APPLIED OPERATING SYSTEM LABORATORY

MODULE 4

LINUX FILE SYSTEM, FILENAMES AND WILDCARDS

OBJECTIVES

Upon completion of this module, the student will be able to:

- Understand LINUX file system/structure, partition, filename format, absolute and relative paths
- Create different file types and display directory content using wildcards in LINUX commands

TOPIC OUTLINE

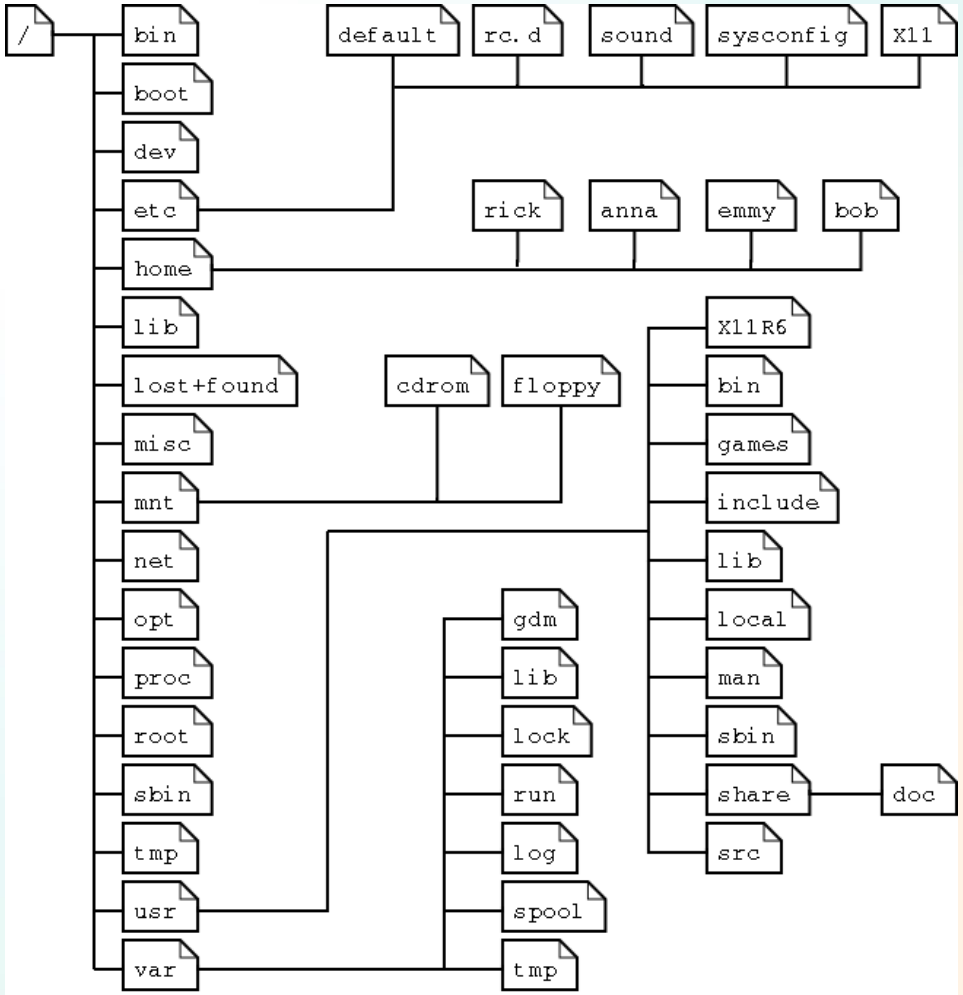
- **Linux File System**
  - Directory Structure / Layout
  - Absolute and Relative Path
  - Filename Convention and File Types
- **How to create different files**
  - Empty file
  - Ordinary/Regular file
  - Hidden files
- **Others**
  - Autocomplete
  - Wildcards

LINUX FILE SYSTEM

Linux File System Layout

The tree of the file system starts at the trunk or **forward slash** (/). This directory, containing all underlying directories and files, is also called the **root directory** or **"the root"** of the file system.

```
vetcha@DESKTOP-U1V5HO4:~$ cd /
vetcha@DESKTOP-U1V5HO4:/$ ls
bin   dev  home  lib    lib64   media  opt   root  sbin  srv  tmp  var
boot  etc  init  lib32  libx32  mnt    proc  run   snap  sys  usr
```



LINUX FILE SYSTEM

Partition

This is a logical division of a hard disk created so that you can have different operating system on the same hard disk, or to create the impression of having separate hard drives for file management, multiple users or other purposes.

