**Max’s Beginner-Friendly DevOps Deep-Dive Roadmap**

Max, we’ll still aim for mastery, but I’ll make the path beginner-friendly: smaller steps, gentler explanations, and more examples. Think of this like leveling up in a game—one skill at a time, but always building toward production-grade power.

**How to use this roadmap**

* **Daily cadence (60–90 mins to start):** 30% reading/watching basics, 40% practicing small commands, 30% journaling what you learned.
* **Proof of work:** Keep one repo: realred-devops/ with subfolders per topic. Commit every day, even if it’s just notes or screenshots.
* **Explain it simply:** At the end of each day, write a 3-line note: *What I learned, How I tested it, Where I got stuck.*

Repo scaffold:

realred-devops/

00-notes/

10-linux-git/

20-docker/

30-kubernetes/

40-terraform/

50-ansible/

60-cicd/

90-capstone-realred/

**Day 0 – Set up your arena**

**Goal:** Install tools + verify they run.

1. Install:
   * Git
   * Docker Desktop (includes Kubernetes option)
   * Terraform
   * Ansible (can be later if hard)
2. Verify:

git --version

docker --version

terraform -version

ansible --version # optional for now

1. Success check: All commands return versions.

**Days 1–2 — Linux basics + Git**

**Goal:** Get comfortable with the shell + version control.

* Practice:
  + Navigate with cd, ls, pwd
  + Manage files with cp, mv, rm, cat, less
  + Processes: ps aux, top
* Git basics:
* git init
* git add .
* git commit -m "first commit"
* git log
* git checkout -b feature-x
* git merge feature-x
* Journal: “How does Git help in team projects?”

**Days 3–5 — Docker fundamentals**

**Goal:** Understand containers and images.

* Learn: Image vs Container (image = recipe, container = running instance).
* Practice:
* docker pull nginx
* docker run -d -p 8080:80 nginx
* docker ps
* docker exec -it <container\_id> bash
* docker stop <container\_id>
* Journal: Difference between docker run and docker exec.

**Days 6–8 — Kubernetes basics**

**Goal:** Learn Pods, Deployments, Services.

* Enable Kubernetes in Docker Desktop.
* Practice:
* kubectl get nodes
* kubectl create deployment hello --image=nginx
* kubectl expose deployment hello --type=NodePort --port=80
* kubectl get svc
* Journal: How does a Service help access Pods?

**Days 9–11 — Terraform basics**

**Goal:** Write first IaC file.

* Install Terraform if not done.
* main.tf:

provider "docker" {}

resource "docker\_container" "nginx" {

image = "nginx"

name = "tf-nginx"

}

* Run:

terraform init

terraform apply

terraform destroy

* Journal: What does Terraform state do?

**Days 12–14 — Ansible basics**

**Goal:** Run first playbook.

* Write playbook.yml:

- hosts: localhost

tasks:

- name: Print Hello

debug:

msg: "Hello from Ansible"

* Run:

ansible-playbook playbook.yml

* Journal: What’s the difference between Ansible and Terraform?

**Days 15–17 — CI/CD basics**

**Goal:** Automate builds + test simple workflow.

* Use GitHub Actions.
* Create .github/workflows/docker.yml:

name: Build Docker

on: [push]

jobs:

build:

runs-on: ubuntu-latest

steps:

- uses: actions/checkout@v3

- name: Build Docker Image

run: docker build . -t test:latest

* Journal: Why automate builds?

**Days 18–21 — Put it together**

**Goal:** Deploy a small app end-to-end.

1. **App**: Python FastAPI with /health endpoint.
2. **Docker**: Build + run locally.
3. **K8s**: Deploy via kubectl create deployment.
4. **Terraform**: (optional) Manage infra.
5. **Ansible**: (optional) Install deps.
6. **CI/CD**: GitHub Actions builds + runs Docker.

Deliverable: A README showing step-by-step how you deployed your first app.

**What unstoppable looks like (for beginners)**

* You can explain what containers, Pods, playbooks, and IaC are.
* You can show one working demo app deployed via Docker and Kubernetes.
* You have a repo of daily notes proving your growth.

**Takeaway:** Master the basics first. Once you’re comfortable, we’ll crank up the labs to production-grade blue/green, rollback drills, observability, and chaos testing.

👉 Start with \*\*Day