**🧠 What is vim? (Visual editor – Vi IMproved)**

vim (short for **Vi IMproved**) is a **powerful text editor** used in Linux to **create and edit files from the command line**.

It's an improved version of the older vi editor, and is **very popular among system admins and DevOps engineers**.

**💡 Why Use vim?**

* It's **lightweight**, fast, and works on **any Linux server** (no GUI needed).
* Very helpful to **edit config files**, **write scripts**, and **troubleshoot logs**.

**🚦Modes in vim:**

| **Mode** | **Purpose** |
| --- | --- |
| Normal Mode | Default mode for navigation & commands |
| Insert Mode | For typing text |
| Command Mode | For saving, quitting, searching |

**✅ How to Open a File:**

vim filename.txt

* If file exists → it opens for editing.
* If not → creates a new file.

**🔁 Switching Between Modes:**

| **Press** | **Action** |
| --- | --- |
| i | Enter Insert mode (to type) |
| Esc | Go back to Normal mode |
| : (colon) | Enter Command mode |

**🔹 1. Normal Mode Commands (Default mode after pressing Esc)**

| **Command** | **Action** |
| --- | --- |
| i | Switch to **Insert mode** (before cursor) |
| a | Switch to Insert mode (after cursor) |
| o | New line below & switch to Insert mode |
| **dd** | **Delete the current line** |
| **yy** | **Copy the current line** |
| **p** | **Paste below** |
| x | Delete character under cursor |
| **u** | **Undo** |
| **Ctrl + d** | **Scroll a half page down** |
| **Ctrl + u** | **Scroll a half page up** |
| Ctrl + r | Redo |
| **/text** | **Search for "text"** |
| n | Jump to next search result |
| 0 | Move to beginning of line |
| $ | Move to end of line |
| **gg** | **Go to first line** |
| **G** | **Go to last line** |

**🔹 2. Command Mode (Start by pressing : from Normal mode)**

| **Command** | **Action** |
| --- | --- |
| :w | Save (write) the file |
| :q | Quit |
| **:wq** | **Save and quit** |
| :x | Save and quit (same as :wq) |
| **:q!** | **Quit without saving** |
| **:set nu** | **Show line numbers** |
| :set nonu | Hide line numbers |
| :s/old/new/g | Replace ‘old’ with ‘new’ word in the current line |
| :%s/old/new/g | Replace ‘old’ with ‘new’ word in the whole file |
| :%d | Deletes the whole text in file |
| :noh | No highlight |
| :5 | Jump to line 5 |
| :help | Open help |
| :!ls | Run a shell command like ls |

**Use Named Registers to Copy Multiple Lines Randomly**

To copy and store each line separately, use **named registers** ("a, "b, "c, etc.) like this:

vim

:5y a " copies line 5 to register a

:10y b " copies line 10 to register b

:13y c " copies line 13 to register c

:15y d " copies line 15 to register d

“a p

“10 p

“13 p

“15 p

**🔹 3. Insert Mode (After pressing i, a, or o)**

| **Key** | **Action** |
| --- | --- |
| Esc | Return to Normal mode |

📌 In Insert Mode, you just **type text** like in a normal text editor.

**LINUX ADMINISTRATION**

**USER MANAGEMENT – GROUP MANAGEMENT – FILE PERMISSIONS AND OWNERSHIP**

**🧑‍💻 What is User Management in Linux?**

User Management is how Linux handles **who can access the system**, **what they can do**, and **how their access is controlled**. It includes:

* **Creating/Deleting users**
* **Assigning permissions**
* **Managing groups**
* **Setting passwords**
* **Restricting access**

This is essential for **system security**, especially in real-time **DevOps or admin work**.

**🗂️ System Files Involved in User Management**

| **File Name** | **Description** |
| --- | --- |
| /etc/passwd | Contains user account information |
| /etc/shadow | Stores encrypted passwords |
| /etc/group | Defines groups |
| /etc/gshadow | Secure group info |
| /etc/login.defs | Default settings for new user accounts |
| /etc/skel | Skeleton directory (default files for new users) |

**📄 Example: /etc/passwd**

Each line in this file looks like:

koushik:x:1001:1001:Koushik Dharavath:/home/koushik:/bin/bash

| **Field** | **Meaning** |
| --- | --- |
| Username | koushik |
| Password | x (stored in /etc/shadow) |
| UID | 1001 (Unique User ID) |
| GID | 1001 (Group ID) |
| Comment | Full name or description |
| Home Directory | /home/koushik |
| Shell | /bin/bash |