File System

This refers to the way in which files are named and where they are placed logically for storage and retrieval.

LINUX FILE SYSTEM

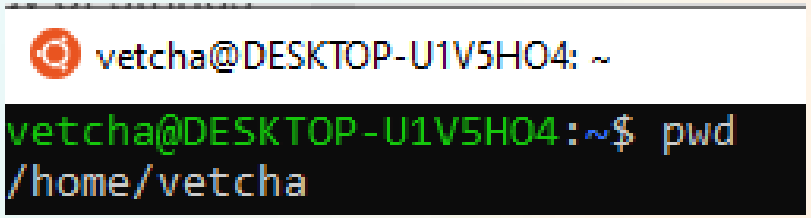
A **path** is a unique location to a file or a folder in a file system of an OS.

Absolute Path

The highest directory in the Linux directory tree is the /. An absolute path is defined as specifying the location of a file or directory from the root directory(/).

To write an absolute path-name:

- Start at the root directory (/) and work down
- Write a slash (/) after every directory name.

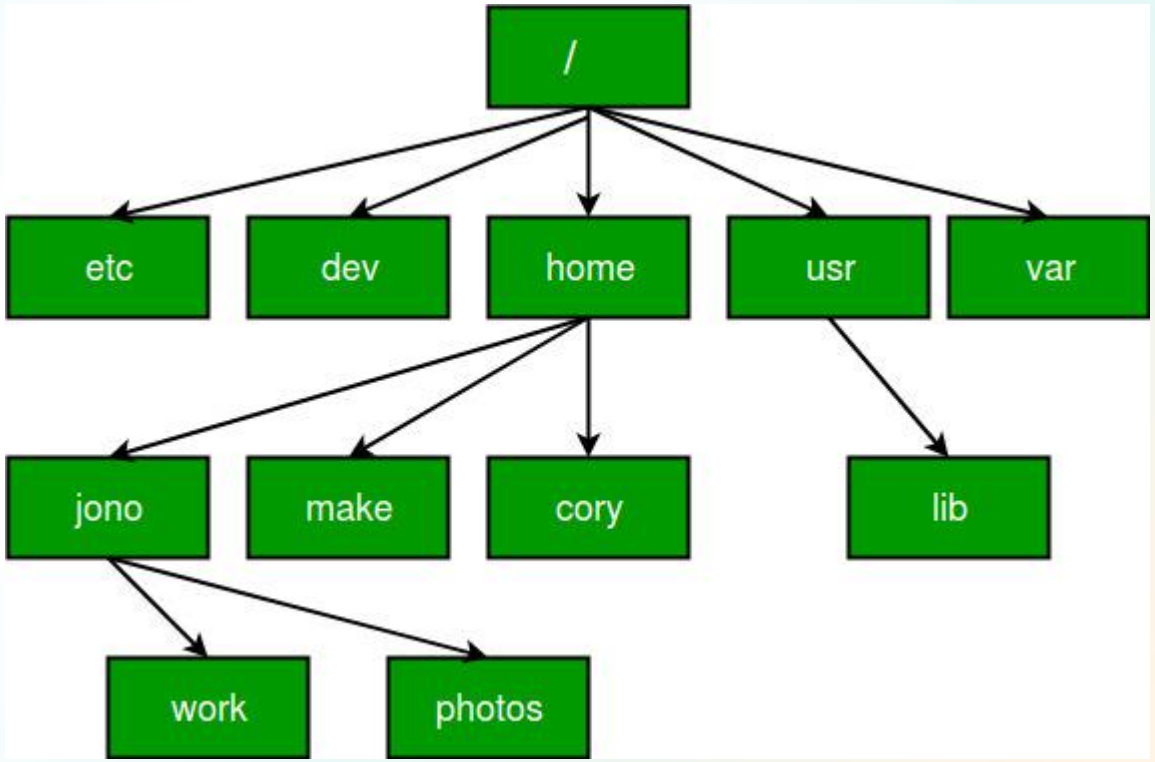


LINUX FILE SYSTEM

Relative Path

It is defined as the path related to the present working directory. It starts at your current directory and never starts with a / .

