|  |  |  |  |
| --- | --- | --- | --- |
|  | **Description** | **Example** | **Try it** |
| [] | A set of characters | "[a-m]" | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_regex_meta1) |
| \ | Signals a special sequence (can also be used to escape special characters) | "\d" | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_regex_meta2) |
| . | Any character (except newline character) | "he..o" | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_regex_meta3) |
| ^ | Starts with | "^hello" | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_regex_meta4) |
| $ | Ends with | "planet$" | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_regex_meta5) |
| \* | Zero or more occurrences | "he.\*o" | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_regex_meta6) |
| + | One or more occurrences | "he.+o" | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_regex_meta7) |
| ? | Zero or one occurrences | "he.?o" | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_regex_meta10) |
| {} | Exactly the specified number of occurrences | "he.{2}o" | [Try it »](https://www.w3schools.com/python/trypython.asp?filename=demo_regex_meta8) |
| | | Either or | "falls|stays" |  |

Task 1:

RegEX Symbols in linux

| **Pattern** | **Meaning** |
| --- | --- |
| \\d | A digit (0–9) |
| \\D | A non-digit |
| \\w | A word character (letter, digit, \_) |
| \\W | A non-word character |
| \\s | A whitespace |
| ^ | Start of line |
| $ | End of line |
| . | Any character |
| \* | 0 or more times |
| + | 1 or more times |
| ? | 0 or 1 time |
| {n} | Exactly n times |

In Linux, **regular expressions (regex)** are powerful patterns used to search, match, or manipulate text. They are commonly used with tools like grep, sed, awk, and find. Here's a summary of **regular expression symbols and usage in Linux**

|  |  |  |
| --- | --- | --- |
| **Symbol** | **Description** | **Example** |
| **.** | **Any single character (except newline)** | **gr.y matches gray, grey** |
| **^** | **Start of line** | **^Hello matches lines starting with Hello** |
| **$** | **End of line** | **end$ matches lines ending with end** |
| **\*** | **Zero or more of the previous character** | **lo\* matches l, lo, loo** |
| **[]** | **Any one character in the set** | **[aeiou] matches any vowel** |
| **[^]** | **Any one character not in the set** | **[^0-9] matches non-digits** |
| **\** | **Escape special characters** | **\. matches literal dot .** |
| **\{n\}** | **Exactly n repetitions** | **a\{3\} matches aaa** |
| **\{n,\}** | **n or more repetitions** | **a\{2,\} matches aa, aaa,...** |
| **\{n,m\}** | **Between n and m repetitions** | **a\{1,3\} matches a, aa, aaa** |
| **\(...\)** | **Grouping (for sed, not grep)** | **\(ab\)\* matches repeated ab** |
| **+** | **One or more of the previous character** | **lo+ matches lo, loo, etc.** |
| **?** | **Zero or one of the previous character** | **colou?r matches color or colour** |
| **`** | **`** | **Logical OR** |
| **()** | **Group expressions** | **(ab)+ matches ab, abab** |
| **{n}** | **Exactly n repetitions** | **a{3} matches aaa** |
| **{n,}** | **n or more repetitions** | **a{2,} matches aa, aaa, ...** |
| **{n,m}** | **Between n and m repetitions** | **a{1,3} matches a, aa, aaa** |

|  |  |
| --- | --- |
| **Command** | **Description** |
| **grep "^A" file.txt** | **Lines starting with A** |
| **grep "ed$" file.txt** | **Lines ending in ed** |
| **grep "[0-9]" file.txt** | **Lines with any digit** |
| **`grep -E "foo** | **bar"` file.txt** |
| **grep -E "a{2,4}" file.txt** | **Matches aa, aaa, or aaaa** |

Task 2:

If you are aware of Linux OS.. can you tell me the feature of Linux.

Core Features of Linux:

* **Open Source:**

Linux is open-source, meaning its source code is freely available and can be modified, distributed, and contributed to by anyone.

* **Free and Open Source:**

Linux is available for free and open source, allowing users to modify, distribute, and contribute to its development.

* **Multiuser and Multitasking:**

Linux supports multiple users concurrently, each with their own accounts and access levels. It also excels at multitasking, allowing users to run multiple programs simultaneously without interfering with each other.

* **Strong Security:**

Linux is known for its strong security features, including user authentication, authorization, and encryption, making it a secure choice for various applications.

* **Portability and Hardware Support:**

Linux can run on a wide range of hardware platforms, from embedded devices to supercomputers, making it highly versatile.

* **Hierarchical File System:**

Linux uses a hierarchical file system structure, which provides a logical and organized way to store and manage files.

* **Command-Line Interface (Shell):**

Linux relies heavily on a command-line interface, allowing users to interact with the system through text-based commands.

* **Graphical User Interface (GUI):**

While Linux primarily uses a command-line interface, it also supports graphical user interfaces (GUIs) like X Window System and Wayland, providing a more user-friendly experience.

* **Customization:**

Linux offers a high degree of customization, allowing users to tailor their operating system to their specific needs and preferences.

* **Stability and Reliability:**

Linux is known for its stability and reliability, often operating for extended periods without requiring reboots.

* **Community Support:**

Linux benefits from a large and active community that provides support and contributions, ensuring continuous development and improvement.

Task 3:

What is Kernal ? can you explain about it in your words..

Kernel in Linux

The Linux kernel is the core of the Linux operating system, acting as a critical interface between hardware and software. It's a free, open-source, Unix-like kernel used in many computer systems worldwide. The kernel manages hardware resources, processes, and system calls, effectively serving as the link between the physical machine and the software running on it.

Core Function:

The kernel is the fundamental part of the Linux OS, responsible for managing all the hardware and software components.

Interface:

It acts as a bridge between the hardware and the software, allowing the OS to interact with the machine.

**Linux** is an **open-source**, **Unix-like operating system** based on the Linux kernel. It powers everything from smartphones and laptops to servers, supercomputers, and even TVs and routers.

Key components are Kernel, shell, File system,Utilities(tools1s,grep,cp in terminal)

Characteristics of Linux – open source,secure,stable,customizable,multiuser.

Linux uses- Webservers,cloudsystems,smartphones,embeddeddevices,desktops,cybersecurity.

Popular linux distributions are, Ubuntu,Debian,fedora,Arch,Redhat

Task 4:

BASH in Linux full form and Explanation

Bash in Linux Bourne Again Shell

It’s a command line interpreter /shell used by most Linux

BASH allows users to Run commands,write scripts,Navigate file systems,Manage process,control flow using loops

BASH is a fundamental tool for Linux system administration, scripting, DevOps, and more

Task 5:

Now that you know Linux is also an Operating System like Windows..

Linux is ideal for developers, power users, servers, and security-focused environments.

Windows is great for casual users, gaming, and business applications that depend on Microsoft tools.

The primary difference between Linux and Windows is that Linux is an open-source operating system, meaning its source code is available for public access and modification. Windows, on the other hand, is a commercial operating system developed by Microsoft, and its source code is not accessible to the public. This open-source nature of Linux allows for greater flexibility, customization, and community-driven development compared to the closed-source nature of Windows.

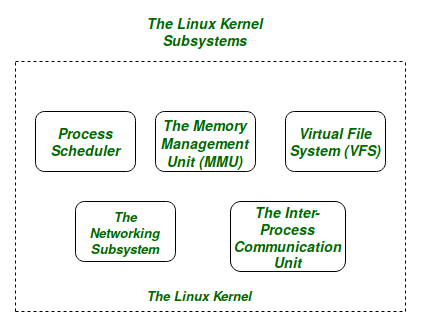
Task 6:

What are the basic components of Linux? Describe each in detail with diagrams

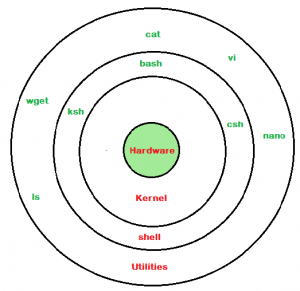
Components of Linux

Kernel,shell,SystemLibraries,System utilities

Kernel is the main core component of Linux, it controls the activity of other hardware components. It visualizes the common hardware resources and provide each process with necessary virtual resources. It makes the process to wait in the ready queue and execute in consequently to avoid any kind of conflict.

Kernel sub sytems

Shell can be determined as the interface to the kernel, which hides the internal execution of functions of kernel from the user. Users can just enter the commend and using the kernel's function that specific task is performed accordingly.



SystemLibrary**:**

System libraries are some predefined functions by using which any application programs or system utilities can access kernel's features. These libraries are the foundation upon which any software can be built.

Hardware Layer:

Hardware layer of Linux is the lowest level of operating system track. It plays a vital role in managing all the hardware components. It includes device drivers, kernel functions, memory management, CPU control, and I/O operations. This layer generalizes hard complexity, by providing an interface for software by assuring proper functionality of all the components.

System utility:

System utilities are the commend line tools that preforms various tasks provided by user to make system management and administration better. These utilities enables user to perform different tasks, such as file management, system monitoring, network configuration, user management etc.

Task 7:

Is it legal to edit Kernal ? when do you think we have to in case?

