# Enablement

Goal is **to mirror developer workflows and learning paths**
1. Foundation first - the fundamentals needed to get up and running, no more and no less.
2. Core skills needed for committing asap - any subjects that are going to be frequently referenced and needed for regular development
3. Automated workflows - topics that get your developed code integrated and released
4. Specialized items - sections with a smaller audience or which do not have a parent section
 a. Docker guides
 b. Security

"I need to get set up" → "I need to write good code" → "I need to ship code" → specific items, "I need to use buildkit" or "I need to understand security"

**Novice developers** should be able to move from section to section in order, building on previous topics.
**Experienced developers** should be able to move through setup and select what they need for their work.

## Landing page
- Overview
- Reference links
- Should not be needed for non-developers

---

**"day 1" env setup, you should be running at the end (and no more)**

### Local and remote development
- Overview
- Differences and use cases
- Tools and requirements

**Set up a local development environment**
- "Quickstart" or checklist
- Link out to longer-form guides as needed
1. Git (link to guides)
2. Common/recommended installs (python, golang, kubernetes, k9s...)
   a. Note that your team docs have this
3. VSCode (and guide)
4. Orbstack
5. AI workflows (link to AI dev section)
6. Scripts or other common resources

**Configure git and GitHub**
- Auth and MFA requirements for GitHub
- SSH requirements
- Answer questions like:
  ○ Do I need to make a new account?
  ○ How can I forward emails to my CW address for CW repos?

**Configure Orbstack for local development** — Existing long-form guide

**Set up your VDI** — Existing long-form guide

**Coder guide when available, etc** — Add additional long-form guides as needed to augment set up

---

**topics that will get you committing code**

### Development tools and practices
- Overview
- **Link out** to git guides, don't replicate

**Best practices**
- CoreWeave best practices
- Existing "Working with GitHub at CoreWeave" guide
- Fold in most of current **"gotchas"** guide

**Code Review Guidelines**

**Use AI Agents in development**

**Commit signing**

**Testing environments** — This can become its own section as we add more content

**Git and GitHub for non-engineers**
- Optonal
- This is where the basic "git exercises" content could live if needed

**Additional guides…** — Add additional long-form guides to match evolving developer need

---

**topics that will get you integrating and releasing**

### CI/CD and automation

**GitHub actions**

**semantic-release**

**GitHub rulesets**

**Additional guides…**

---

**Docker section reduced, could fold into tools/practices or stay standalone**

### BuildKit and Docker images

**Docker images**

**Docker best practices**

---

### Software transparency and security

**SBOM and in-toto**

**Chainguard**

**Artifact management**