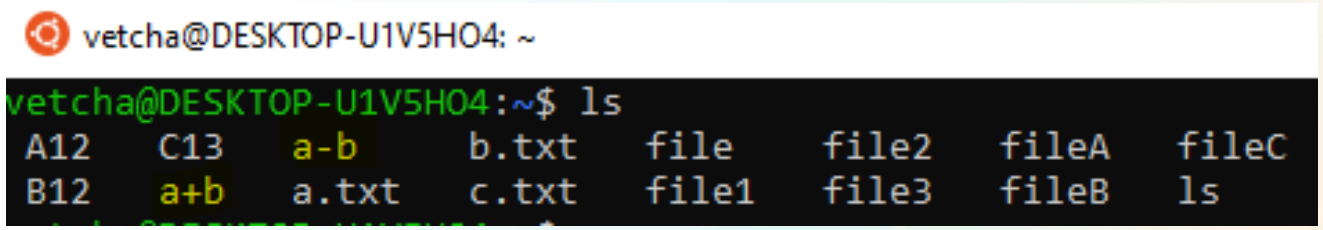
```
vetcha@DESKTOP-U1V5HO4:~$ ls
 A12        'Folder 1'    empty.txt    fileA
 B           a+b          empty1       fileB
 B12         a-b          empty2       fileC
 C           a.txt        file         first
 C13         b.txt        file1        hello.txt
 FOLDER      c.txt        file2        hi.txt
 FOLDER2     elif         file3        his.txt
vetcha@DESKTOP-U1V5HO4:~$ cd FOLDER
vetcha@DESKTOP-U1V5HO4:~/FOLDER$ cd /usr/lib
vetcha@DESKTOP-U1V5HO4:/usr/lib$
```



LINUX FILE SYSTEM

Filename Convention

- All file names are **case sensitive**. So filenames **sam.txt Sam.txt SAM.txt** all are three different files.
- You can use upper and lowercase letters, numbers, “.” (dot), and “_” (underscore) symbols.
- You can use other special characters such as blank space except / (root directory), but they are hard to use and it is better to avoid them.
- Most modern Linux and UNIX limit filename to 255 characters (255 bytes). However, some older version of UNIX system limits filenames to 14 characters only.

