

# MS 클라우드 개발환경 깃허브 코드스페이스 기본 생성과 설정 맞추



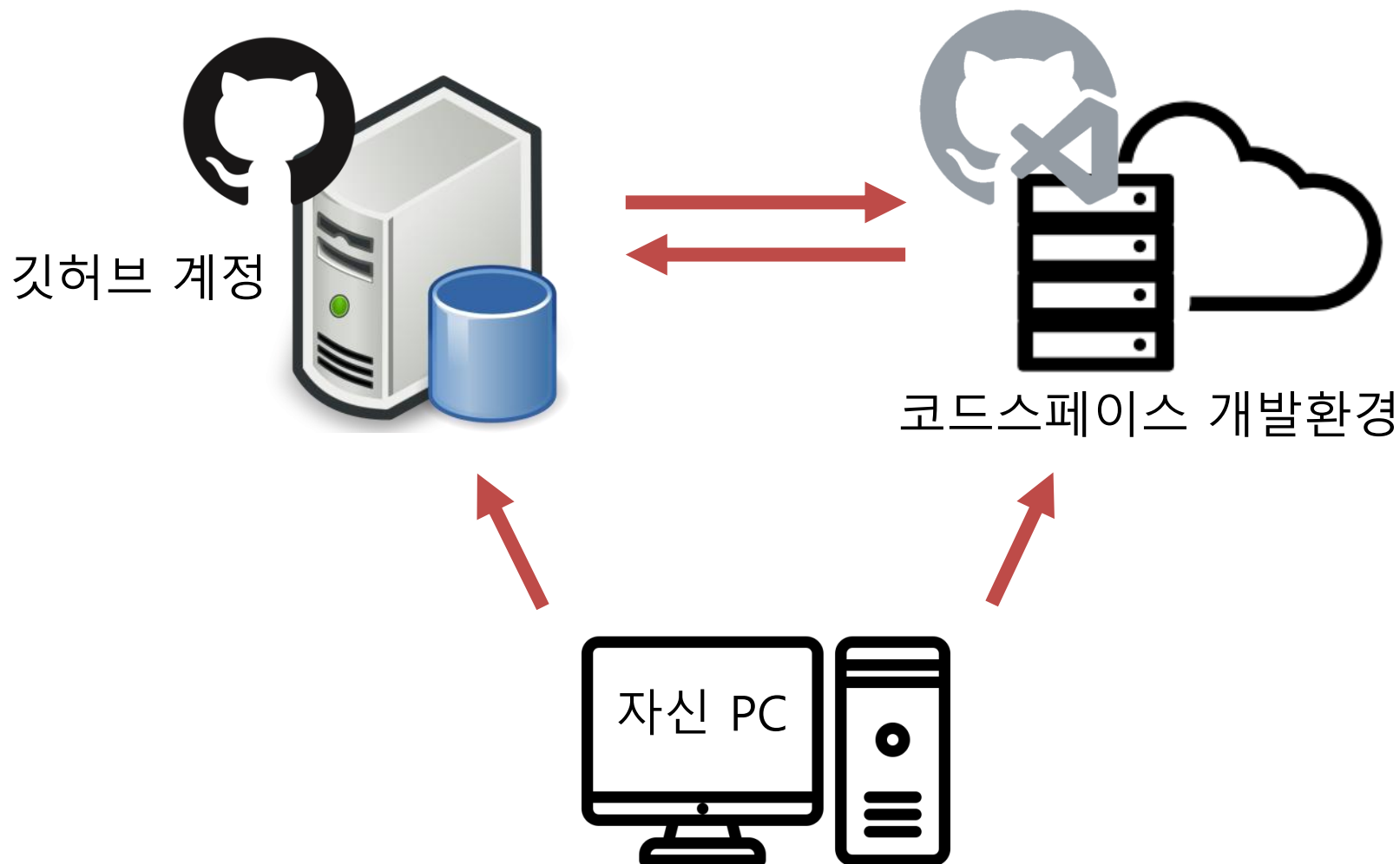
강환수 교수

AI Experts  
Who Lead  
The Future

# 01

## 깃허브 코드스페이스 기본 생성

- 깃허브에서 실행하는 클라우드 개발환경



- chatkang/naked-space

Codespaces x Create new codespace x Create a New Repository x +

github.com/new

Search or jump to... Pull requests Issues Codespaces Marketplace Explore

### Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Repository template

Start your repository with a template repository's contents.

No template

Owner \* Repository name \*

chatkang / naked-space

Great repository names are short and memorable. Need inspiration? How about [super-doodle?](#)

Description (optional)

기본 코드스페이스 생성

☒ Public  
Anyone on the internet can see this repository. You choose who can commit.

☐ Private  
You choose who can see and commit to this repository.

Initialize this repository with:

Skip this step if you're importing an existing repository.

☒ Add a README file  
This is where you can write a long description for your project. [Learn more.](#)

Add .gitignore

Choose which files not to track from a list of templates. [Learn more.](#)

.gitignore template: None

Choose a license

A license tells others what they can and can't do with your code. [Learn more.](#)

License: None

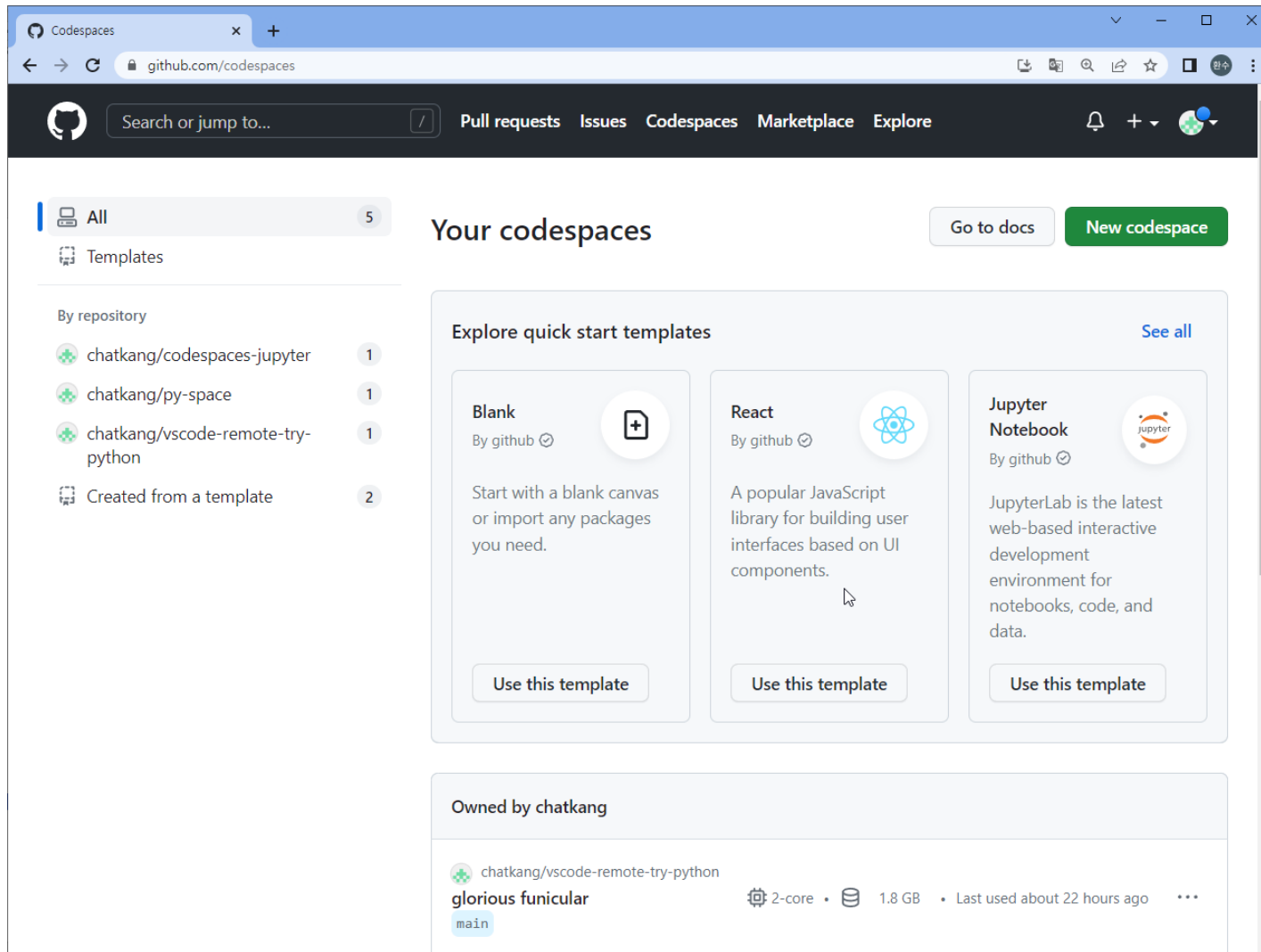
This will set `main` as the default branch. Change the default name in your [settings](#).

① You are creating a public repository in your personal account.

Create repository

© 2023 GitHub, Inc. Terms Privacy Security Status Docs Contact GitHub Pricing API Training Blog About

