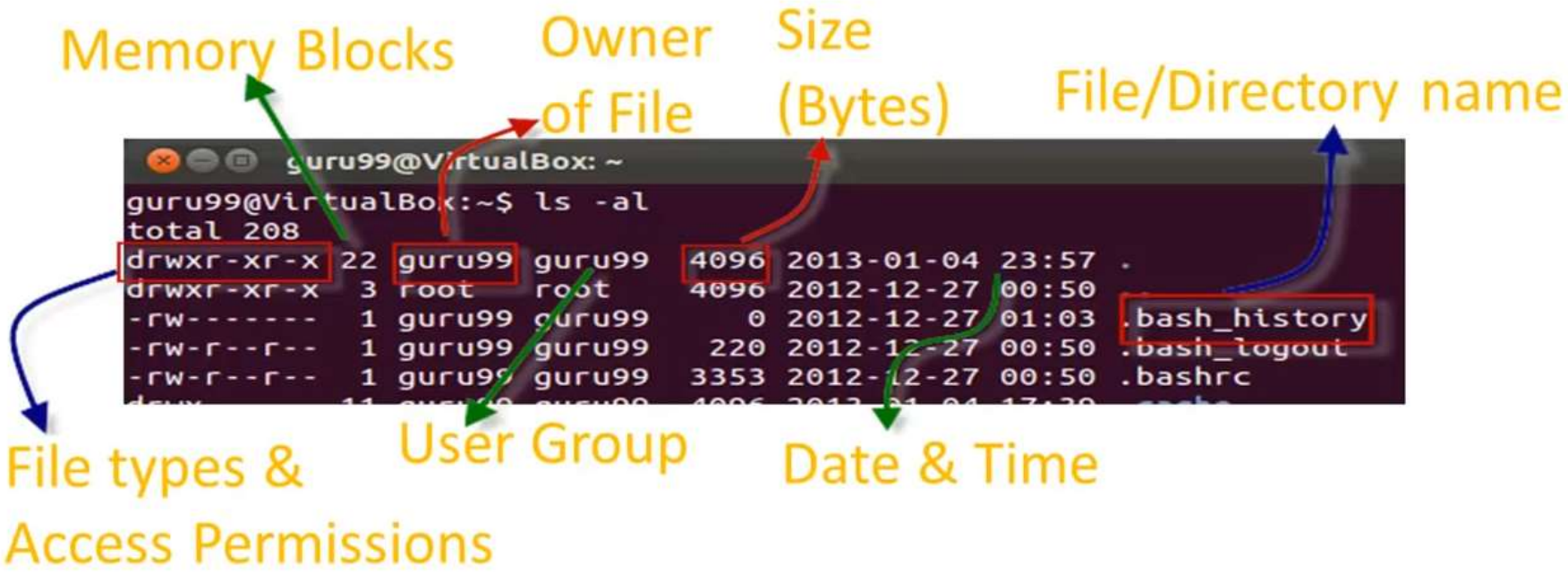
# Viewing Hidden Files

Hidden files start with '.' period symbol

**ls -a command - list contents including hidden files**

```
guru99@VirtualBox:~$ ls -a
.                .dbus              .fontconfig      .libreoffice      Publ
..               Desktop            .gconf           .local            .pul
.bash_history    .dmrc              .gnome2          .mission-control  .pul
.bash_logout     Documents          .gstreamer-0.10  .mozilla          .suc
.bashrc          Downloads          .gtk-bookmarks   Music             Temp
.cache           .esd_auth          .gvfs            Pictures          Test
.config          examples.desktop   .ICEauthority   .profile          .thu
guru99@VirtualBox:~$
```

**ls -al command - combines ls -l and ls -a (full details and attributes, including hidden files):**



# Listing Subdirectories/Files


**ls -R command - used to list all files contained in a nested directory structure. It will list all the files as well as list the directories and their sub-directories.**

# Manipulating Files

***NOTE: Remember that capitalization matters extensively here !!!***

**Hint: With many commands, you can target the results per a different directory while still in your current directory: $ command /<target directory>**