Open Source Kernels

Yes, it is legal to edit a kernel, specifically the Linux kernel, as it is open-source software released under the [General Public License (GPL)](https://www.google.com/search?client=firefox-b-e&channel=entpr&cs=0&sca_esv=b445a9df1f419e25&q=General+Public+License+%28GPL%29&sa=X&ved=2ahUKEwiUqNzs-cyNAxXYyjgGHTENOGoQxccNegQIAhAB&mstk=AUtExfDiobYjpVWqFegw2rCS7S_1BuTrQPzTF1eAN5NnoaYyQBZPFllj63W6ZDVhnLjTbEfynjI4b9UPBZkZs-zHYtQkf2pmDF52HBXGMKMLEZuDL2fTuFXfRcFdB7dNyYFpSW8&csui=3). This means anyone can modify, distribute, and redistribute the kernel under the terms of the GPL

Legally allowed to

Modify source code

Compile own session

Use it for personal purpose

Illegal when

Could breach copyright laws.

Task  8:

What is LILO? Explain

LILO

|  |
| --- |
| **LILO** |
| What is LILO?  Explain ? LILO, short for Linux Loader, is an older bootloader program used in Linux systems What does LILO do? • It helps your computer choose and start the operating system. • You can set it to: • Boot Linux • Boot Windows (if you have both installed) • Boot other OSes Features of LILO: • Can boot multiple operating systems • Simple and fast • Doesn’t depend on file systems (works directly with disk) Note: • LILO is old and not used much today. • New systems usually use GRUB (a more powerful boot loader). |

|  |
| --- |
| Linux Loader – an older boot loader for Linux |

|  |
| --- |
| **Role** |

|  |  |
| --- | --- |
| Loads the Linux kernel and starts the OS | |
| **Status** |

|  |
| --- |
| Largely replaced by GRUB |
| LILO, short for Linux Loader, is a boot loader program used in Linux operating systems. It's responsible for loading the kernel and other necessary files into memory when a computer starts or restarts, allowing the Linux operating system to begin execution. While LILO was once the dominant boot loader for Linux, it's now less common, with grub and other modern alternatives being more widely used  Task 9:  What is shell? How many shells are there and what are they ? can you explain.  Shell |

A **shell** is a **command-line interface (CLI)** that allows users to interact with the operating system. It takes your input (commands), passes it to the system to execute, and returns the output.

**Most Commonly Used Shells**

1. **Bash** – Default on many Linux distros.
2. **Zsh** – Popular for power users (used with Oh My Zsh).
3. **Fish** – Gaining popularity for its ease of use.

Command line shell

These shells provide a text-based interface where users type commands to interact with the operating system , Ex: Bourne, C Shell

Graphical shell

These shells provide a graphical user interface, allowing users to interact with the operating system through a window-based system with icons, menus, and other visual elements, KDE,GNOME

Task 10:

What is swap space?

**Swap space** is a portion of a storage device (usually a disk or SSD) that is used as **virtual memory** when your system runs out of physical **RAM (Random Access Memory)**.

When the system’s RAM is full, the kernel **moves inactive pages of memory to the swap space** to free up RAM for active processes. This prevents the system from crashing due to memory exhaustion.

Swap partition,swap file

Task 11

Mount

In Linux, **mounting** means making a filesystem **accessible** to the system by **attaching it to a directory** (called a **mount point**) in the main file hierarchy.

You can mount:

* Internal partitions (e.g., /dev/sda1)
* USB drives (e.g., /dev/sdb1)
* ISO images
* Network filesystems (e.g., NFS, SMB)
* Virtual devices

What is Mount ? how do you mount and unmount file system in Linux?

What is Mount ? how do you mount and unmount file system in Linux? Mount means connecting a storage device (like a pen drive, hard disk, or partition) to your Linux system so you can use it. Mount = Make a drive usable Unmount = Remove it safely

Task 12

What is chmod command ? how to use it?

Chmod is change mode

It is used to **change file or directory permissions** (who can read, write, or execute a file).

Owner u – user who owns

Group g – in file group

Others o – everyonelese

Read,write ,execute

The "chmod" command modifies the read, write, and execute permissions of specified files and the search permissions of specified directories. [who] refers to who you are giving permissions to. Specified in the order: user, group, others. The permissions that can be given are : read, write or execute.

|  |  |
| --- | --- |
| **Symbol** | **Meaning** |
| r | Read |
| w | Write |
| x | Execute (or open for folders) |
| **Category** | **Who it is** |
| u | User (owner) |
| g | Group |
| o | Others |
| a | All (user + group + others) |

Task 13

Can you add a new user account? Crate a new user in different ways and paste ss

**Can you add a new user account? Crate a new user in different ways and paste ss**

**Method 1: Using adduser Command (Simpler & Interactive)**

sudo adduser newusername

**Method 2: Using useradd Command (More Manual)**

sudo useradd -m -s /bin/bash newusername

sudo passwd newusername

**Explanation:**

* -m creates the home directory.
* -s sets the default shell.
* passwd sets the password for the new user.

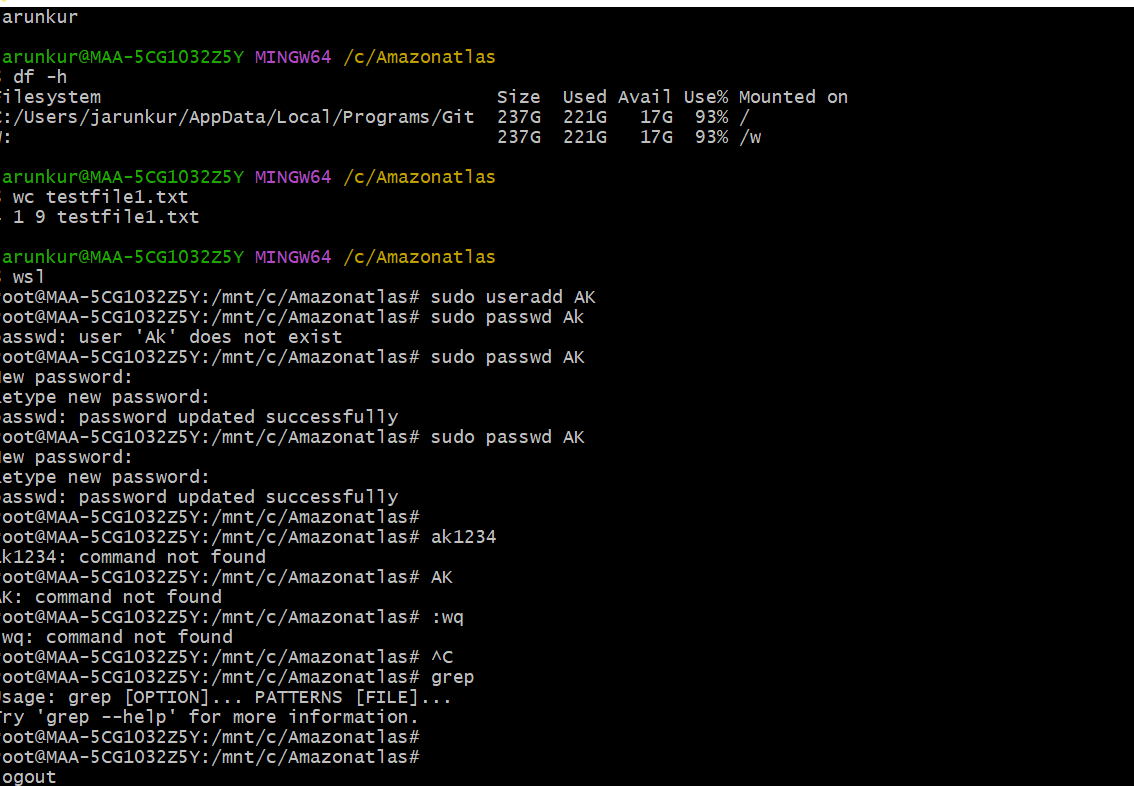
**Method 3: Creating User with Custom Home Directory and Shell**

sudo useradd -m -d /custom/home/path -s /bin/zsh newusername

sudo passwd newusername

**Verify the User**

id newusername



Task 14

Can you change the password of a user?

How do you do that? Plz share ss

**For your own account:**

passwd

You’ll be prompted to:

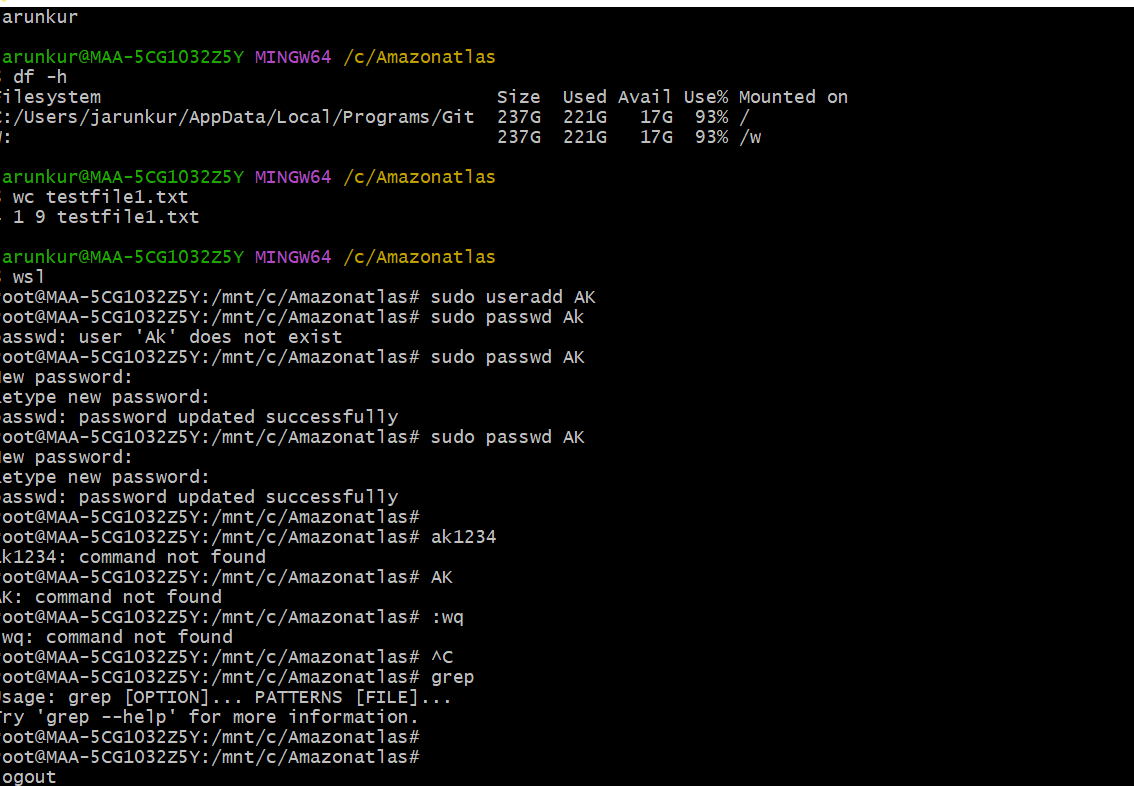
1. Enter your current password.
2. Enter a new password.
3. Confirm the new password.