**🧱 Types of Users**

1. **Root user (Superuser)**
   * Has **UID = 0**
   * Can do anything on the system
   * Command: sudo su or sudo -i
2. **Normal user**
   * Created by admins or yourself
   * Limited permissions
3. **System user**
   * Used by services like mysql, nginx, postfix
   * Usually has **no login shell**

**🔤 What is adduser in Linux?**

adduser is a **higher-level command** in Linux used to **create a new user** along with their **home directory**, **shell**, **password**, and **default files**.

It is more **user-friendly** than useradd, which is more **manual and low-level**.

**🧑‍🏫 Purpose**

When you want to **create a new user account easily**, adduser handles:

* User creation
* Home directory setup
* Password prompt
* Shell assignment
* Basic info (Full name, etc.)

**✅ Basic Syntax**

sudo adduser [username]

Example:

sudo adduser Koushik

| **Option** | **Description** |
| --- | --- |
| **-c** | Adds or changes the comment (full name) |  | scription) |
| -u | User ID |
| -g | Group ID |
| -s | Shell |
| -d | Home directory |
| -m | Create home dir |
| -N | Don’t create group |
| -r | System account |

**Example using useradd:**

sudo useradd -u 1055 -g devteam -d /opt/dev -s /bin/bash -m koushik

**🔄 adduser vs useradd in Linux**

| **Feature** | **adduser** | **useradd** |
| --- | --- | --- |
| **Type** | High-level Perl script | Low-level system binary |
| **Ease of Use** | More user-friendly and interactive | Requires manual setup |
| **Default Home Creation** | Yes (by default) | No (needs -m to create home directory) |
| **Password Prompt** | Yes (asks for it during creation) | No (you must set it manually using passwd) |
| **Shell Setup** | Sets to /bin/bash by default | Needs to be specified with -s |
| **Group Handling** | Automatically creates group with same name | You must specify group manually |
| **Default Files Copy** | Copies /etc/skel files automatically | Doesn’t do this unless scripted |
| **Option Style** | Uses long options with -- | Uses short options with - |
| **Best For** | Beginners, scripts, simple user management | Advanced setups, custom automation scripts |

**✅ adduser Example (Simple & Friendly)**

sudo adduser koushik

This:

* Creates user koushik
* Sets up /home/koushik
* Prompts for password and full name
* Adds to group koushik
* Default shell: /bin/bash

**⚙️ useradd Example (Manual & Precise)**

sudo useradd -m -d /home/koushik -s /bin/bash -g devteam -u 1055 koushik

sudo passwd koushik

This:

* Manually sets home, shell, group, UID
* Requires separate command to set password

**📌 Summary Table**

| **Task** | **adduser** | **useradd** |
| --- | --- | --- |
| Create user with home dir | ✅ Default | ❌ Needs -m |
| Add password automatically | ✅ Prompts | ❌ Use passwd separately |
| Set shell | ✅ Default /bin/bash | ❌ Use -s /bin/bash |
| Add to group | ✅ Same-name group auto | ❌ Use -g groupname |
| Create system user | ✅ --system | ✅ -r |

**🧑‍💼 Interview-Friendly Explanation:**

"adduser is a simple, interactive command used to create new users with all defaults like home directory, shell, and password setup. useradd is a more manual, low-level command used for precise control, often used in automation scripts."

**🔐 What is passwd?**

passwd is a Linux command used to **set, update, or change the password** of a user account.

**✅ Basic Syntax:**

passwd [options] [username]

**🔧 Common Use Cases:**

| **Command** | **Purpose** |
| --- | --- |
| **passwd** | Change your own password |
| **sudo passwd username** | Change another user's password (admin use) |
| **sudo passwd -d username** | Delete password (user can login without one) |
| **sudo passwd -l username** | Lock a user account |
| **sudo passwd -u username** | Unlock a user account |
| **sudo passwd -e username** | Force user to change password at next login |

**What is usermod?**

usermod command in Linux is used to **modify an existing user account**. It’s very useful for admins in real-time scenarios to change user details without deleting or recreating the user.

