# Makefile for RAG Pipeline Project

.PHONY: help install install-dev test test-unit test-integration lint format clean run-index run-query run-evaluate docker-build docker-run

# Default target

help:

@echo "Available commands:"

@echo " make install - Install production dependencies"

@echo " make install-dev - Install development dependencies"

@echo " make test - Run all tests"

@echo " make test-unit - Run unit tests only"

@echo " make test-integration - Run integration tests only"

@echo " make lint - Run linting checks"

@echo " make format - Format code"

@echo " make clean - Clean up generated files"

@echo " make run-index - Index documents"

@echo " make run-query - Run a query"

@echo " make run-evaluate - Run evaluation"

# Installation targets

install:

pip install -r requirements.txt

install-dev: install

pip install -e .

pre-commit install

# Testing targets

test:

pytest tests/ -v --cov=src --cov-report=html --cov-report=term

test-unit:

pytest tests/unit/ -v --cov=src --cov-report=term

test-integration:

pytest tests/integration/ -v

# Code quality targets

lint:

flake8 src/ tests/ --max-line-length=100 --extend-ignore=E203,W503

mypy src/ --ignore-missing-imports

black --check src/ tests/

isort --check-only src/ tests/

format:

black src/ tests/

isort src/ tests/

# Cleanup

clean:

find . -type d -name "\_\_pycache\_\_" -exec rm -rf {} +

find . -type f -name "\*.pyc" -delete

rm -rf .pytest\_cache/

rm -rf htmlcov/

rm -rf .coverage

rm -rf dist/

rm -rf build/

rm -rf \*.egg-info

# Running the application

run-index:

python main.py --mode index --documents data/documents/\*.pdf

run-query:

python main.py --mode query --question "$(QUESTION)" --load-existing

run-evaluate:

python main.py --mode evaluate --load-existing

run-demo:

python main.py --mode demo

# Docker targets

docker-build:

docker build -t rag-pipeline:latest .

docker-run:

docker run -it --rm -v $(PWD)/data:/app/data -v $(PWD)/logs:/app/logs rag-pipeline:latest

# Development helpers

watch-tests:

ptw tests/ -- -v

profile:

python -m cProfile -o profile.stats main.py --mode demo

python -m pstats profile.stats