**To change another user's password (requires sudo):**

sudo passwd username

**Example:**

sudo passwd john



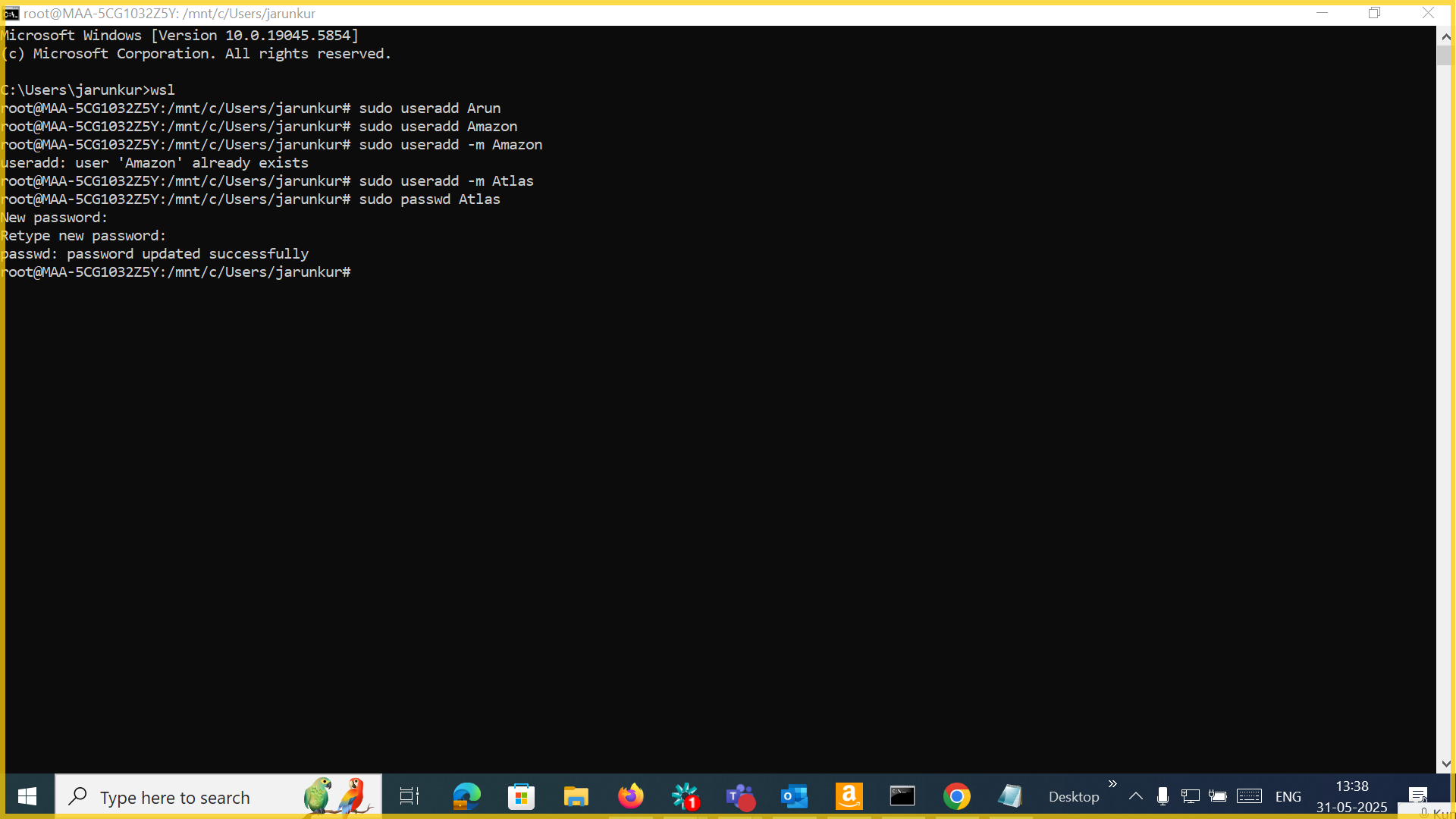
Task 15

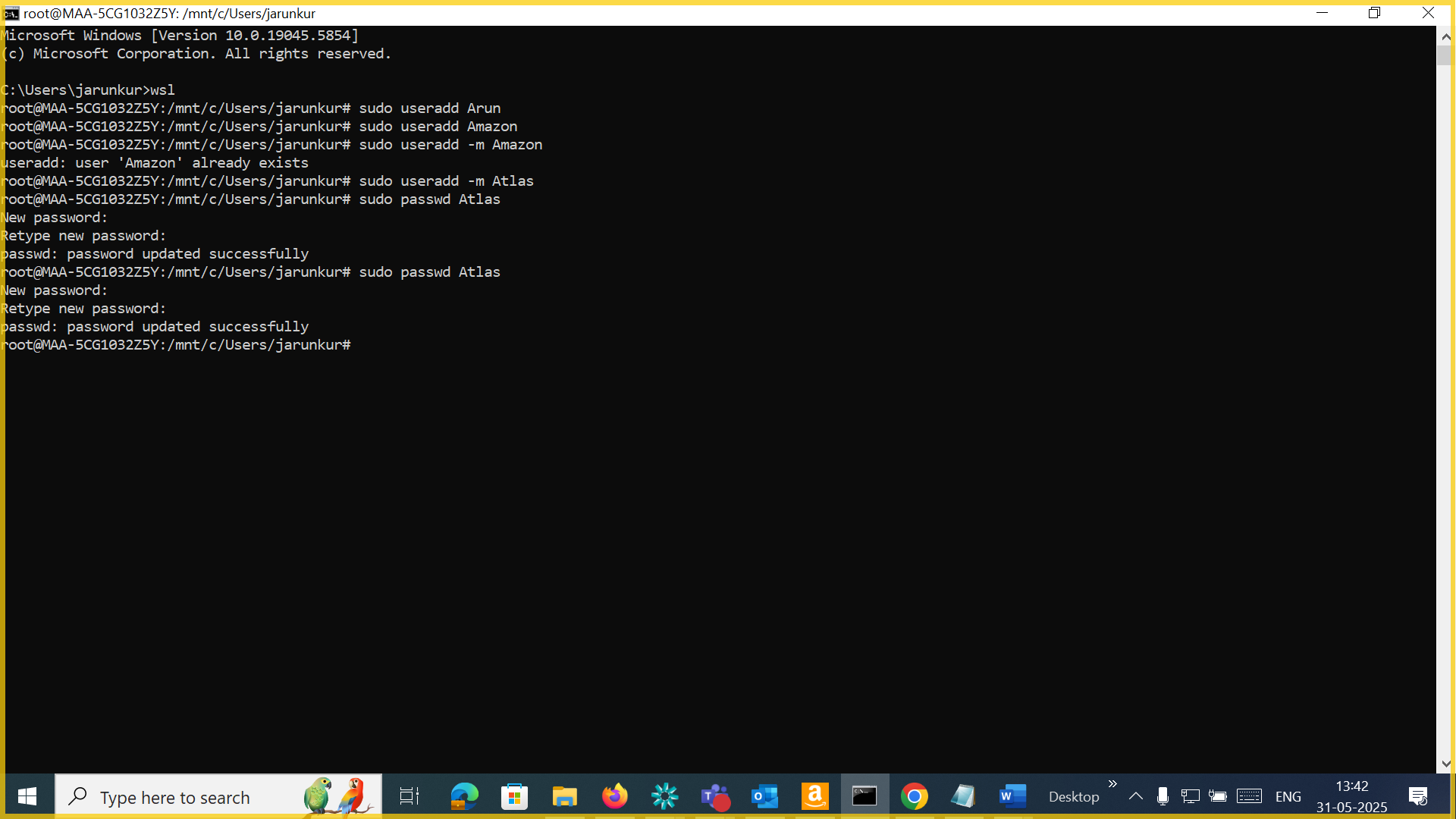
What is diff between Process and Thread?

|  |  |  |
| --- | --- | --- |
| **Feature** | **Process** | **Thread** |
| **Definition** | A program in execution. | A lightweight unit of a process. |
| **Memory** | Each process has its own memory space. | Threads share the same memory space. |
| **Isolation** | Processes are independent; crashing one doesn't affect others. | Threads are interdependent; error in one can affect others. |
| **Communication** | Inter-process communication (IPC) is complex. | Thread communication is easier (shared memory). |
| **Creation Time** | More time and system resources to create. | Faster to create and manage. |
| **Examples** | Opening Chrome and VLC as separate apps. | Opening multiple tabs in Chrome (each tab = thread). |

