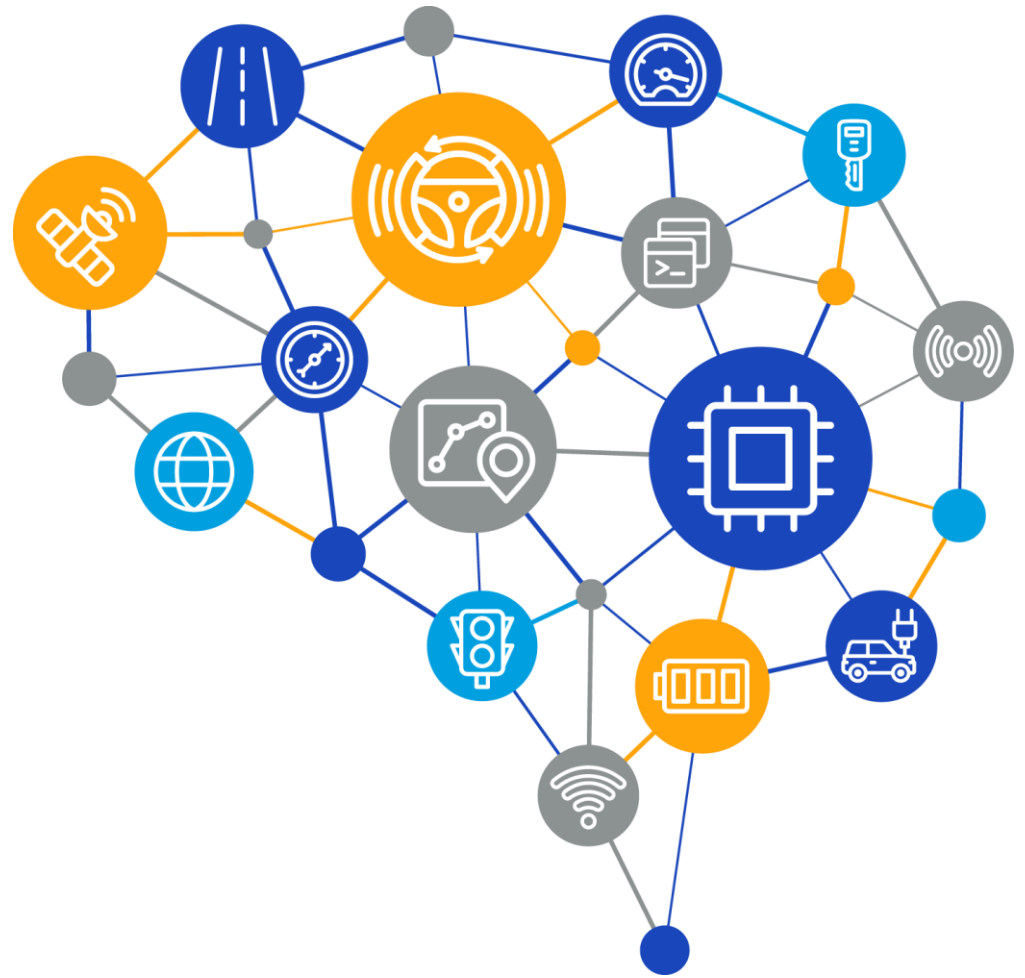


# 코드스페이스 템플릿 개요와 템플릿 코드스페이스 생성 후 깃허브 게시

강환수 교수

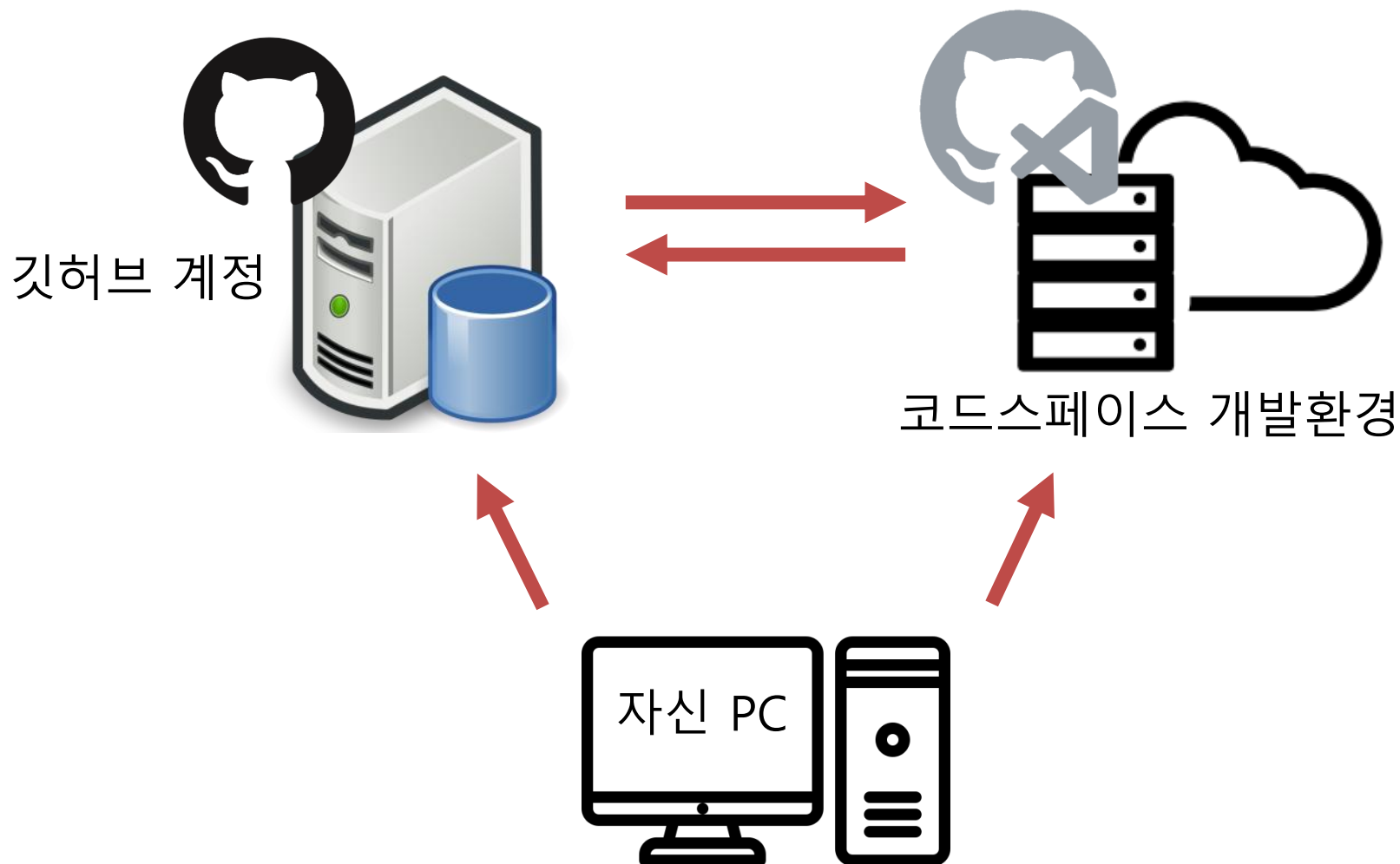


AI Experts  
Who Lead  
The Future

# 01

## 깃허브 코드스페이스 템플릿 개요

- 깃허브에서 실행하는 클라우드 개발환경

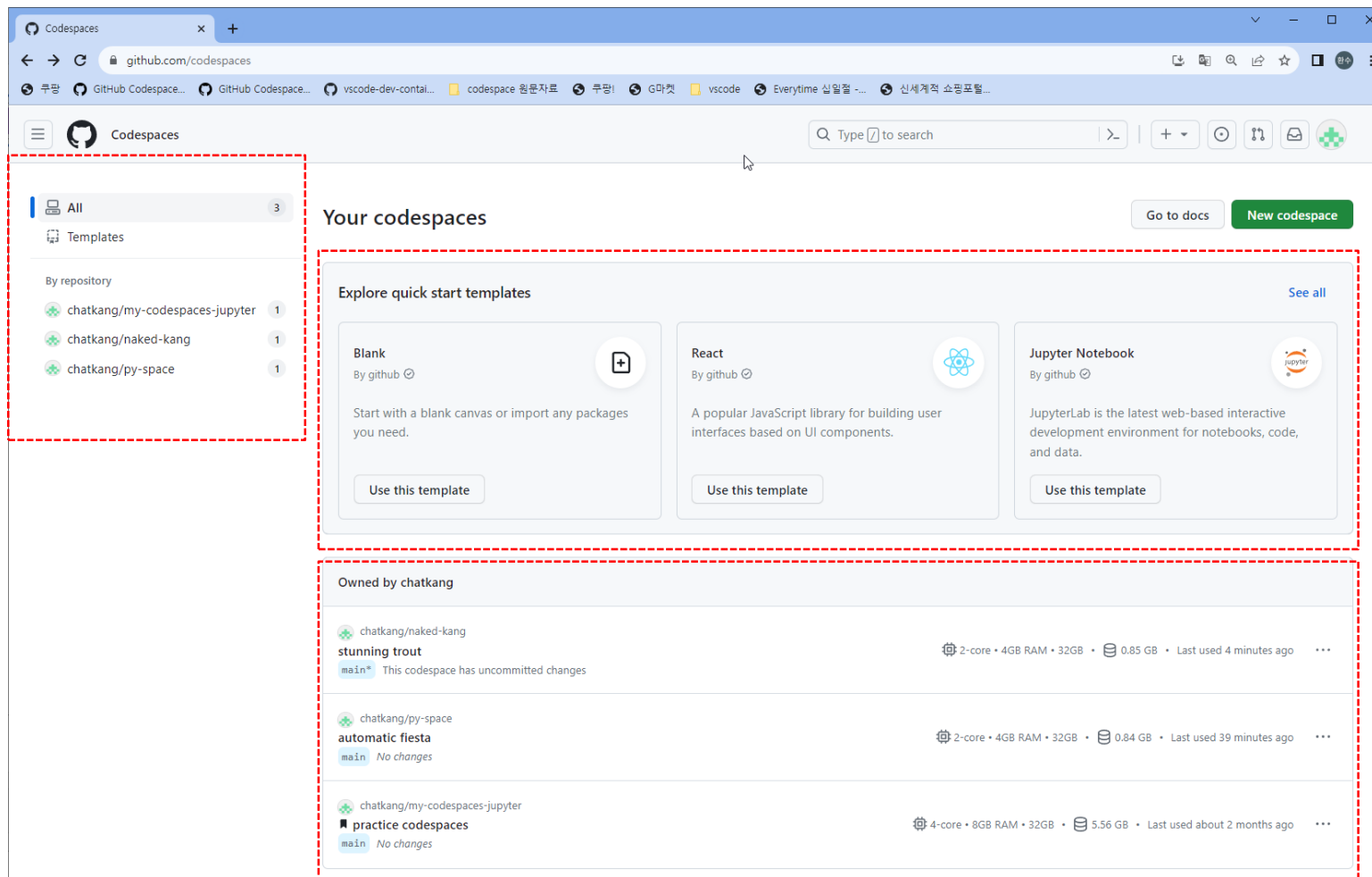


- 특정 언어, 프레임워크 또는 플랫폼을 위한 미리 구성된 개발환경을 제공
- 필요한 개발환경 설정을 가지고 코드페이스를 생성해 보다 쉽고 빠르게 전용 개발환경을 설정해주는 틀
  - 예를 들어, Node.js, Python, Ruby, React, Vue.js, Angular 등 다양한 언어 및 프레임워크를 위한 템플릿이 제공
- 템플릿 사용 장점
  - 개발자는 개발환경을 수동으로 설정할 필요없이 몇 번의 클릭만으로 프로젝트를 시작 가능
    - 코드스페이스를 연결해 원하는 특정 개발환경을 빠르게 설정
  - 프로젝트 구조, 라이브러리 종속성 및 설정 등을 미리 구성해 놓기 때문에, 개발자가 직접 구성하는 것보다 더 효율적이고 일관성 있는 개발환경을 제공

- 화면 좌측



- 자신이 생성한 코드스페이스 목록이 표시
- 우측

- 템플릿과 자신의 코드스페이스의 상세 정보가 표시





## Choose a template

Start a codespace from a template and get to developing with the power of a virtual machine in the cloud.

**Blank**  
By github 



Start with a blank canvas or import any packages you need.

Use this template

**Ruby on Rails**  
By github 



A full-stack web framework for building dynamic websites that deliver a rich user experience.

Use this template

**React**  
By github 



A popular JavaScript library for building user interfaces based on UI components.

Use this template

**Jupyter Notebook**  
By github 



JupyterLab is the latest web-based interactive development environment for notebooks, code, and data.

Use this template

**Express**  
By github 



Express is a minimal and flexible Node.js web application framework.

Use this template

**Next.js**  
By github 



Next.js is a React framework that gives you building blocks to create web applications.

Use this template

**Django**  
By github 



