**Step-by-Step DevOps Practical Guide**

**1. Basic Git Commands**

Initialize a Git repository:

bash

Copy

Download

git init

Clone a repository:

bash

Copy

Download

git clone <repository\_url>

Check status:

bash

Copy

Download

git status

Add files to staging:

bash

Copy

Download

git add <file\_name> # Single file

git add . # All files

Commit changes:

bash

Copy

Download

git commit -m "Commit message"

Push to remote:

bash

Copy

Download

git push origin <branch\_name>

Pull latest changes:

bash

Copy

Download

git pull origin <branch\_name>

**2. Create & Fork Repositories in GitHub**

Create a new repository:

1. Go to **GitHub** → Click **"New"**.
2. Enter **Repository Name**, choose **Public/Private**.
3. Click **"Create Repository"**.

Fork a repository:

1. Navigate to the target repository (e.g., https://github.com/user/repo).
2. Click **"Fork"** (top-right).
3. Select your account to fork into.

**3. Branch, Merge, Rebase**

Create a branch:

bash

Copy

Download

git branch <branch\_name>

git checkout <branch\_name> # Switch to branch

Merge a branch:

bash

Copy

Download

git checkout main

git merge <branch\_name>

Rebase a branch:

bash

Copy

Download

git checkout <branch\_name>

git rebase main

**4. Git Collaboration & Cloning**

Add a collaborator (GitHub):

1. Go to **Repository → Settings → Collaborators**.
2. Add a GitHub username.

Clone a forked repo:

bash

Copy

Download

git clone https://github.com/your-username/repo.git

**5. GitLab Web IDE**

1. Open a project in **GitLab**.
2. Click **"Web IDE"** (top-right).
3. Edit files directly in the browser.
4. Commit changes via the left panel.

**6. Merge Requests in GitLab**

1. Push changes to a new branch:

bash

Copy

Download

git push origin new-feature

1. In GitLab, go to **Merge Requests → New Merge Request**.
2. Select **Source & Target branches**.
3. Add reviewers and submit.

**7. GitLab Workflow Management**

* Use **Issue Boards** for Kanban-style tracking.
* Create **Milestones** for deadlines.
* Use **Labels** for categorization.

**8. Jenkins CI/CD**

Install Jenkins:

bash

Copy

Download

java -jar jenkins.war --httpPort=8080

Create a Jenkins job:

1. Go to **New Item → Freestyle Project**.
2. Configure **Git Repository URL**.
3. Add build steps (e.g., mvn clean install).
4. Save and **Build Now**.

**9. Docker Commands**

List containers:

bash

Copy

Download

docker ps -a

Build an image:

bash

Copy

Download

docker build -t my-app:latest .

Run a container:

bash

Copy

Download

docker run -d -p 8080:80 my-app

Remove a container:

bash

Copy

Download

docker rm <container\_id>

**10. Containerized App with Docker**

1. Create a Dockerfile:

dockerfile

Copy

Download

FROM nginx:alpine

COPY index.html /usr/share/nginx/html

1. Build and run:

bash

Copy

Download

docker build -t my-website .

docker run -d -p 80:80 my-website

**11. Ad-hoc Ansible Commands**

Ping all hosts:

bash

Copy

Download

ansible all -m ping -i inventory.ini

Install a package:

bash

Copy

Download

ansible all -m apt -a "name=nginx state=present" -i inventory.ini

**12. Ansible Playbooks**

Example playbook.yml:

yaml

Copy

Download

---

- hosts: webservers

tasks:

- name: Install Nginx

apt:

name: nginx

state: present

- name: Start Nginx

service:

name: nginx

state: started

Run the playbook:

bash

Copy

Download

ansible-playbook -i inventory.ini playbook.yml

**Final Notes**

* Use git log to check commit history.
* Use docker logs <container\_id> to debug containers.
* Use ansible --version to verify Ansible installation.

**✅ PDF Export Ready** – You can copy this into a Markdown/PDF converter like **Pandoc** or use **Chrome Print (Ctrl+P → Save as PDF)**.

Would you like any section expanded further? 🚀

make more simple and detail

3 / 3

**Simple & Detailed Step-by-Step DevOps Guide**

*(From Basic Git to Ansible Playbooks)*

**1. Basic Git Commands**

**Initialize a Git Repository**

bash

Copy

Download

git init

* Creates a new .git folder (hidden) to track changes.

**Check Git Status**

bash

Copy

Download

git status

* Shows untracked/modified files.

**Add Files to Staging**

bash

Copy

Download

git add file.txt # Single file

git add . # All files

* Prepares files for commit.

**Commit Changes**

bash

Copy

Download

git commit -m "Added new feature"

* Saves changes with a message.

