LAB #4

In this lab, you will use GitHub pipelines and a cloud-based LLMs to automate the entire workflow.

01

Create a new GitHub repository.

02

Add your OpenAI API token to your GitHub repository:

1. Open your repository on GitHub.
2. Go to Settings.
3. In the sidebar, click Secrets and variables → Actions.
4. Click New repository secret.
5. Name the secret (for example: OPENAI_API_KEY) and paste your token as the value.
6. Click Add secret to save.

OpenAI token: EXAMPLE sk-proj-xxxxxxxxxxxx-xxxxxxxxxxxx-xxxxxxxxxxxx-xxxxxxxxxxxx-xxxxx

03

Your repository structure should look like this:

- fetch_data.py
- preprocess_data.py
- use_llm.py
- data/
- data_preprocessed/
- results.csv
- .github/workflows/data.yaml

This means you should have three Python scripts, two folders (data/ and data_preprocessed/), one results file (results.csv), and a GitHub Actions workflow file (data.yaml).

04 In your use_llm.py script, make sure to use the following model:
gpt-4.1-nano-2025-04-14

05 Example workflow file:
Below is a general example of a simple GitHub Actions YAML file.

You can copy this as a starting point, paste it into your repository at .github/workflows/data.yaml, and modify it for your own project.

```
name: Example Workflow

on: push

jobs:
  example_job:
    runs-on: ubuntu-latest

    env:
      EXAMPLE_ENV_VAR: "example value"

    steps:
      - name: Checkout code
        uses: actions/checkout@v2

      - name: Set up Python
        uses: actions/setup-python@v2
        with:
          python-version: '3.x'

      - name: Install dependencies
        run: |
          pip install some-package

      - name: Run example script
        run: python example_script.py
        env:
          ANOTHER_ENV_VAR: "another value"

      - name: Configure Git
        run: |
          git config --global user.name 'github-actions[bot]'
          git config --global user.email 'github-actions[bot]@users.noreply.github.com'

      - name: Commit and push changes
        run: |
          git add .
          git commit -m "Example commit from workflow [skip ci]" || echo "No changes to commit"
          git pull --rebase
          git push
```

06

At the end, your workflow should roughly follow these steps:

- Fetch data from an external source
- Preprocess the data
- Analyze the data using the OpenAI API
- Save results (e.g., results.csv)
- Commit and push changes back to your repository