# Creating & Viewing Files

## cat command

***Note: the cat command is used for text files only.***

***The cat command is used to create, display, and combine text files***

**Do the cat command exercise !!!**
**Best-practice hint: Use ls commands throughout, to see your results as you go.**

- When you create a text file via cat, you will be prompted to enter the desired text into that file.
- After entry of the text, use Ctrl + D to terminate input.

| | |
|---|---|
| 1) cd ~ | [*lets do this cat exercise at home !!* ] |
| 2) $ cat > ***file1*** | [text file **create**, display and combine] |
| 3) Enter text *Hey I wrote this in file1* | |
| 4) $ cat > ***file2*** | [text file **create**, display and combine] |
| 5) Enter text *But I wrote this in file2* | |
| 6) $ cat ***file1*** | [text file create, **display** and combine] |
| 7) $ cat ***file2*** | [text file create, **display** and combine] |
| 8) $ cat ***file1 file2 > newfile*** | [text file create, display and **combine**] |

# Copy Files

## cp command – copy/backup files

**Lets do the cp command exercise !!!**
**Best-practice hint: Use ls commands throughout, to see your results as you go.**

| | |
|---|---|
| 1) cd ~ | [*lets do this cp exercise at home !!* ] |
| 2) $ cp ***file1 file1-backup*** | [**copy** the file to a new file] |
| 3) $ cp ***file1 File1*** | [**copy** the file to a new file] |

# Renaming Files

## mv command – rename files

**Lets do the mv command exercise – for renaming !!!**
**Best-practice hint: Use ls commands throughout, to see your results as you go.**

| | |
|---|---|
| 1) cd ~ | [*lets do this mv exercise at home !!* ] |
| 2) $ mv ***file1-backup file1-rename*** | [**rename** the file with a different file name] |

# Moving Files

**mv command (when used to move files, the mv command requires sudo elevation)**

**Lets do the mv command exercise – for moving !!!**
**Best-practice hint: Use ls commands throughout, to see your results as you go.**

| | |
|---|---|
| 1) cd ~ | [*lets do this mv exercise at home !!* ] |
| 2) $ sudo mv ***file1 Pictures*** | [**move** the file to your Pictures directory] |

Create a slide for exception when using the mv command, if a file or particularly a directory by the same name already exists,  eg: sudo mv file1 Pictures, if file or directory Pictures already exists.

# Deleting Files

## rm command – delete files

**Lets do the rm command exercise – for deleting !!!**
**Best-practice hint: Use ls commands throughout, to see your results as you go.**