**Push to Remote (GitHub/GitLab)**

bash

Copy

Download

git push origin main

* Uploads commits to the remote repository.

**Pull Latest Changes**

bash

Copy

Download

git pull origin main

* Downloads updates from the remote.

**2. Create & Fork GitHub Repositories**

**Create a New Repository**

1. Go to [GitHub](https://github.com/" \t "_blank) → Click **"New"**.
2. Enter **Repository Name** (e.g., my-project).
3. Choose **Public/Private**.
4. Click **"Create Repository"**.

**Fork a Repository (Copy Someone Else’s Project)**

1. Go to the repo you want to fork (e.g., github.com/user/repo).
2. Click **"Fork"** (top-right).
3. Select your account.
4. Now you have a copy (github.com/your-username/repo).

**3. Branching, Merging & Rebasing**

**Create a New Branch**

bash

Copy

Download

git branch new-feature # Creates branch

git checkout new-feature # Switches to branch

(or)

bash

Copy

Download

git checkout -b new-feature # Creates & switches

**Merge a Branch into**main

bash

Copy

Download

git checkout main

git merge new-feature

**Rebase (Cleaner History than Merge)**

bash

Copy

Download

git checkout new-feature

git rebase main

**4. Git Collaboration (Cloning & Pull Requests)**

**Clone a Repository**

bash

Copy

Download

git clone https://github.com/user/repo.git

**Collaborate on GitHub**

1. **Owner**: Go to **Repo → Settings → Collaborators → Add People**.
2. **Contributor**: Clone, make changes, push to a new branch.
3. Open a **Pull Request (PR)** on GitHub.

**5. GitLab Web IDE (Edit Code in Browser)**

1. Open your GitLab project.
2. Click **"Web IDE"** (top-right).
3. Edit files directly.
4. Click **"Commit"** (left panel).

**6. Merge Requests in GitLab**

1. Push changes to a new branch:

bash

Copy

Download

git push origin new-feature

1. In GitLab:
   * Go to **Merge Requests → New Merge Request**.
   * Select **Source (new-feature)** & **Target (main)**.
   * Add reviewers & submit.

**7. GitLab Workflow (Issue Tracking)**

* **Issues**: Track bugs/features.
* **Boards**: Kanban-style workflow (To Do, Doing, Done).
* **Milestones**: Group issues for releases.

**8. Jenkins CI/CD (Automate Builds)**

**Run Jenkins Locally**

bash

Copy

Download

java -jar jenkins.war --httpPort=8080

* Open http://localhost:8080.

**Create a Jenkins Job**

1. Click **New Item → Freestyle Project**.
2. Under **Source Code Management**, select **Git** → Enter repo URL.
3. Under **Build Steps**, add:

bash

Copy

Download

echo "Hello Jenkins!"

1. Click **Save → Build Now**.

**9. Basic Docker Commands**

**List Running Containers**

bash

Copy

Download

docker ps

**Build a Docker Image**

bash

Copy

Download

docker build -t my-app:latest .

**Run a Container**

bash

Copy

Download

docker run -d -p 80:80 my-app

**Stop & Remove a Container**

bash

Copy

Download

docker stop <container\_id>

docker rm <container\_id>

**10. Simple Containerized App**

**Step 1: Create**Dockerfile

dockerfile

Copy

Download

FROM nginx:alpine

COPY index.html /usr/share/nginx/html

**Step 2: Build & Run**

bash

Copy

Download

docker build -t my-website .

docker run -d -p 80:80 my-website

* Open http://localhost to see your website.

**11. Ad-hoc Ansible Commands**

**Ping All Servers**

bash

Copy

Download

ansible all -m ping -i inventory.ini

**Install Nginx on Servers**

bash

Copy

Download

ansible all -m apt -a "name=nginx state=present" -i inventory.ini

**12. Ansible Playbook Example**

**Create**playbook.yml

yaml

Copy

Download

---

- hosts: webservers

tasks:

- name: Install Nginx

apt:

name: nginx

state: present

- name: Start Nginx

service:

name: nginx

state: started

**Run the Playbook**

bash

Copy

Download

ansible-playbook -i inventory.ini playbook.yml

**Final Tips**

✅ **Git**: Use git log to see commit history.  
✅ **Docker**: Use docker logs <container\_id> for debugging.  
✅ **Ansible**: Use ansible --version to check installation.

**📥 Save as PDF**:

1. Copy this guide into a **Markdown file (**devops-guide.md**)**.
2. Convert to PDF using:

bash

Copy

Download

pandoc devops-guide.md -o devops-guide.pdf

(or) **Print (Ctrl+P) → Save as PDF** in Chrome.