Django is a high-level Python web framework that encourages rapid development and clean, pragmatic design.

Use this template

**Flask**  
By github 

Flask is a lightweight web application framework.

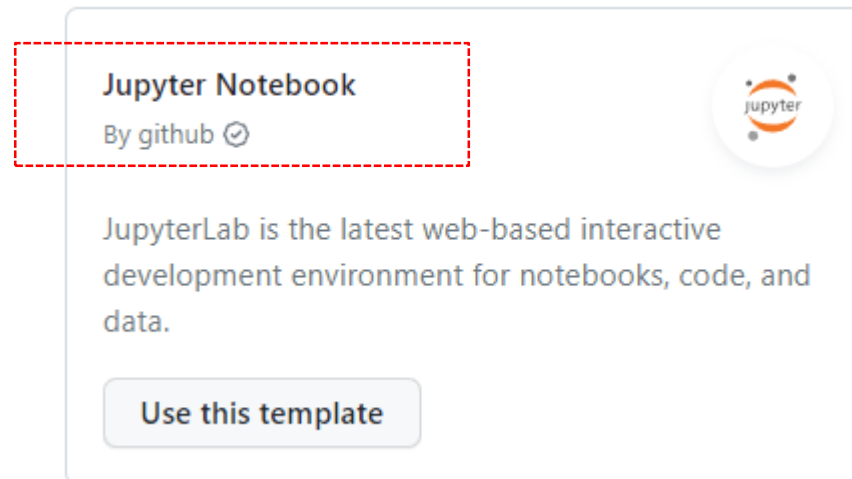
Use this template

**Preact**  
By github 

A fast 3kB alternative to React with the same modern API.

Use this template

- 파이썬의 주피터 노트북을 지원하는 템플릿
  - Jupyter Notebook 클릭



The screenshot shows the GitHub repository page for 'codespaces-jupyter'. The repository is a public template with 61 watches, 701 forks, and 378 stars. It has 2 branches and 0 tags. The main branch is selected. The file list includes .devcontainer, data, notebooks, .gitignore, LICENSE, README.md, and requirements.txt. The README.md file is open, showing a welcome message and instructions for using the codespace. The repository is described as a place to explore machine learning and data science with Codespaces. The README content is as follows:

**GitHub Codespaces ❤️ Jupyter Notebooks**

Welcome to your shiny new codespace! We've got everything fired up and running for you to explore Python and Jupyter notebooks.

You've got a blank canvas to work on from a git perspective as well. There's a single initial commit with what you're seeing right now - where you go from here is up to you!

Everything you do here is contained within this one codespace. There is no repository on GitHub yet. If and when you're ready you can click "Publish Branch" and we'll create your repository and push up your project. If you were just exploring then and have no further need for this code then you can simply delete your codespace and it's gone forever.