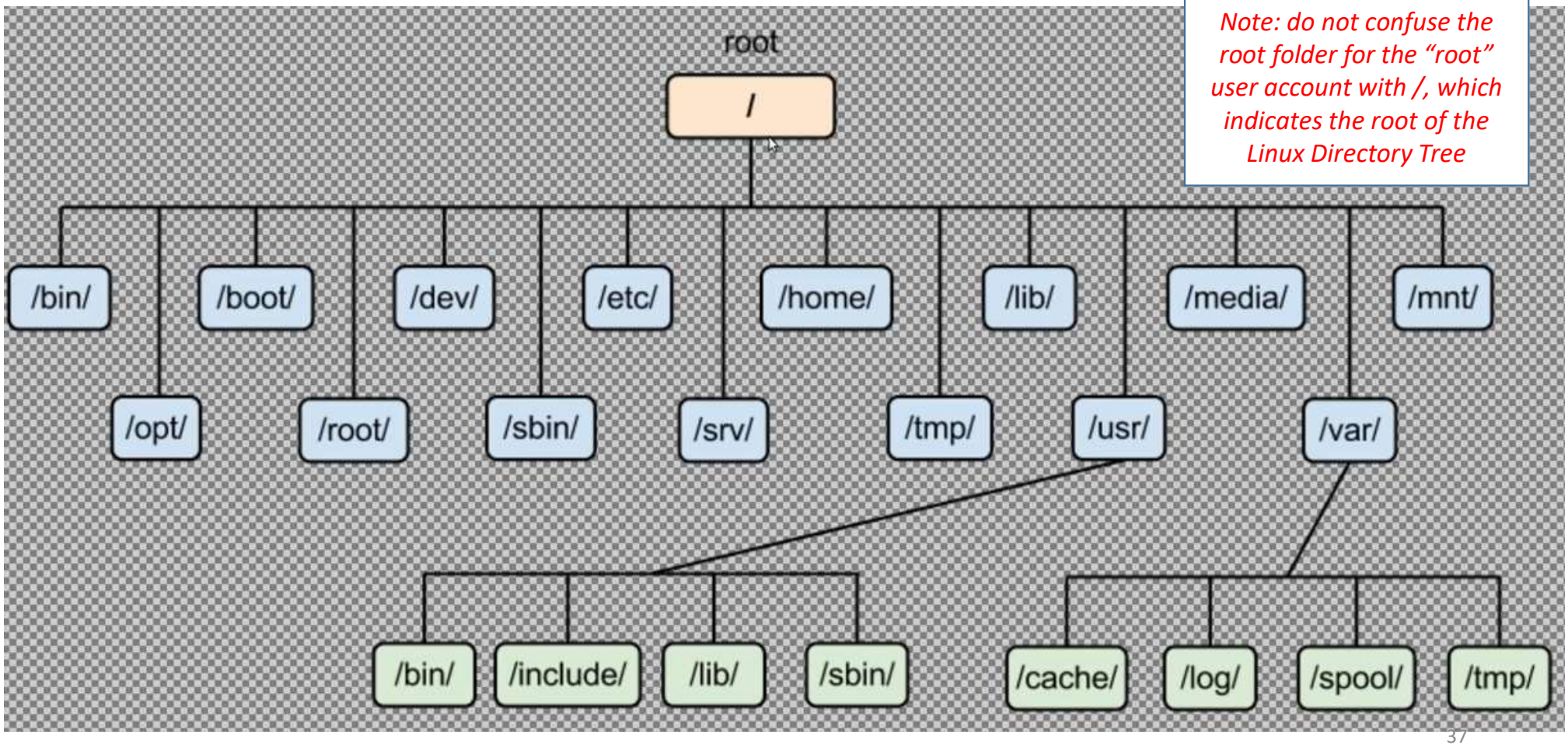
1) ~~cd ~~ [*remember, this file we are deleting is in now in your Pictures directory!!* ]
2) $ rm ***file1*** [**delete** the file from the directory]

# Manipulating Directories

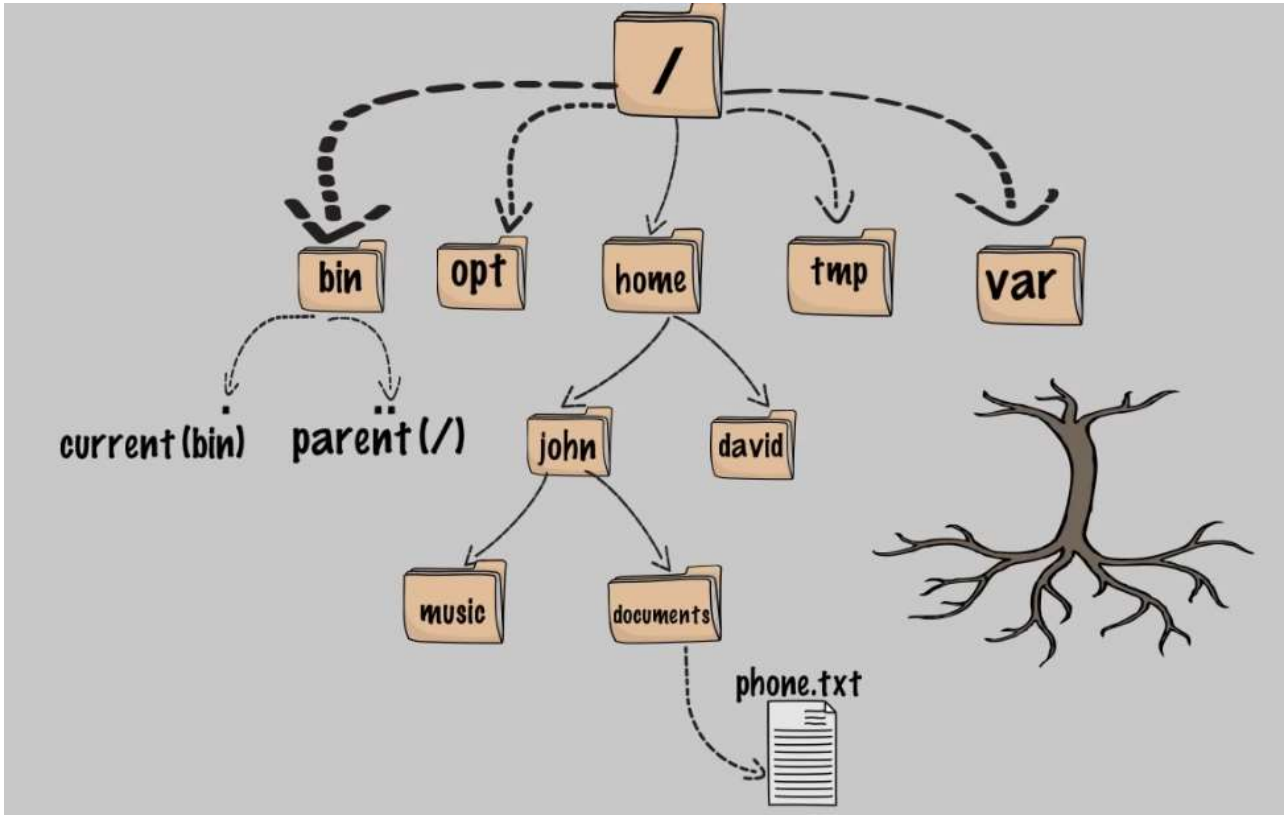
***NOTE: Remember that capitalization matters extensively here !!!***