Process vs Thread (in simple words):

Process: • A program that is running. • Has its own memory and resources. • Example: Opening Chrome is a process. Thread: • A smaller part inside a process. • Shares memory with other threads in the same process. • Example: Chrome loading one tab is a thread, loading another tab is another thread. • Process = Big program • Thread = Small tasks inside the program

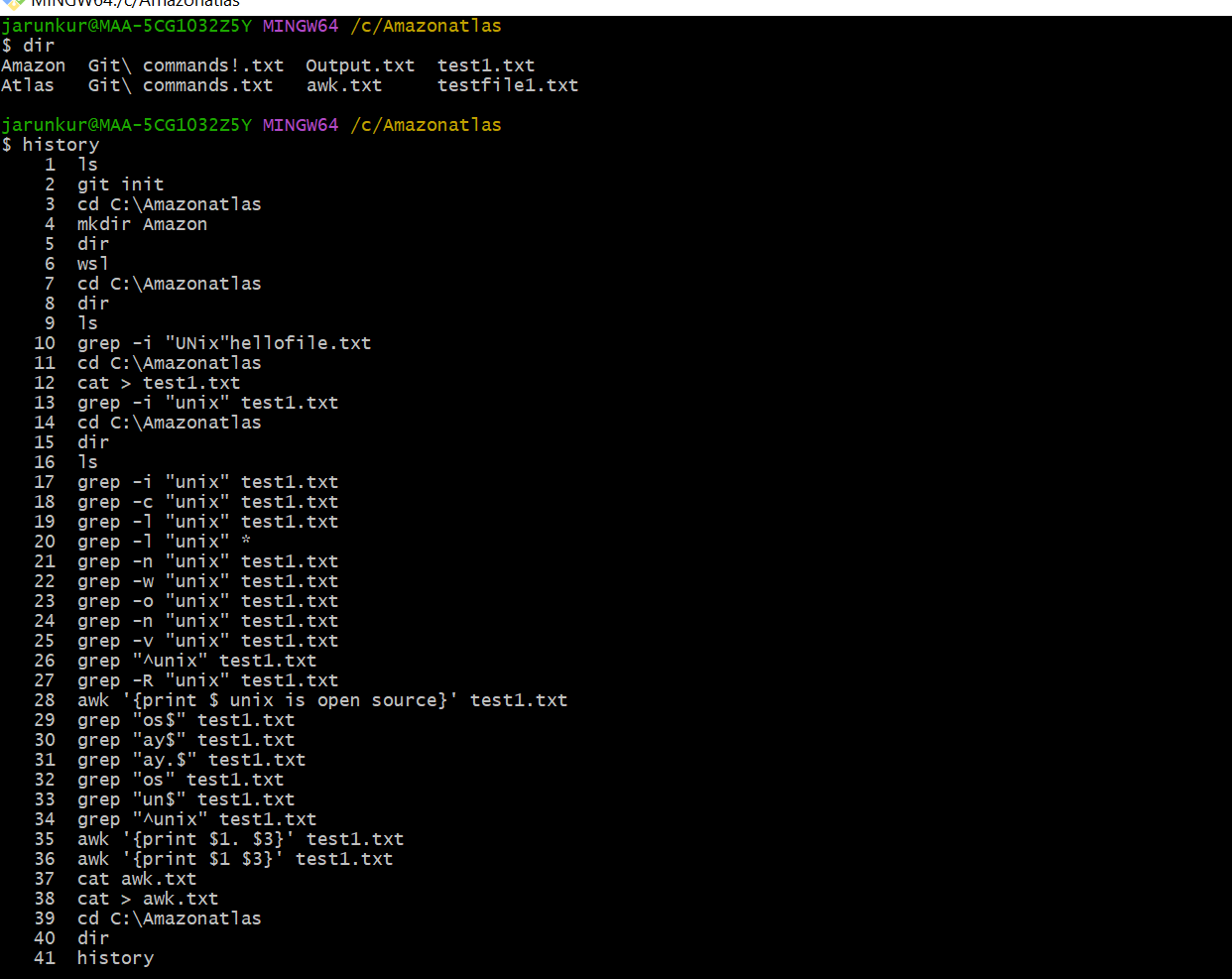


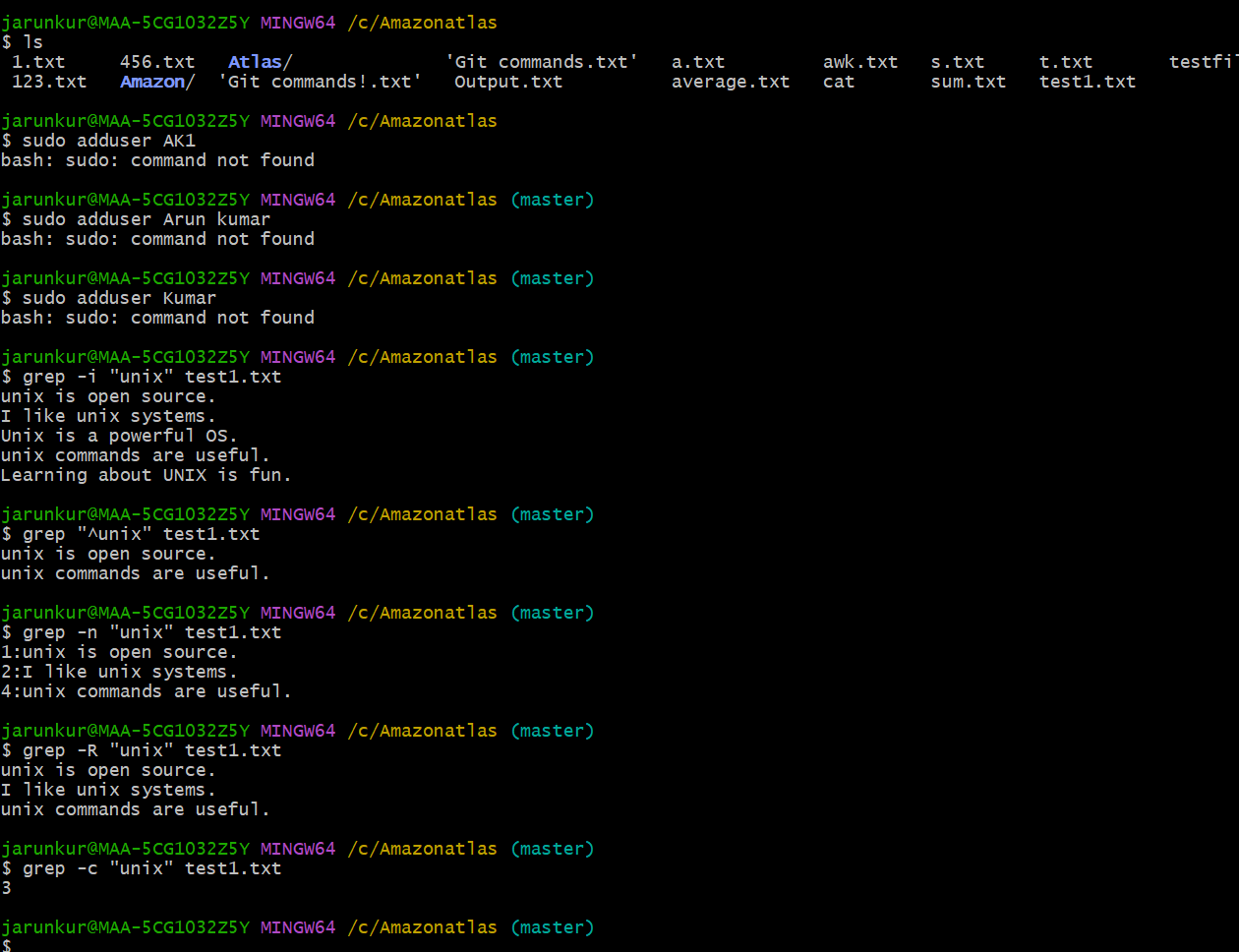


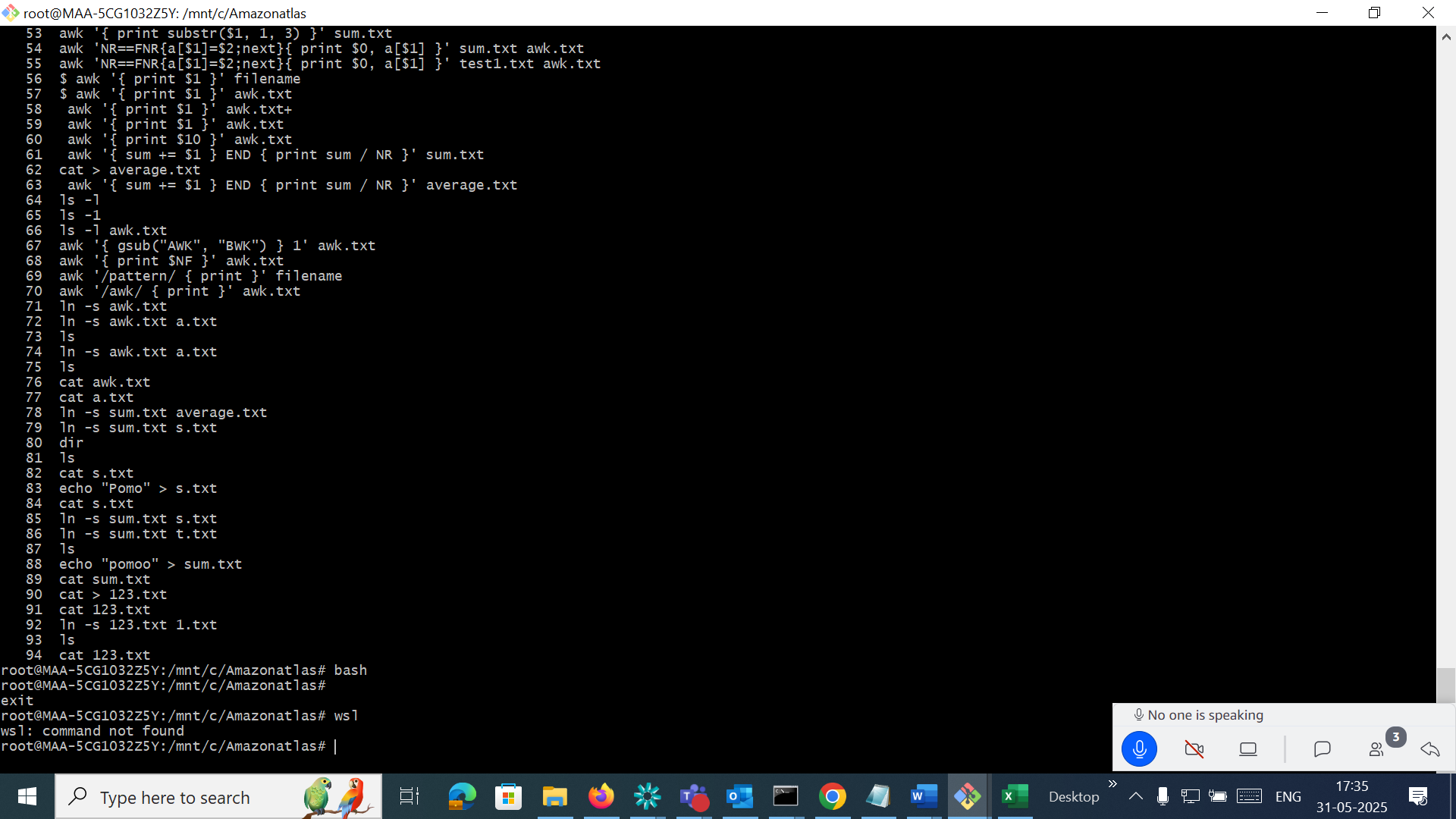
Task 16

Doc 14 Linux Grep commands in docs to study folder .. plz work on it..

<https://docs.google.com/document/d/1ZV5l14lKa6SZc8IJD-HxogqMmTuRzMOV/edit?tab=t.0>