- 버튼 New codespace



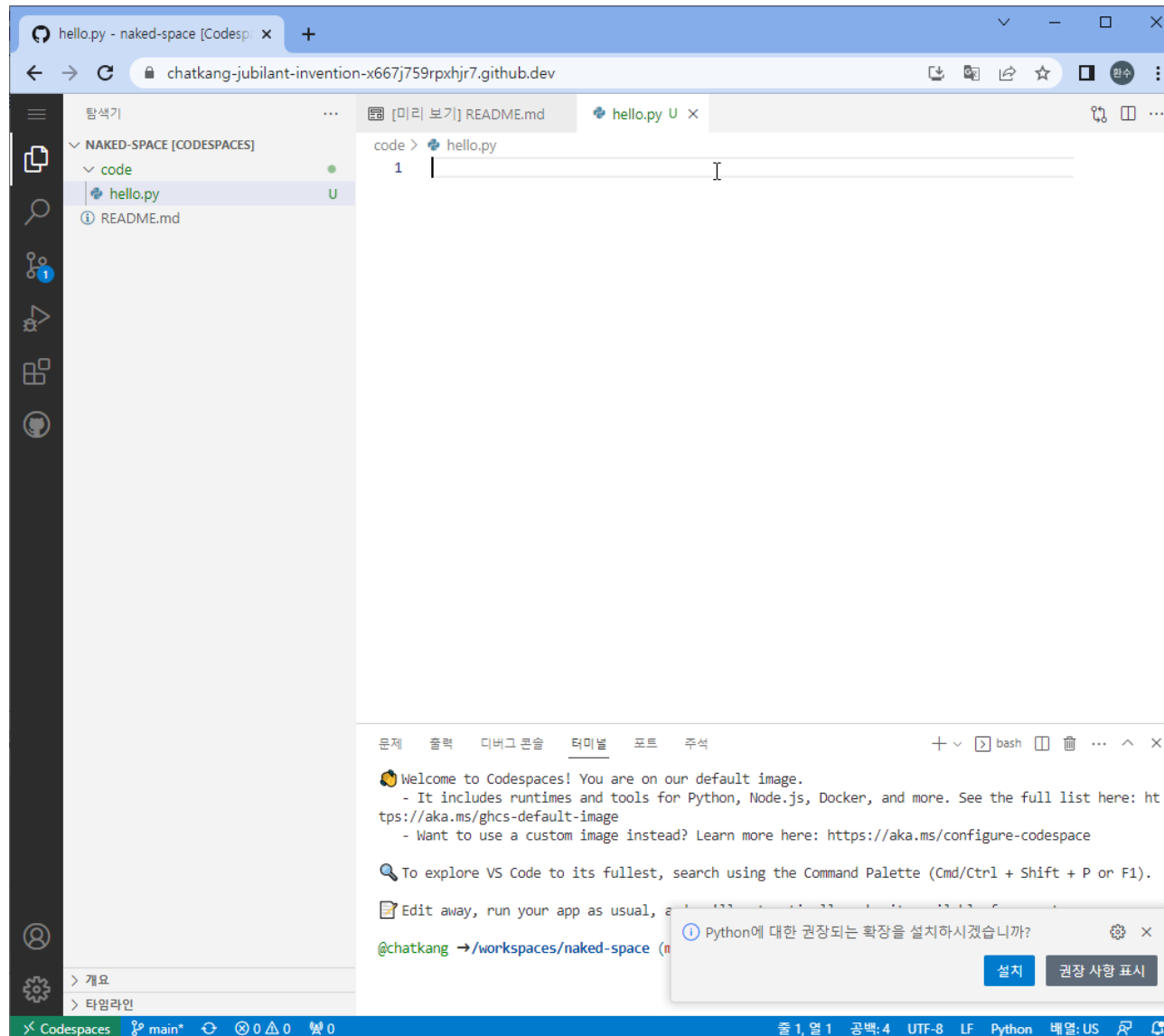
## 4개 선택

오픈소스 소프트웨어를 위한 깃과 깃허브 Python language

- 저장소, 브랜치, 지역, 하드웨어 머신

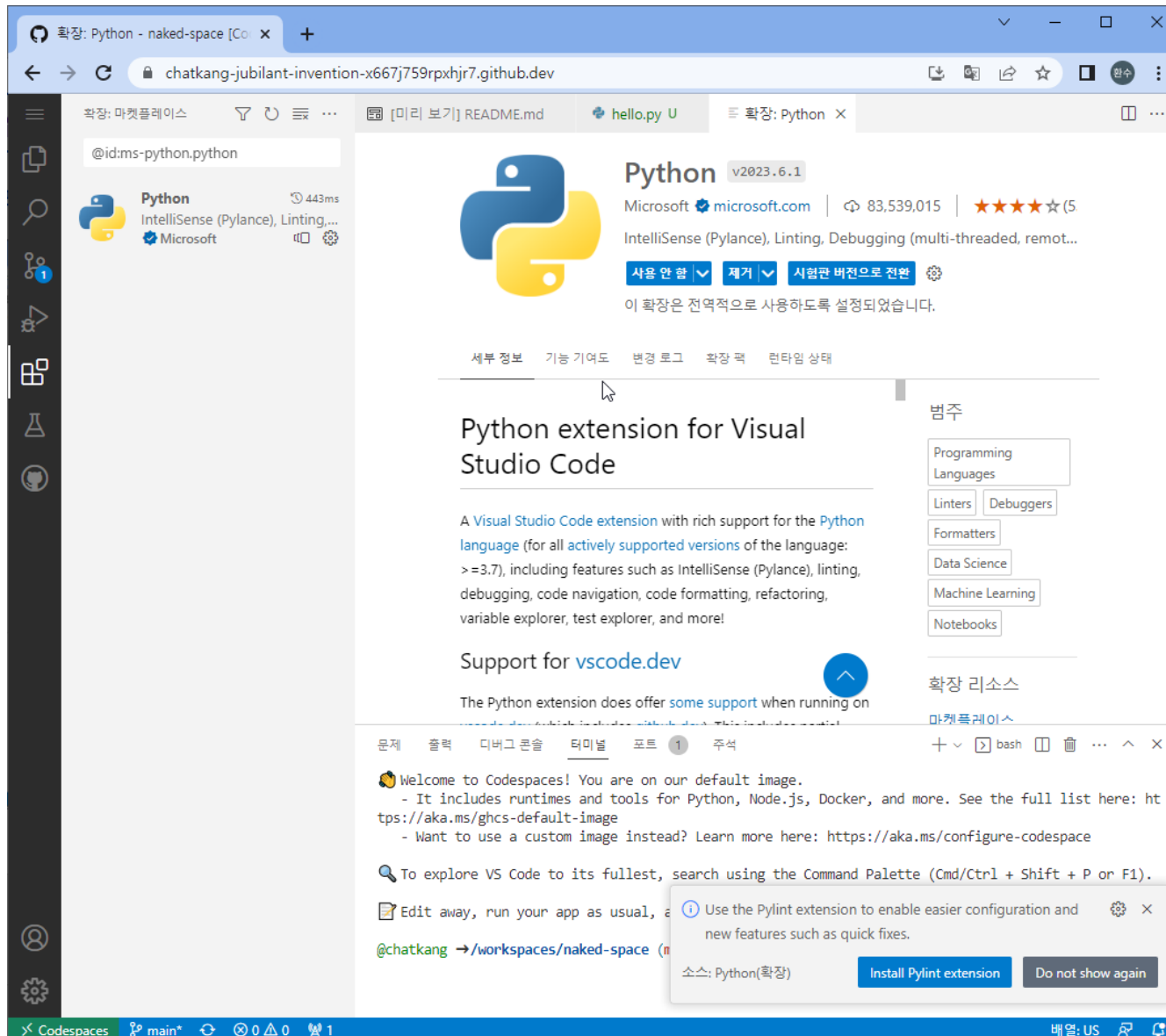
The screenshot shows the GitHub Codespaces interface for creating a new codespace. The browser tabs at the top include 'Codespaces' and 'Create new codespace'. The URL is 'github.com/codespaces/new?machine=basicLinux32gb&repo=631142464&ref=main&location=Sou...'. The page title is 'Create a new codespace'. A blue notification bar states 'Codespace usage for this repository is paid for by chatkang'. Below this, there are four selection sections: 'Repository' (To be cloned into your codespace) with a dropdown showing 'chatkang/naked-s...'; 'Branch' (This branch will be checked out on creation) with a dropdown showing 'main'; 'Region' (Your codespace will run in the selected region) with a dropdown showing 'Southeast Asia'; and 'Machine type' (2-core • 4GB RAM • 32GB storage) with a dropdown showing '2-core'. A link 'Contact our team' is provided for more power options. At the bottom right is a green 'Create codespace' button. The footer includes links for Terms, Privacy, Security, Status, Docs, Contact GitHub, Pricing, API, Training, Blog, and About, along with the GitHub logo and copyright notice '© 2023 GitHub, Inc.'.

- 기본 파이썬 설치

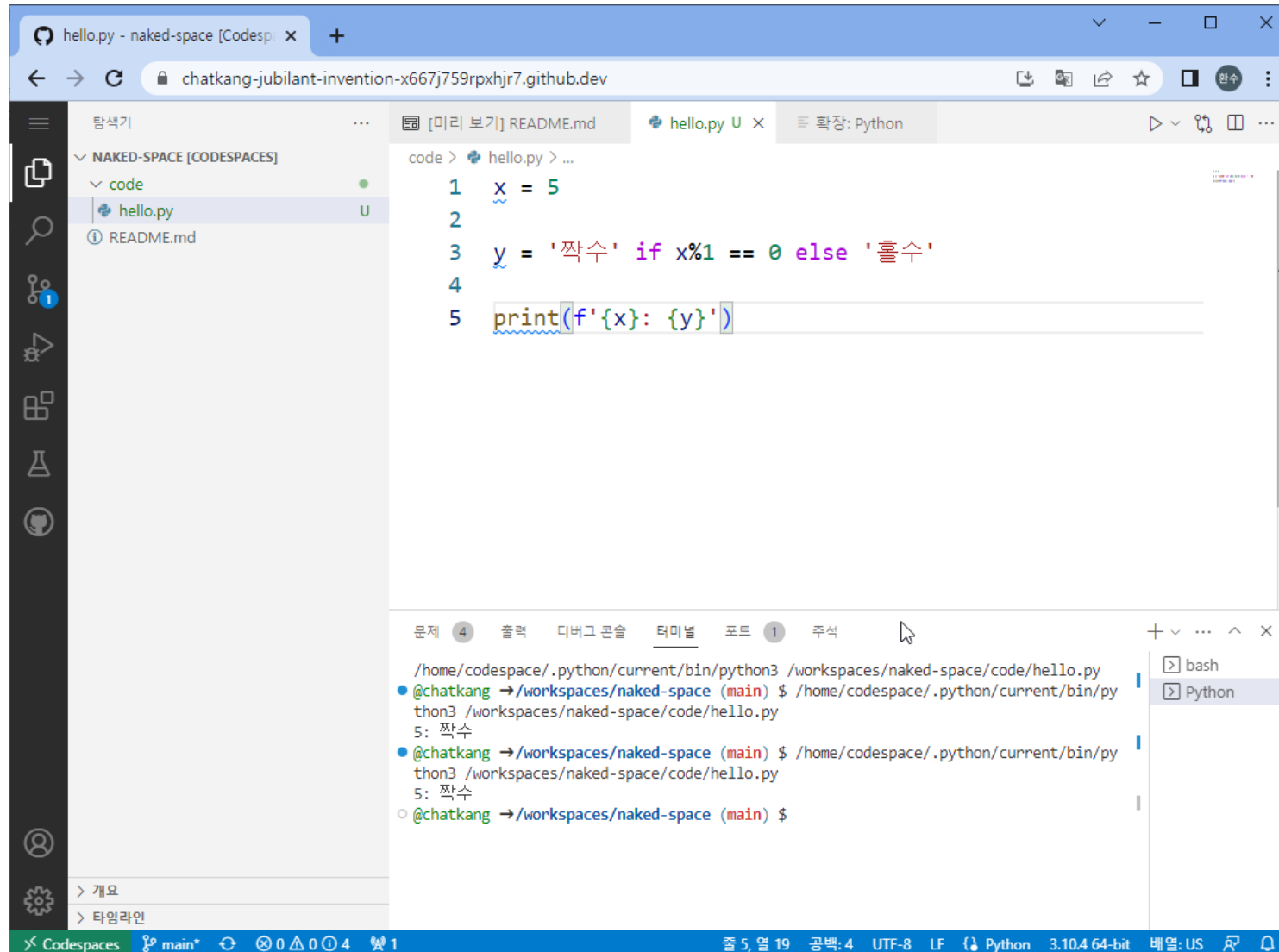


# 파이썬 설치된 화면

오픈소스 소프트웨어를 위한 깃과 깃허브 Python language







The screenshot displays a GitHub Codespace interface. The top bar shows the file 'hello.py' and the language 'Python'. The left sidebar contains a file explorer with 'code' and 'hello.py' selected. The main editor area shows the following Python code:

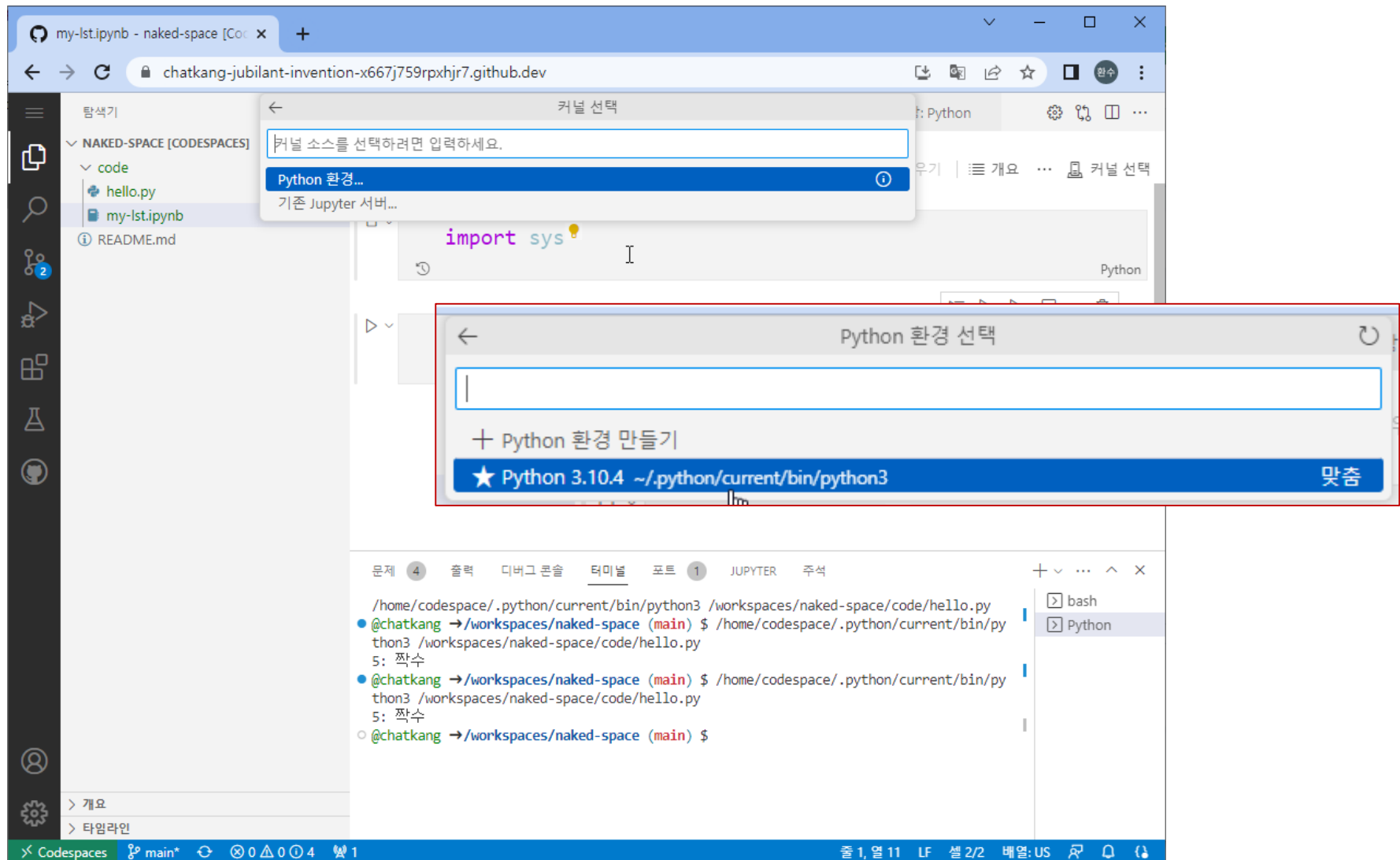
```
1 x = 5
2
3 y = '짝수' if x%1 == 0 else '홀수'
4
5 print(f'{x}: {y}')
```

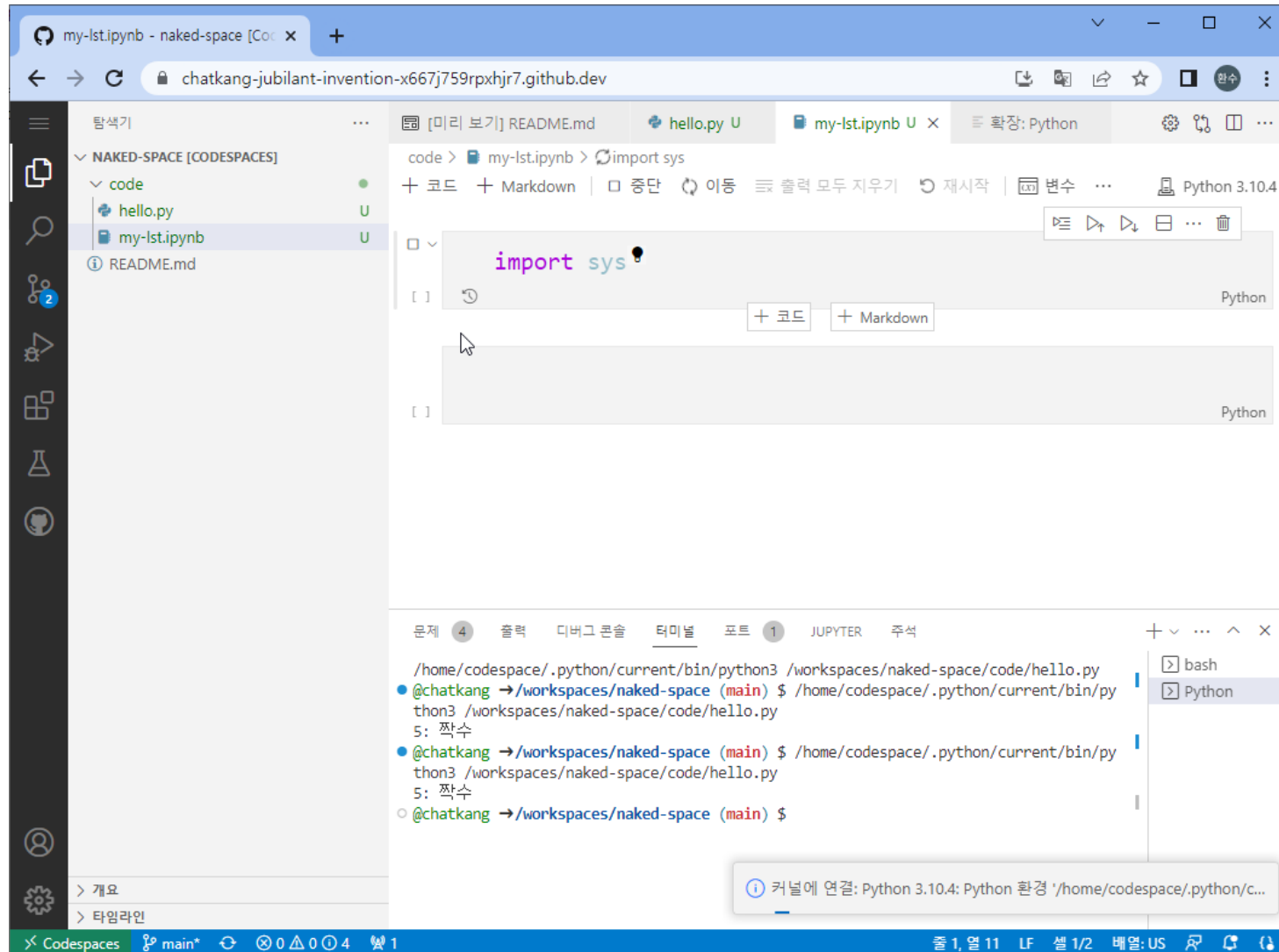
Below the code editor is a terminal window with the following output:

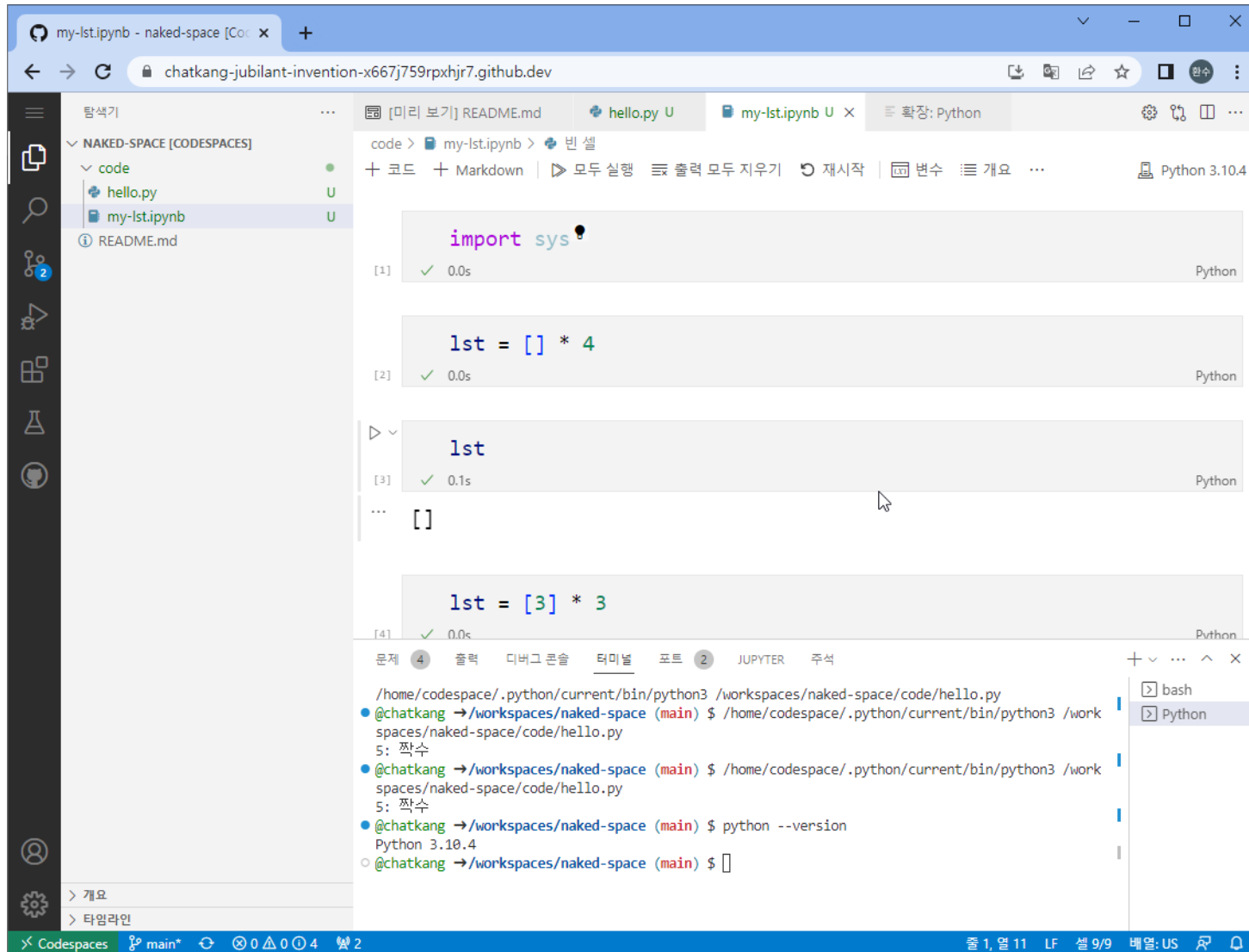
```
/home/codespace/.python/current/bin/python3 /workspaces/naked-space/code/hello.py
@chatkang →/workspaces/naked-space (main) $ /home/codespace/.python/current/bin/py
thon3 /workspaces/naked-space/code/hello.py
5: 짝수
@chatkang →/workspaces/naked-space (main) $ /home/codespace/.python/current/bin/py
thon3 /workspaces/naked-space/code/hello.py
5: 짝수
@chatkang →/workspaces/naked-space (main) $
```

The bottom status bar indicates the environment is 'Python 3.10.4 64-bit' with 'UTF-8' encoding and 'LF' line endings.

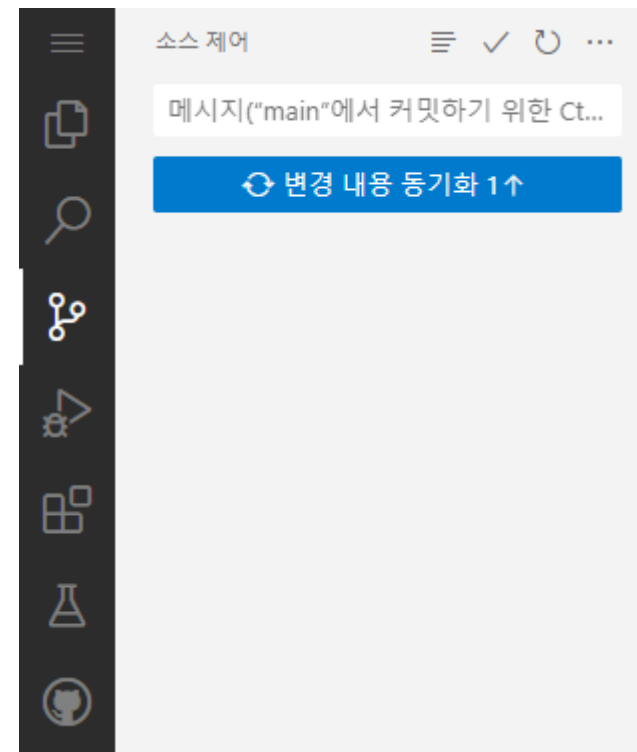
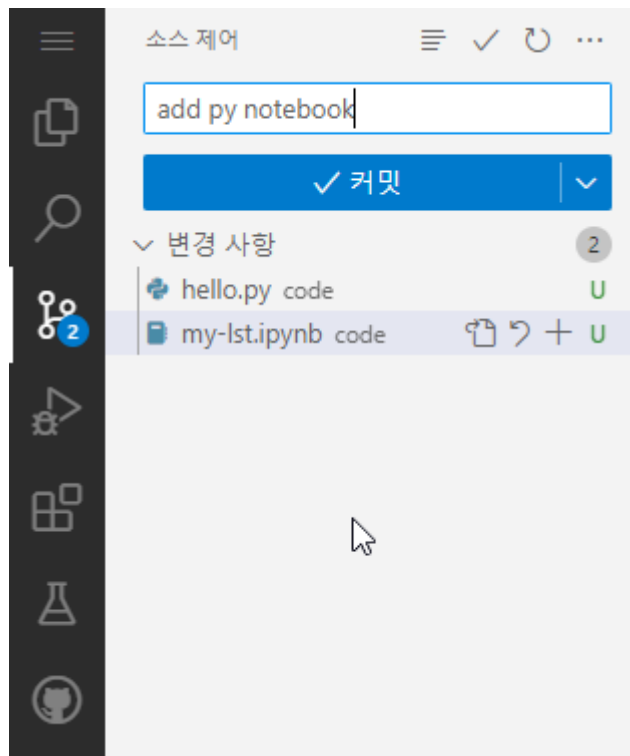
- Shift + Enter 실행
  - Python 환경 | Python 3.10.4 ~/.python/current/bin/python3







- 커밋 메시지 입력 후 커밋
- 동기화



# 자신의 저장소 확인

오픈소스 소프트웨어를 위한 깃과 깃허브 Python language

The screenshot shows a web browser displaying a GitHub repository page. The address bar shows the URL: `github.com/chatkang/naked-space/blob/main/code/my-lst.ipynb`. The repository name is `chatkang / naked-space` and it is marked as `Public`. The repository has 1 watch, 0 forks, and 0 stars. The file `my-lst.ipynb` is selected, showing its commit history and contributors. The file content is a Jupyter notebook with the following code cells:

```
In [1]: import sys

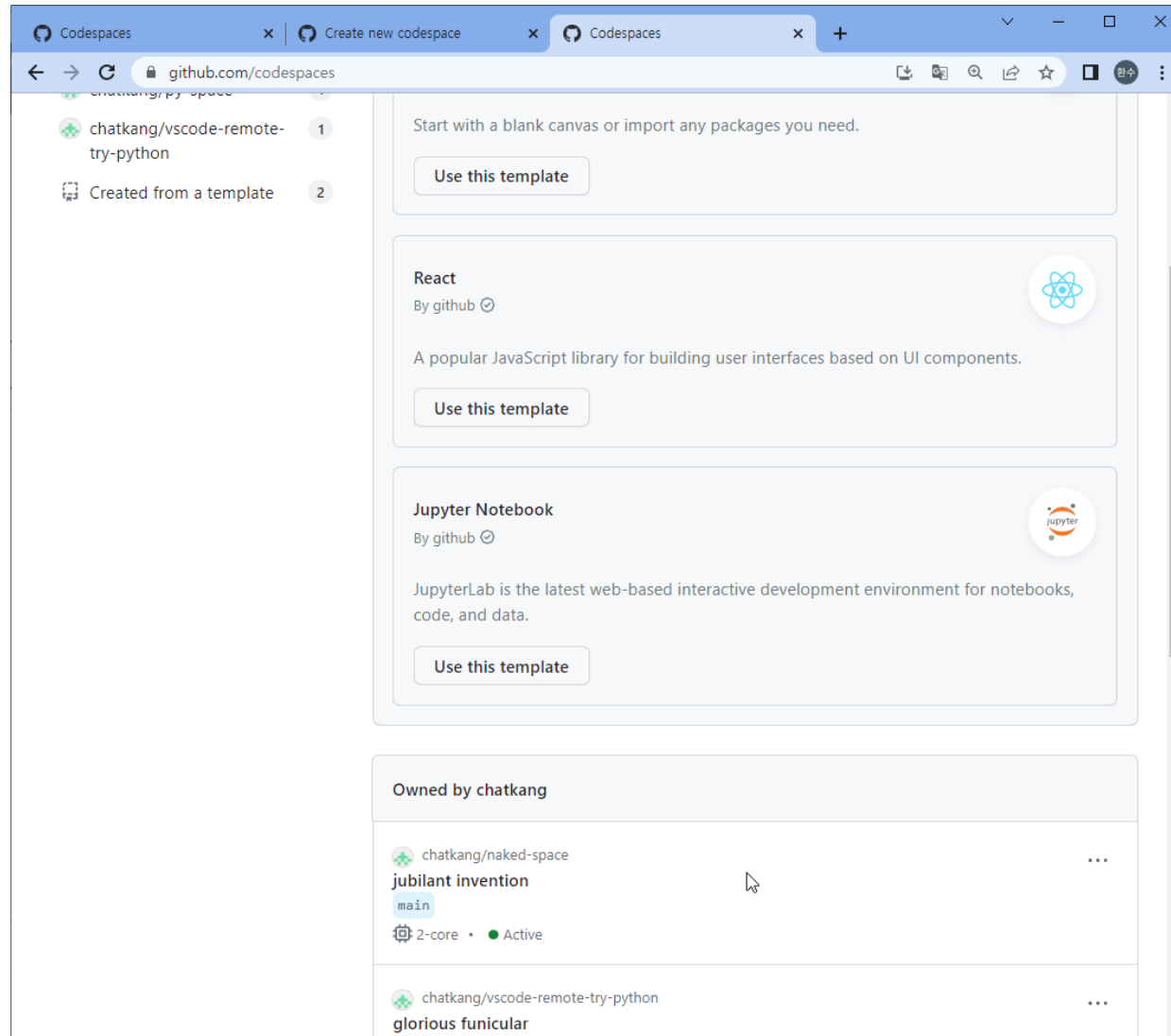
In [2]: lst = [] * 4

In [3]: lst

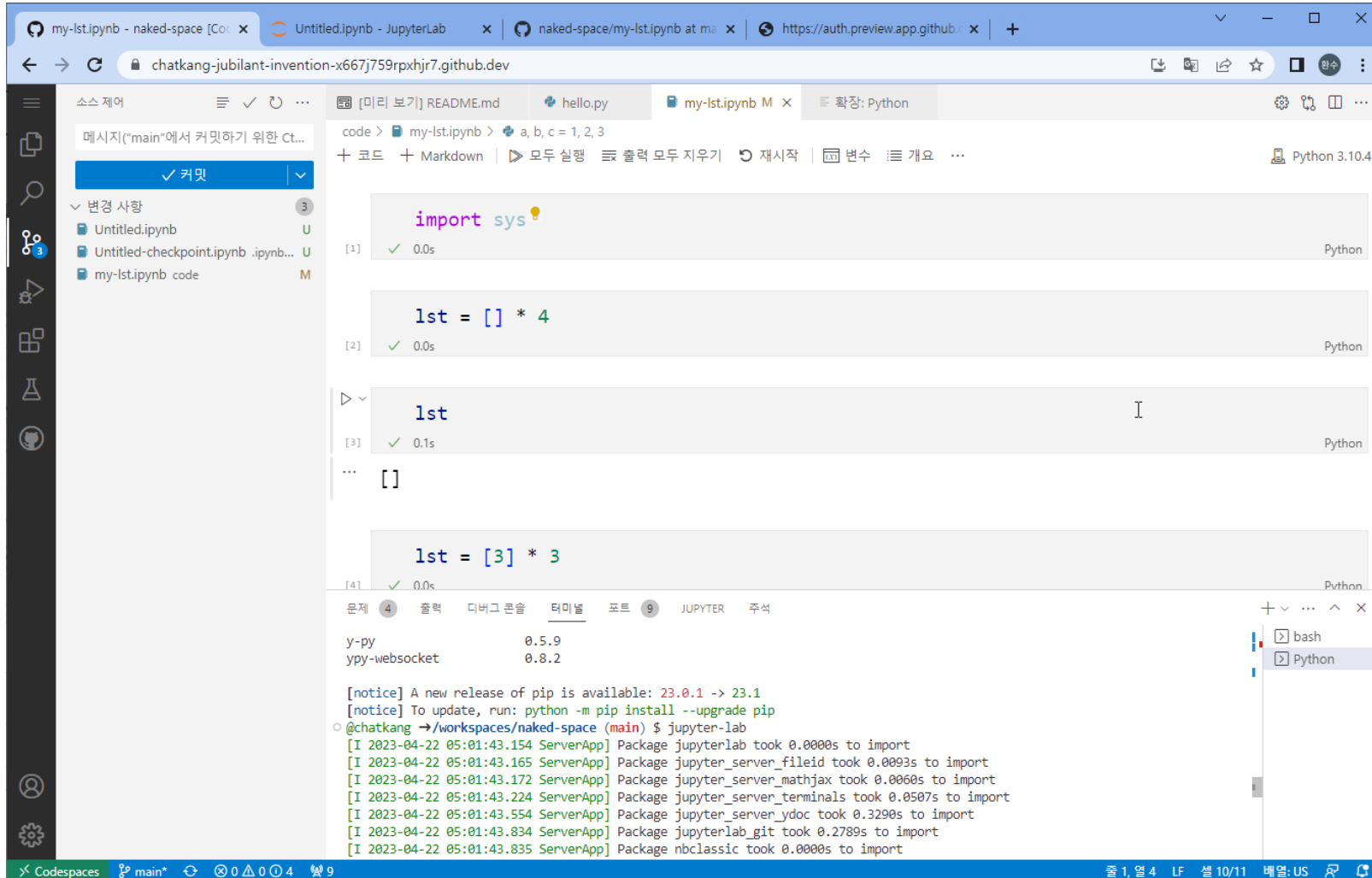
Out[3]: []

In [4]: lst = [3] * 3
```

- 자신의 코드스페이스 확인 가능



- 코드스페이스 터미널에서 명령  
- \$ jupyter-lab





- 다음이 토큰 암호

```
[I 2023-04-22 05:01:45.742 ServerApp]
```

```
http://localhost:8888/lab?token=9c14b70bb84851e885968123415b09de00330dd56a4dbafa
```

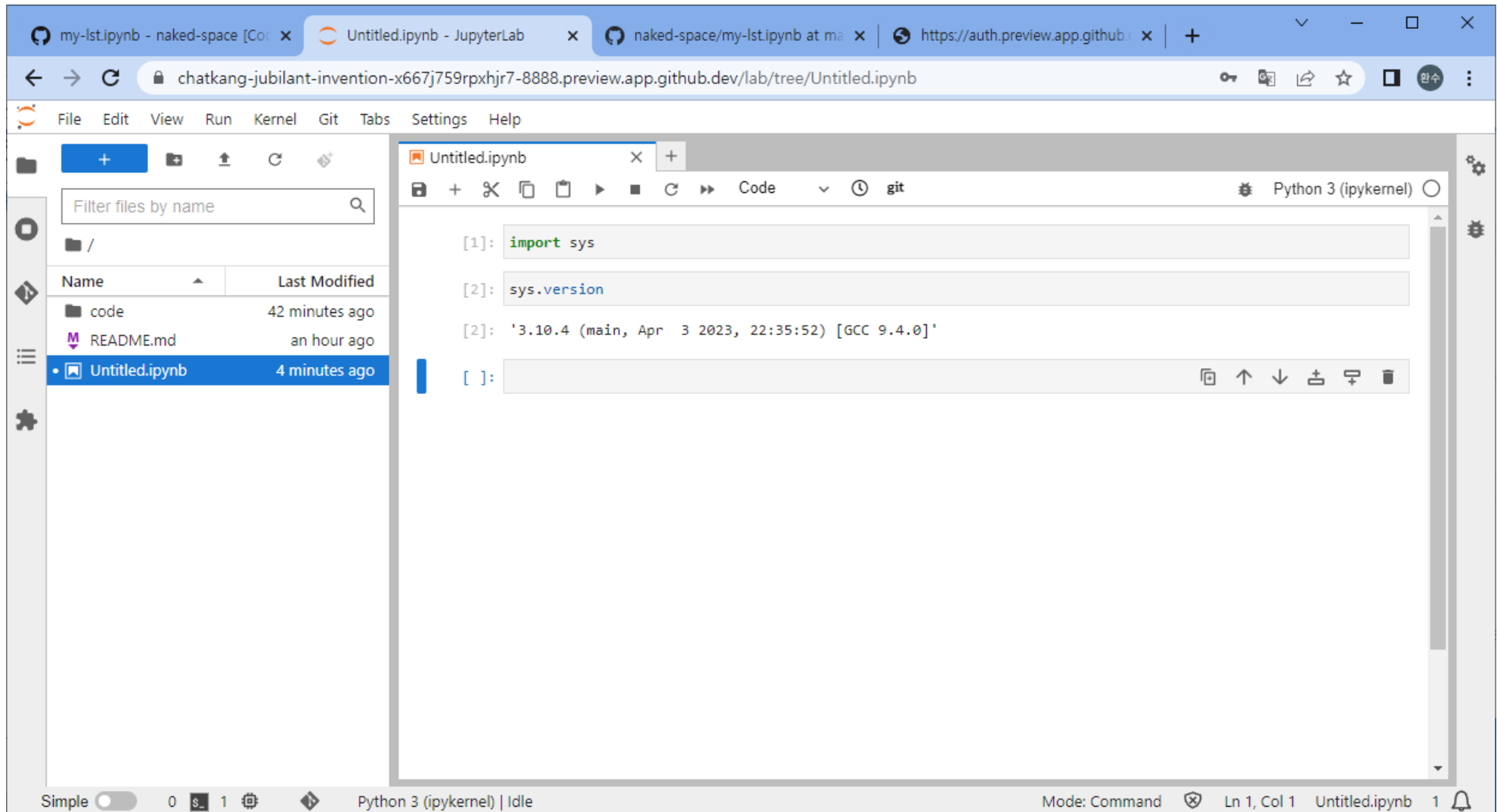
```
[I 2023-04-22 05:01:45.742 ServerApp]
```

```
http://127.0.0.1:8888/lab?token=9c14b70bb84851e885968123415b09de00330dd56a4dbafa
```

```
문제 4 출력 디버그 콘솔 터미널 포트 9 JUPYTER 주석
[I 2023-04-22 05:01:45.480 ServerApp] nbclassic | extension was successfully loaded.
[I 2023-04-22 05:01:45.741 ServerApp] nbdime | extension was successfully loaded.
[I 2023-04-22 05:01:45.742 ServerApp] Serving notebooks from local directory: /workspaces/naked-space
[I 2023-04-22 05:01:45.742 ServerApp] Jupyter Server 2.5.0 is running at:
[I 2023-04-22 05:01:45.742 ServerApp] http://localhost:8888/lab?token=9c14b70bb84851e885968123415b09de00330dd56a4dbafa
[I 2023-04-22 05:01:45.742 ServerApp] http://127.0.0.1:8888/lab?token=9c14b70bb84851e885968123415b09de00330dd56a4dbafa
[I 2023-04-22 05:01:45.742 ServerApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation
).
[C 2023-04-22 05:01:45.771 ServerApp]
```

# 코드스페이스의 주피터랩 실행

오픈소스 소프트웨어를 위한 깃과 깃허브 Python language

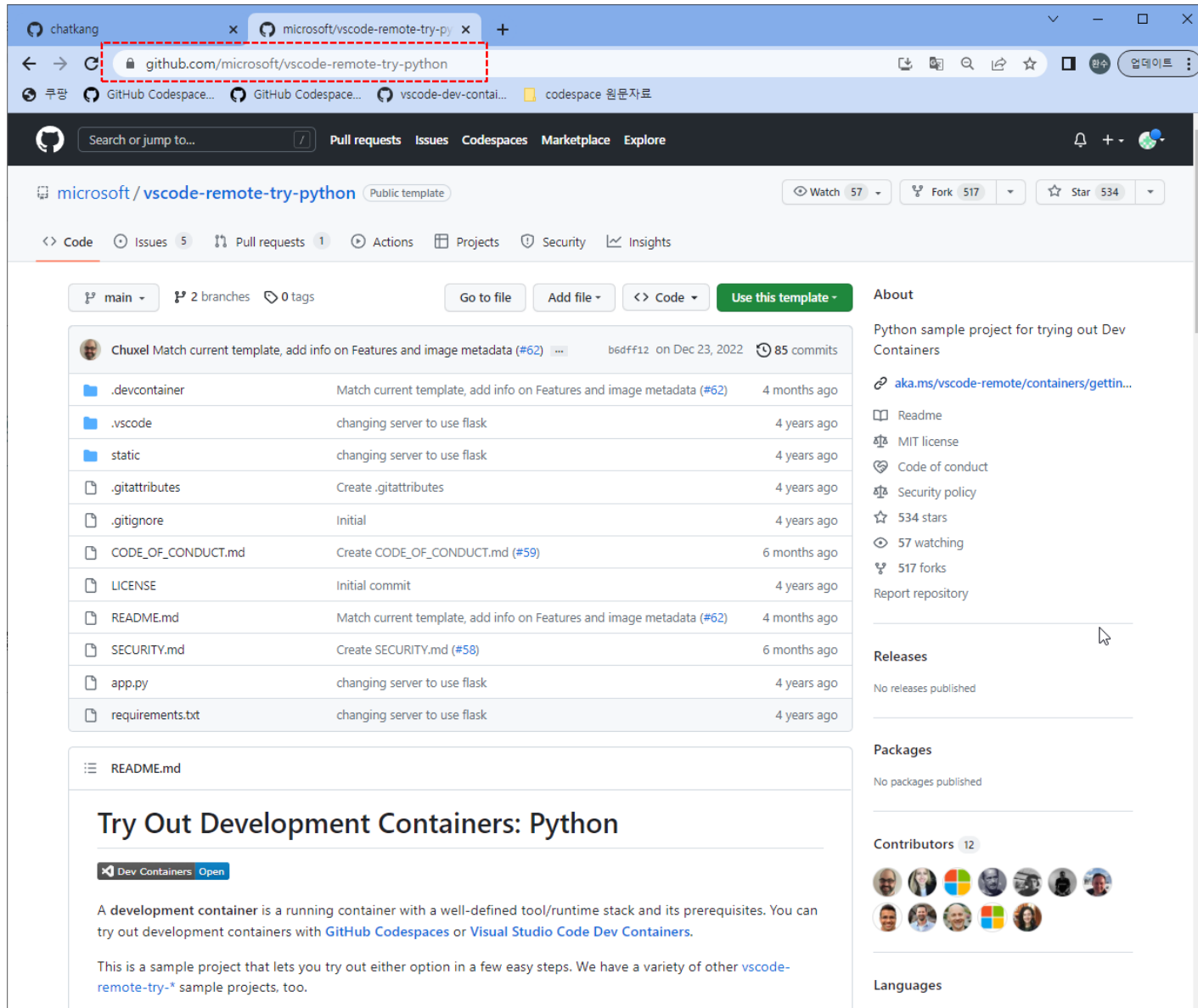


AI Experts  
Who Lead  
The Future

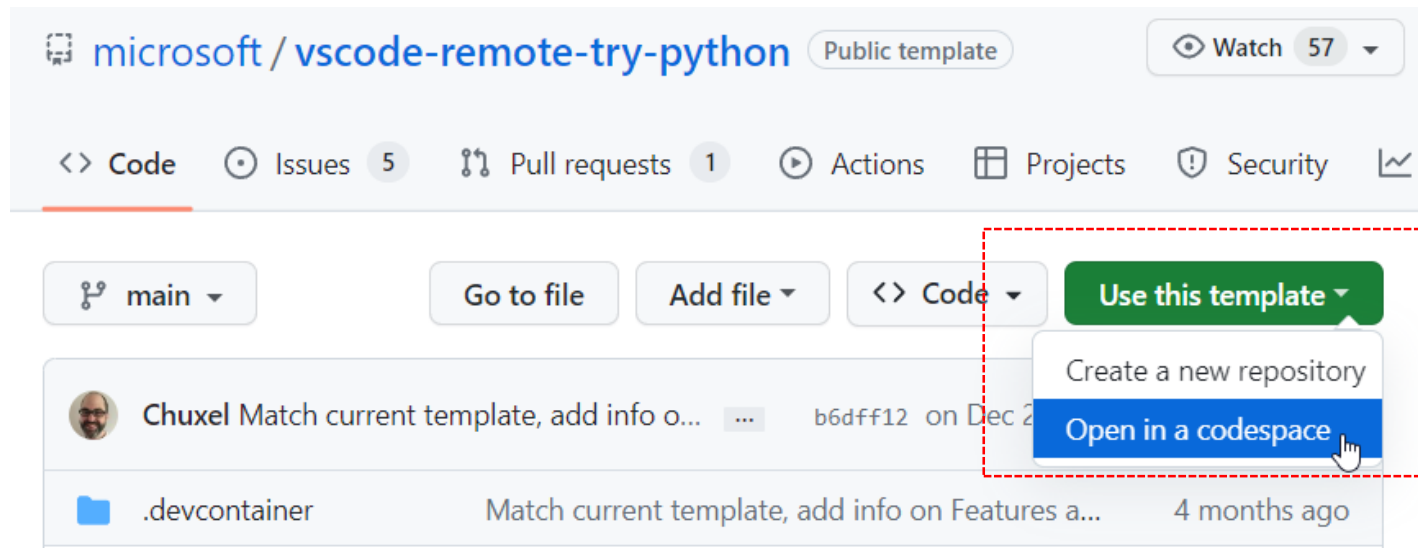
## 02

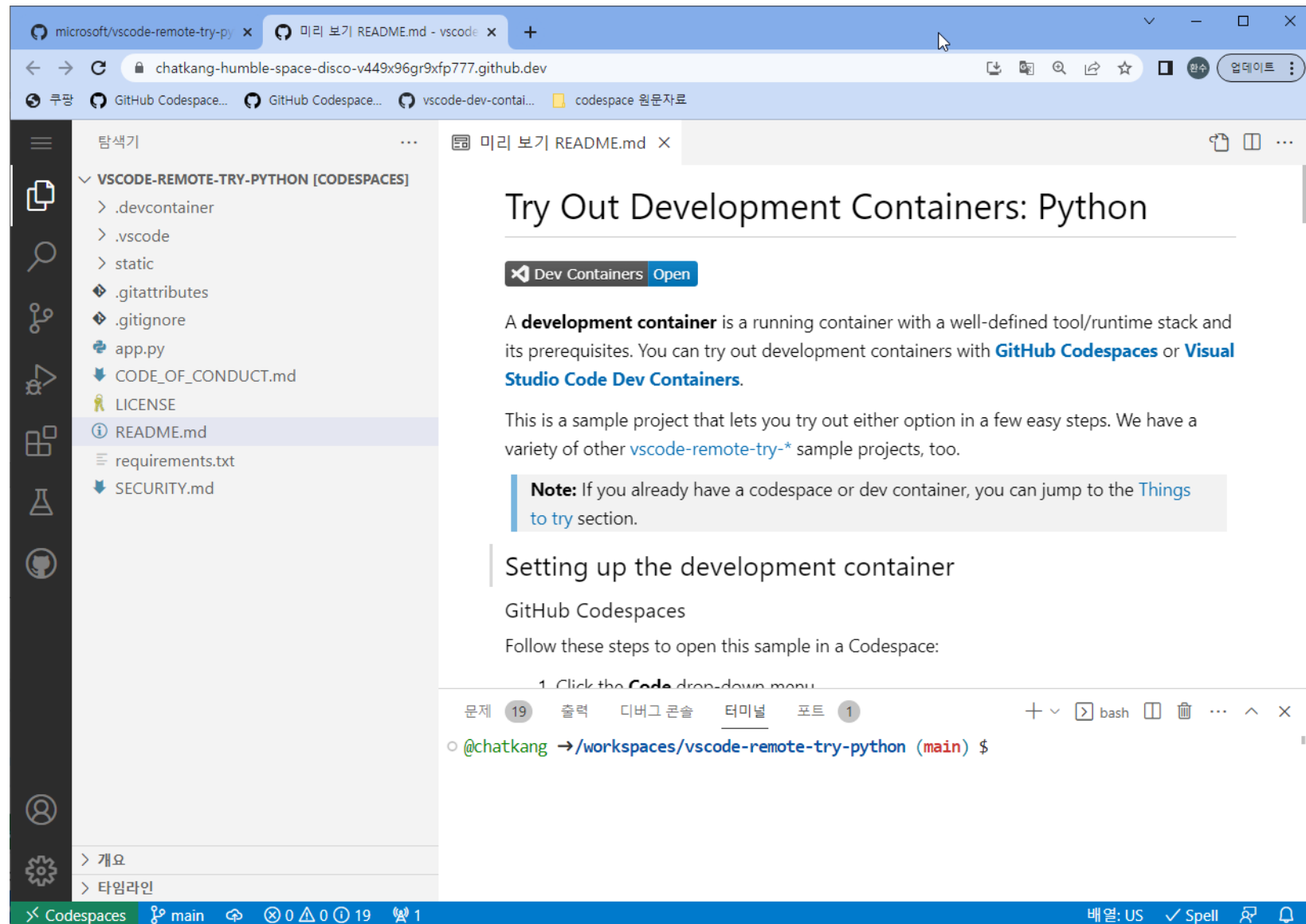
### 코드스페이스 템플릿의 설정 파일 devcontainer.json

- <https://github.com/microsoft/vscode-remote-try-python>



- 템플릿의 모든 파일을 이용해 코드 스페이스 생성





- `.devcontainer/devcontainer.json`

```
// For format details, see https://aka.ms/devcontainer.json. For config options, see the
// README at: https://github.com/devcontainers/templates/tree/main/src/python
{
  "name": "Python 3",
  // Or use a Dockerfile or Docker Compose file. More info: https://containers.dev/guide/dockerfile
  "image": "mcr.microsoft.com/devcontainers/python:0-3.11",

  // Features to add to the dev container. More info: https://containers.dev/features.
  // "features": {},

  // Configure tool-specific properties.
  "customizations": {
    // Configure properties specific to VS Code.
    "vscode": {
      "settings": {},
      "extensions": [
        "streetsidesoftware.code-spell-checker"
      ]
    }
  },

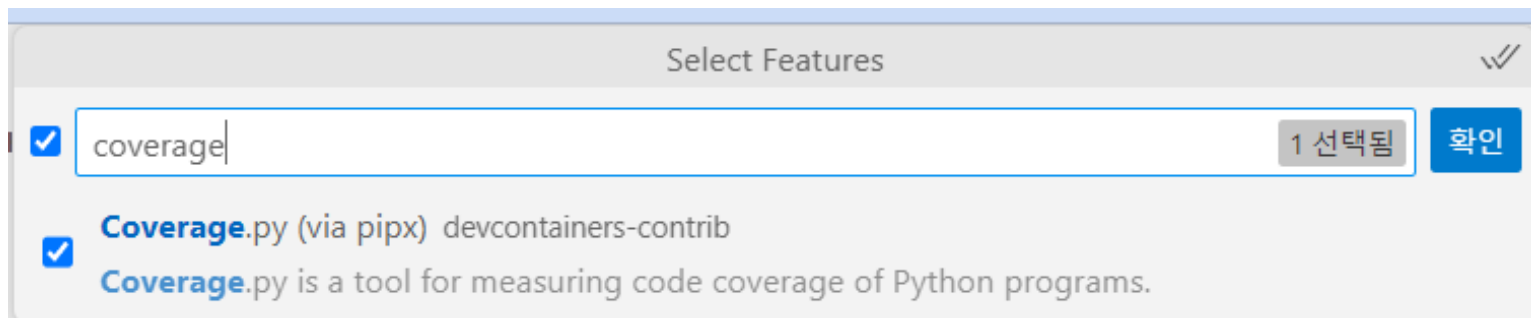
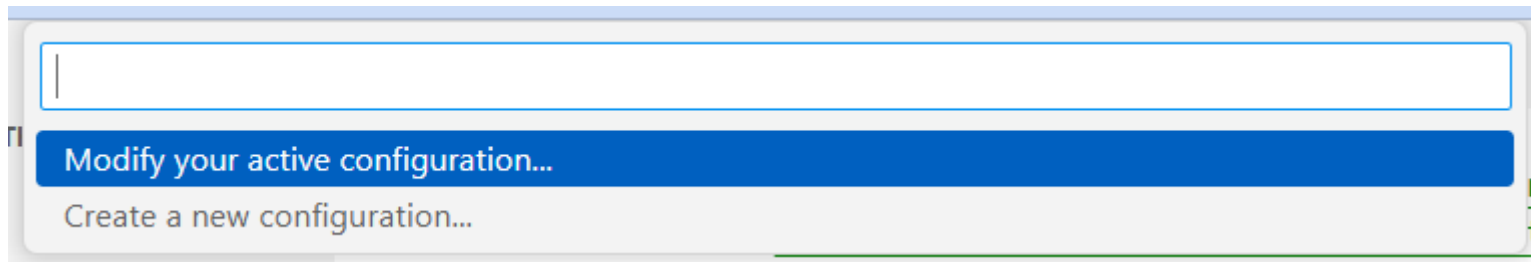
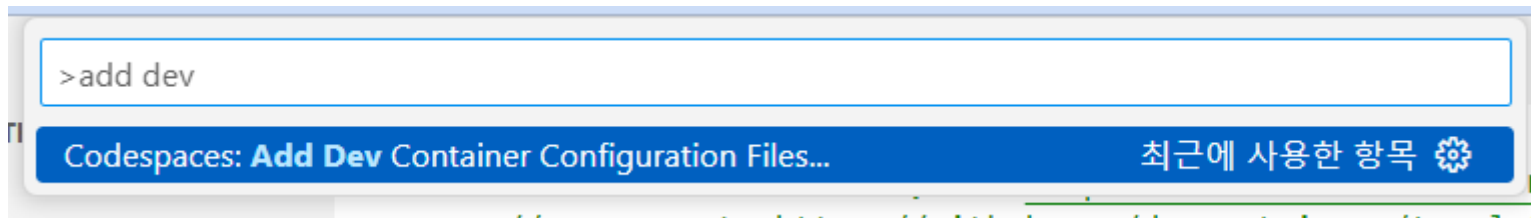
  // Use 'forwardPorts' to make a list of ports inside the container available locally.
  // "forwardPorts": [9000],

  // Use 'portsAttributes' to set default properties for specific forwarded ports.
  // More info: https://containers.dev/implementors/json_reference/#port-attributes
  "portsAttributes": {
    "9000": {
      "label": "Hello Remote World",
      "onAutoForward": "notify"
    }
  },

  // Use 'postCreateCommand' to run commands after the container is created.
  "postCreateCommand": "pip3 install -r requirements.txt"

  // Uncomment to connect as root instead. More info: https://aka.ms/dev-containers-non-root.
  // "remoteUser": "root"
}
```

- **Ctrl + Shift + P**
  - >add dev





# Configure Options

오픈소스 소프트웨어를 위한 깃과 깃허브 Python language

Keep Feature defaults or configure options?

  
**Keep Defaults**  
Configure Options

```
// For format details, see https://aka.ms/devcontainer.json. For config options, see the
// README at: https://github.com/devcontainers/templates/tree/main/src/python
{
  "name": "Python 3",
  // Or use a Dockerfile or Docker Compose file. More info: https://containers.dev/guide/dockerfile
  "image": "mcr.microsoft.com/devcontainers/python:0-3.11",

  // Features to add to the dev container. More info: https://containers.dev/features.
  // "features": {},

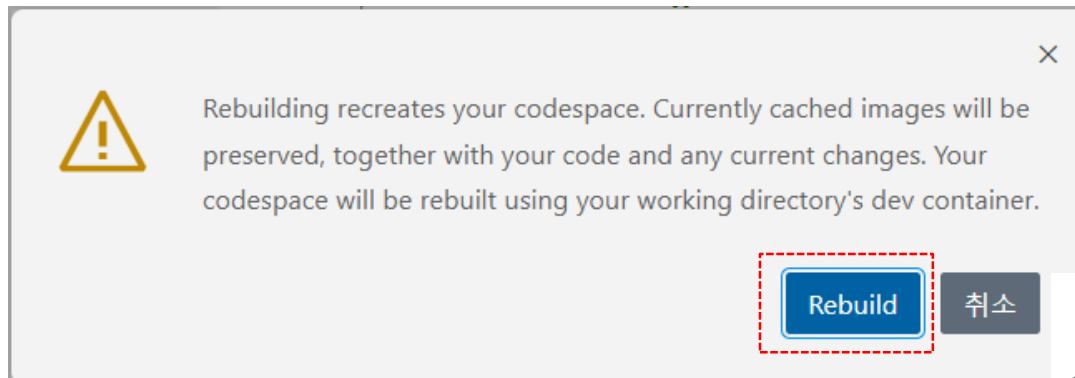
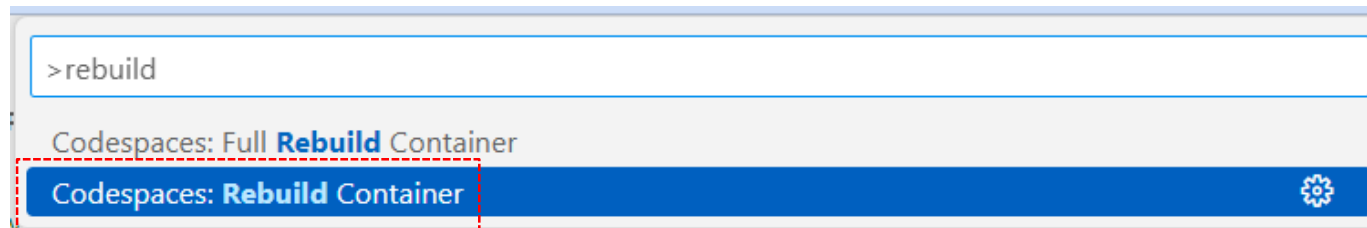
  // Configure tool-specific properties.
  "customizations": {
    // Configure properties specific to VS Code.
    "vscode": {
      "settings": {},
      "extensions": [
        "streetsidesoftware.code-spell-checker"
      ]
    }
  },

  // Use 'forwardPorts' to make a list of ports inside the container available locally.
  // "forwardPorts": [9000],

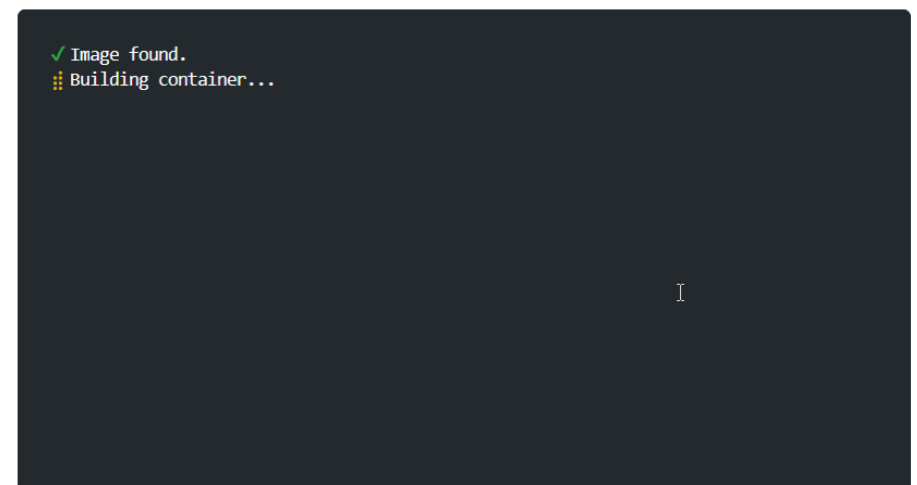
  // Use 'portsAttributes' to set default properties for specific forwarded ports.
  // More info: https://containers.dev/implementors/json_reference/#port-attributes
  "portsAttributes": {
    "9000": {
      "label": "Hello Remote World",
      "onAutoForward": "notify"
    }
  },

  // Use 'postCreateCommand' to run commands after the container is created.
  "postCreateCommand": "pip3 install -r requirements.txt",
  "features": {
    "ghcr.io/devcontainers-contrib/features/coverage-py:2": {
      "version": "latest"
    }
  },

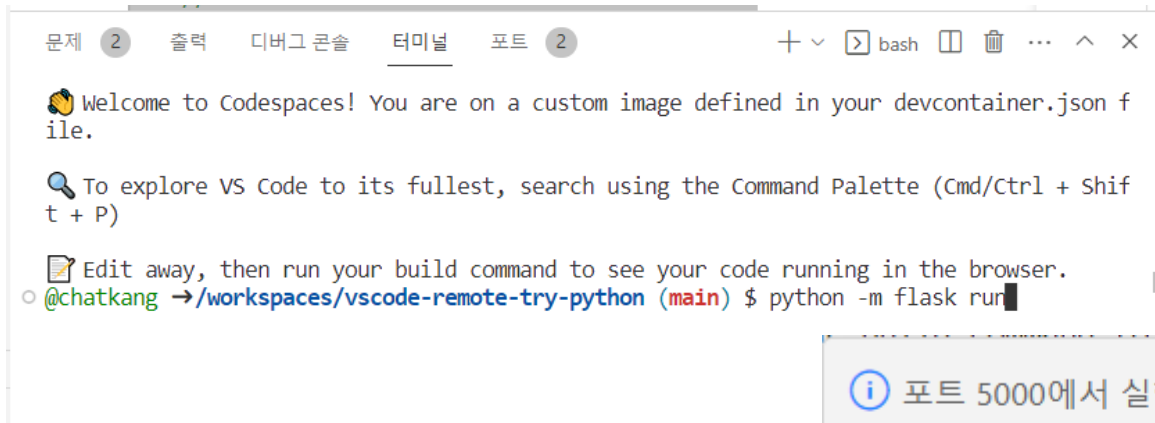
  // Uncomment to connect as root instead. More info: https://aka.ms/dev-containers-non-root.
  // "remoteUser": "root"
}
```



## Setting up your codespace



- `$ python -m flask run`



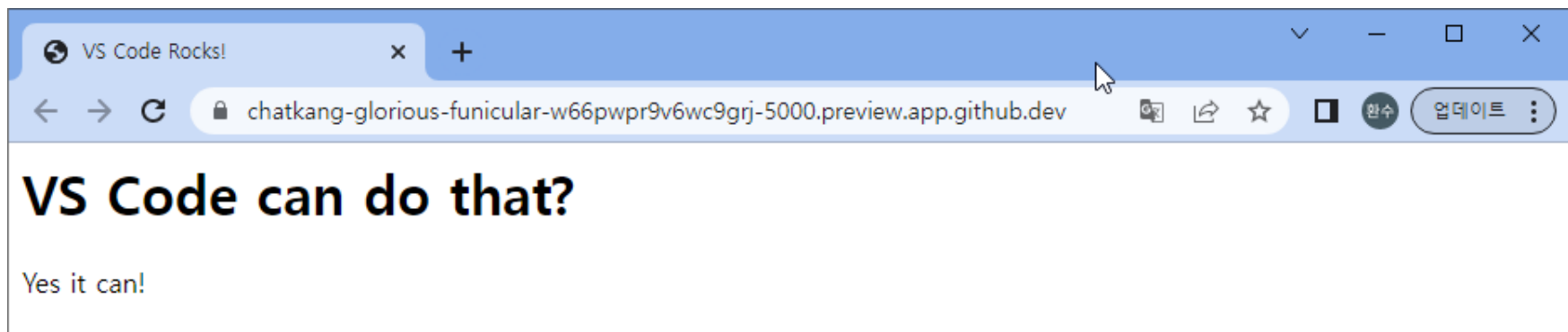
```
문제 2 출력 디버그 콘솔 터미널 포트 2 + bash [icon] [icon] ... ^ x
👋 Welcome to Codespaces! You are on a custom image defined in your devcontainer.json f
ile.

🔍 To explore VS Code to its fullest, search using the Command Palette (Cmd/Ctrl + Shif
t + P)

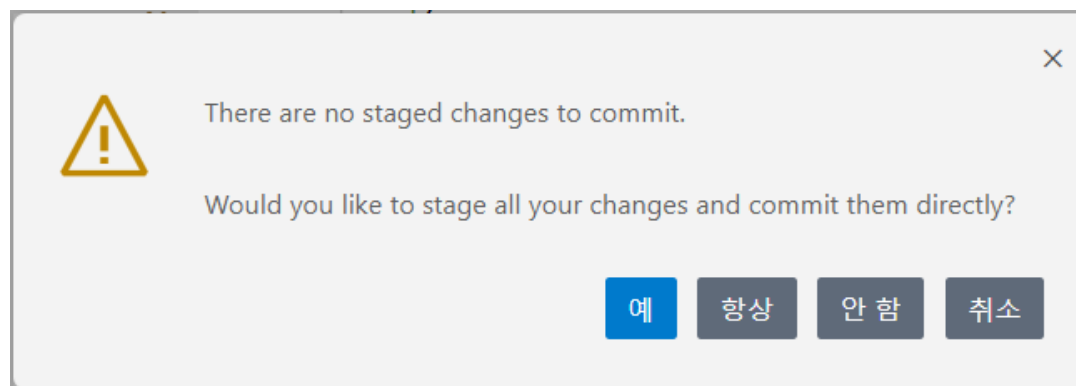
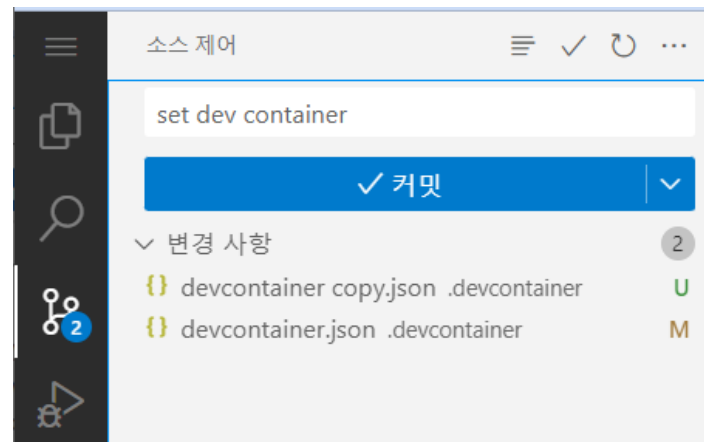
📄 Edit away, then run your build command to see your code running in the browser.
@chatkang →/workspaces/vscode-remote-try-python (main) $ python -m flask run
```

📘 포트 5000에서 실행되는 애플리케이션을 사용할 수 있습니다. ⚙️ ×  
전달된 모든 포트 보기

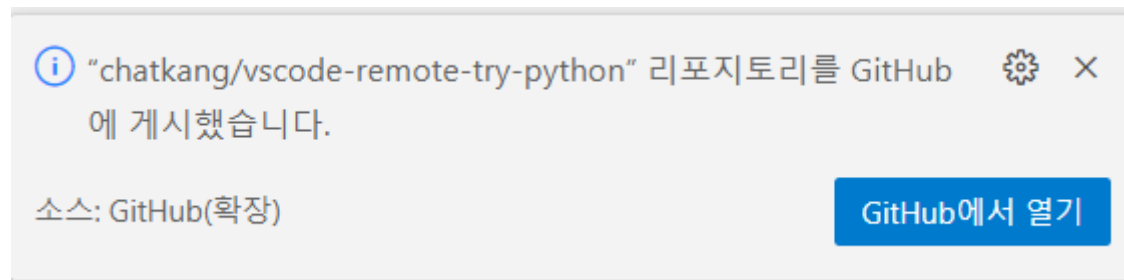
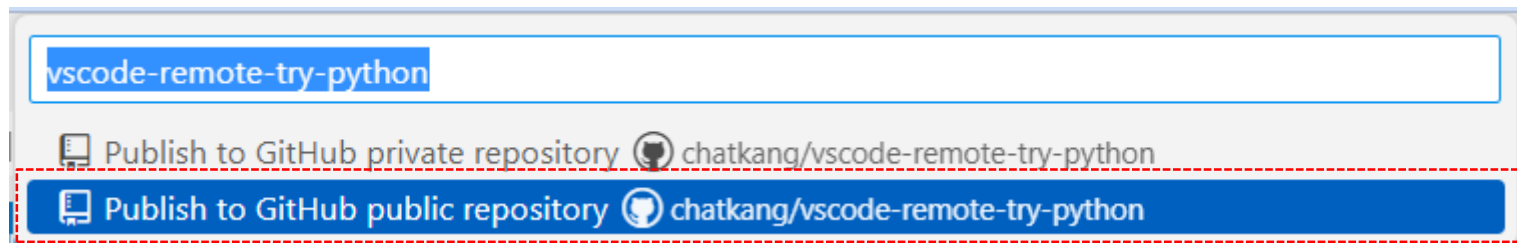
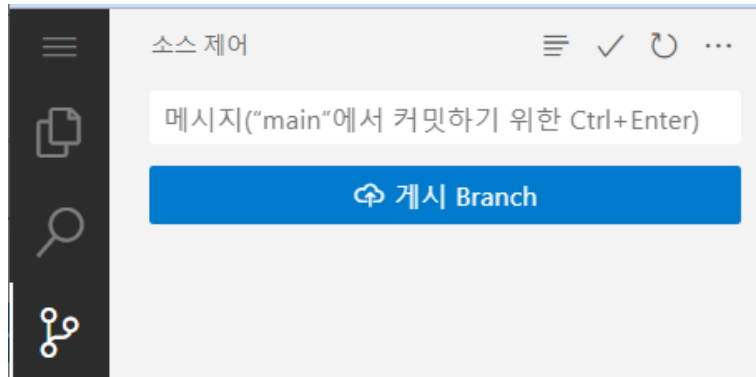
브라우저에서 열기



