**Name Shreya**

**Roll no 09**

**USN ENG24CY0051**

**SECTION C**

**Linux Programming: Assignment-2**

**1. What does the command pwd, whoami, and hostname display?**

Print Working Directory (pwd)  
  
shows the current working directory's entire path.  
  
An example of output  
  
/home/user/Documents  
  
  
Use Case: Assists in locating you within the filesystem while using the terminal.  
  
2. Who Am I?  
  
Displays: The username of the active user.  
  
An illustration of output  
  
  
user  
  
  
Use Case: Useful for confirming which user is logged in when logging in with sudo or another user.  
3. Hostname: Host Name  
  
Displays: The network system (computer) name.  
  
An example of output  
  
My computer's name  
  
  
Use Case: Beneficial for machine identification in a networked setting.

**2. Write the command to create a directory named “project” inside the**

**/home/student folder and keep three .txt file into it. Give output snapshot.**

bash

mkdir /home/student/project

touch /home/student/project/file1.txt /home/student/project/file2.txt /home/student/project/file3.txt

shell

$ mkdir /home/student/project

$ touch /home/student/project/file1.txt /home/student/project/file2.txt /home/student/project/file3.txt

$ ls /home/student/project

file1.txt file2.txt file3.txt

**3. Explain the difference between absolute path and relative path with proper examples.**

Difference Between Absolute Path and Relative Path

| Feature | Absolute Path | Relative Path |
| --- | --- | --- |
| Definition | The full path to a file or directory starting from the root /. | The path to a file or directory relative to the current working directory. |
| Starts With | / (root directory) | Folder or file name (no leading /) |
| Depends on Current Location? | No | Yes |
| Always Unique? | Yes | No (depends on context) |

**Absolute Path Example**

/home/student/project/file1.txt

* This path starts from the root /.
* No matter where you are in the system, this path will always point to the same file.

**Relative Path Example**

Assume you are currently inside /home/student.

project/file1.txt

* This means "go to the project folder in the current directory, then open file1.txt."
* The same command won't work if you're not in /home/student.

**4. What command will give you the already executed command traces in the terminal. Give output snapshot**

Command to Show Previously Executed Commands in the Terminal

The command used to view the history of previously executed commands is:

bash

History

**Simulated Output Snapshot:**

$ history

1 pwd

2 whoami

3 hostname

4 mkdir /home/student/project

5 touch /home/student/project/file1.txt file2.txt file3.txt

6 ls /home/student/project

7 cat /etc/hostname

8 history

**5. Compare the working functionality of find and locate command. Which one is**

**faster and why?**

Comparison of find and locate

| Feature | find | locate |
| --- | --- | --- |
| Search Method | Scans the filesystem in real-time | Searches a pre-built database |
| Speed | Slower (because it checks each file live) | Faster (uses a cached index) |
| Updated Information | Always up-to-date | May be outdated if database isn't updated |
| Flexibility | Highly flexible (supports size, date, type, etc.) | Limited (only searches by name) |
| Database Required | No | Yes (updatedb creates it) |
| Typical Use | Complex, filtered searches | Quick filename searches |

**6. Which command is used to modify file permissions in Linux? Give an example.**

**Command to Modify File Permissions in Linux**

The command used to **modify file permissions** is:

chmod

Example:

Suppose you want to give the owner read, write, and execute permissions, the group read and execute, and others only read permission on a file named script.sh.

You can use:

chmod u=rwx,g=rx,o=r script.sh

**Explanation:**

* u=rwx → User: read, write, execute
* g=rx → Group: read, execute
* o=r → Others: read only

**7. A file has permissions -rw -r- -r- -. What does this mean?**

**Breaking it down:**

The permissions string has 10 characters:

1 2 3 4 5 6 7 8 9 10

- r w - r - r - -

* **1st character:**  
  - means it’s a **regular file** (not a directory or special file).
* **Next 3 characters (2-4): User (owner) permissions**  
  rw- means the owner has **read (r)** and **write (w)** permissions, **no execute (-)**.
* **Next 3 characters (5-7): Group permissions**  
  r-- means the group has **read (r)** permission only.
* **Last 3 characters (8-10): Others (world) permissions**  
  r-- means others have **read (r)** permission only.

**Summary:**

| **User (owner)** | **Group** | **Others** |
| --- | --- | --- |
| Read, Write | Read only | Read only |

**What it means practically:**

* The **owner** can read and modify the file.
* Members of the **group** can only read the file.
* **Everyone else** can only read the file.
* **No one** can execute the file.

**8. Explain the difference between chown and chgrp with an example.**

Difference Between chown and chgrp

Both commands are used to change ownership of files or directories in Linux, but they modify different parts of ownership:

| Command | Purpose | What It Changes |
| --- | --- | --- |
| chown | Change user (owner) of a file or folder | Changes the owner user |
| chgrp | Change group ownership of a file or folder | Changes the group ownershi |

**9. A file needs to be accessible by multiple users but only writable by the owner. How will you set permissions?**

**How to Set File Permissions So That:**

* **Multiple users can read/access the file**
* **Only the owner can write to it**

**Solution:**

Use the **chmod** command to set permissions as:

chmod 644 filename

**Explanation of chmod 644:**

**Permissions Breakdown:**

| **Who** | **Permissions** | **Meaning** |
| --- | --- | --- |
| Owner | rw- | Can **read and write** |
| Group | r-- | Can **only read** |
| Others | r-- | Can **only read** |

So, **everyone can access the file (read it), but only the owner can write to it.**

**10. How do you check the manual page for any Linux commands?**

How to Check the Manual Page for Linux Commands

In Linux, you can view the manual (man) page for any command using the:

bash

man <command>

**Example:**

**bash**

man ls

* This displays the manual (help) page for the ls command.
* It shows **usage, options, description**, and **examples**.