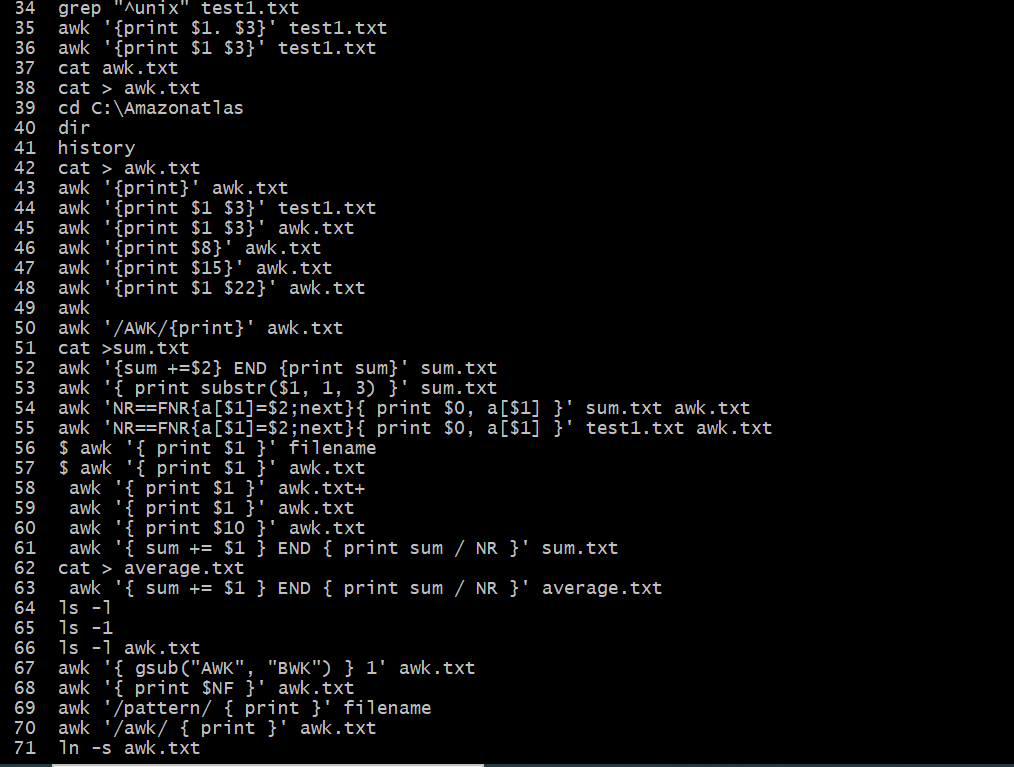


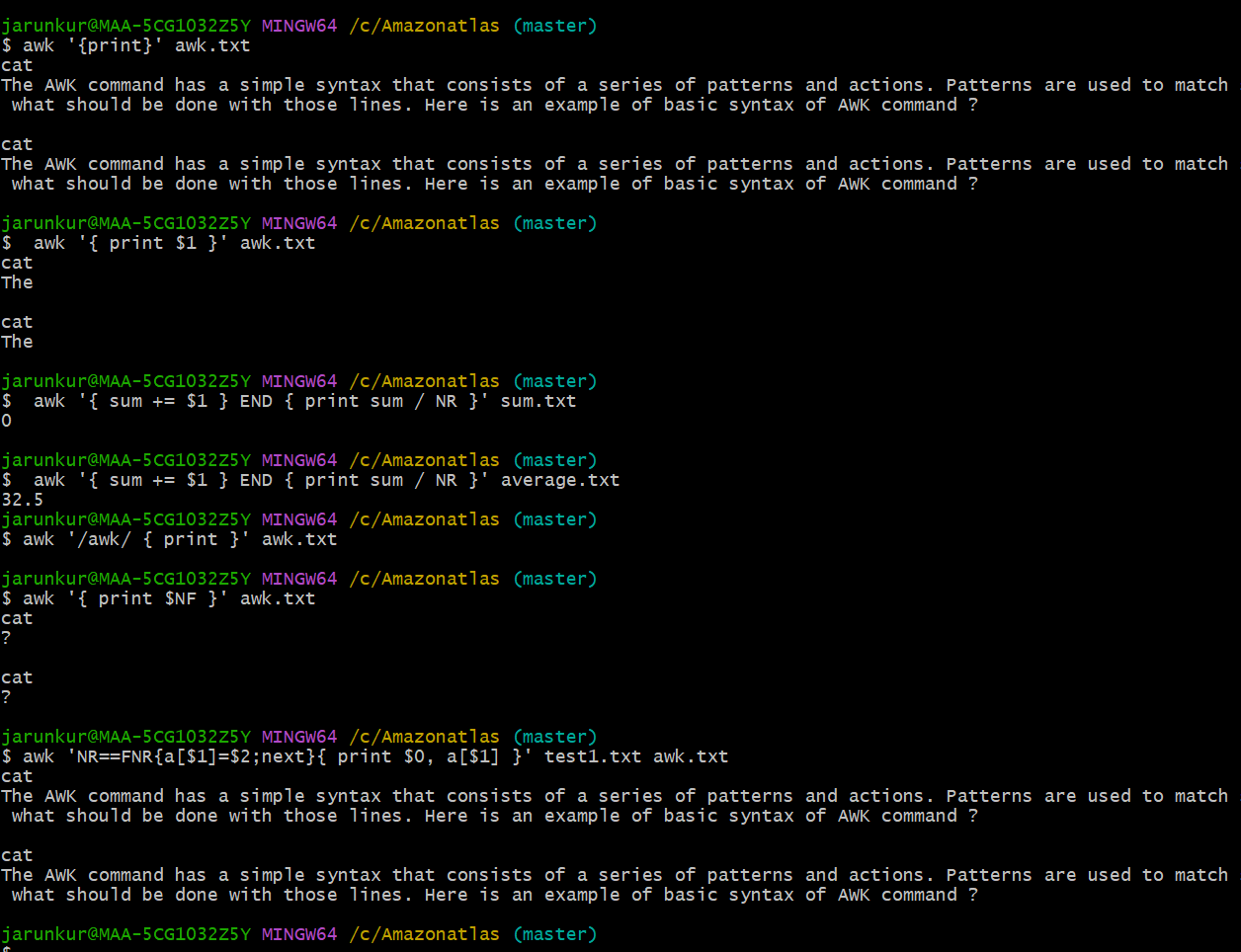




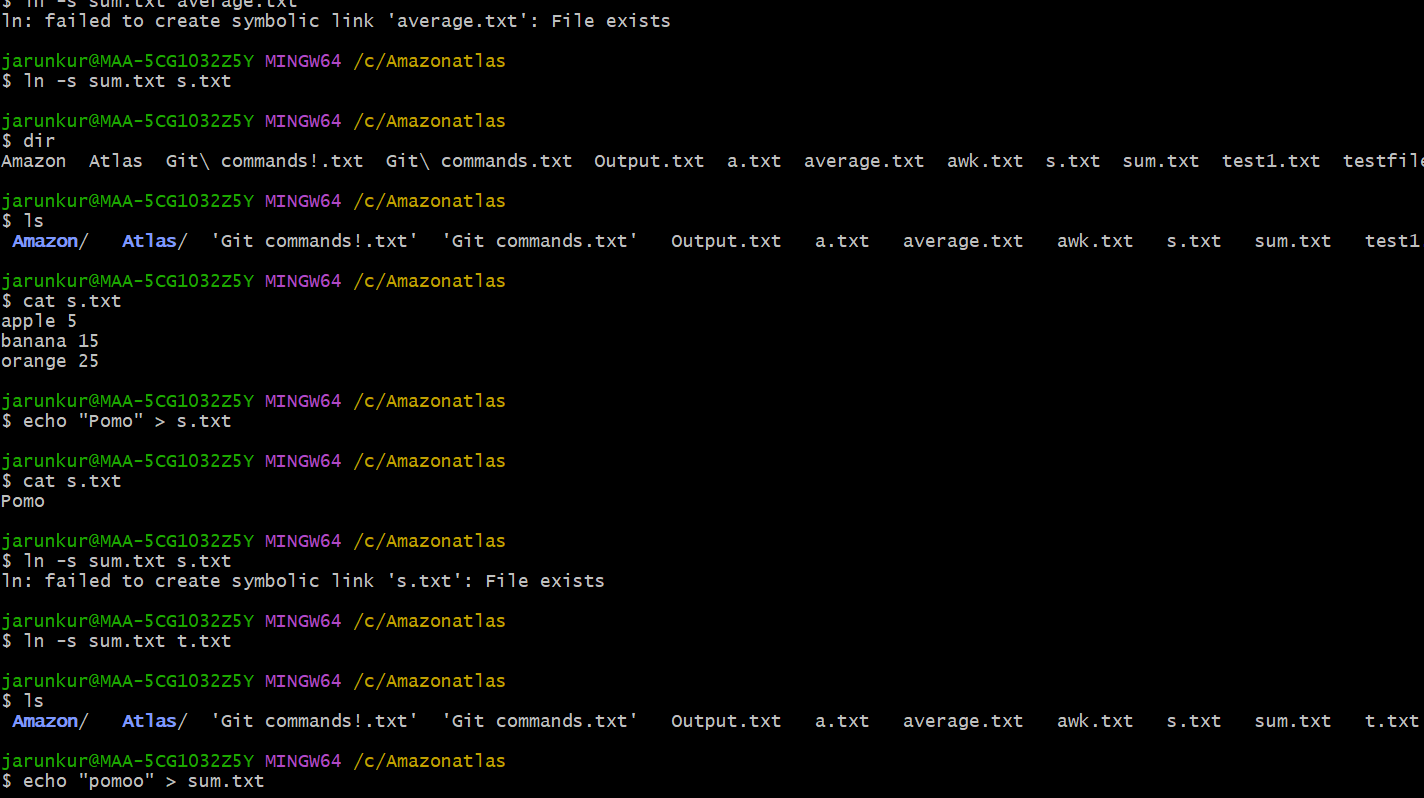
Task 17

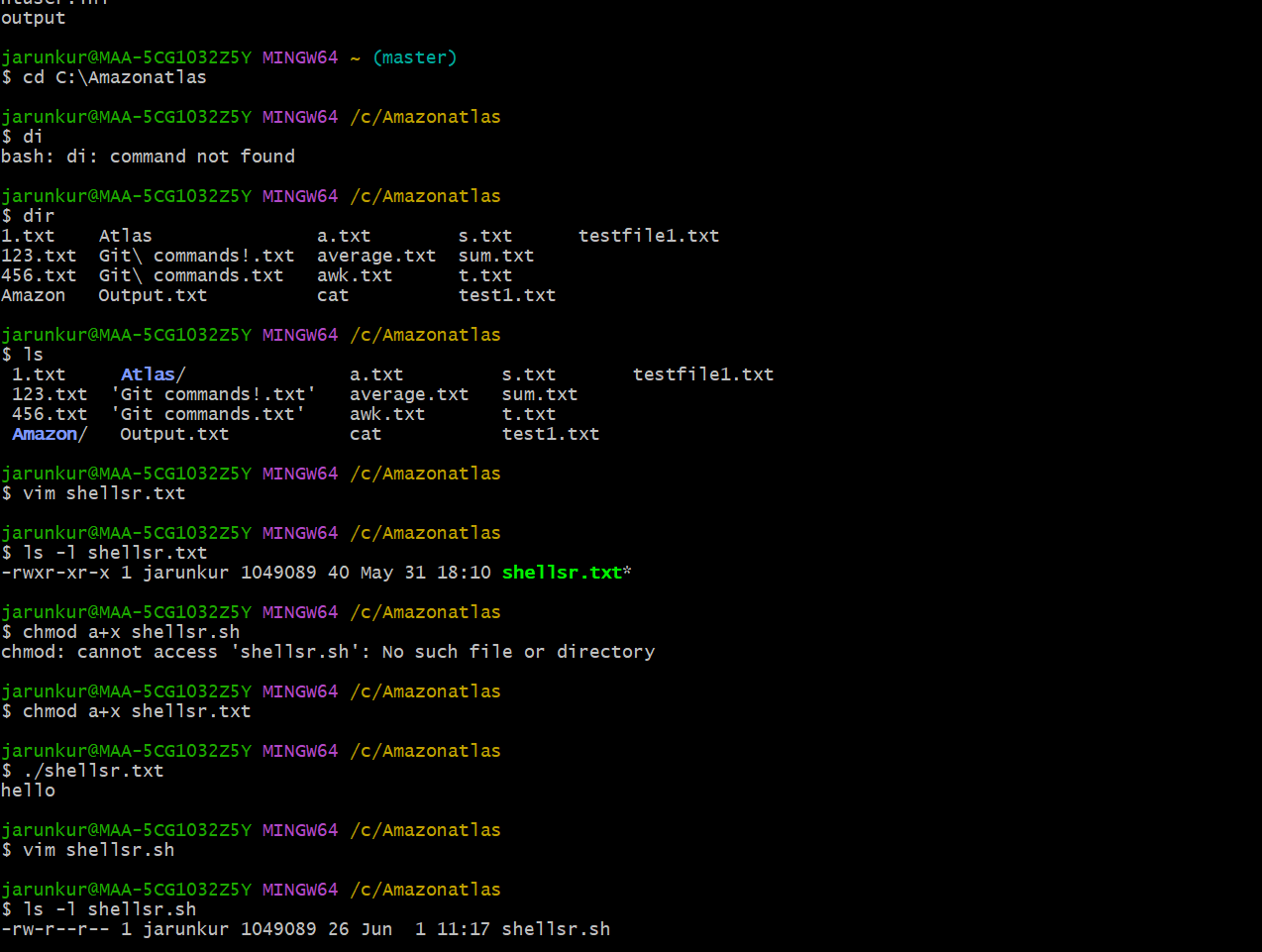