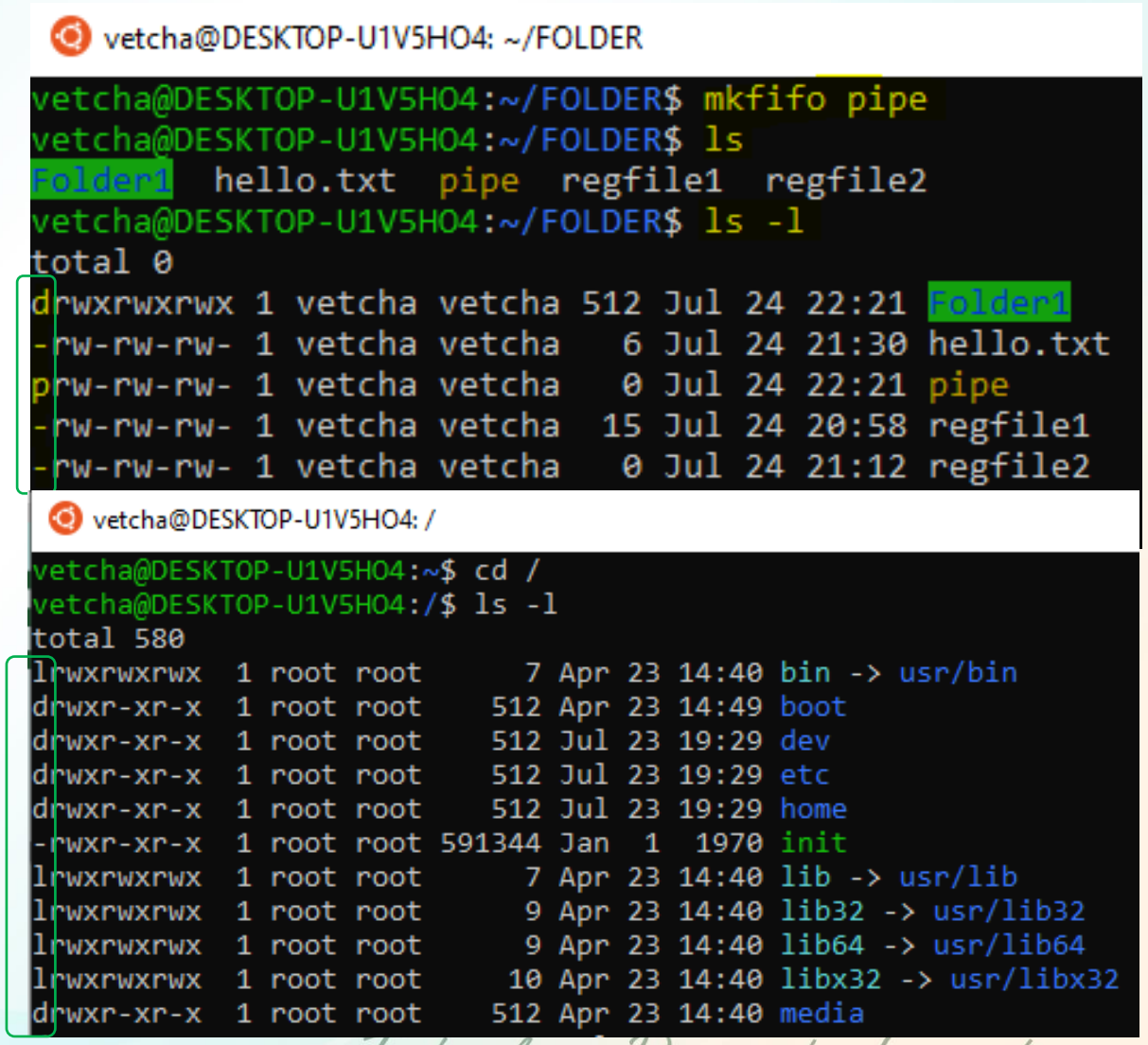


LINUX FILE SYSTEM

File Types

Seven different types of Linux file types and **ls** command identifiers:

- **-** : regular file
- **d** : directory
- **c** : character device file
- **b** : block device file
- **s** : local socket file
- **p** : named pipe
- l : symbolic link