**Hint: With many commands, you can target the results per a different directory while still in your current directory: $ command /<target directory>**

# The Linux Directory Tree - graphical breakdown

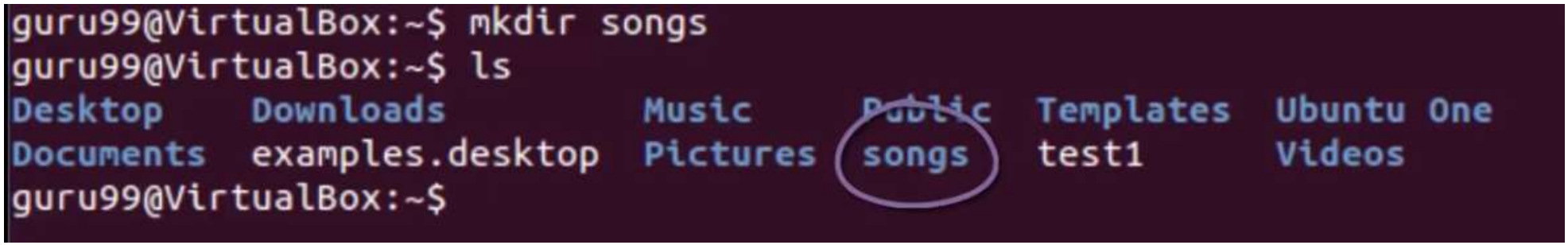



*Note: do not confuse the root folder for the "root" user account with /, which indicates the root of the Linux Directory Tree*


/
bin
opt
home
tmp
var
current (bin)
parent (/)
john
david
music
documents
phone.txt