AWK commands in doc 15 Linux AWK commands.



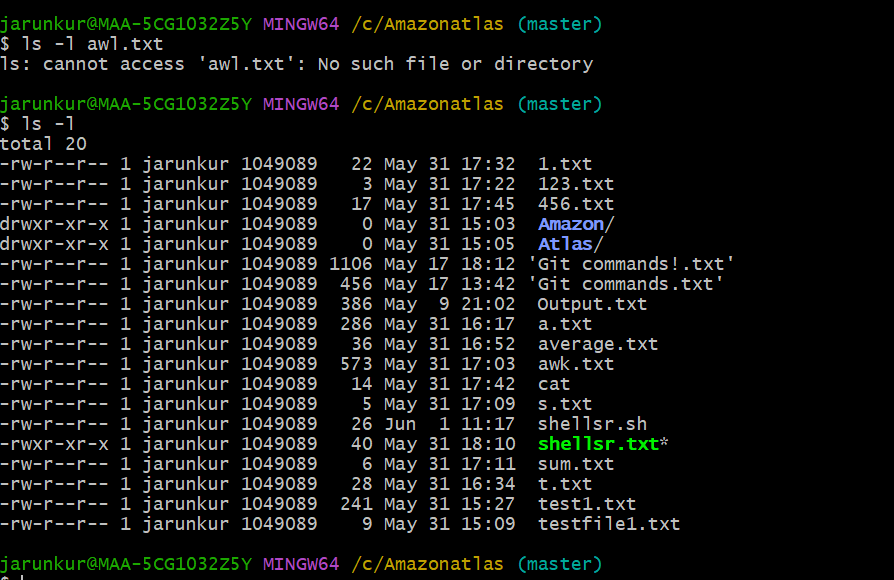








Task 18:



Owner-group-all&others

r- read, w- write, x-execute

Task 19:

What are the default permissions for a new file ?