**About**

Explore machine learning and data science with Codespaces

data-science machine-learning  
jupyter-notebook codespaces

Readme  
MIT license  
Code of conduct  
Security policy  
Activity  
378 stars  
61 watching  
701 forks  
Report repository

**Contributors** 7

**Languages**

Jupyter Notebook 100.0%



- Jupyter Notebook 템플릿

- <https://github.com/github/codespaces-jupyter>인 깃허브 저장소
  - 코드스페이스로 기계 학습 및 데이터 과학 분야를 개발하기 위한 템플릿

- Readme.md 내용

- 반짝이는 새 codespace에 오신 것을 환영합니다! Python 및 Jupyter 노트북을 탐색할 수 있는 모든 기능이 준비되어 있습니다.
- git 관점에서도 작업할 수 있는 빈 캔버스가 있습니다. 지금 보고 있는 것과 관련된 단일 초기 커밋이 있습니다. 여기서 어디로 가는지는 당신에게 달려 있습니다!
- 여기에서 수행하는 모든 작업은 이 하나의 코드 공간에 포함됩니다. GitHub에는 아직 리포지토리가 없습니다. 준비가 되었으면 "게시 지점(publish branch)"을 클릭하면 저장소가 생성되고 프로젝트가 푸시됩니다. 방금 탐색 중이었고 이 코드가 더 이상 필요하지 않은 경우 코드 공간을 삭제하면 영원히 사라집니다.

# 커밋 이력 보기

오픈소스 소프트웨어를 위한 깃과 깃허브 Python language

- 다음 중앙 '54 commits' 클릭

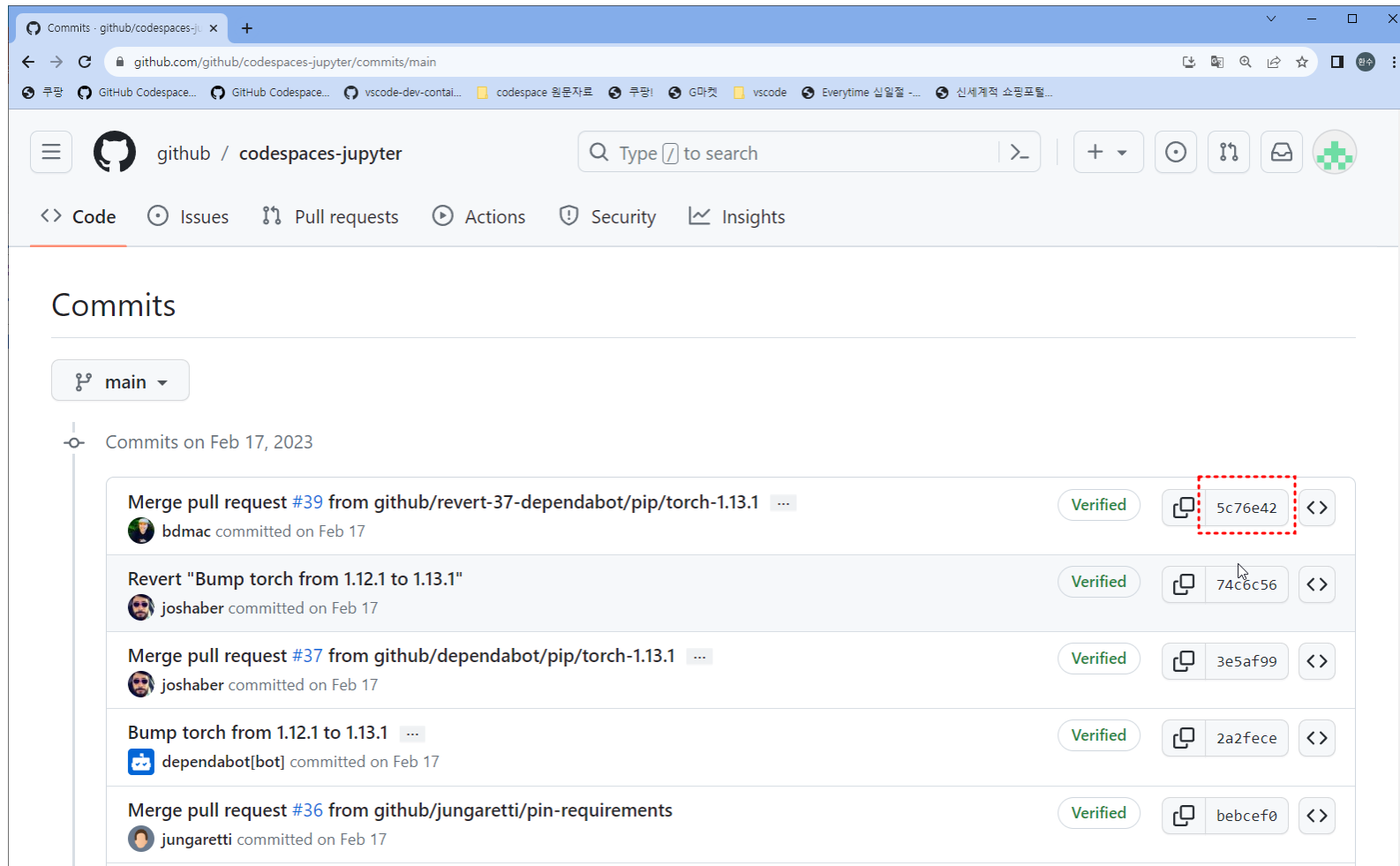
The screenshot shows the GitHub repository page for `codespaces-jupyter`. The repository is a public template. The main branch is `main`. The repository structure includes the following files and folders:

File/Folder	Description	Commit Date
<code>.devcontainer</code>	Update devcontainer.json	7 months ago
<code>data</code>	Add interactive notebook	9 months ago
<code>notebooks</code>	Fix axis titles	8 months ago
<code>.gitignore</code>	Add .ipynb_checkpoints/ to .gitignore	8 months ago
<code>LICENSE</code>	Initial commit	last year
<code>README.md</code>	Update SCM tip.	9 months ago
<code>requirements.txt</code>	Revert "Bump torch from 1.12.1 to 1.13.1"	5 months ago

The commit history table shows a merge pull request #39 from `github/revert...` on Feb 17 with 54 commits. The '54 commits' link is highlighted with a dashed blue box.

On the right side, the 'About' section provides information about the repository, including tags like `data-science`, `machine-learning`, `jupyter-notebook`, and `codespaces`. It also lists the repository's README, MIT license, Code of conduct, Security policy, Activity, 378 stars, and 61 watching.

- 40개의 16진수 클릭



- 파이썬 자동 설치 모듈 목록
  - 모듈이름==00.00.0

The screenshot shows a GitHub web interface for a commit. The browser tab is 'Merge pull request #39 from gi...'. The address bar shows 'github.com/github/codespaces-jupyter/commit/5c76e42e748d3f9085a6fe216d1e934a63e6c79d'. The page title is 'Commit'. The commit message is 'Merge pull request #39 from github/revert-37-dependabot/pip/torch-1.13.1' with a sub-message 'Revert "Bump torch from 1.12.1 to 1.13.1"'. The commit is by 'bdmac' on Feb 17, marked as 'Verified'. It shows '2 parents 3e5af99 + 74c6c56' and 'commit 5c76e42'. Below, it says 'Showing 1 changed file with 1 addition and 1 deletion.' and shows a diff for 'requirements.txt'. The diff highlights a change from 'torch==1.13.1' to 'torch==1.12.1'. At the bottom, it says '0 comments on commit 5c76e42' and has 'Write' and 'Preview' buttons.

Commit

Merge pull request #39 from github/revert-37-dependabot/pip/torch-1.13.1

Revert "Bump torch from 1.12.1 to 1.13.1"

main (#39)

bdmac committed on Feb 17 Verified 2 parents 3e5af99 + 74c6c56 commit 5c76e42

Showing 1 changed file with 1 addition and 1 deletion.

