

*# Pattern 1: Basic build and test workflow*

**.PHONY:** build test clean

**build:**

`docker build -t myapp:latest .`

**test:** build

`docker run --rm myapp:latest npm test`

**clean:**

`docker rmi myapp:latest`

*# Pattern 2: Environment-specific configurations*

**.PHONY:** deploy-dev deploy-prod

**deploy-dev:**

`kubectl apply -f k8s/dev/`

**deploy-prod:**

`@echo "Deploying to production..."`

`@read -p "Are you sure? [y/N] " confirm && [[ $$confirm == [yY] ]] || exit 1`

`kubectl apply -f k8s/prod/`

*# Pattern 3: Parameterized targets*

**.PHONY:** deploy

ENV ?= dev

TAG ?= latest

**deploy:**

`@echo "Deploying to $(ENV) with tag $(TAG)"`

`docker build -t myapp:$(TAG) .`

`docker push myapp:$(TAG)`

`kubectl set image deployment/myapp container=myapp:$(TAG) -n $(ENV)`

*# Pattern 4: Self-documenting Makefile*

**.PHONY:** help

**.DEFAULT\_GOAL :=** help

**help:**

`@grep -E '^[a-zA-Z_-]+:.*?## .*$$' $(MAKEFILE_LIST) | sort | awk 'BEGIN {FS = ":.*?##"}'`

**setup: ##** Install dependencies

`npm install`

**lint: ##** Run linters

```
eslint . && stylelint "**/*.css"
```

*# Pattern 5: Local development environment*

```
.PHONY: dev down logs
```

**dev:** *## Start local development environment*

```
docker-compose up -d
```

**down:** *## Stop local development environment*

```
docker-compose down
```

**logs:** *## View logs*

```
docker-compose logs -f
```