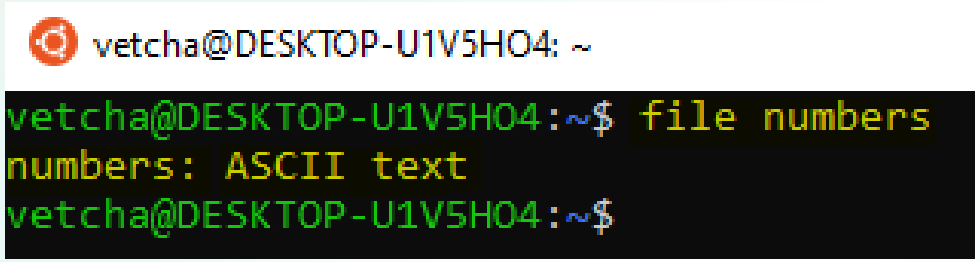


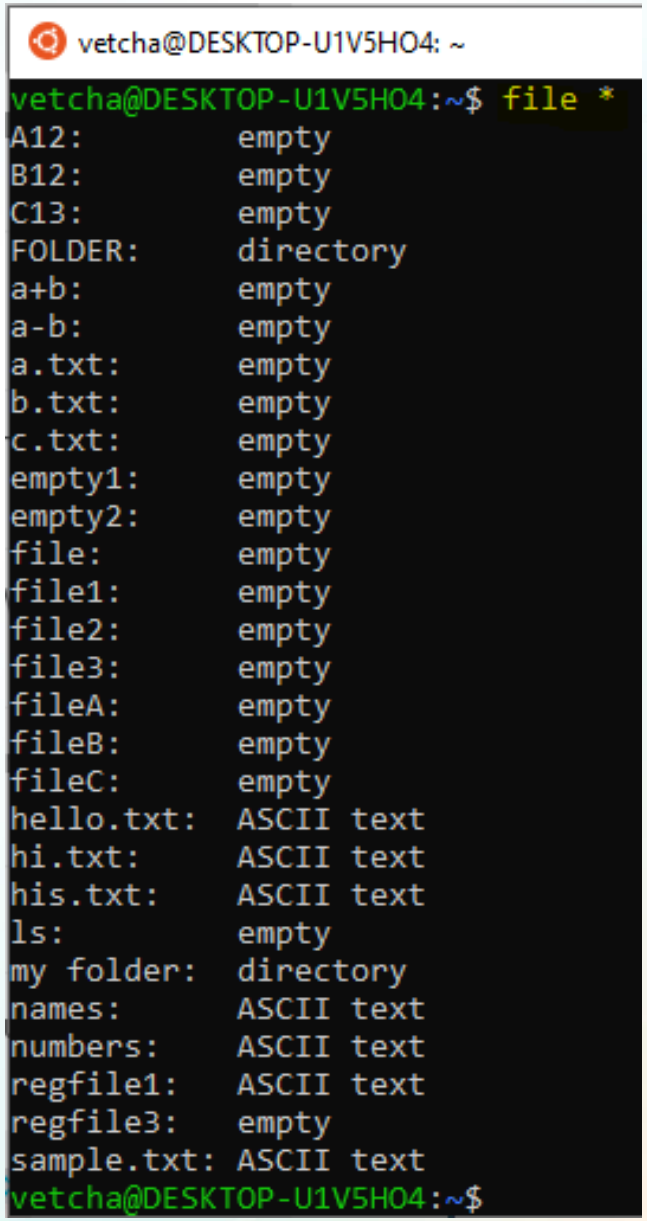
FILE TYPE

file command

- It is used to determine the type of a file.

Syntax: **file** <filename>

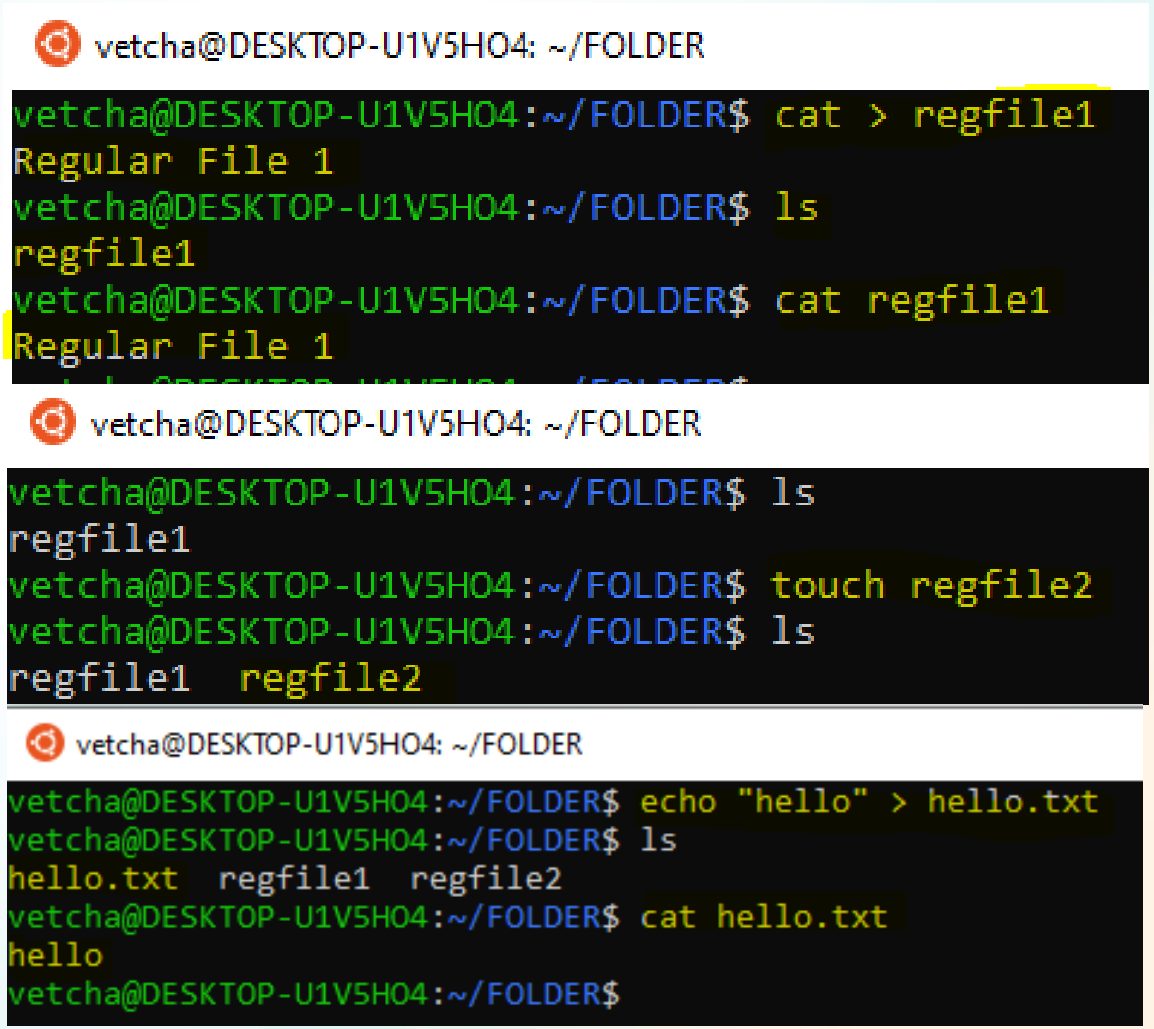




HOW TO CREATE DIFFERENT FILES

Regular file

- **cat, touch,** and **echo** commands can be used to create regular or ordinary files.
- It governs all different files such us text files, images, binary files, shared libraries, etc.



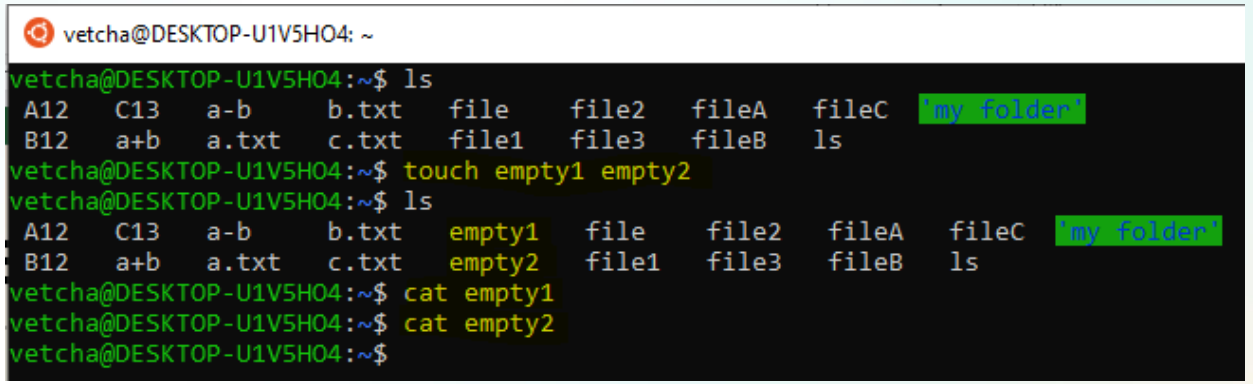
HOW TO CREATE DIFFERENT FILES

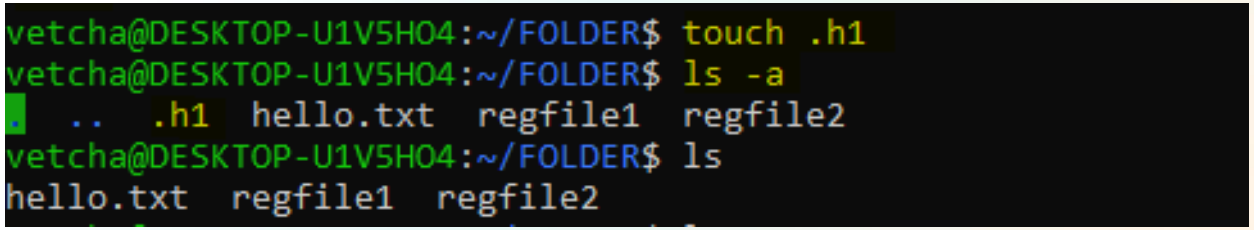
Empty file

- The **touch** command is the easiest way to create new, empty files.

Hidden file

- You can recognize the **hidden** files and folders with the **dot (.)** before their names.
- Use ***ls –a*** to display all the files, including the hidden ones





OTHERS

Autocomplete

Autocomplete works on files and directories. Just press **Tab** key to complete the directory or file that you are looking for.