Split Unified

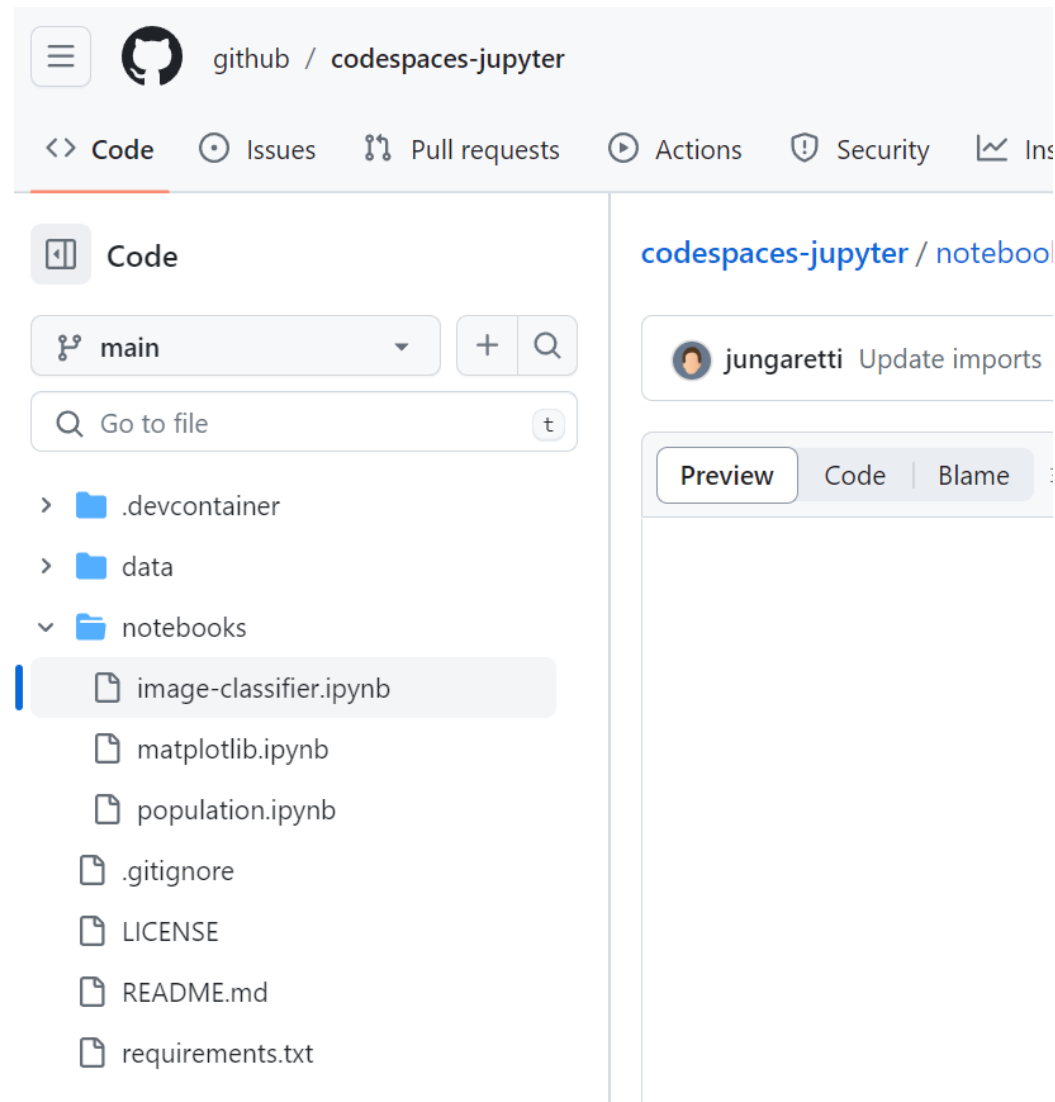
requirements.txt

@@ -2,6 +2,6 @@ ipywidgets==7.7.1	
2 matplotlib==3.7.0	2 matplotlib==3.7.0
3 numpy==1.24.2	3 numpy==1.24.2
4 pandas==1.5.3	4 pandas==1.5.3
5 - torch==1.13.1	5 + torch==1.12.1
6 torchvision==0.13.1	6 torchvision==0.13.1
7 tqdm==4.64.0	7 tqdm==4.64.0

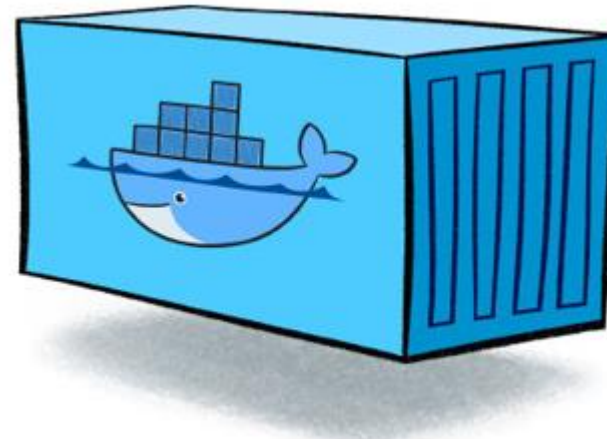
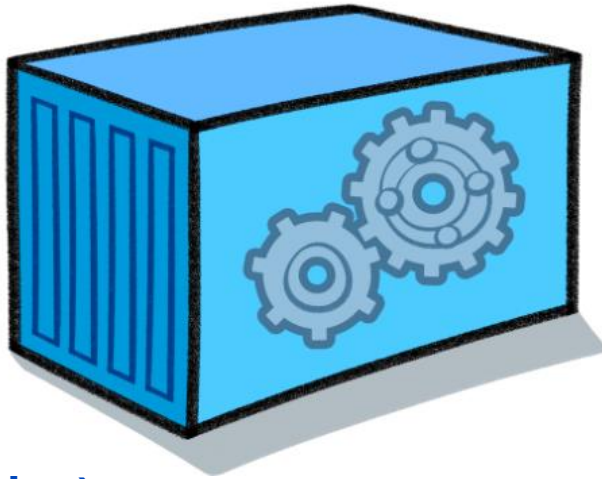
0 comments on commit 5c76e42

