1 One-time setup (on a new machine)

```
# Go to where you want the project
cd /Users/aarushimahajan/Desktop

# Clone your agentic project
git clone https://github.com/AarushiMahajan001/agentic-doc-maintainer.git
cd agentic-doc-maintainer

# Create and activate virtual environment
python3 -m venv .venv
source .venv/bin/activate

# Install dependencies
pip install -r requirements.txt

# Create .env with your Groq key
cat > .env << 'EOF'
GROQ_API_KEY=your_groq_api_key_here
GROQ_MODEL_NAME=llama-3.1-70b-versatile
EOF
```

---

2 Per-session setup (every time you open a new terminal)

```
cd /Users/aarushimahajan/Desktop/agentic-doc-maintainer
source .venv/bin/activate
```

---

3 Run on a GitHub repo

3.1 (Optional) Clear old repos/index/docs

```
cd /Users/aarushimahajan/Desktop/agentic-doc-maintainer
source .venv/bin/activate

rm -rf data/repo/*
rm -rf data/index/*
rm -rf data/docs/*
```

3.2 Clone target GitHub repo into `data/repo`

Pattern:

```
cd /Users/aarushimahajan/Desktop/agentic-doc-maintainer

git clone https://github.com/USERNAME/REPO_NAME.git data/repo/REPO_NAME

ls data/repo
ls data/repo/REPO_NAME
```

Example (Calgary crime repo):

```
git clone
https://github.com/AarushiMahajan001/Calgary_Crime_Data_Analysis_and_Neural_Ne
twork_Prediction.git \
  data/repo/Calgary_Crime_Data_Analysis_and_Neural_Network_Prediction

ls data/repo
ls data/repo/Calgary_Crime_Data_Analysis_and_Neural_Network_Prediction
```

3.3 (If needed) Convert a notebook → `.py`

```
cd data/repo/Calgary_Crime_Data_Analysis_and_Neural_Network_Prediction

Jupyter nbconvert --to script
Calgary_Crime_Data_Analysis_and_Neural_Network_Prediction.ipynb \
  --output Calgary_Crime_Data_Analysis_and_Neural_Network_Prediction.py

# If it ever writes .txt instead of .py, fix:
# mv something.txt something.py
# or:
# mv something.py.txt something.py

cd /Users/aarushimahajan/Desktop/agentic-doc-maintainer
```

3.4 Build / rebuild FAISS index

```
cd /Users/aarushimahajan/Desktop/agentic-doc-maintainer
source .venv/bin/activate

python scripts/ingest_repo.py

ls data/index    # should show: code.index  metadata.json
```

3.5 Run the agentic pipeline on a module

Pattern (VERY IMPORTANT):

- Real file: `data/repo/REPO_NAME/path/to/module.py`

- Arg to script: `REPO_NAME/path/to/module.py`

```
cd /Users/aarushimahajan/Desktop/agentic-doc-maintainer
source .venv/bin/activate

python scripts/run_cli_demo.py REPO_NAME/path/to/module.py
```

Example:

```
python scripts/run_cli_demo.py \

Calgary_Crime_Data_Analysis_and_Neural_Network_Prediction/Calgary_Crime_Data_A
nalysis_and_Neural_Network_Prediction.py
```

Inspect generated docs:

```
ls data/docs
cat
data/docs/Calgary_Crime_Data_Analysis_and_Neural_Network_Prediction_Calgary_Cr
ime_Data_Analysis_and_Neural_Network_Prediction.py.md
```

---

4 Run on local code (project on your machine)

Assume you have a local project:

```
/Users/aarushimahajan/Desktop/my_local_project
```

with `.py` files inside.

4.1 Copy local project into `data/repo`

```
cd /Users/aarushimahajan/Desktop/agentic-doc-maintainer
source .venv/bin/activate
```

```
cp -R /Users/aarushimahajan/Desktop/my_local_project data/repo/
```

```
ls data/repo
ls data/repo/my_local_project
```

4.2 Rebuild the index (now includes this project)

```
python scripts/ingest_repo.py
```

```
ls data/index     # check code.index + metadata.json
```

4.3 Run the pipeline on a module from that local project

Pattern:

- Real file: `data/repo/my_local_project/path/to/module.py`

- Arg: `my_local_project/path/to/module.py`

```
python scripts/run_cli_demo.py my_local_project/path/to/module.py
```

Example:

```
python scripts/run_cli_demo.py my_local_project/src/model.py
```

Inspect generated docs:

```
ls data/docs
cat data/docs/my_local_project_src_model.py.md
```