Plz find out for

Owner   →  rw- → read, write

Group → r-- → read only

All and others →  r-- → read only

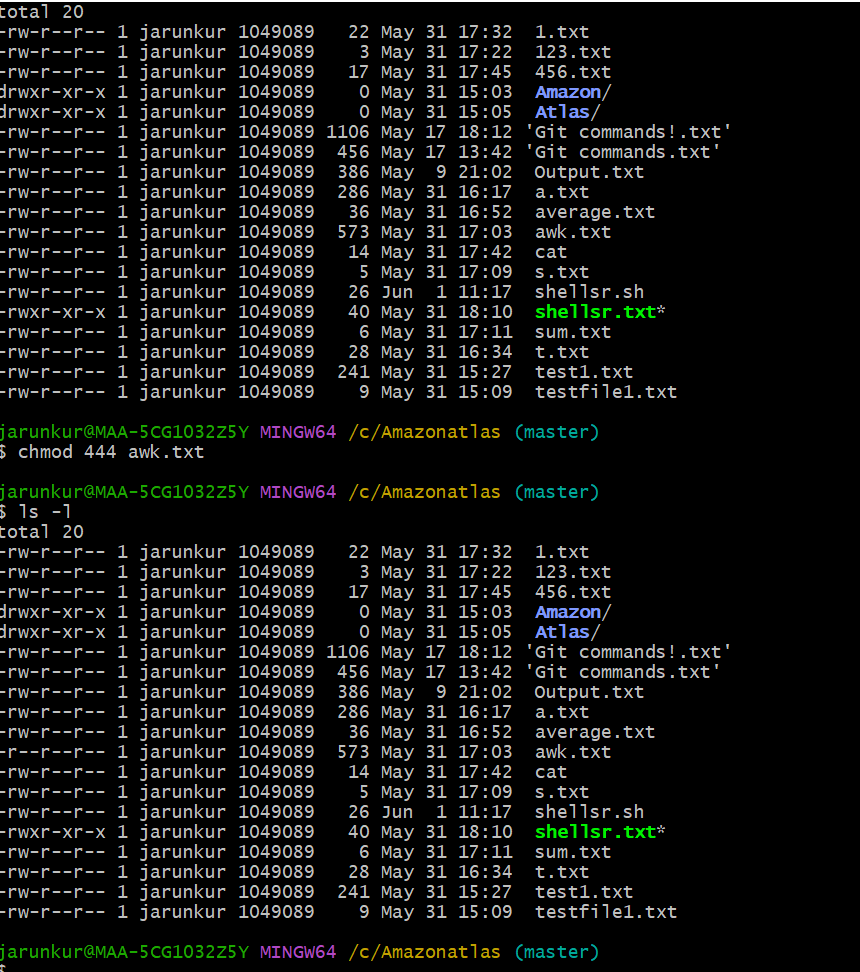
Using ls -l

-rw-r--r-- 1 prathesh pratheesh  30 May  1 12:34 testfile.txt

Task 20:

What is the command to change the permisssion to read only for the owner, group and all other users

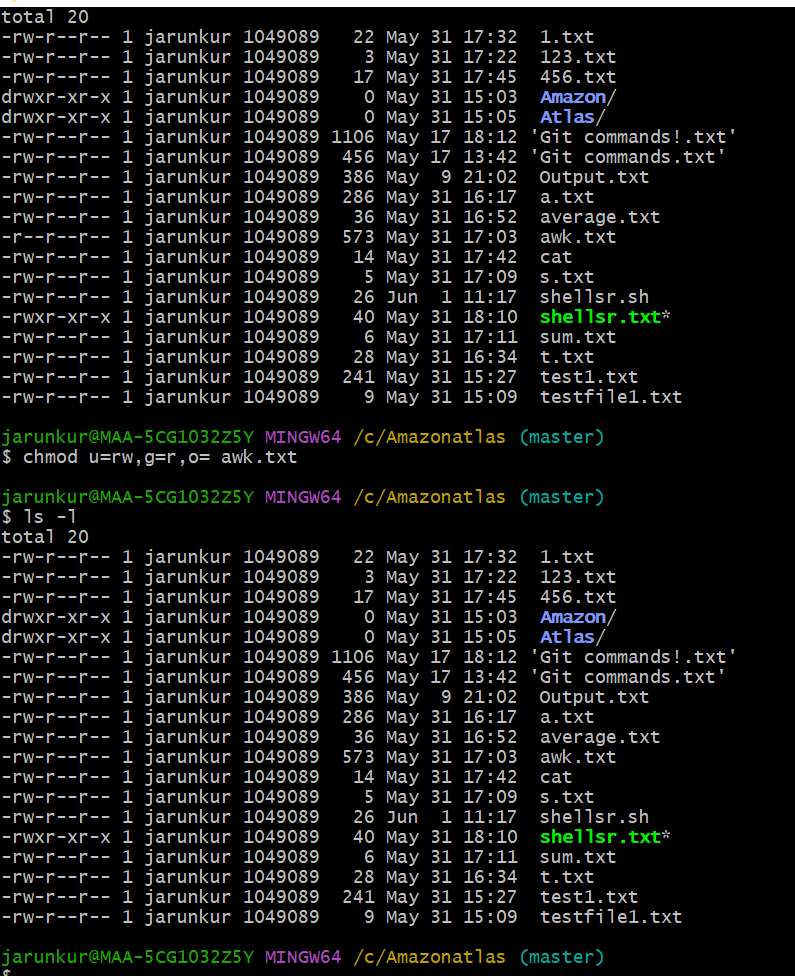
HInt: chmod 444 filename



Task 21:

Can you change the file permissions to match the following:

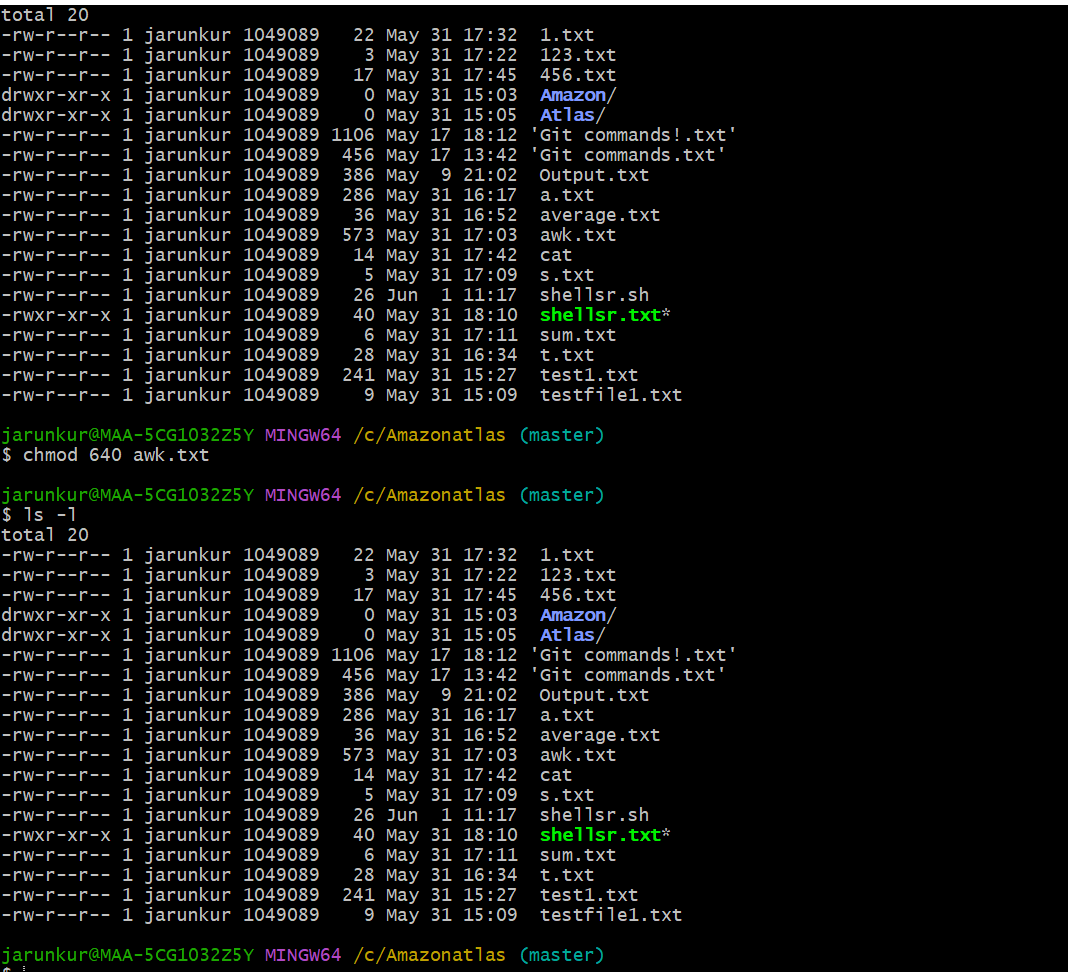
* owner: Read and Write
* group: Read
* other: no permissions (None)



Task 22:

What was the command for changing the file permissions to -rw-r-----?