Write Preview

- **설정 폴더**
  - **.devcontainer**
- **코드와 자료 파일**
  - **Data**
  - **Notebooks**



- 소프트웨어를 실행하기 위한 가벼운, 독립적이고 이식 가능한 환경을 제공하는 기술



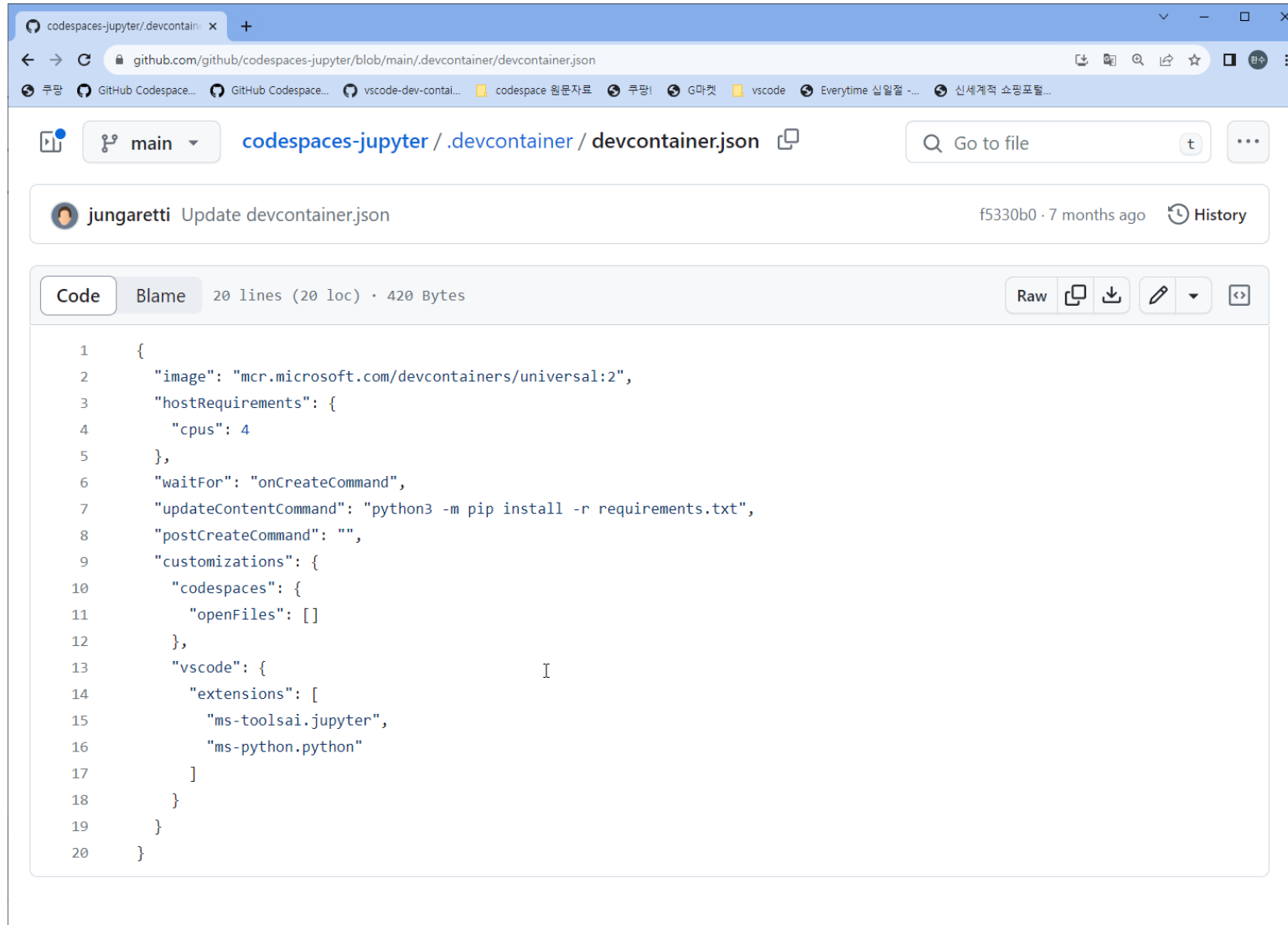
- 도커(Docker)

- 컨테이너화된 애플리케이션을 개발, 배포 및 실행하기 위한 오픈 소스 플랫폼



- 1. **\*\*독립성\*\***:
  - 각각의 컨테이너는 다른 컨테이너와 분리되어 있기 때문에, 개발 환경에서의 변경이 다른 컨테이너에 영향을 미치지 않습니다. 이는 개발 환경 간의 충돌을 최소화하고 개발자가 애플리케이션을 격리된 상태에서 개발할 수 있게 합니다.
- 2. **\*\*이식성\*\***:
  - 컨테이너는 어떤 환경에서도 동일하게 실행될 수 있습니다. 개발자가 로컬 환경에서 개발한 애플리케이션을 클라우드나 다른 서버에서도 똑같이 실행할 수 있으며, 이는 배포 및 테스트 단계에서 많은 혜택을 제공합니다.
- 3. **\*\*확장성\*\***:
  - 컨테이너 기술은 쉽게 확장할 수 있습니다. 필요에 따라 개발 환경의 컨테이너 수를 늘리거나 줄일 수 있으며, 이로 인해 리소스를 효율적으로 사용할 수 있습니다.
- 4. **\*\*빠른 시작\*\***:
  - 컨테이너는 가볍고 빠르게 시작할 수 있습니다. 따라서 개발자는 개발 환경을 빠르게 설정하고 바로 개발에 집중할 수 있습니다.

## • 개발 컨테이너 설정 파일

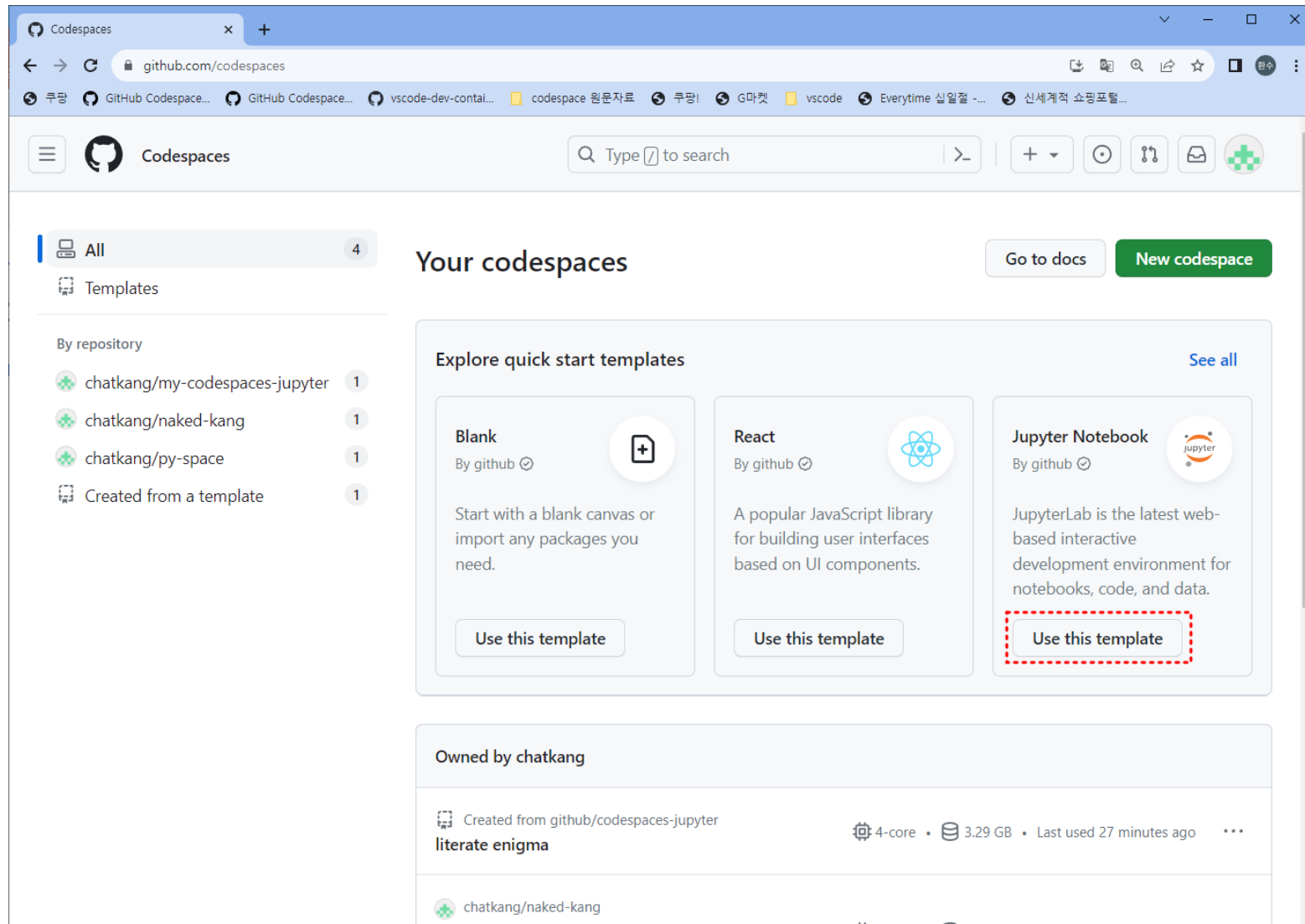


The screenshot shows a web browser displaying the GitHub repository for `codespaces-jupyter`. The file `devcontainer.json` is selected, showing its content. The file is a JSON configuration for a development container, specifying the image, host requirements, and customizations for VS Code.

```
1  {
2    "image": "mcr.microsoft.com/devcontainers/universal:2",
3    "hostRequirements": {
4      "cpus": 4
5    },
6    "waitFor": "onCreateCommand",
7    "updateContentCommand": "python3 -m pip install -r requirements.txt",
8    "postCreateCommand": "",
9    "customizations": {
10     "codespaces": {
11       "openFiles": []
12     },
13     "vscode": {
14       "extensions": [
15         "ms-toolsai.jupyter",
16         "ms-python.python"
17       ]
18     }
19   }
20 }
```



- 바로 [Use this template] 클릭



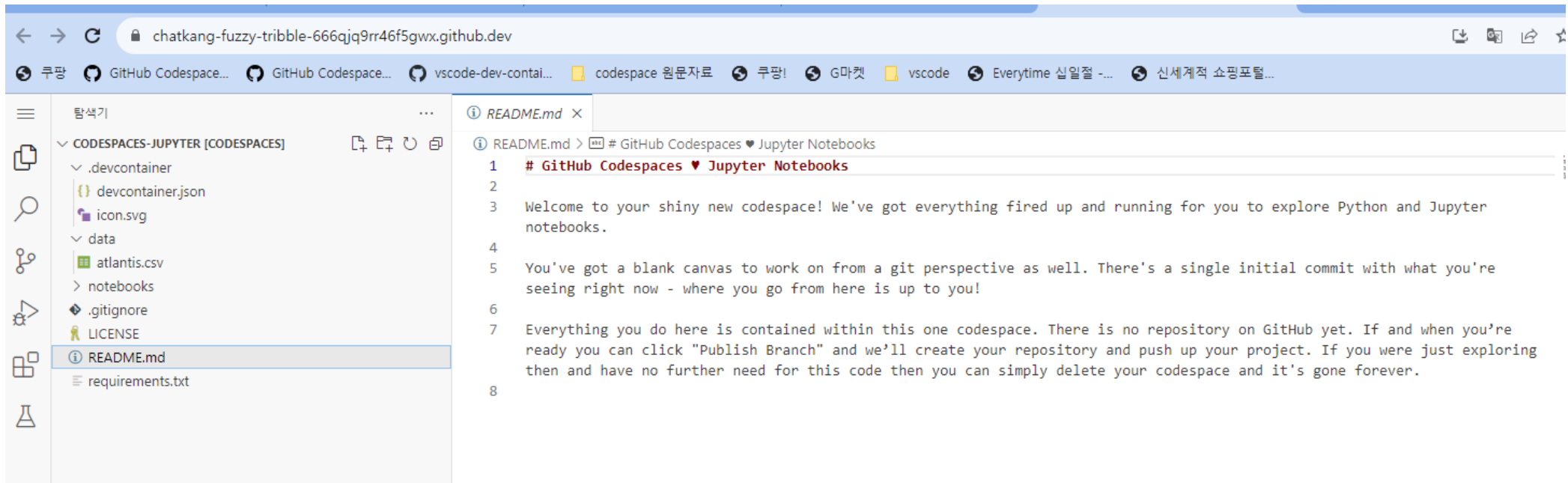
AI Experts  
Who Lead  
The Future

## 02

---

코드스페이스 템플릿을 사용해  
작업 코드스페이스 생성

- 파일 구성
  - 폴더이름: CODESPACES-JUPYTER [CODESPACES]
  - 3개의 폴더
  - 여러 개의 파일



## • 리눅스 터미널

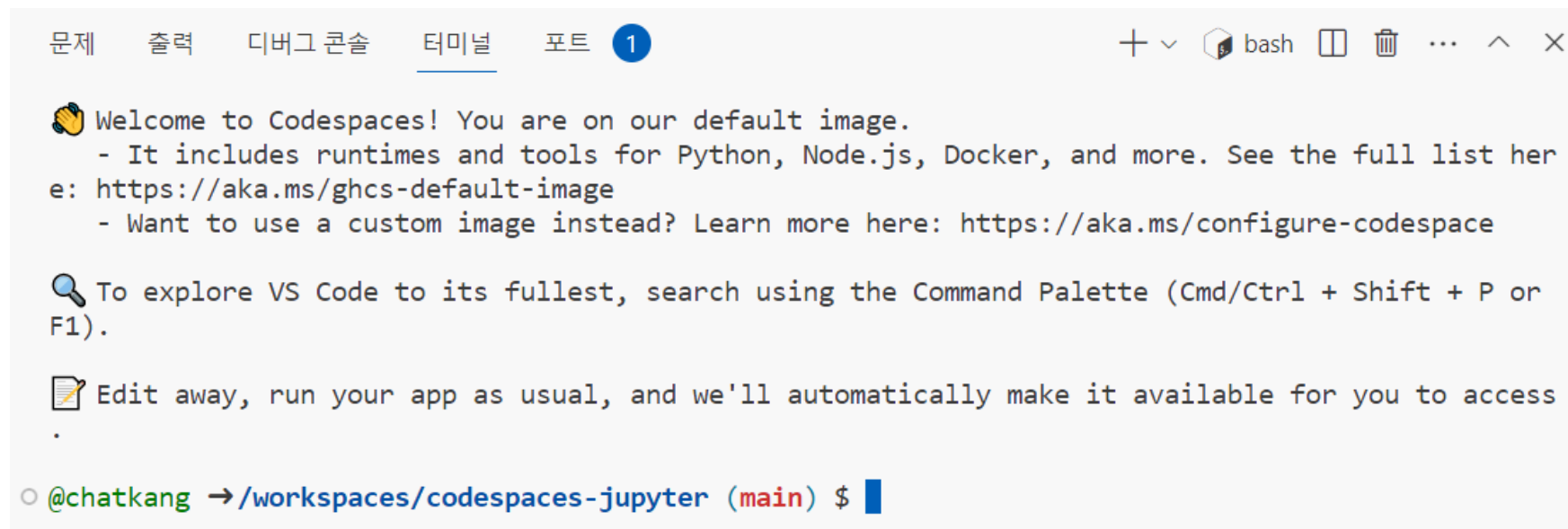
Welcome to Codespaces! You are on our default image.

- 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.

@chatkang → /workspaces/codespaces-jupyter (main) \$



```
문제  출력  디버그 콘솔  터미널  포트 1  +  bash  [icon]  [icon]  ...  ^  X

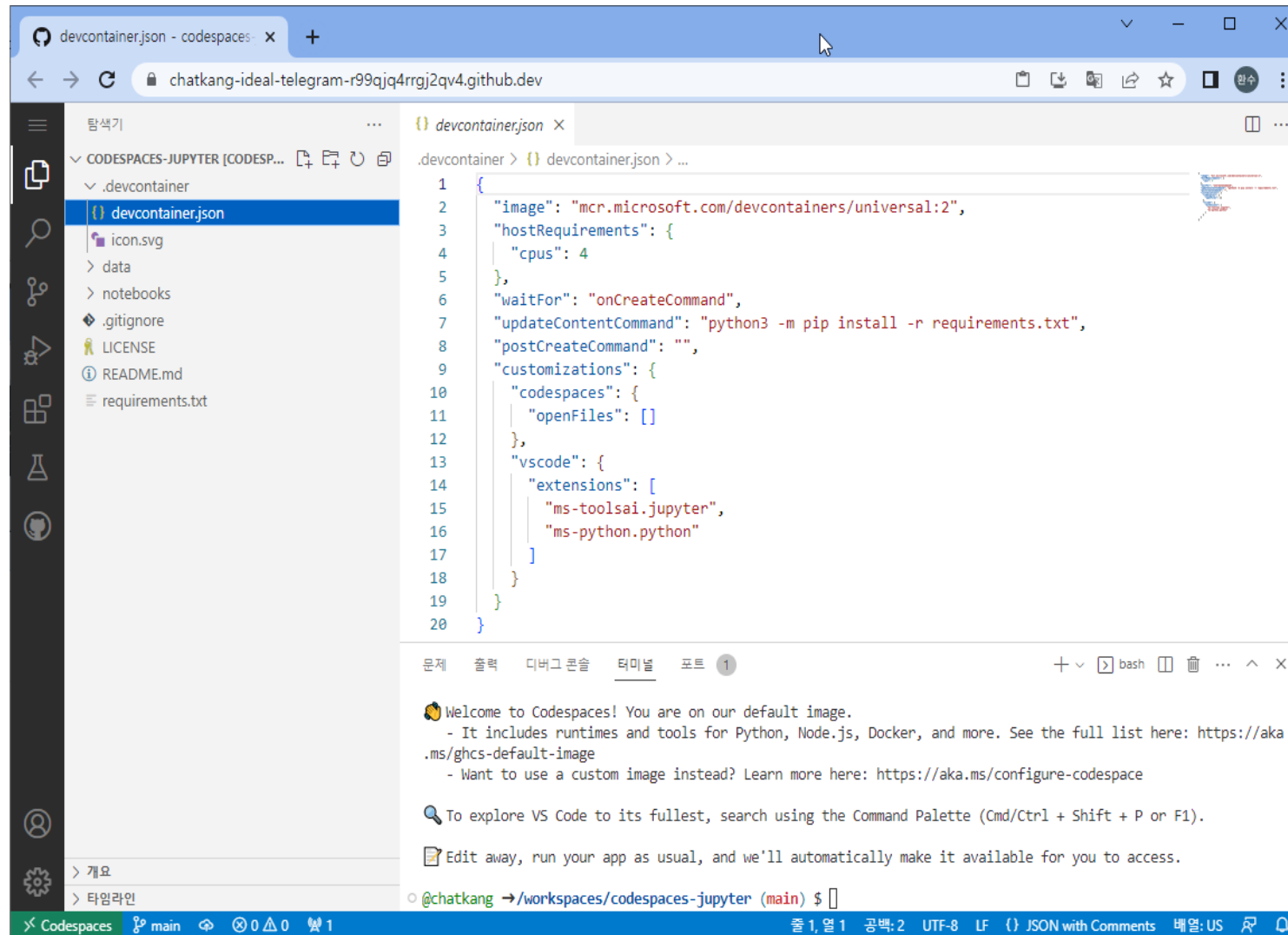
👋 Welcome to Codespaces! You are on our default image.
  - It includes runtimes and tools for Python, Node.js, Docker, and more. See the full list here: https://aka.ms/ghcs-default-image
  - Want to use a custom image instead? Learn more here: https://aka.ms/configure-codespace

🔍 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
.

@chatkang → /workspaces/codespaces-jupyter (main) $
```

- **.devcontainer/devcontainer.json**



- 모듈 Faker 설치
  - `pip install Faker`

- 폴더 code
- 파일 fake-dict.py

```
from faker import Faker

def generate_fake_data():
    faker = Faker('ko_KR') # 한국어 데이터를 생성하기 위해 'ko_KR' locale을 설정합니다.
    data_list = []

    for _ in range(10): # 10개의 키와 값이 들어가도록 반복합니다.
        name = faker.name()
        job = [faker.job()] # 리스트로 만들기 위해 job을 대괄호로 묶습니다.
        address = [faker.address()] # 리스트로 만들기 위해 address를 대괄호로 묶습니다.

        data = {
            'name': name,
            'job': job,
            'address': address
        }
        data_list.append(data)

    return data_list

if __name__ == "__main__":
    fake_data_list = generate_fake_data()
    print(fake_data_list)
```

- 실행
  - Ctrl + F5

## • 다음으로 코드 작성

- 파이썬 모듈 Faker를 사용해 다음의 목록을 만드는 코드를 작성해 줘
- 한글로 하고 사전 형식으로 name이 키이고 리스트 job, address를 만들어
- 사전에는 10개의 키와 값이 들어가도록

HS

파이썬 모듈 Faker를 사용해 다음의 목록을 만드는 코드를 작성해 줘  
- 한글로 하고 사전 형식으로 name이 키이고 리스트 job, address를 만들어  
- 사전에는 10개의 키와 값이 들어가도록



먼저, 'Faker' 모듈을 설치해야 합니다. 아래와 같이 'Faker'를 설치하는 방법은 pip를 이용하는 것입니다:

bash

Copy code

```
pip install Faker
```

이후에 파이썬 코드에서 'Faker'를 사용하여 목록을 생성할 수 있습니다. 다음은 요구사항에 맞게 코드를 작성한 예시입니다:

python

Copy code

```
from faker import Faker

def generate_fake_data():
    faker = Faker('ko_KR') # 한국어 데이터를 생성하기 위해 'ko_KR' locale를 설정
    data_list = []

    for _ in range(10): # 10개의 키와 값이 들어가도록 반복합니다.
        name = faker.name()
        job = [faker.job()] # 리스트로 만들기 위해 job을 대괄호로 묶습니다.
        address = [faker.address()] # 리스트로 만들기 위해 address를 대괄호로 묶습니다.

        data = {
            'name': name,
            'job': job,
            'address': address
        }
        data_list.append(data)

    return data_list

if __name__ == "__main__":
    fake_data_list = generate_fake_data()
    print(fake_data_list)
```

위 코드를 실행하면 한글로 된 이름과 해당 사람들의 직업과 주소를 가지는 10개의 사전이 담긴 리스트가 생성됩니다. 출력된 결과는 각 실행마다 다를 수 있습니다.

Regenerate



AI Experts  
Who Lead  
The Future

## 04

### 대화형 창과 주피터 노트북 실행

- 파일 fake-basic.py
  - 셀 구성
    - # %%
  - 셀 실행
    - shift + Enter

```
code > fake-dict.py U fake-basic.py 1, U x ▶ ▼ 🔍 □ ... Interactive-1 x ...
```

code > fake-basic.py > ...

셀 실행 | 아래에서 실행 | 디버그 셀

```
1 # %%
2 from faker import Faker
3
4 f = Faker('ko_KR')
5 f.name()
6 f.job()
7 f.address()
8
9 셀 실행 | 위에서 실행 | 디버그 셀 | [5](으)로 이동
10 # %%
11 from faker import Faker
12
13 f = Faker()
14 f.name()
15 f.job()
16 f.address()
17
18 셀 실행 | 위에서 실행 | 디버그 셀
19 # %%
```

중단 | x 모두 지우기 | 재시작 | Python 3.10.8

✓ from faker import Faker ...

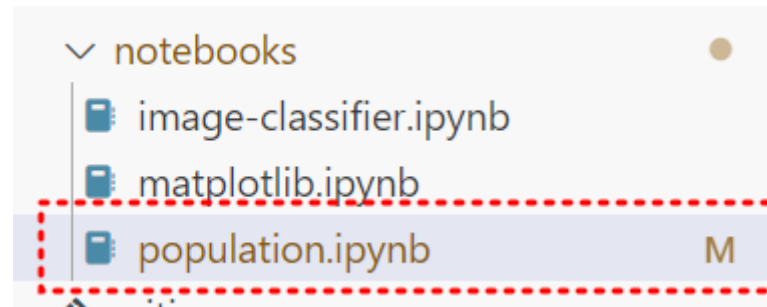
... '049 Rodriguez Motorway\nChelseafort, MI 03800

✓ from faker import Faker ...