**🔧 usermod – Basic Syntax:**

sudo usermod [options] username

**✅ Commonly Used Options with Syntax and Explanation:**

| **Option** | **Syntax Example** | **Explanation** |
| --- | --- | --- |
| -c | -c "Koushik Dharavath" | Adds or changes the comment (full name/description) |
| -d | -d /opt/dev\_home koushik | Changes the user's home directory **(without moving files)** |
| -m | -d /opt/dev\_home -m koushik | **Moves existing files** to new home directory as well |
| -e | -e 2025-12-31 koushik | Sets account expiry date (format: YYYY-MM-DD) |
| -g | -g developers koushik | Changes the **primary group** of the user |
| -G | -G docker,sudo koushik | Adds user to **supplementary groups** (comma-separated, no spaces) |
| -aG | -aG docker koushik | Adds to secondary group of the user, Appends user to new groups **without removing existing ones** |
| -L | -L koushik | Locks the user account (disables password login) |
| -U | -U koushik | Unlocks a previously locked account |
| -s | -s /bin/zsh koushik | Changes the user’s default shell |
| -l | -l newname koushik | Changes the **username** from koushik to newname |
| -u | -u 2022 koushik | Changes the **UID** of the user |

**🧹 userdel**

**📘 Definition:**

userdel is used to **delete a user account** and optionally remove the user's home directory and mail spool.

🔐 Only the **root user** or users with sudo privileges can run userdel.

**🛠️ Basic Syntax:**

sudo userdel [options] username

**🔧 Common Options with Syntax and Meaning:**

| **Option** | **Syntax Example** | **Description** |
| --- | --- | --- |
| -r | sudo userdel -r koushik | **Removes the user** along with their **home directory** and **mail spool** |
| (no option) | sudo userdel koushik | Deletes the user, but **keeps the home directory and files** |
| --force or -f | sudo userdel -f koushik | Force deletion even if the user is logged in or has running processes *(use with caution)* |

**🧪 Examples:**

**1. ❌ Delete user but keep home directory:**

sudo userdel koushik

**2. ❌ Delete user and their home directory:**

sudo userdel -r koushik

**3. ⚠️ Force delete user even if logged in:**

sudo userdel -f koushik

**Group Management in Linux**

In Linux, **groups** are used to manage and assign permissions to multiple users at once. This makes it easier to control access to files, directories, and system features.

**📌 Why Use Groups?**

* To **organize users** based on department, role, or project.
* To give **multiple users access** to the same files or resources.
* To apply **permissions** more efficiently.

**🔍 Types of Groups**

| **Type** | **Description** |
| --- | --- |
| **Primary Group** | Automatically assigned when a user is created. Usually matches the username. |
| **Secondary (Supplementary) Groups** | Additional groups a user can belong to for extra access rights. |

**Basic Syntax:**

sudo groupadd [options] groupname

**1. ✅ Create a new group with default GID:**

sudo groupadd devs

**2. ✅ Create a new group with a specific GID:**

sudo groupadd -g 2000 devs

📘 groupdel

groupdel is used to **delete a group** from the Linux system. This is helpful when the group is no longer needed.

🔐 Only the **root user** or a user with sudo privileges\*\* can use this command.

**🛠️ Basic Syntax:**

sudo groupdel groupname

**If devs is a primary group for a user:**

You’ll see an error like:

groupdel: cannot remove the primary group of user 'koushik'

You must change the user's primary group first using:

sudo usermod -g newgroup koushik

Then try deleting the group again.

**gpasswd**

**🔑 Definition:**

gpasswd is used to **administer /etc/group**, allowing you to:

* Set a **group password**
* Add or remove **users from a group**
* Assign a **group administrator**

🔐 You must be the **root user** or the **group administrator** to use this command for modifying groups.

**🛠️ Basic Syntax:**

sudo gpasswd [options] groupname

Or to manage users:

sudo gpasswd -a username groupname # Add user to group

sudo gpasswd -d username groupname # Remove user from group

**1. 🔐 Set a password for a group:**

sudo gpasswd devs

(You will be prompted to enter a password. This password is needed when someone uses newgrp to switch to this group.)

**2. ➕ Add user to a group:**

sudo gpasswd -a koushik devs

**3. ➖ Remove user from a group:**

sudo gpasswd -d koushik devs

**4. 👤 Make a user group admin:**

sudo gpasswd -A koushik devs

**5. 👥 Set multiple members (overwrites existing):**

sudo gpasswd -M koushik,ram devs