Hint : use chmod 640 filename



Task 23:

Change chmod.exercises permissions to -rwxr-x--x

Change the file permissions to match the following:

owner: Read, Write and Execute

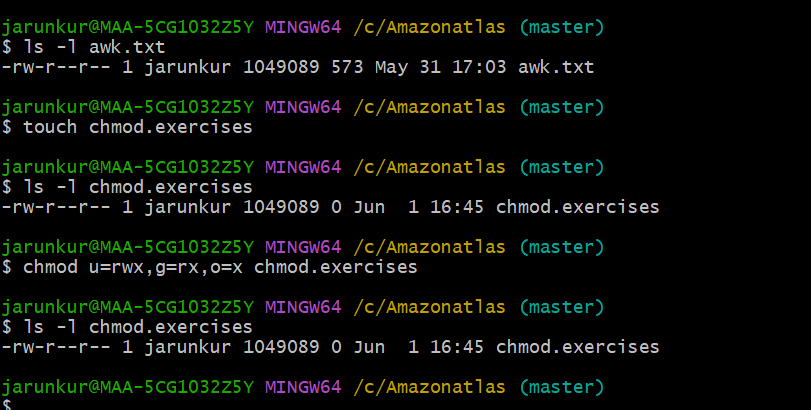
group: Read and Execute

other: Execute

Task 24:

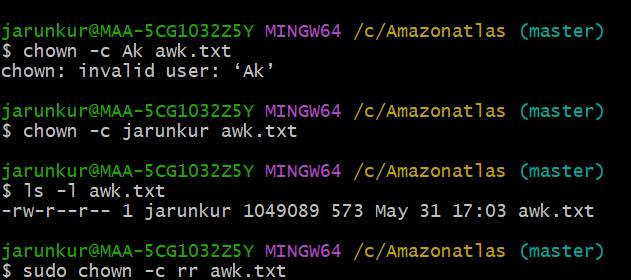
What was the command for changing the file permissions to -rwxr-x--x

Hint : use chmod 751 filename

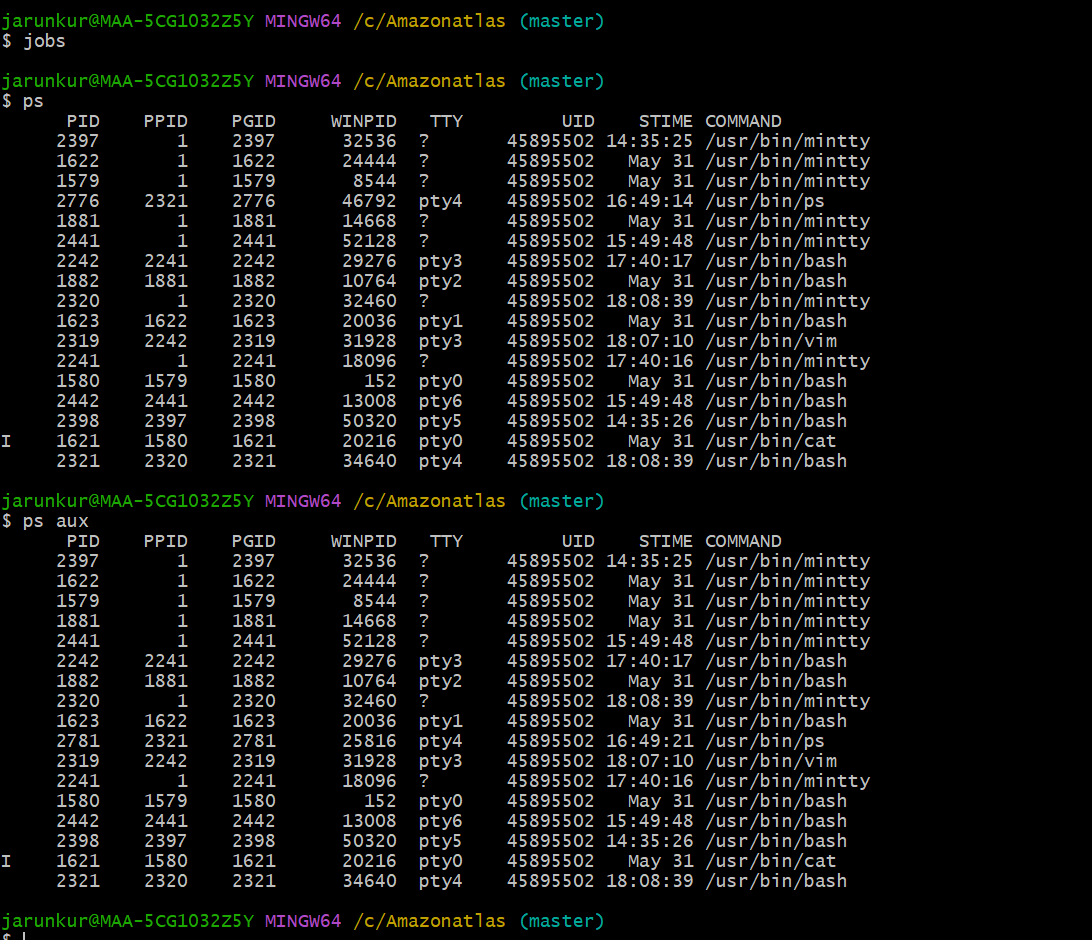


Task 25:

Change ownership of a file.

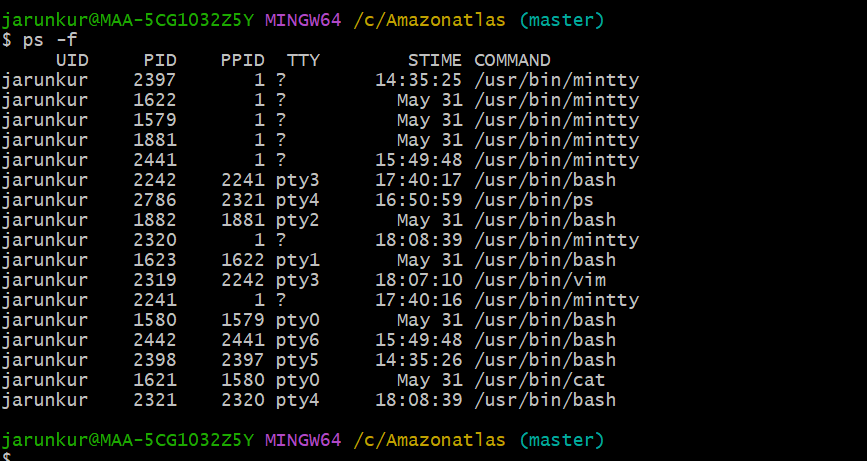


Task 27 & 28: ps and ps aux for foreground and running processes and jobs for background processes

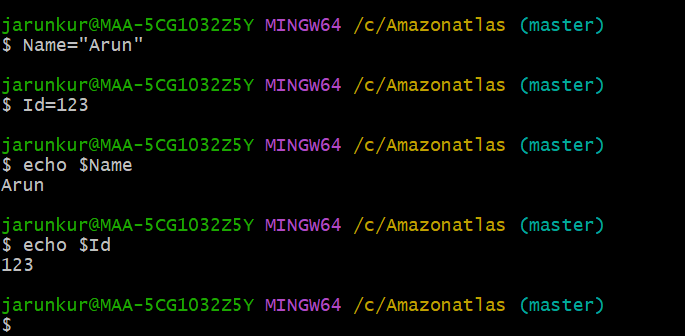


Task 29:

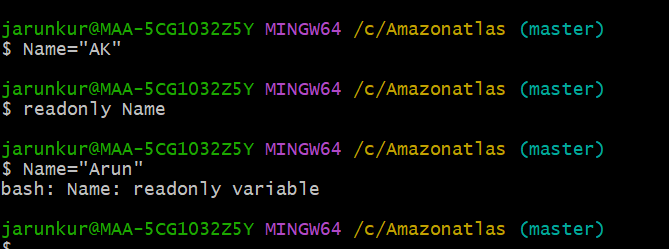
Ps -f : displays information about currently running processes.



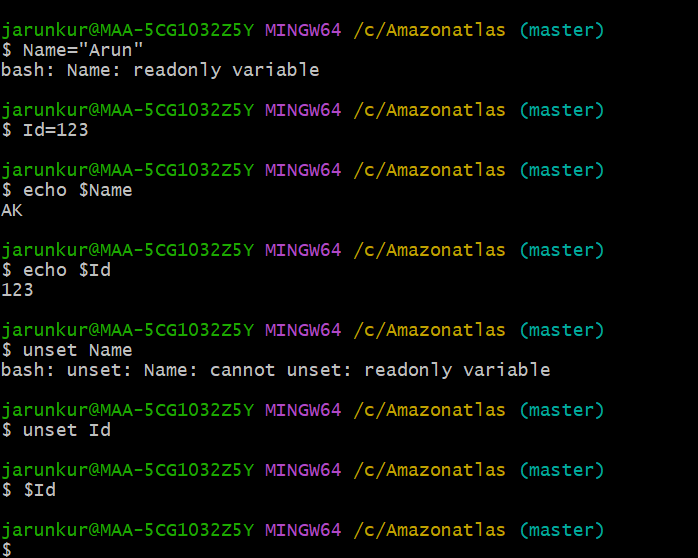
Task 30:



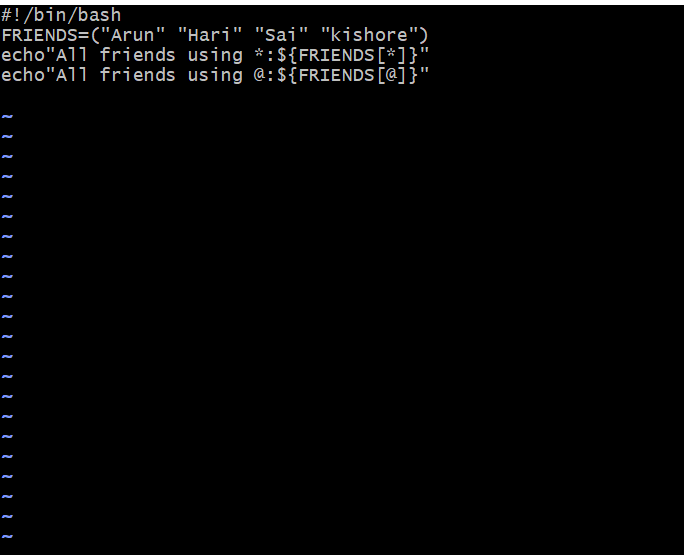
Task 31:



Task 32:



Task 33, 34:



Task 35:

