

# MS 클라우드 개발환경 깃허브 코드스페이스 활용



강환수 교수

AI Experts  
Who Lead  
The Future

# 01

## 깃허브 코드스페이스 개요와 연결 실행

- **Github Codespaces**

- 깃허브 저장소 기반 클라우드 개발환경
  - 간단히 코드스페이스라고 명명
- 여러 개발자가 협업하며 개발할 수 있는 클라우드 기반 통합개발환경(cloud integrated development environments)
  - 브라우저에서 개발환경 실행
- 장점
  - 서버에서 웹개발이나 응용프로그램과 머신러닝, 딥러닝 개발 등 다양한 분야의 개발환경을 지원하거나 설정 가능
  - 자신의 PC 등의 지역 시스템에 소프트웨어를 설치하거나 유지 관리할 필요가 없음
    - Linux에서 컨테이너 개발환경 구성 및 실행
    - 개인적으로 별도의 개발환경을 설치할 필요가 없이 클라우드에서 제공하는 개발환경을 사용
- 2022년 11월부터 무료 지원 확대

- 자신의 깃허브에서 서버에 연결, 수정 내용은 그대로 깃허브저장소 저장 가능
  - 자신의 깃허브 저장소에서 마우스 클릭으로
    - 손쉽게 브라우저에서 원하는 코드를 작성해 컴파일과 디버깅, 실행 등의 작업을 수행
  - 다양한 방법으로 코드 공간에 연결 가능
    - 브라우저, Visual Studio Code, JetBrains Gateway 응용 프로그램 또는 GitHub CLI를 사용
  - 클라우드에 저장된 데이터와 코드는 버전관리 깃을 활용
    - 소스와 구성 파일을 저장소에 커밋하여 GitHub 코드 공간에 대한 프로젝트를 저장 가능

# 코드를 저장할 저장소 생성

오픈소스 소프트웨어를 위한 깃과 깃허브 Python language

- **py-space**
  - README.md 체크해 생성

Create a New Repository

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 \*** chatkang / **Repository name \*** py-space ✓

Great repository names are short and memorable. Need inspiration? How about [bug-free-rotary-phone?](#)

**Description (optional)**  
기초 파이썬 코딩 활용

☒ **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](#).

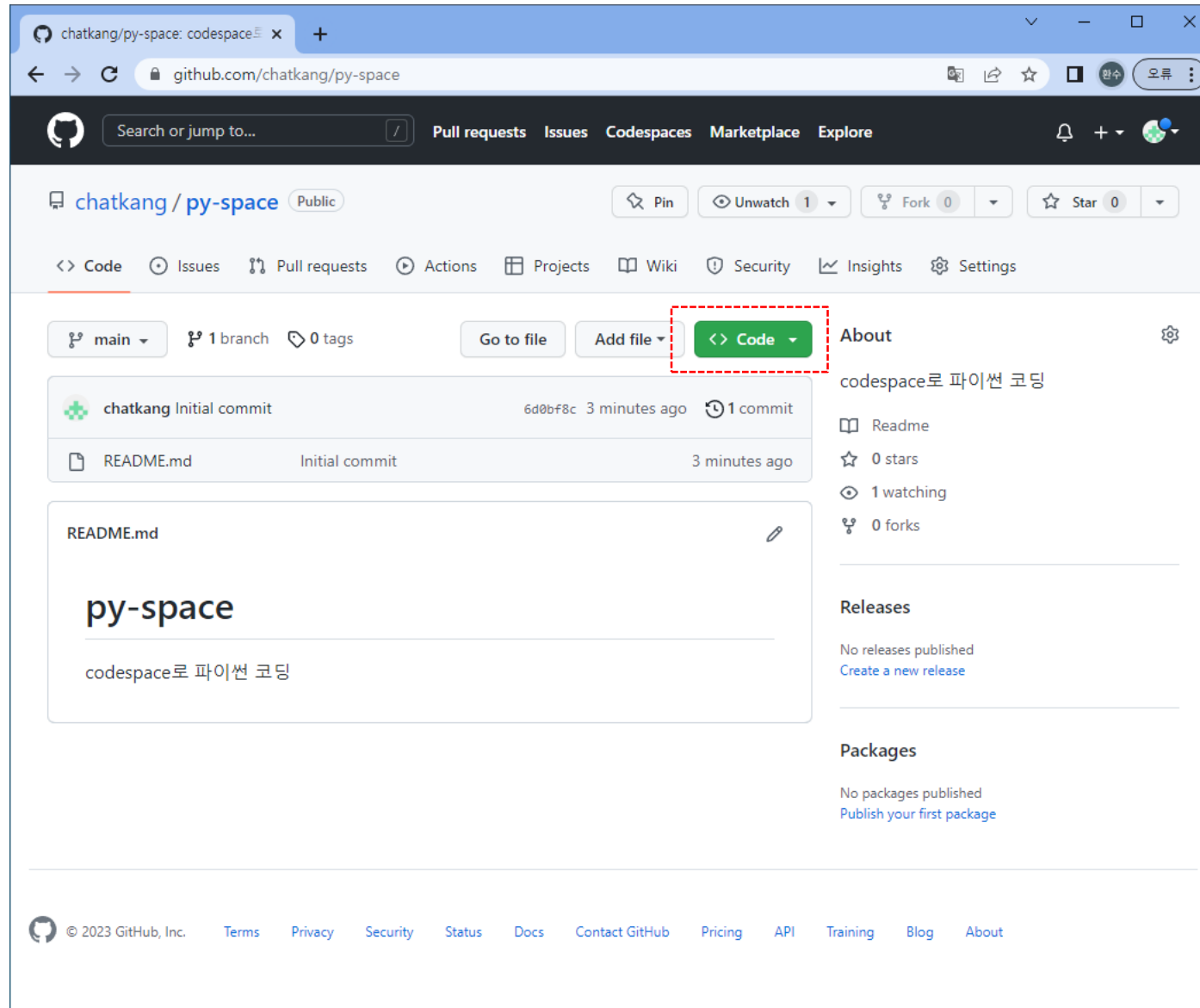
ⓘ You are creating a public repository in your personal account.

**Create repository**

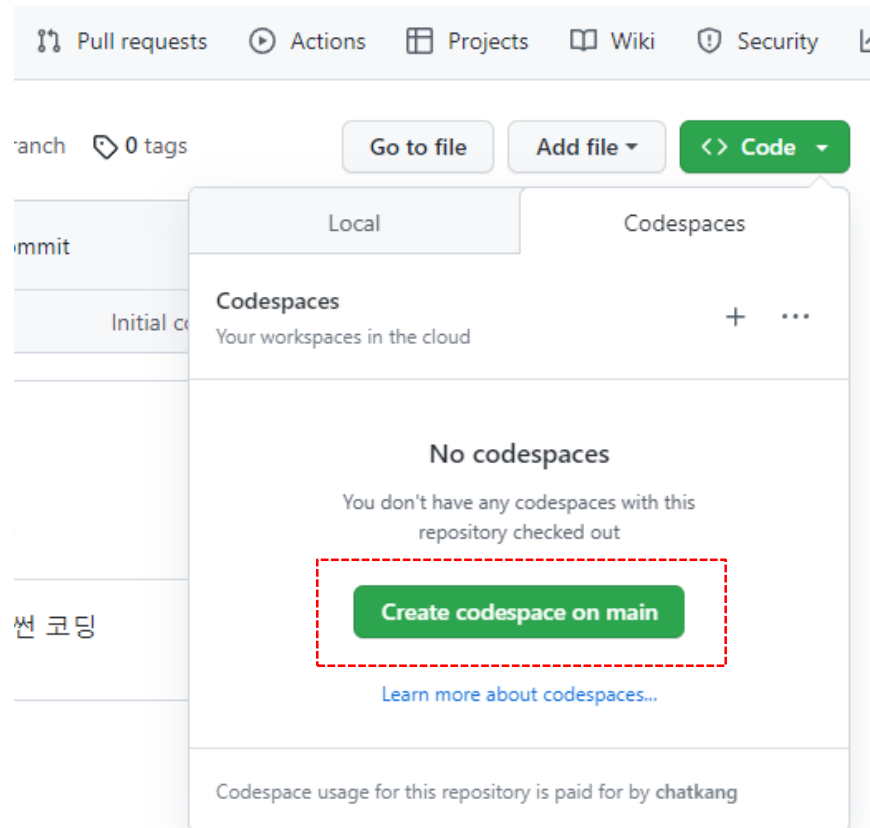
# 코드를 저장할 저장소 py-space

오픈소스 소프트웨어를 위한 깃과 깃허브 Python language

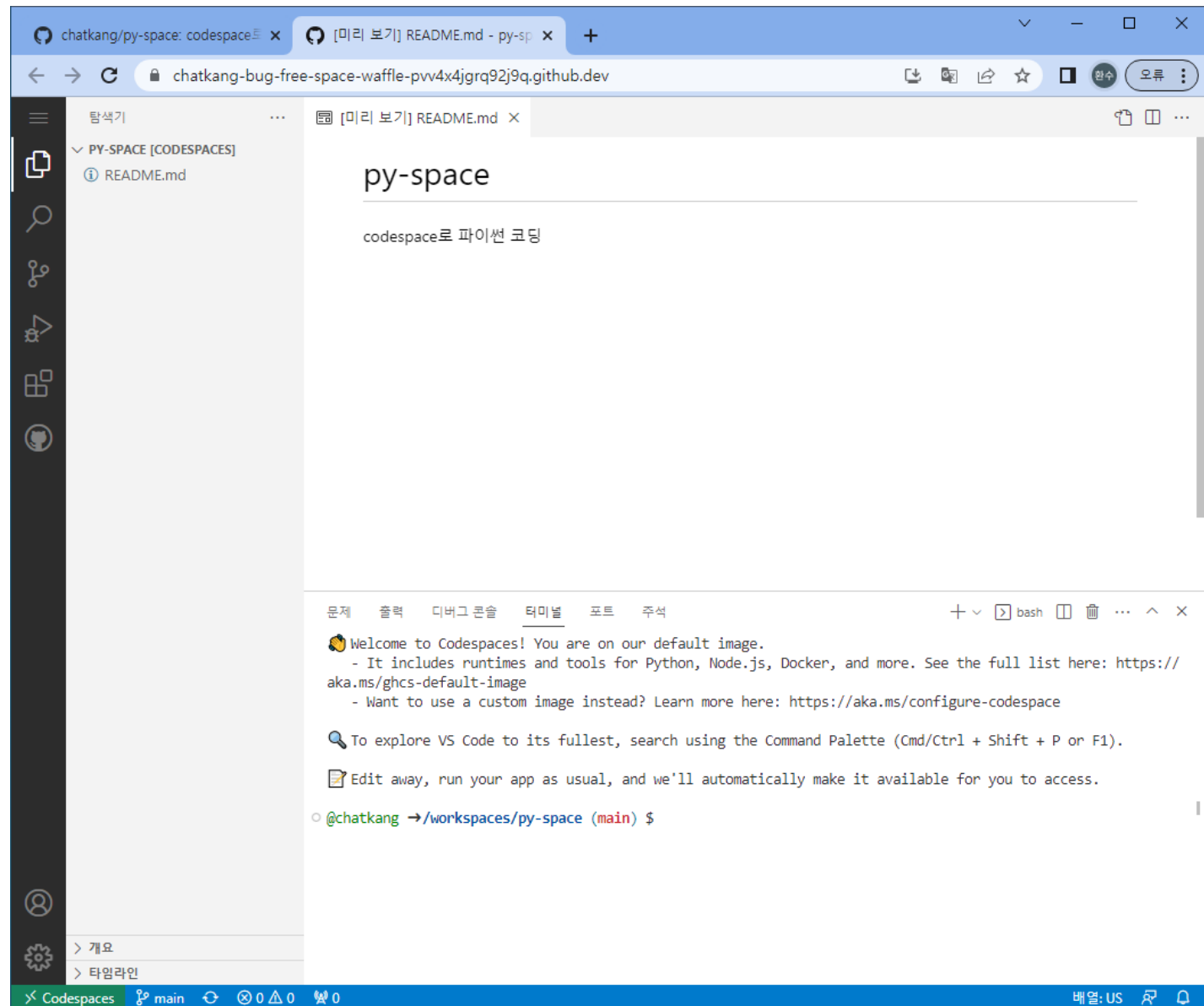
- py-space



- 버튼 Code 하부
  - Create codesapce on main
    - 브랜치 메인에 코드스페이스 생성
      - 기본 설정으로 코드스페이스 생성해 연결
        - » 간단한 설정으로 파이썬 활용 가능



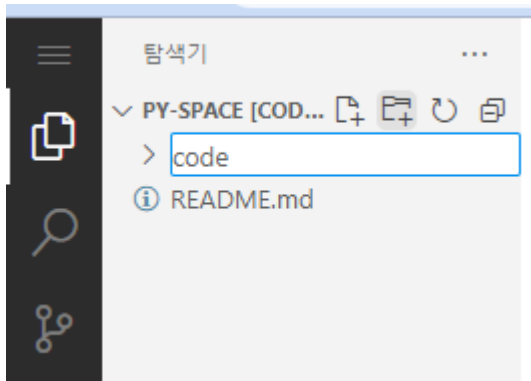
- 브라우저에 생성





- 파일 생성

- 폴더 code 생성 후 하부에 파일 hello.py 생성



- 실행 준비
  - 확장 python 설치

문지 출력 디버그 콘솔 터미널 포트 주석

+ v bash [ ] [ ] ... ^ x

👋 Welcome to Codespaces! You are on our default image.

- It includes runtimes and tools for Python, Node.js, Docker, and more. See the full list here: <https://aka.ms/ghcs-default-image>
- Want to use a custom image instead? Learn more here: <https://aka.ms/configure-codespace>

🔍 To explore VS Code to its fullest, search using the Command Palette (Cmd/Ctrl + Shift + P or F1).

📝 Edit away, run your app as usual, and we'll automatically make it available for you to access.

o @chatkang → /workspaces/py-space (main) \$

Python에 권장되는 확장을 설치하시겠습니까?

설치 권장 사항 표시

확장: 마켓플... [미리 보기] README.md hello.py U 확장: Python x

@id:ms-python.python

Python 81M ★ 4

IntelliSense (Pylance) by Microsoft 설치 중

Python v2023.4.1

Microsoft microsoft.com 81,069,345 ★★★★★ (534)

IntelliSense (Pylance), Linting, Debugging (multi-threaded, remote), Ju...

설치 중

세부 정보 기능 기여도 변경 로그 확장 팩

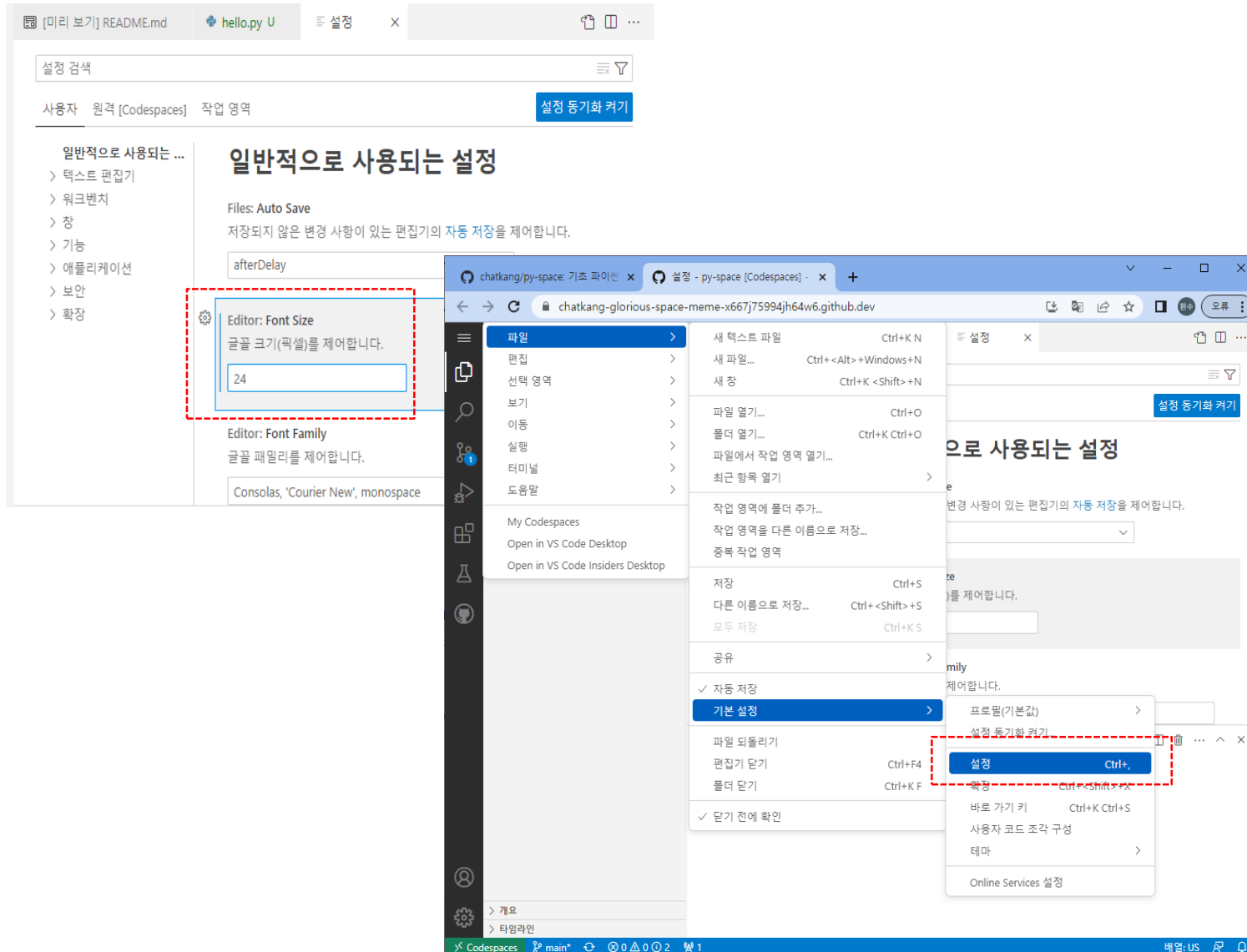
Python extension for Visual Studio Code

A Visual Studio Code extension with rich support for the Python language (for all actively supported versions of the language: > including features such as IntelliSense (Pylance), linting, debugging, code navigation, code formatting, refactoring, variable explorer, test...

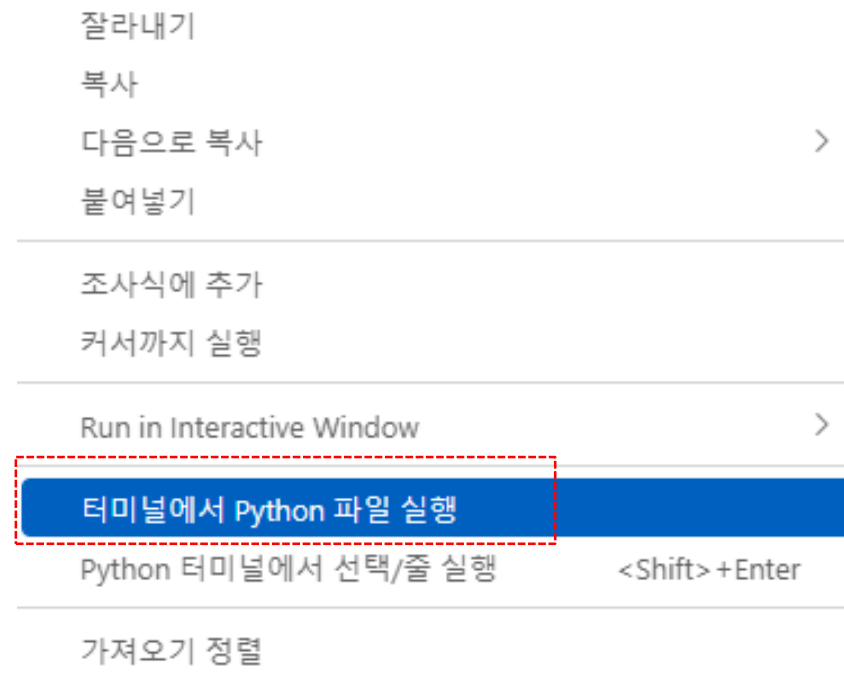
범주

- Programming Languages
- Linters
- Debuggers
- Formatters
- Data Science
- Machine Learning

- 단축키 **ctrl + ,**



- 실행
  - 소스에서 팝업 메뉴

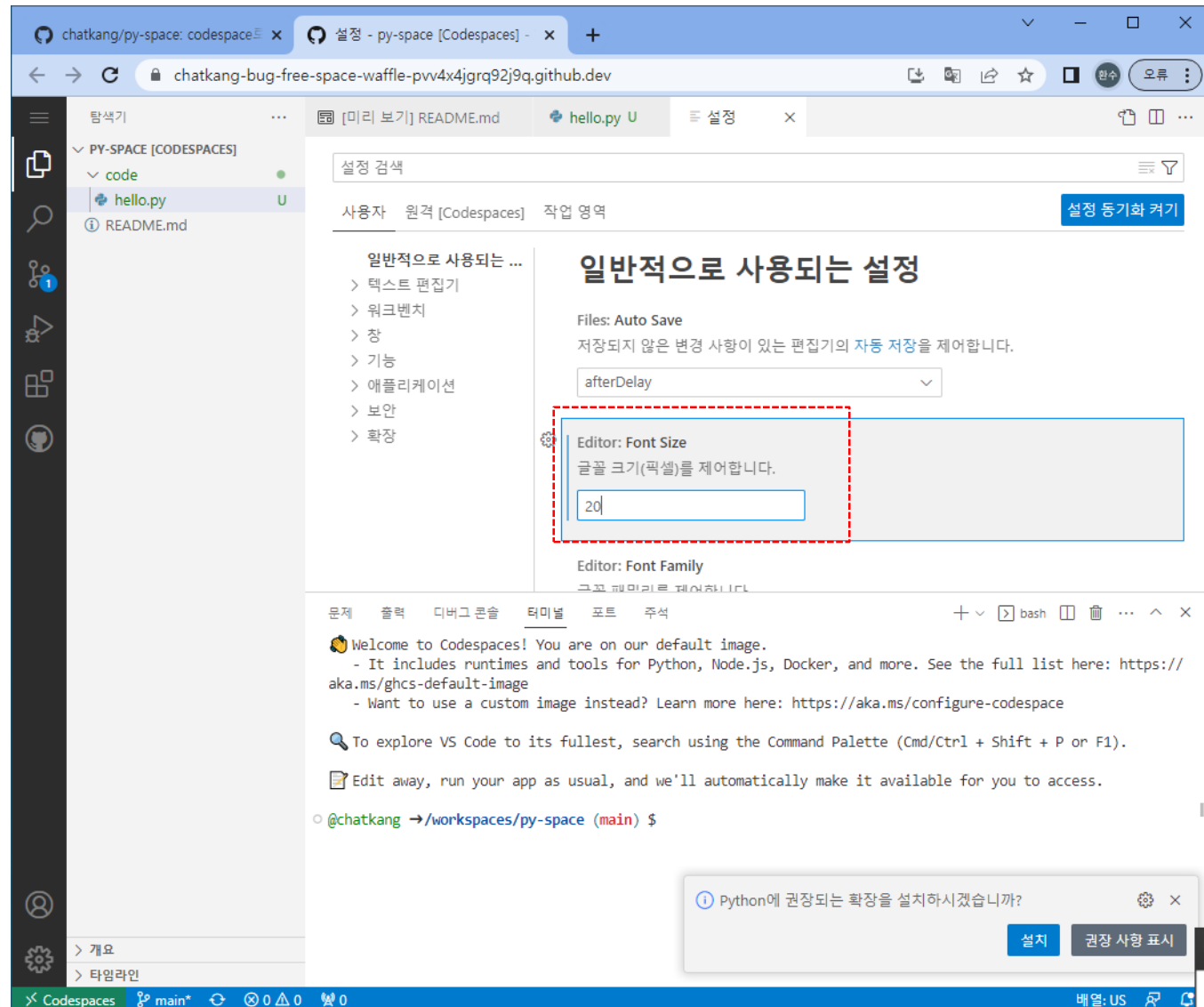


- 실행 결과

```

문제 2 출력 디버그 콘솔 터미널 포트 1 주석
/home/codespace/.python/current/bin/python3 /workspaces/py-space/code/hello.py
• @chatkang →/workspaces/py-space (main) $ /home/codespace/.python/current/bin/python3 /workspaces/py-space/code/hello.py
Hello, Codespaces!
['p', 'y', 't', 'h', 'o', 'n']
○ @chatkang →/workspaces/py-space (main) $
    
```

- 단축키 **ctrl + ,**

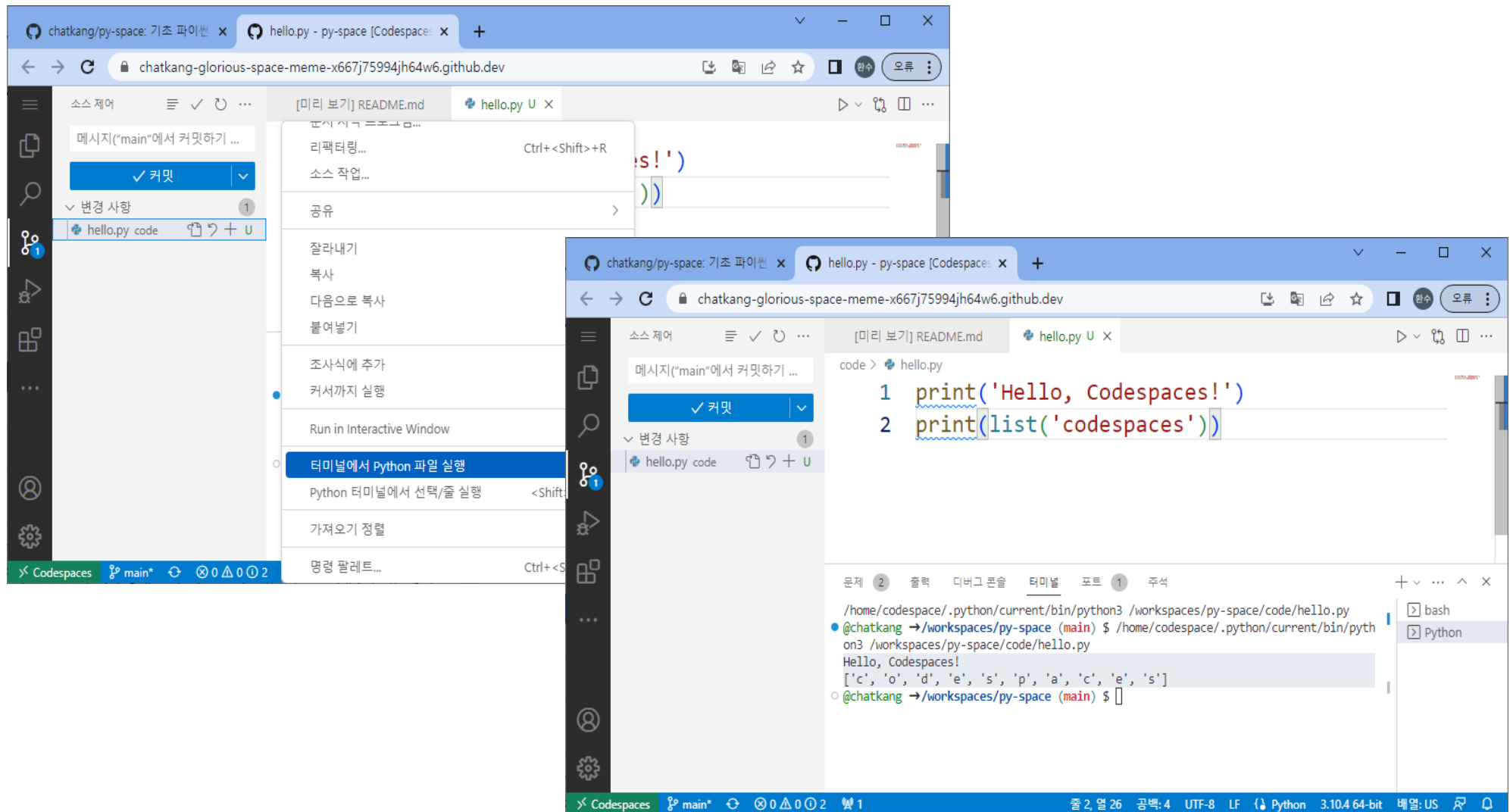


AI Experts  
Who Lead  
The Future

## 02

### 다양한 파이썬 실행

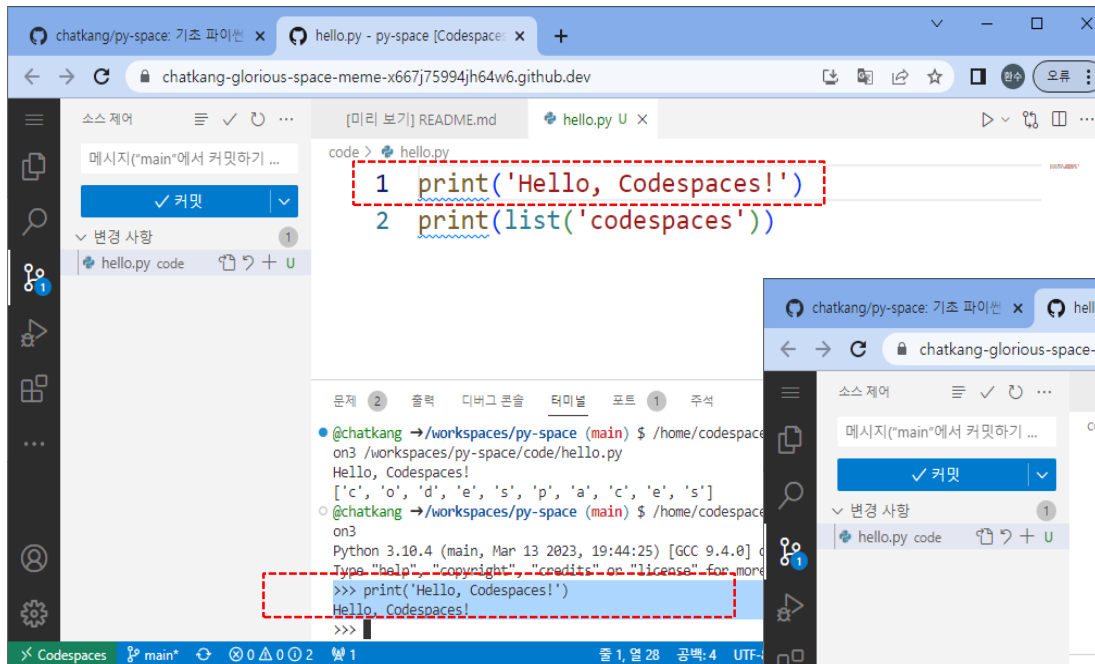
- 파이썬 파일 전체를 실행
  - 메뉴 [터미널에서 Python 파일 실행]



# 파이썬 터미널에서 선택/줄 실행

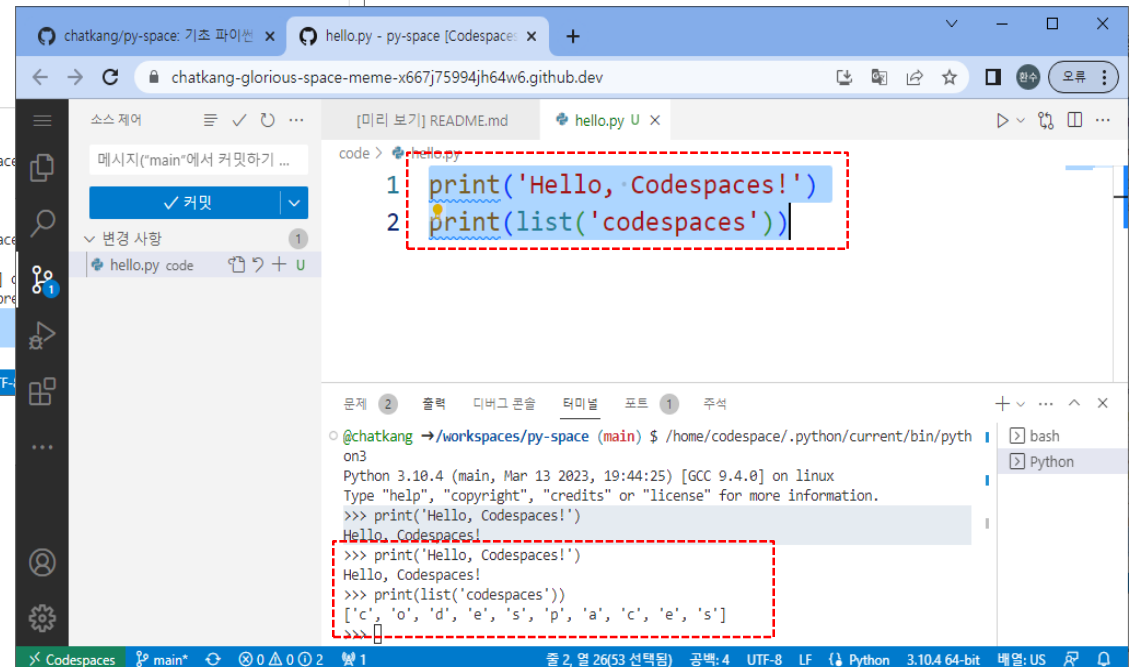
오픈소스 소프트웨어를 위한 깃과 깃허브 Python language

- 선택한 줄/블록이 실행
  - 실행을 원하는 줄에 커서를 놓고 **shift + enter**
  - Python 쉘로 실행



```
1 print('Hello, Codespaces!')
2 print(list('codespaces'))
```

```
@chatkang →/workspaces/py-space (main) $ /home/codespace/on3 /workspaces/py-space/code/hello.py
Hello, Codespaces!
['c', 'o', 'd', 'e', 's', 'p', 'a', 'c', 'e', 's']
@chatkang →/workspaces/py-space (main) $ /home/codespace/on3 Python 3.10.4 (main, Mar 13 2023, 19:44:25) [GCC 9.4.0]
Type "help", "copyright", "credits" or "license()" for more
>>> print('Hello, Codespaces!')
Hello, Codespaces!
>>>
```



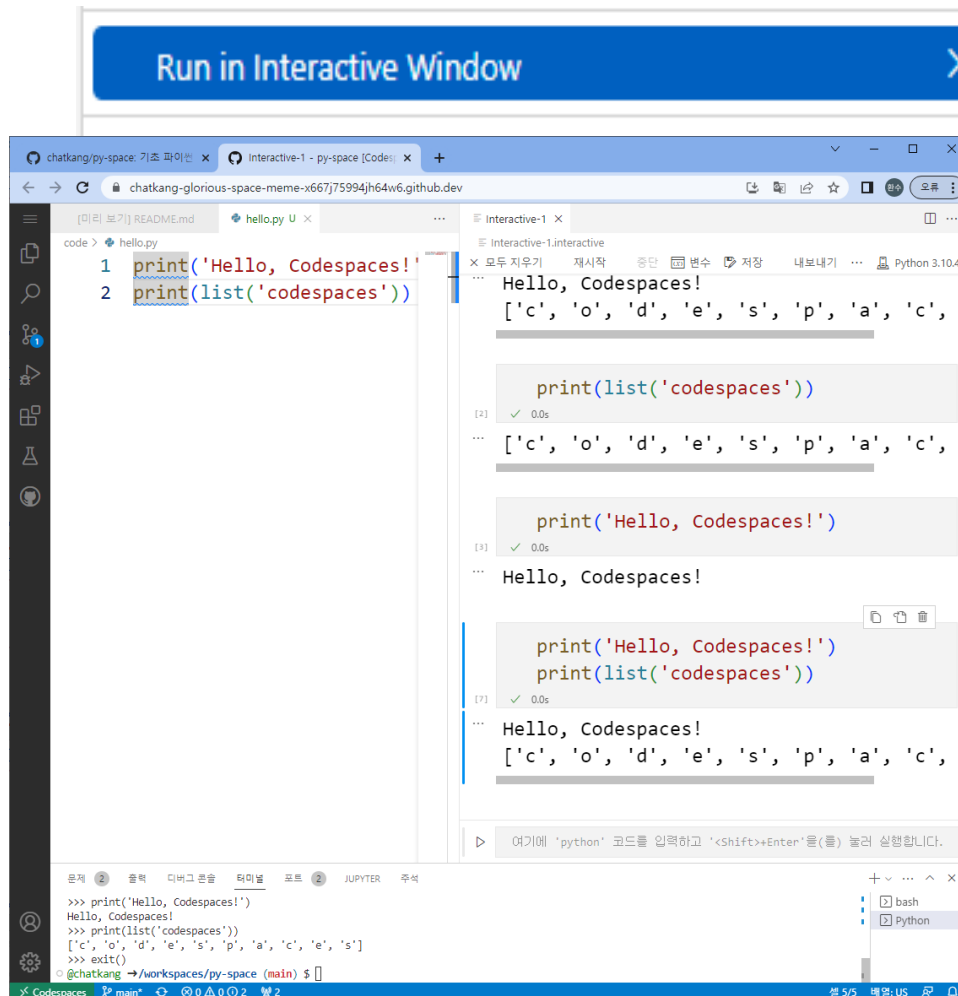
```
1 print('Hello, Codespaces!')
2 print(list('codespaces'))
```

```
@chatkang →/workspaces/py-space (main) $ /home/codespace/on3 /workspaces/py-space/code/hello.py
Hello, Codespaces!
['c', 'o', 'd', 'e', 's', 'p', 'a', 'c', 'e', 's']
@chatkang →/workspaces/py-space (main) $ /home/codespace/on3 Python 3.10.4 (main, Mar 13 2023, 19:44:25) [GCC 9.4.0]
Type "help", "copyright", "credits" or "license()" for more
>>> print('Hello, Codespaces!')
Hello, Codespaces!
>>> print(list('codespaces'))
['c', 'o', 'd', 'e', 's', 'p', 'a', 'c', 'e', 's']
>>>
```



## • ipython 실행

- 대화형 창에서의 실행은 주피터 커널인 ipython을 사용
- 메뉴 [Run in interactive Window] 하부에 여러 방법 제공



대화형 창에서 현재 파일 실행

대화형 창의 줄에서 실행

대화형 창에서 선택/행 실행

<Shift>+Enter

대화형 창에서 라인까지 실행

- Shift + Enter 로 실행
  - 대화형 창의 마지막 줄 아랫 부분에는 코드를 입력한 후

```
print('Hello, Codespaces!')
```

[3] ✓ 0.0s

```
... Hello, Codespaces!
```

```
print('Hello, Codespaces!')
```

```
print(list('codespaces'))
```

[7] ✓ 0.0s

```
... Hello, Codespaces!
```

```
['c', 'o', 'd', 'e', 's', 'p', 'a', 'c',
```

```
▷ dict(((1, 'Jan'), (2, 'Feb')))
```

```
dict(((1, 'Jan'), (2, 'Feb')))
```

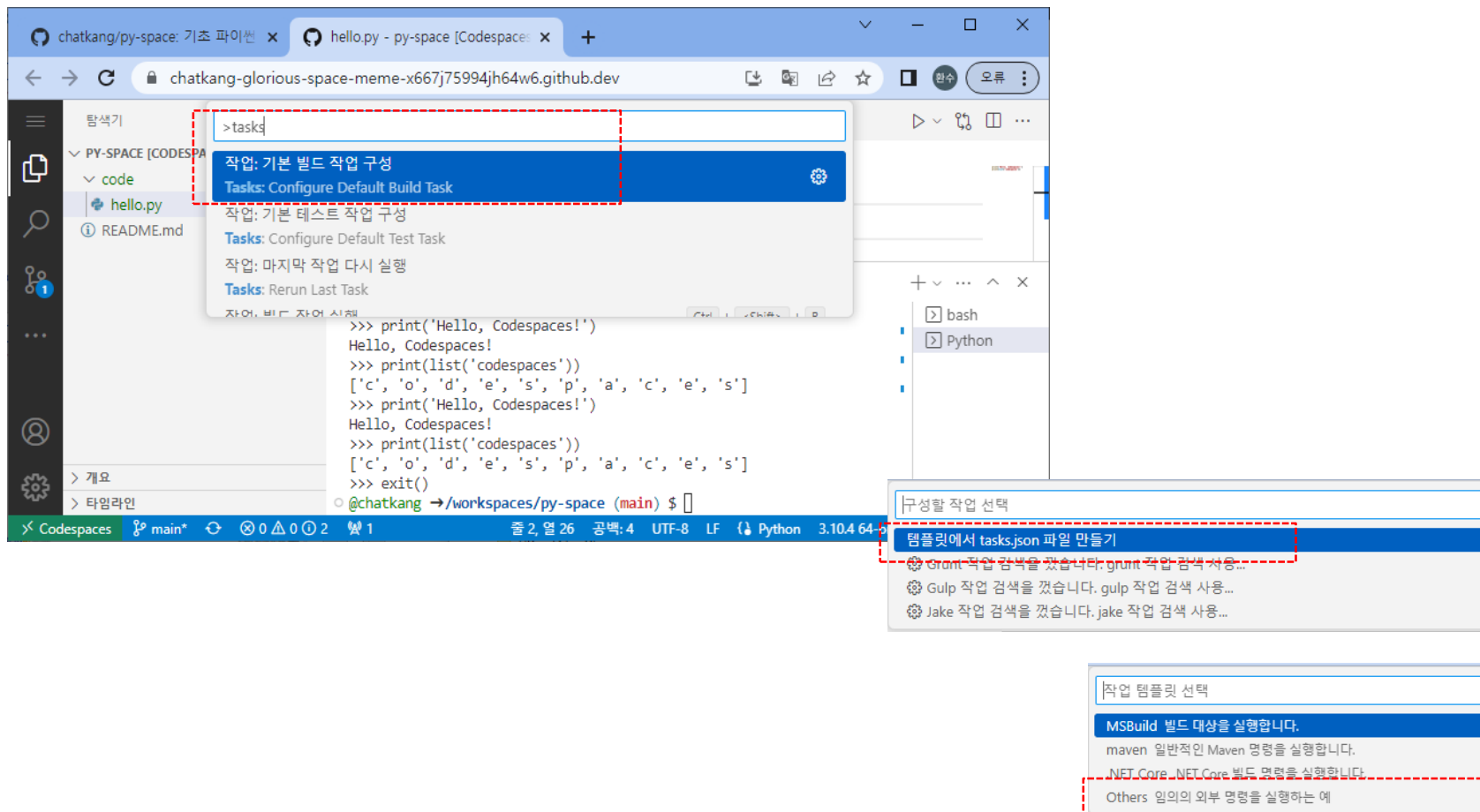
[11] ✓ 0.1s

```
... {1: 'Jan', 2: 'Feb'}
```

# 실행할 빌드 작업으로 파이썬 실행 설정

오픈소스 소프트웨어를 위한 깃과 깃허브 Python language

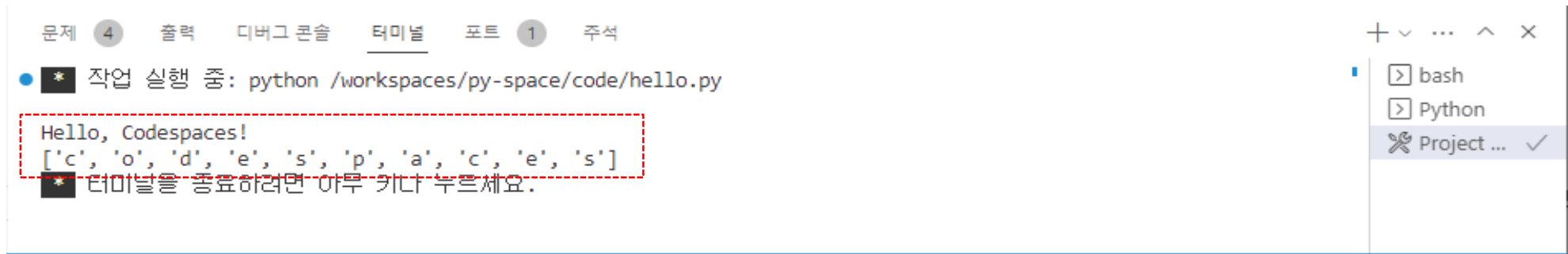
- Vs code에서 실행할 빌드를 설정하기 위해 먼저 ctrl + shift + P로 팔레트를 실행
  - Tasks 입력
    - [템플릿에서 tasks.json 파일 만들기]
    - 다음으로 마지막 항목 [Others 임의의 외부 명령을 실행하는 예]



- 파일 `.vscode/tasks.json` 수정 후 저장(ctrl + S)

```
{  
  // See https://go.microsoft.com/fwlink/?LinkId=733558  
  // for the documentation about the tasks.json format  
  "version": "2.0.0",  
  "tasks": [  
    {  
      "label": "Project Label",  
      "type": "shell",  
      "command": "python",  
      "args": [  
        "${file}"  
      ],  
      "presentation": {  
        "reveal": "always",  
        "panel": "new"  
      },  
      "options": {  
        "env": {  
          "PYTHONIOENCODING": "UTF-8"  
        }  
      },  
      "group": {  
        "kind": "build",  
        "isDefault": true  
      }  
    }  
  ]  
}
```

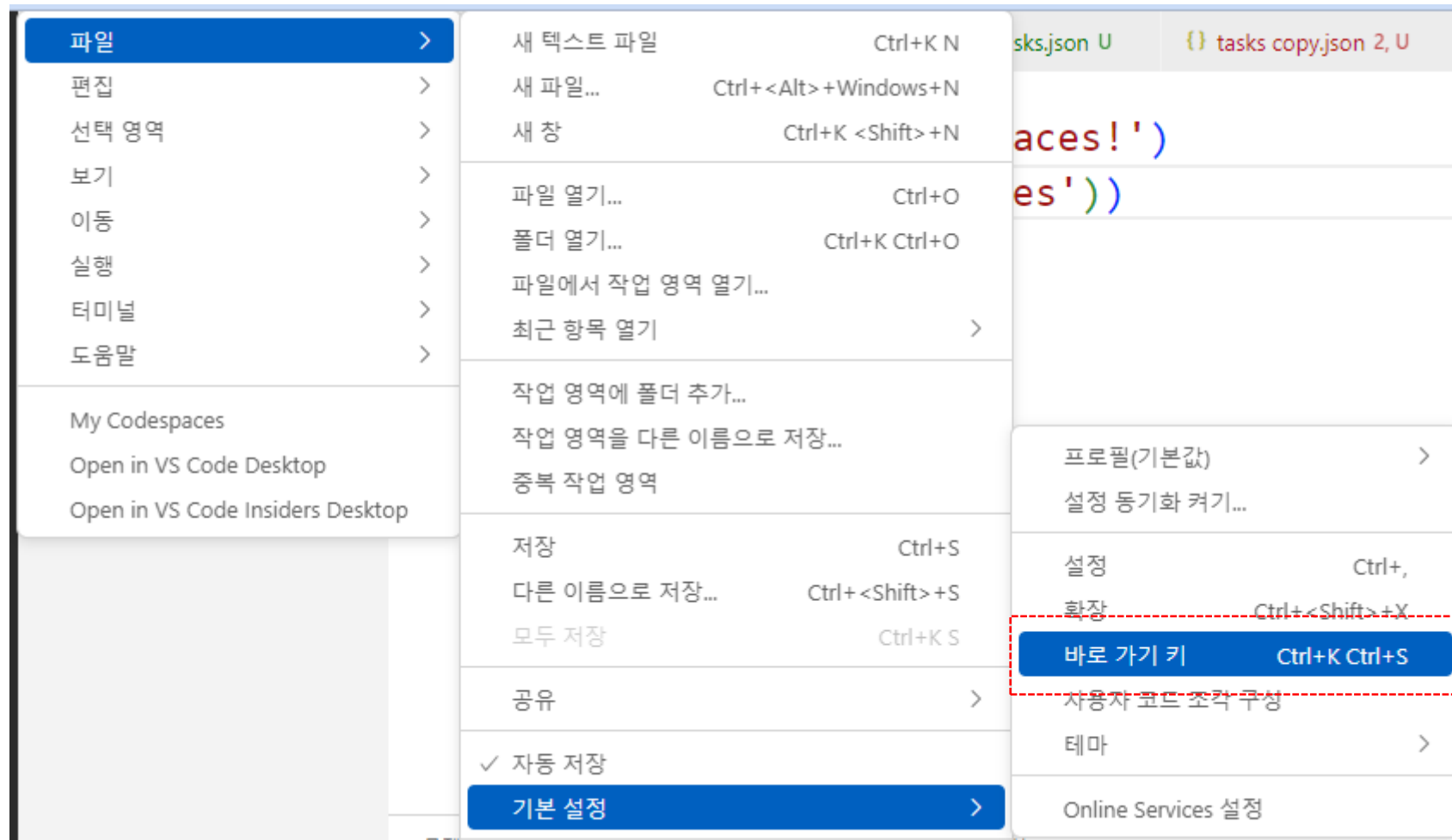
- 파이썬 소스 파일에서 설정한 태스크를 Ctrl + Shift + B로 실행



```
문제 4 출력 디버그 콘솔 터미널 포트 1 주석
* 작업 실행 중: python /workspaces/py-space/code/hello.py
Hello, Codespaces!
['c', 'o', 'd', 'e', 's', 'p', 'a', 'c', 'e', 's']
* 터미널을 종료하려면 아무 키나 누르세요.
```

- 단축 키를 설정

- 메뉴 [파일 | 기본 설정 | 바로 가기 키], 단축 키는 **ctrl + K ctrl + S**



# 현재 설정된 바로 가기 키가 표시

오픈소스 소프트웨어를 위한 깃과 깃허브 Python language

[미리 보기] README.md hello.py U 바로 가기 키 X tasks.json U tasks copy.json 2, U			
키 바인딩에서 검색하려면 입력			
명령	키 바인딩	언제	소스
<Tab> 키로 포커스 이동 설정/해제	Ctrl + M	-	시스템
가리키기 또는 포커스 표시	Ctrl + K Ctrl + I	editorTextFocus	시스템
가져오기 구성	<Shift> + <Alt> + O	editorTextFocus && !editorReadOnly && suppo...	시스템
개발자: 창 다시 로드	Ctrl + R	isDevelopment	시스템
검색 편집기: 검색 편집기 입력 포커스	Escape	inSearchEditor	시스템
검색 편집기: 다시 검색	Ctrl + <Shift> + R	inSearchEditor	시스템
검색 편집기: 단어 단위로 토글	<Alt> + W	inSearchEditor && searchInputBoxFocus	시스템
검색 편집기: 대/소문자 구분 토글	<Alt> + C	inSearchEditor && searchInputBoxFocus	시스템
검색 편집기: 모든 일치 항목 선택	Ctrl + <Shift> + L	inSearchEditor	시스템
검색 편집기: 정규식 사용 토글	<Alt> + R	inSearchEditor && searchInputBoxFocus	시스템
검색 편집기: 컨텍스트 줄 늘이기	<Alt> + =	inSearchEditor	시스템
검색 편집기: 컨텍스트 줄 줄이기	<Alt> + -	inSearchEditor	시스템
검색 편집기: 컨텍스트 줄 토글	<Alt> + L	inSearchEditor	시스템
검색 편집기: 파일 결과 삭제	Ctrl + <Shift> + Backspace	inSearchEditor	시스템
검색 편집기: 편집기에서 결과 열기	<Alt> + Enter	hasSearchResult && searchViewletFocus	시스템
검색: 검색 취소	Escape	listFocus && searchViewletVisible && !input...	시스템
검색: 다음 검색 결과에 포커스	F4	hasSearchResult    inSearchEditor	시스템
검색: 이전 검색 결과에 포커스	<Shift> + F4	hasSearchResult    inSearchEditor	시스템
검색: 파일에서 바꾸기	Ctrl + <Shift> + H	-	시스템
검색: 파일에서 찾기	Ctrl + <Shift> + F	-	시스템
구현으로 이동	Ctrl + F12	editorHasImplementationProvider && editorTe...	시스템

- 검색 입력 창에 '터미널에서 ...' 입력
  - 단축키를 설정하고 싶은 마지막 항목에서 더블 클릭

터미널에서

명령	키 바인딩	언제	소스
Python: Python 터미널에서 선택/줄 실행	<Shift> + Enter	editorTextFocus && !findInputFocussed && !j...	Python
터미널: 활성 터미널에서 선택			
터미널: 활성 터미널에서 활			
Python: 터미널에서 Python			

터미널에서

명령	키 바인딩	언제	소스
Python: Python 터미널에서 선택/줄 실행	<Shift> + Enter	editorTextFocus && !findInputFocussed && !j...	Python
터미널: 활성 터미널에서 선택한 텍스트 실행		-	시스템
터미널: 활성 터미널에서 활성 파일 실행		-	시스템
+ Python: 터미널에서 Python 파일 실행		-	Python

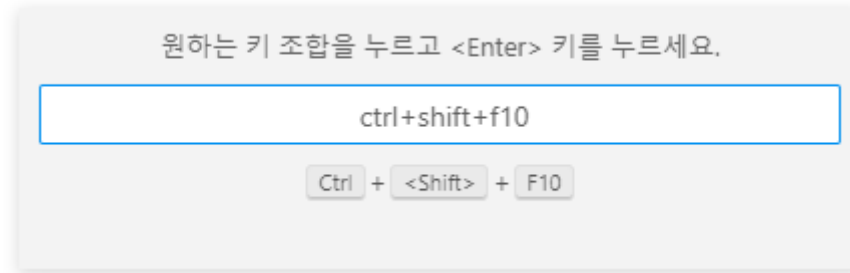
원하는 키 조합을 누르고 <Enter> 키를 누르세요.

ctrl+shift

Ctrl + <Shift> +



- 직접 원하는 조합 키 Ctrl + Shift + F10를 누른 후 Enter키



명령	키 바인딩	언제	소스
Python: <b>터미널에서</b> Python 파일 실행	Ctrl + <Shift> + F10	-	사용자
Python: Python <b>터미널에서</b> 선택/줄 실행	<Shift> + Enter	editorTextFocus && !findInputFocussed && !j...	Python
터미널: 활성 <b>터미널에서</b> 선택한 텍스트 실행		-	시스템
터미널: 활성 <b>터미널에서</b> 활성 파일 실행		-	시스템

# 설정된 단축 키 실행

오픈소스 소프트웨어를 위한 깃과 깃허브 Python language

- ctrl + shift + F10 으로 실행

The screenshot displays a web-based IDE interface for a GitHub Codespace. The browser address bar shows the URL `chatkang-glorious-space-meme-x667j75994jh64w6.github.dev`. The left sidebar contains a file explorer with the following structure:

- PY-SPACE [CODESPACES]
  - .vscode
    - tasks copy.json
    - tasks.json
  - code
    - hello.py
    - README.md

The main editor area shows the `hello.py` file with the following code:

```
1 print('Hello, Codespaces!')
2 print(list('codespaces'))
```

Below the editor, the 'TERMINAL' tab is active, showing the execution output:

```
Hello, Codespaces!
>>> print(list('codespaces'))
['c', 'o', 'd', 'e', 's', 'p', 'a', 'c', 'e', 's']
>>> exit()
@chatkang → /workspaces/py-space (main) $ /home/codespace/.python/current/bin/python3 /workspaces/py-space/code/hello.py
Hello, Codespaces!
['c', 'o', 'd', 'e', 's', 'p', 'a', 'c', 'e', 's']
@chatkang → /workspaces/py-space (main) $
```

The bottom status bar indicates the environment is 'Python 3.10.4 64-bit' and the file encoding is 'UTF-8'.

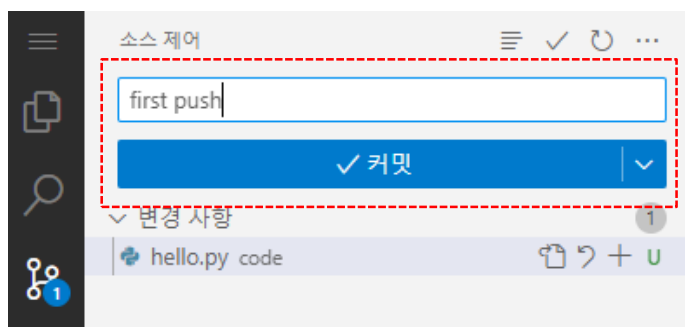
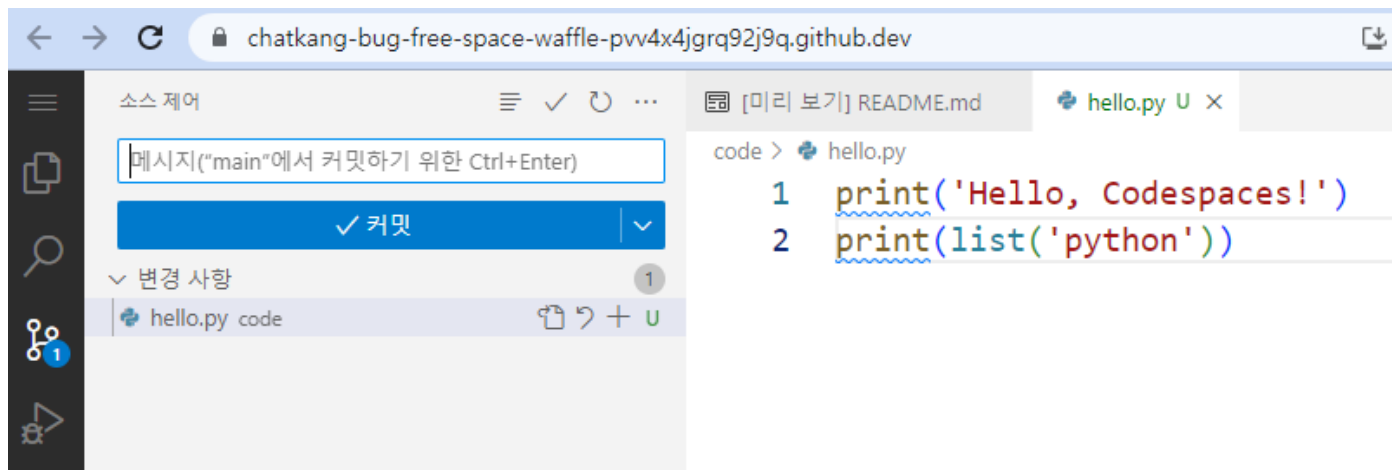
AI Experts  
Who Lead  
The Future

## 03

### 소스 파일 깃허브 저장

- 커밋

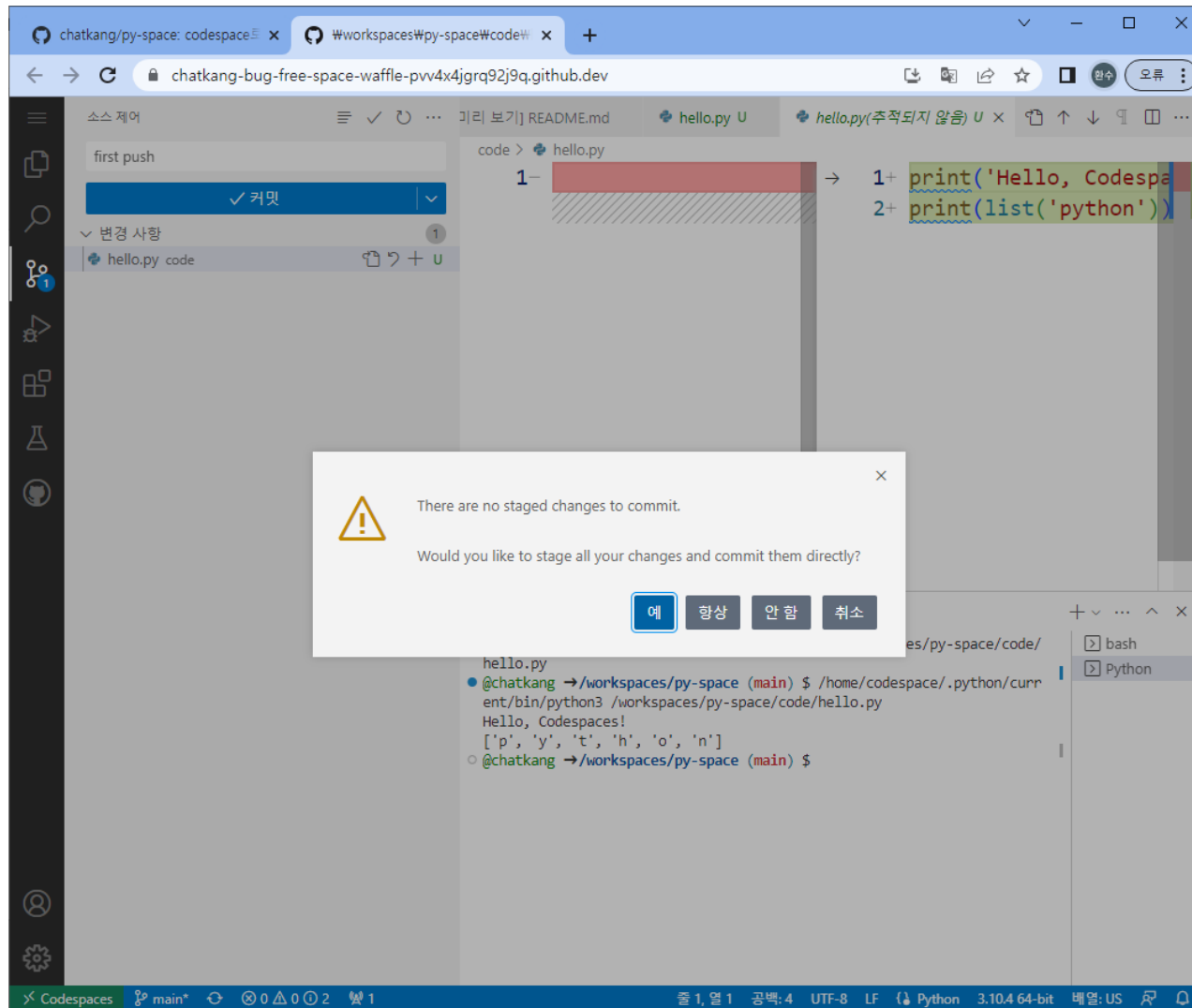
- 버전 제어 활동바 선택
- 커밋 메시지 입력 후
  - 커밋 버튼 클릭



# Add와 커밋 함께

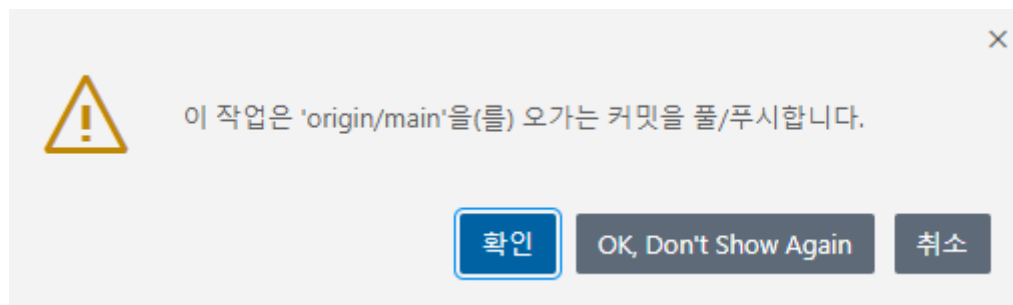
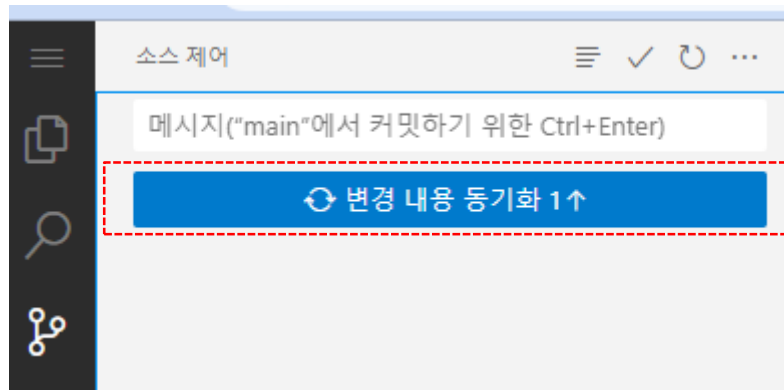
오픈소스 소프트웨어를 위한 깃과 깃허브 Python language

- 예 선택

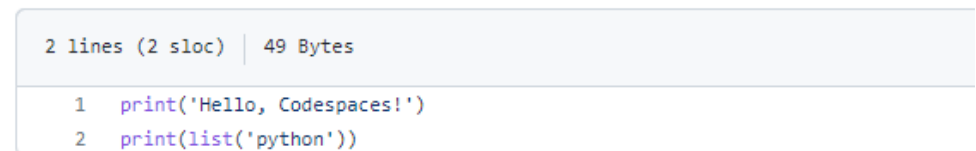
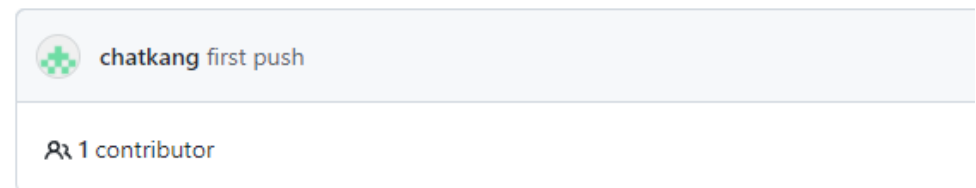
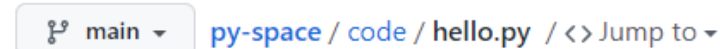
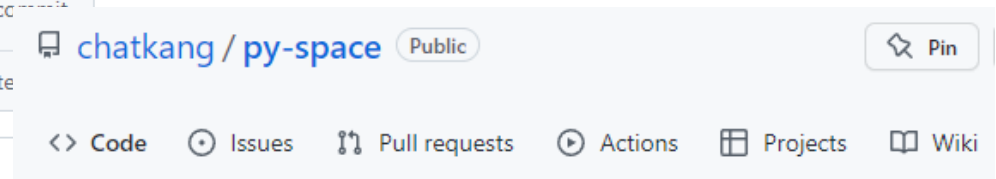
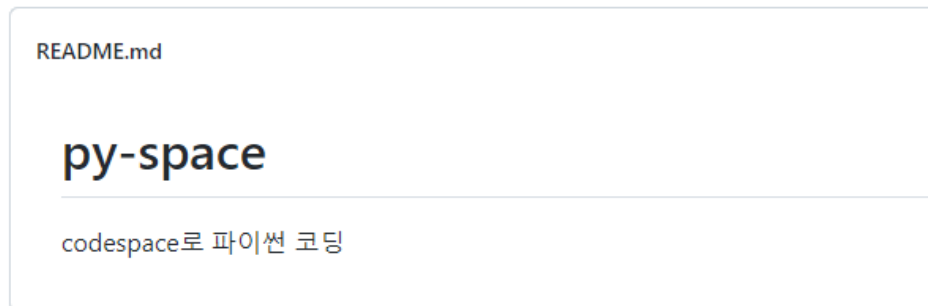
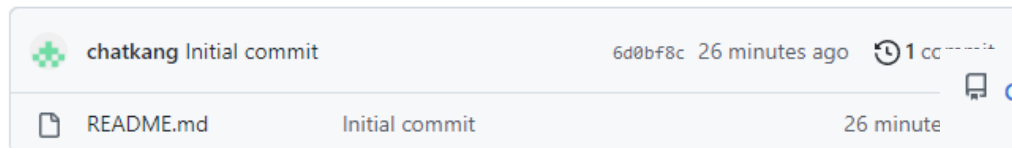
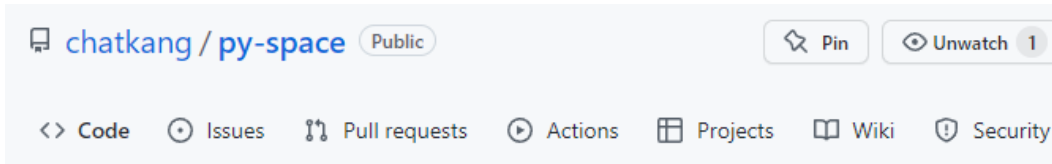


- **Push**

- 변경 내용 동기화 클릭



- 깃허브에서 refresh



[Give feedback](#)

AI Experts  
Who Lead  
The Future

# 04

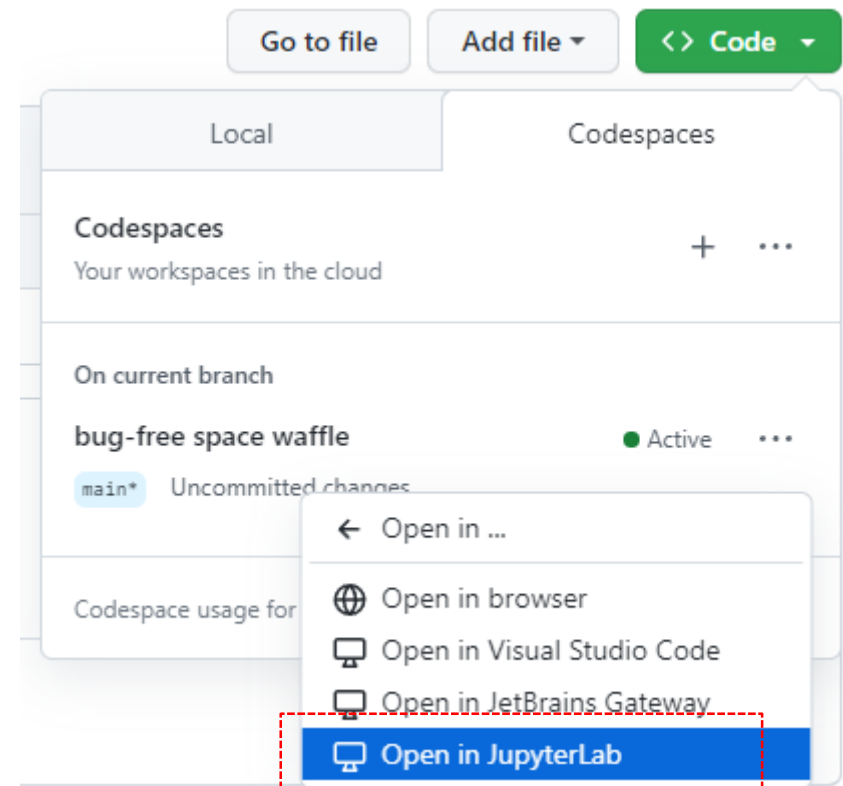
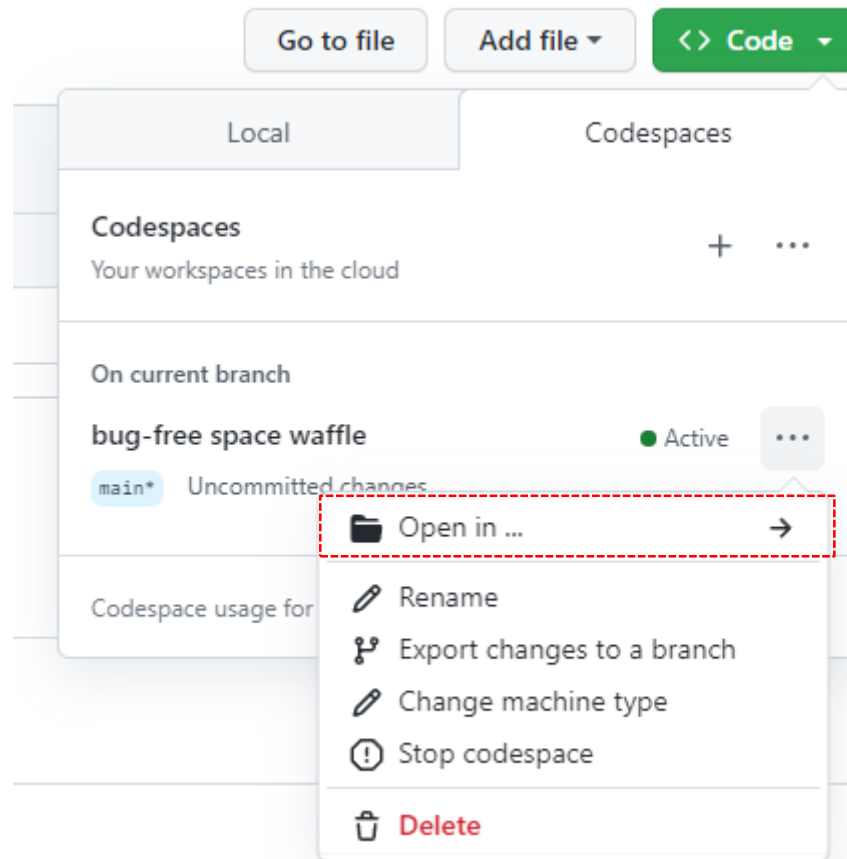
## 주피터 랩 활용



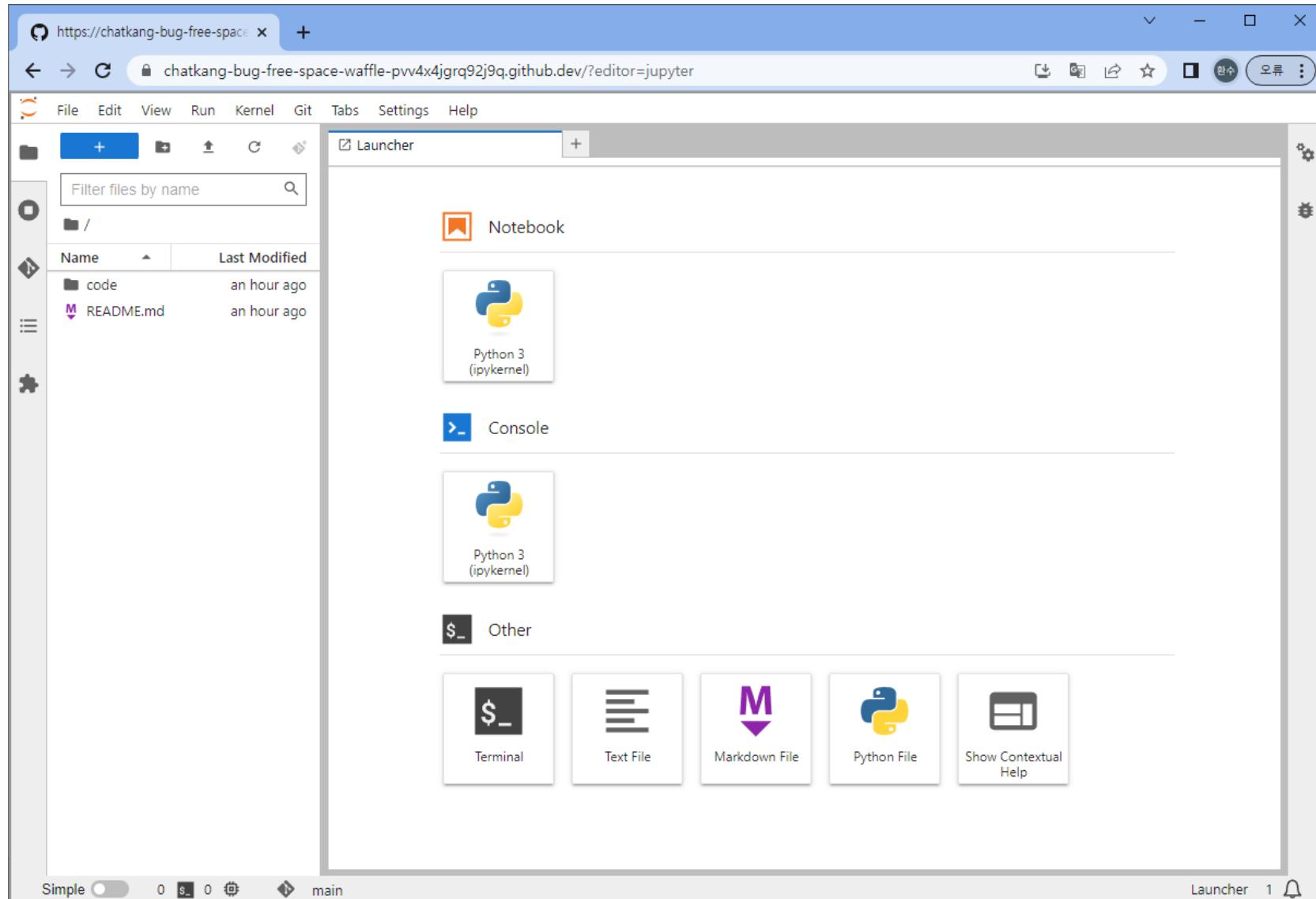
- 버튼 Code

- Open in ... -> 클릭

- Open in JupyterLab 선택

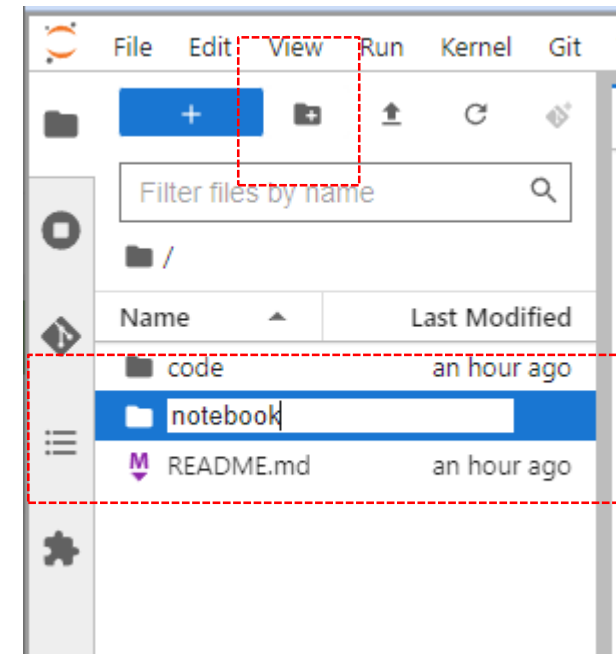
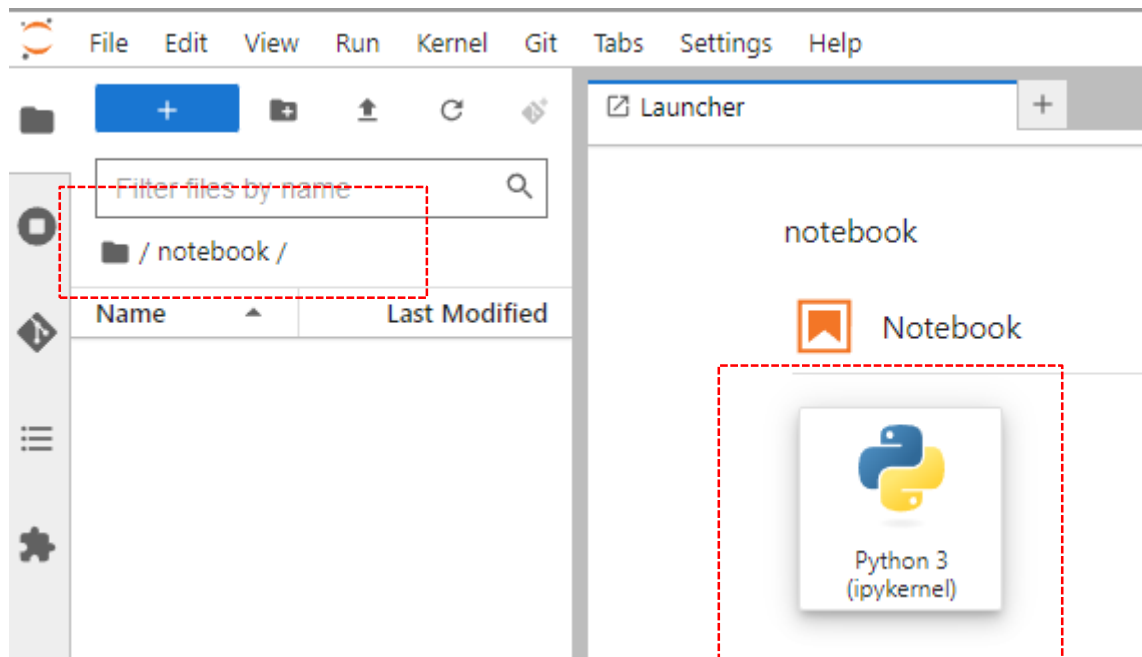


- 브라우저에서 실행

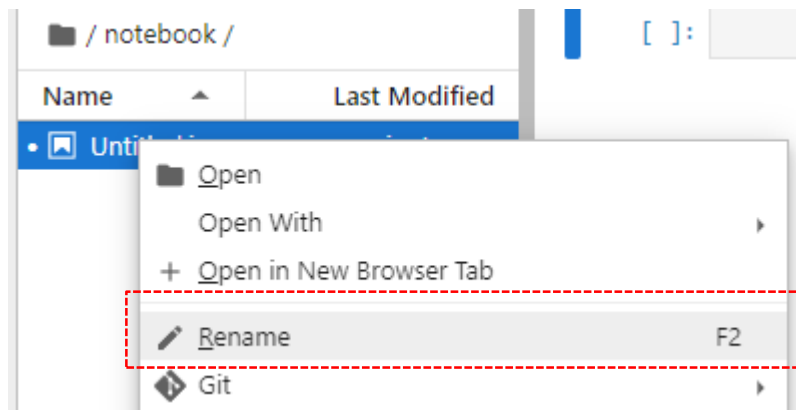
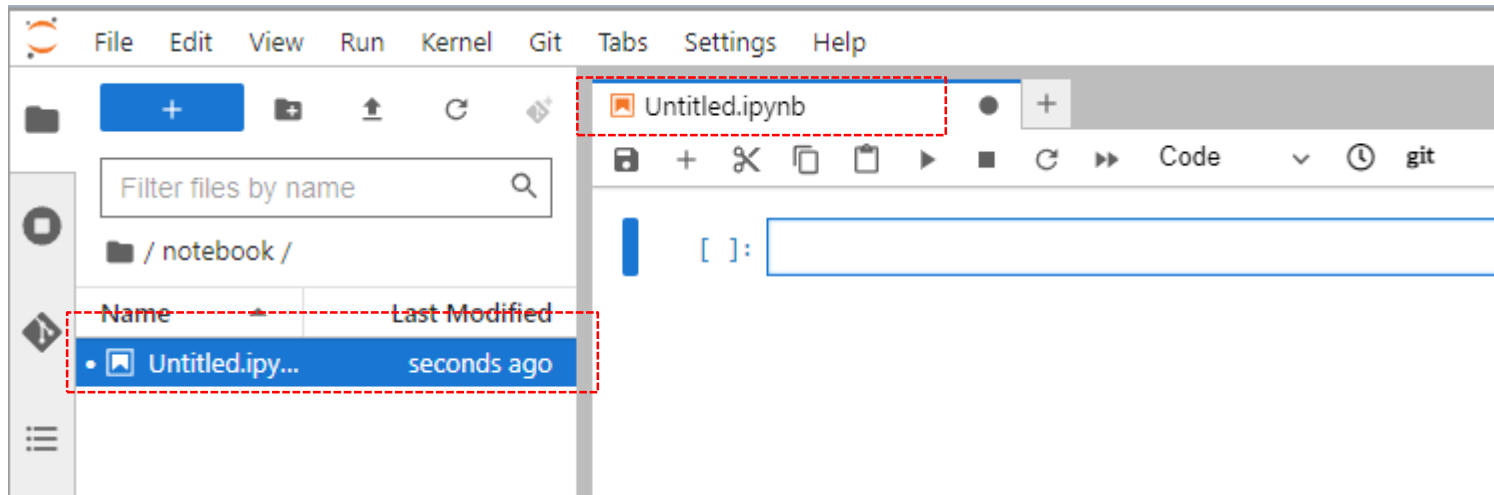


- 파일 생성

- 폴더 notebook 생성 후 하부에 파일 mylst.ipynb 생성



- 메뉴 Rename
  - 단축키 F2



- mylst.ipynb

https://chatkang-bug-free-space-waffle-pvv4x4jgrq92j9q.github.dev/?editor=jupyter

File Edit View Run Kernel Git Tabs Settings Help

Filter files by name

/ notebook /

Name Last Modified

• mylst.ipynb seconds ago

mylst.ipynb

Python 3 (ipykernel)

```
[1]: import sys
     sys.version

[1]: '3.10.4 (main, Mar 13 2023, 19:44:25) [GCC 9.4.0]'
```

```
[2]: lst = []
```

```
[5]: lst.append(int(input('정수 입력 > ')))
     정수 입력 > 329
```

```
[6]: lst

[6]: [329]
```

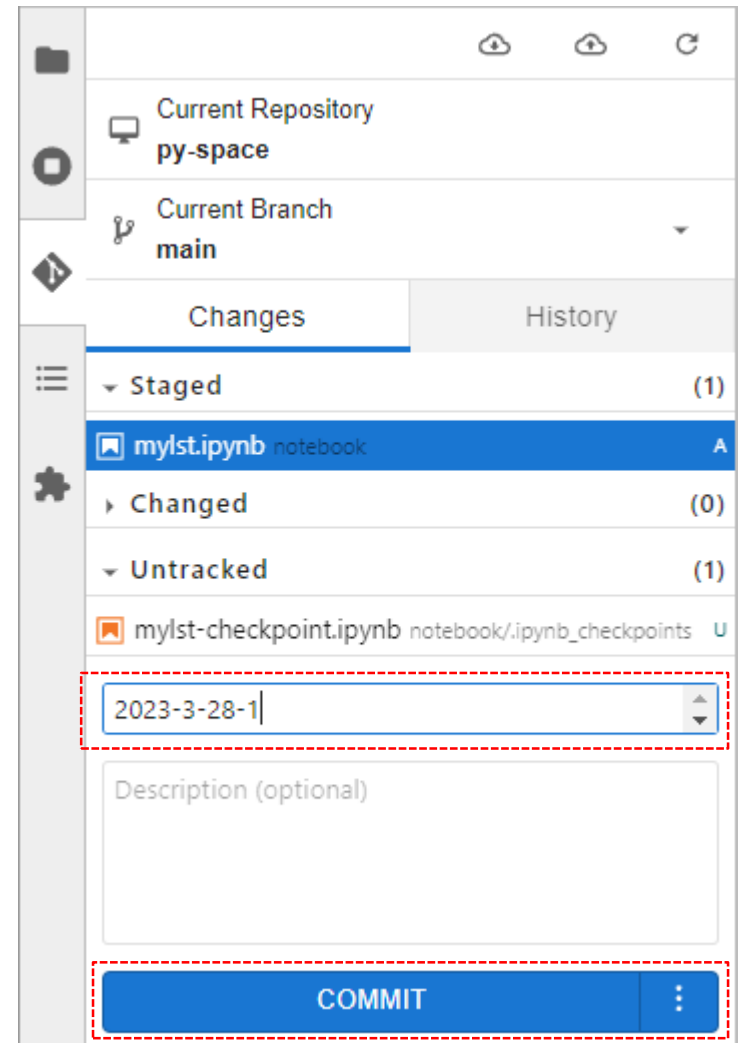
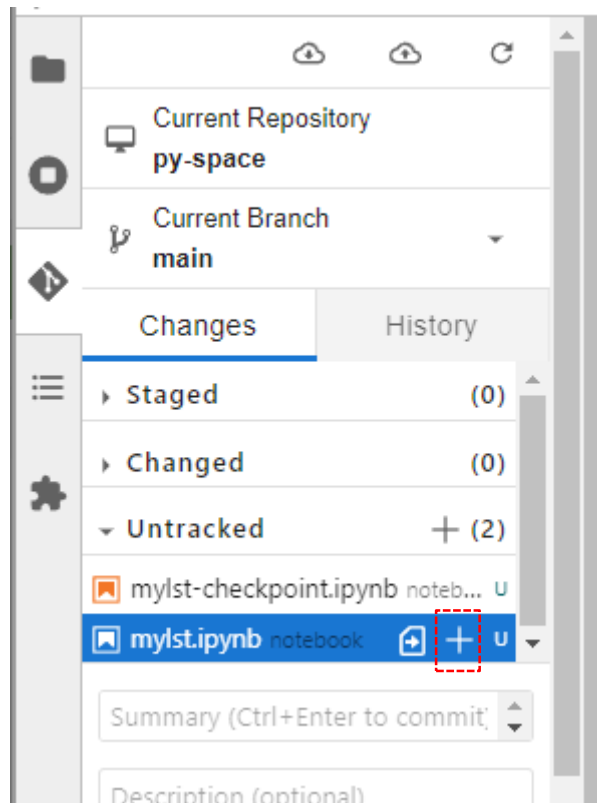
[ ]:

Simple 0 1 main Python 3 (ipykernel) | Connecting Mode: Edit Ln 1, Col 1 mylst.ipynb

# Add, commit

오픈소스 소프트웨어를 위한 깃과 깃허브 Python language

- Untracked 소스의 + 클릭
- Staged 소스 클릭
  - 커밋 메시지 입력 후 COMMIT 버튼 클릭



- **Push 구름 클릭 후**
  - 깃허브에서 확인