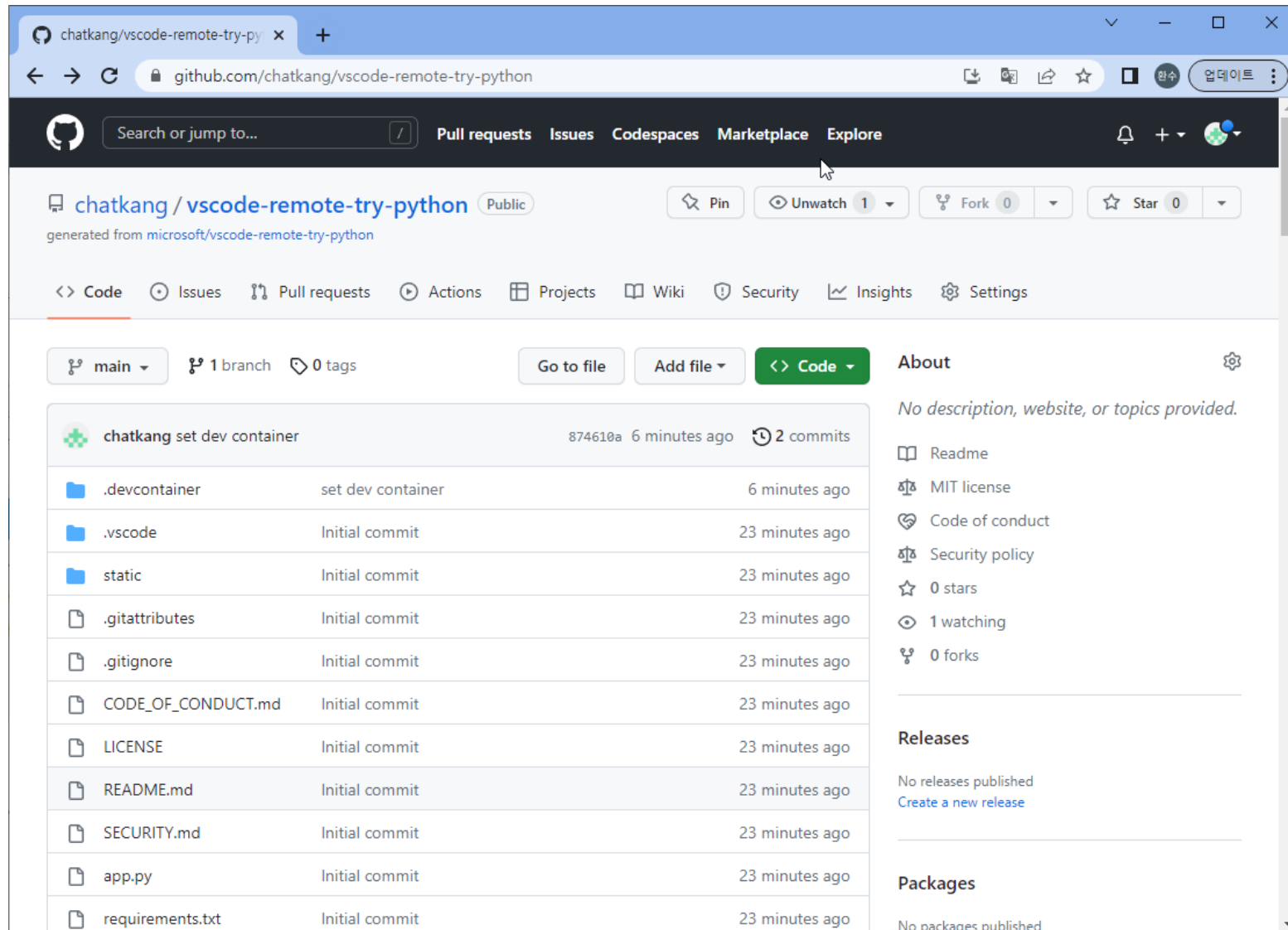
- 코드스페이스에 저장



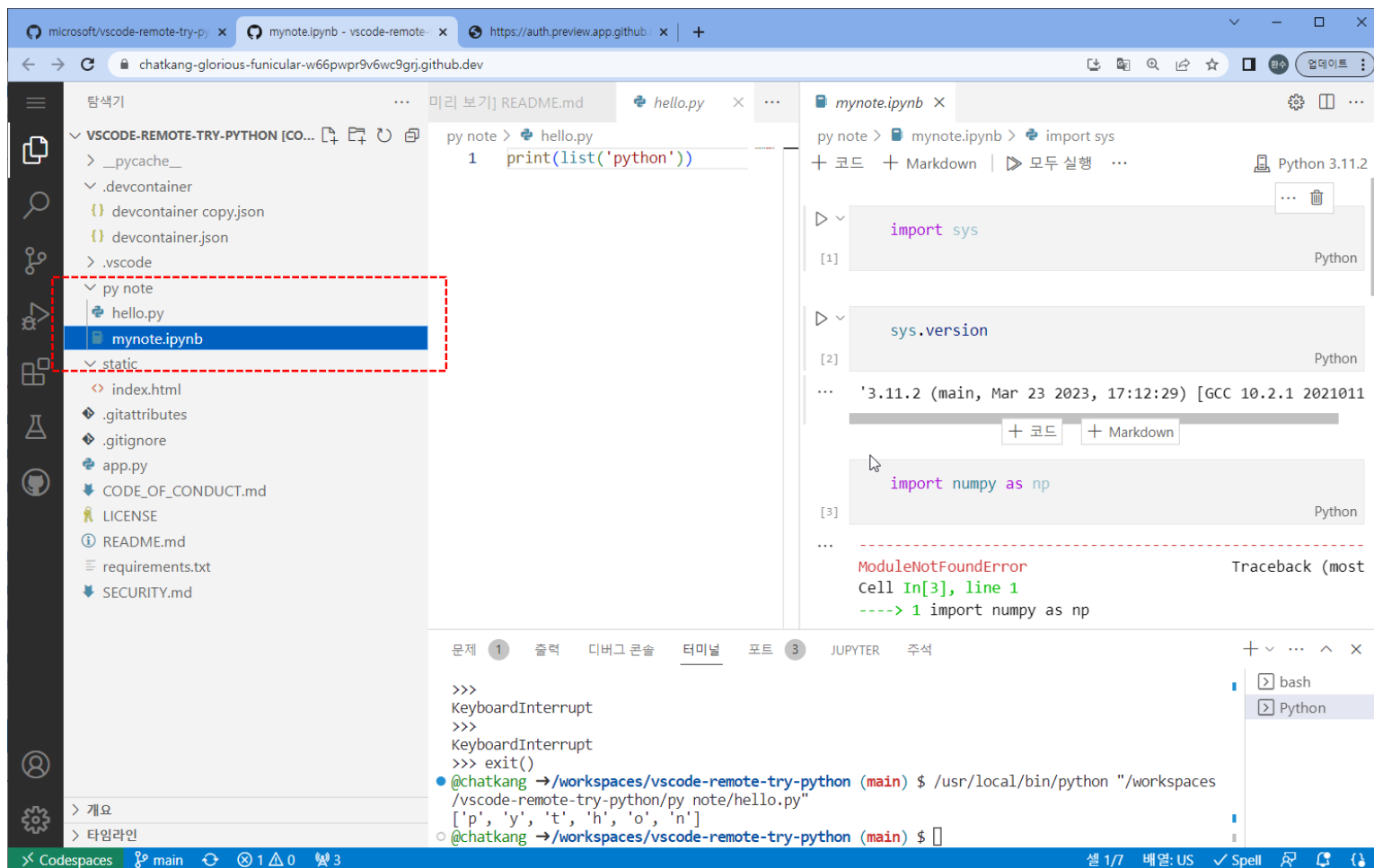
- Publish: 게시
  - 깃허브에 저장소를 생성하고 복사(push)하는 작업



- chatkang/vscode-remote-try-python



- 폴더 py note 생성해 하부에 코딩
  - \*.py
  - \*.ipynb
    - 자동으로 ipykernel 설치

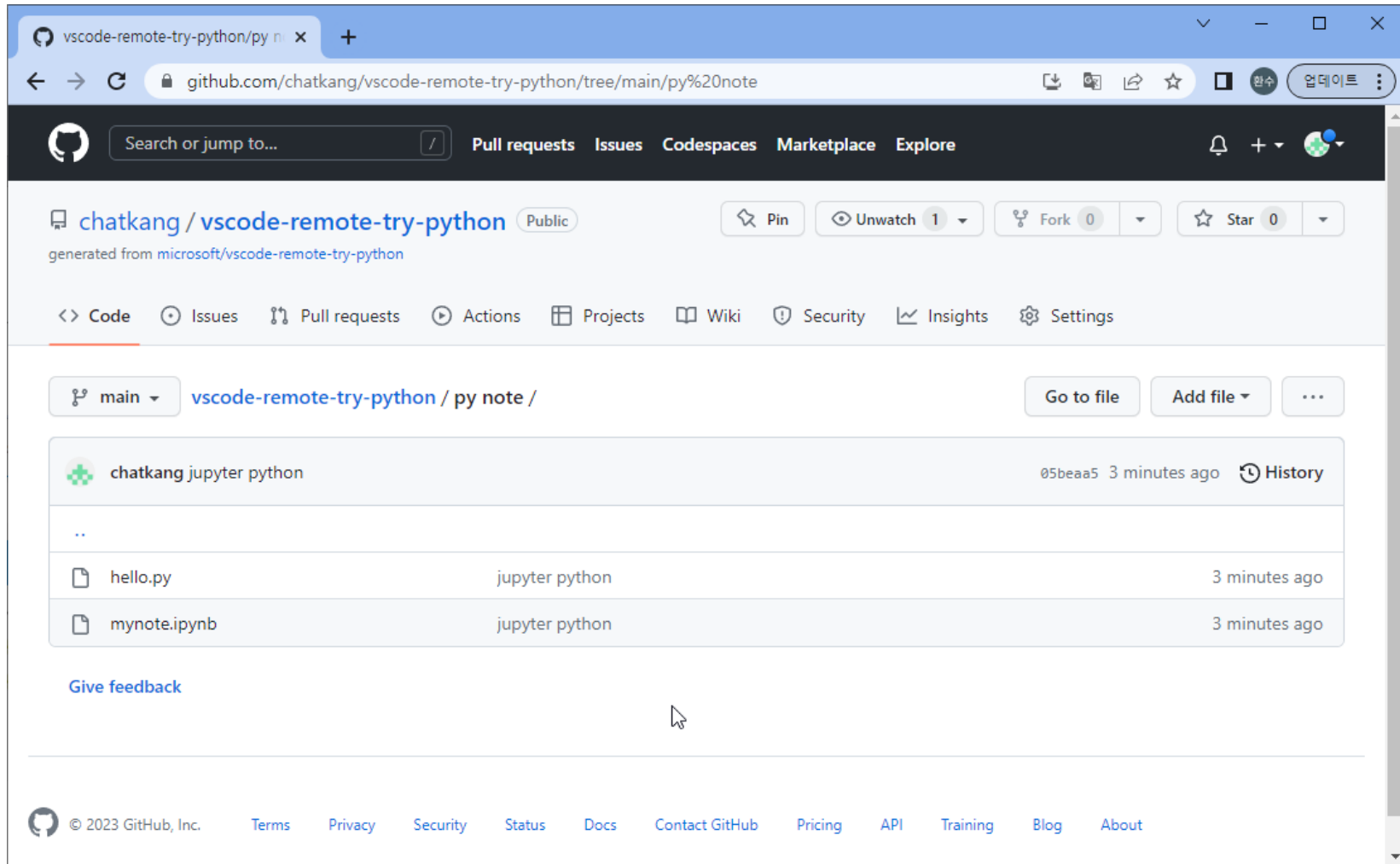




# 다시 커밋 후 push

오픈소스 소프트웨어를 위한 깃과 깃허브 Python language

- 자신의 깃허브에 동기화



AI Experts  
Who Lead  
The Future

## 03

### 소스 파일 깃허브 저장

