Ziel: Du baust dir eine moderne Self-Hosting-Plattform – sauber, wartbar und update-fähig.
Enthalten: Docker + Compose (offiziell), gemeinsames Netzwerk, Stacks für Postgres, n8n, WikiCore (DokuWiki), Gitea, code-server, Filebrowser + optional Cockpit.

Für wen ist das? Für Einsteiger, die eine stabile Basis wollen – ohne Chaos bei Ordnern, Ports und Berechtigungen. **Empfehlung:** 64-Bit Raspberry Pi OS auf SSD (nicht SD-Karte).

---

1) Schnellstart (was du am Ende kannst)

- Docker & Docker Compose sauber installieren (Debian-Style)
- Eine Standard-Ordnerstruktur nutzen: `/srv/docker/corestack/`
- Ein gemeinsames Docker-Netz verwenden: `dc-net`
- Mehrere Stacks getrennt betreiben und trotzdem miteinander verbinden
- Updates und Backups pro Dienst sauber durchführen

---

2) Video-Reihe (Übersicht)

#	Video	Inhalt	Ergebnis
01	**Host vorbereiten + Docker installieren**	Architektur-Check, Storage-Check, System-Update, Cockpit (optional), Docker/Compose (offiziell), Test	Docker läuft, Compose läuft, keine sudo-Pflicht
02	**CoreStack deployen: Ordner + Netzwerk + Stacks**	CoreStack-Ordner, Rechte, dc-net, Postgres, n8n, WikiCore, Gitea, code-server, Filebrowser	Alle Dienste laufen + Browser-Tests

Hinweis: Die Reboots und langen Downloads werden im Video geschnitten. Du bekommst trotzdem alle Befehle 1:1.

---

3) Kursmaterial (PDF / Handout)

Dokument	Zweck
Kurs-Handout	Für Teilnehmer: Schritt-für-Schritt, druckfreundlich.

- **PDF-Export:** Du erzeugst die PDFs selbst (z. B. via dw2pdf).

4) Zielbild: Dienste & Ports

Dienst	Container	Port (Beispiel)	URL
WikiCore (DokuWiki)	`dc-wikicore`	8080	`http://<PI-IP>:8080`
n8n	`dc-n8n`	5678	`http://<PI-IP>:5678`
PostgreSQL	`dc-postgres`	intern	(nur im dc-net)
Gitea	`dc-gitea`	3000	`http://<PI-IP>:3000`
code-server	`dc-codeserver`	8443	`https://<PI-IP>:8443`
Filebrowser	`dc-filebrowser`	8081	`http://<PI-IP>:8081`
Cockpit (optional)	Host	9090	`https://<PI-IP>:9090`

5) Quicklinks (Copy & Paste)

Architektur & Storage

```
getconf LONG_BIT
dpkg --print-architecture
df -h
lsblk
```

System-Update

```
sudo apt update && sudo apt full-upgrade -y
sudo reboot
```

Docker ohne sudo + Test

```
sudo usermod -aG docker $USER
newgrp docker
docker version
docker compose version
```

6) Nächste Schritte

Bitte starte mit Video 01.
Danach Video 02 – erst dann werden Stacks deployed.

- → **Video 01:** Host vorbereiten + Docker installieren
- → **Video 02:** CoreStack Ordner + Netzwerk + Stacks starten

Digitalcraft Standard:

- pro Dienst ein eigener Ordner
- ein gemeinsames Netz (dc-net)
- Updates pro Stack
- Backups der Datenordner – nicht der Container