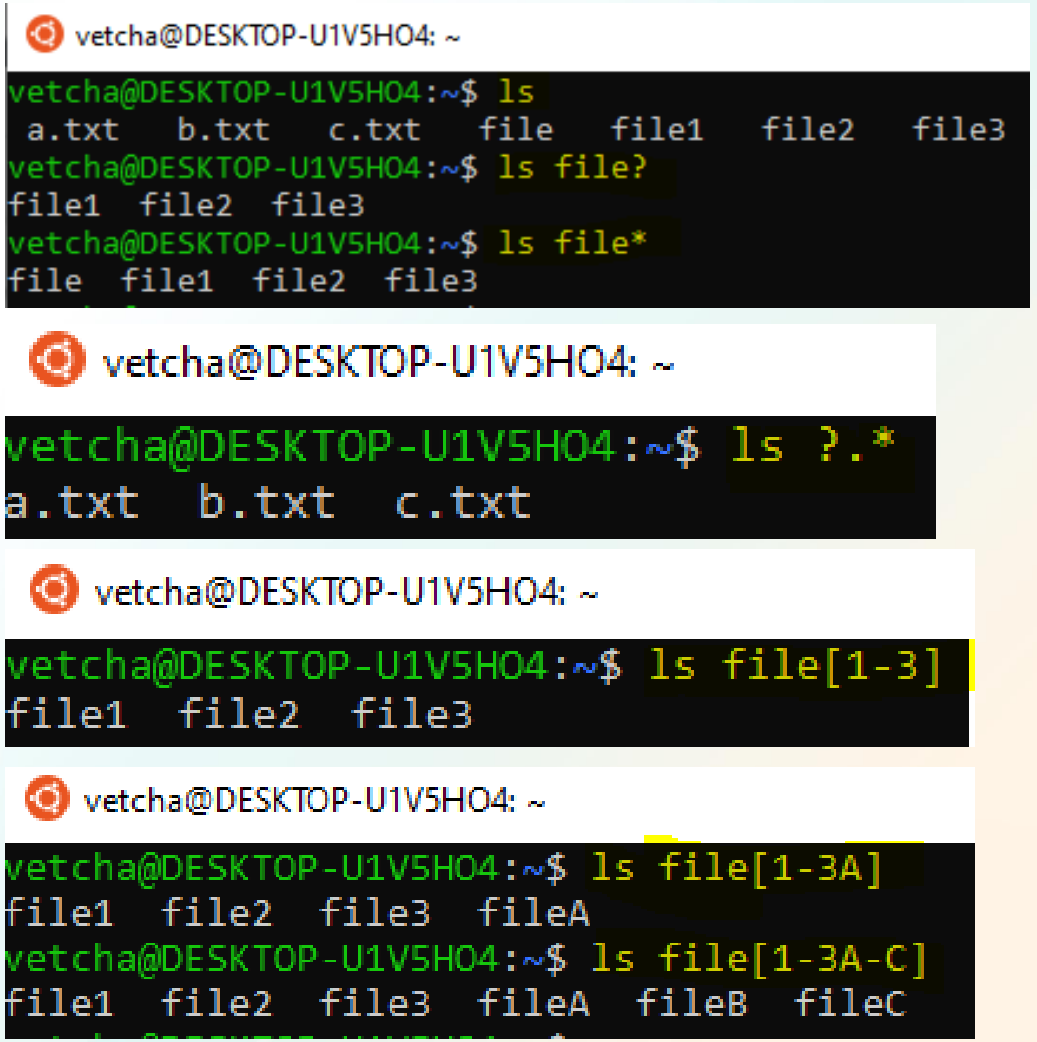
Wildcards

A wildcard is a symbol or a set of symbols that stands in for other characters. It can be used to substitute for any other character or characters in a string.

***** - represents zero or more characters

? - represents a single character

[ ] - represents a range of characters



REFERENCES

- Sobell, M., et al. (2017). A Practical Guide to Linux Commands, Editors, and Shell Programming, 4th Ed. Addison-Wesley Professional
- Cobbaut, P. (2016). Mastering Linux- Networking
- Blum, R., (2015). Linux Command Line and Shell Scripting Bible