# Creating Directories

**mkdir command – create a directory within the current directory**



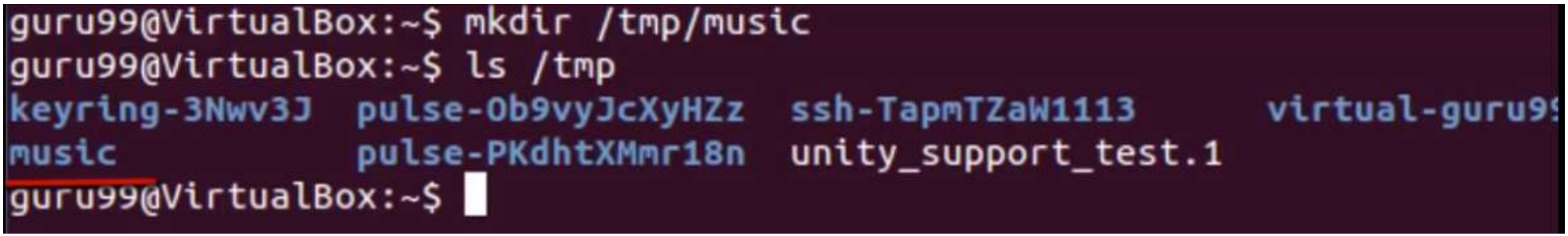
# Creating Multiple Directories

**mkdir command – create multiple directories within the current directory.**

```
guru99@VirtualBox:~$ mkdir direc direc1 direc2
guru99@VirtualBox:~$ ls
Desktop  direc1  Documents  examples.desktop  Pictures  songs      test1
direc    direc2  Downloads  Music              Public    Templates  Ubun1
guru99@VirtualBox:~$
```

# Creating directory at a new location

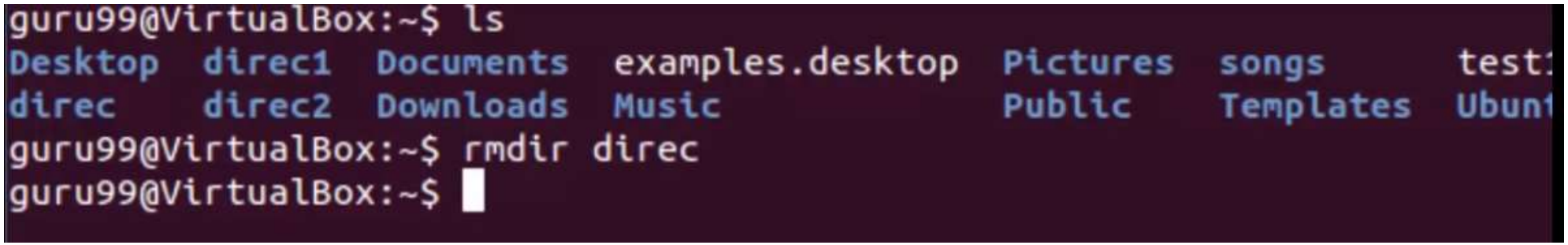
**mkdir command – create a directory in location other than the current directory**




**Hint:** With many commands, you can target the results per a different directory while still in your current directory: $ command /<target directory>

# Removing Directories

## rmdir command – delete directory




**Note: you CANNOT delete a directory that has content (files or sub-directories). You must remove all files and sub-directories BEFORE deleting the parent directory.**

# Renaming Directory

**mv command – rename directories**

```
guru99@VirtualBox:~$ ls
Desktop  direc2     Downloads         Music     Public  Templates  Ubunt
direc1   Documents  examples.desktop  Pictures  songs   test1      Videc
guru99@VirtualBox:~$ mv direc2 direc4
guru99@VirtualBox:~$
```

# Must Know Commands Summarized

Summarized by the order in this lesson

| Command | Description |
| --- | --- |
| ls | Lists all files and directories in the present working directory |
| ls – R | Lists files in sub-directories as well |
| ls –a | Lists hidden files as well |
| ls –la | Lists files and directories with detailed information |

| Command | Description |
| --- | --- |
| cat > filename | Creates a new file |
| cat filename | Displays the file content |
| cat file file2 > file3 | Joins two files (file1, file2) and stores the output in a new file (file3) |
| mv file "new file path" | Moves the files to the new location |

| Command | Description |
| --- | --- |
| mv filename new_file_name | Renames the file to a new filename |
| sudo | Allows regular users to run programs as superuser or root |
| rm <filename> | Deletes a file |
| mkdir <directoryname> | Creates a new directory in the present working directory |

| Command | Description |
| --- | --- |
| mkdir <path/directory> | Create a new directory at the specified path |
| rmdir <directoryname> | Deletes a directory |
| mv <name> <new_name> | Renames a directory |
| man <command> | Gives help information on a command |

| Command | Description |
| --- | --- |
| history | Gives a list of all past commands typed in the current terminal session |
| cp <name> <new_name> | Copy a file |
| clear | Clears the terminal |