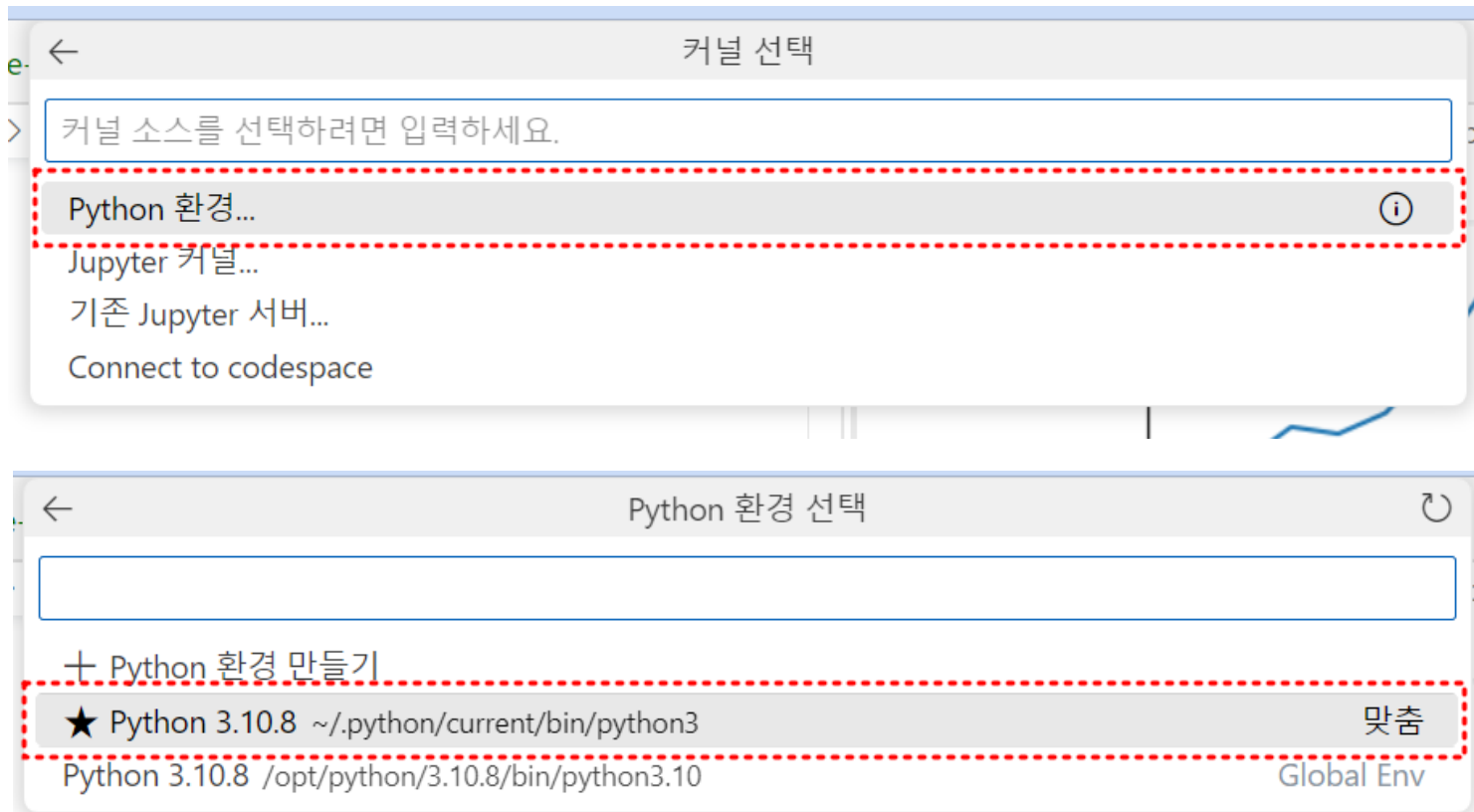
... '95094 Jones Trafficway Apt. 099\nLake Sarahsic

▶ 여기에 'python' 코드를 입력하고 '<Shift>+Enter'

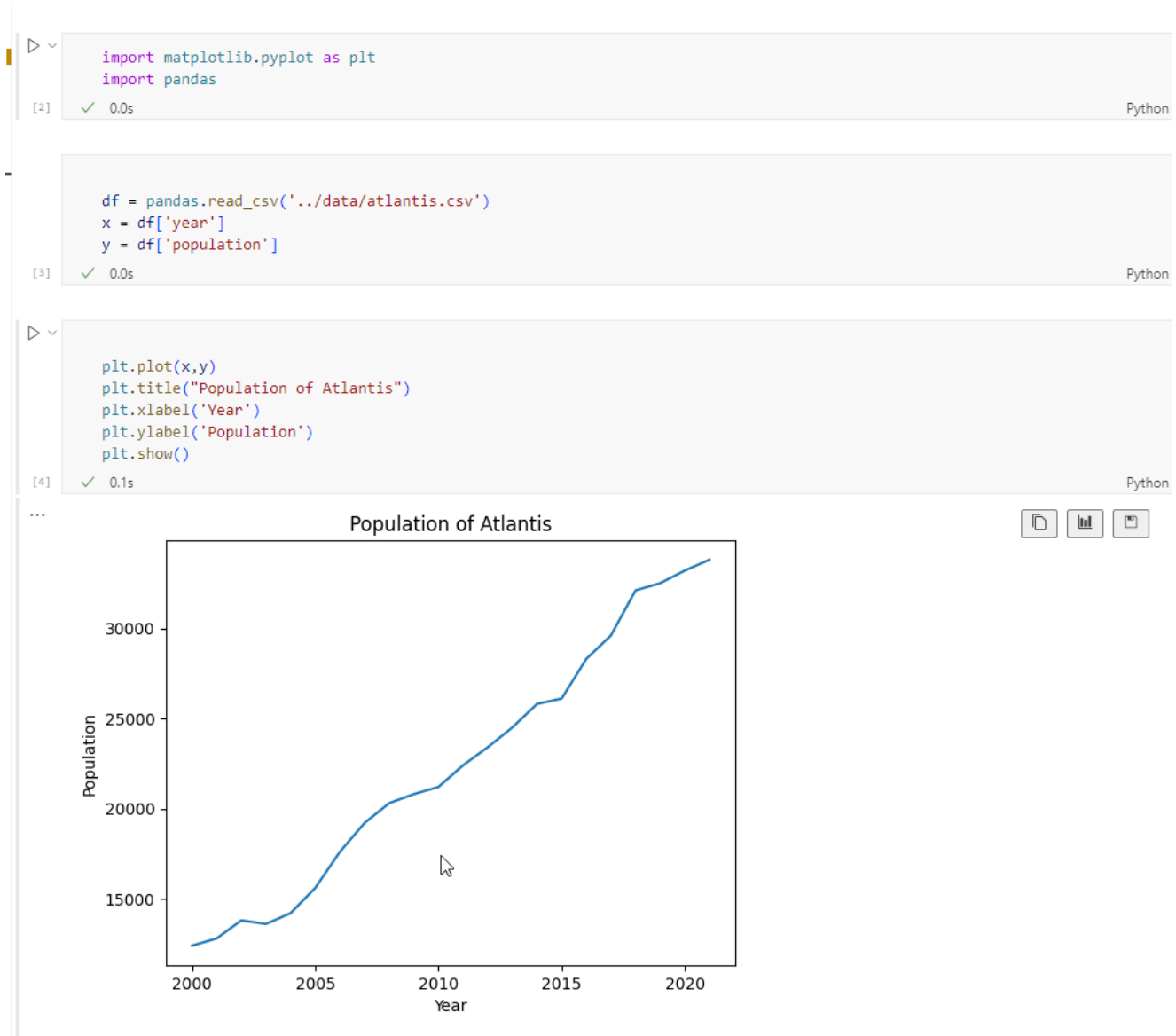
- 파일
  - Code/population.ipynb



- 셀에서
  - Shift + enter



- 명령 모드
  - C v
- 셀 나누기
  - Ctrl + shift + -
- 셀 실행
  - Ctrl + shift

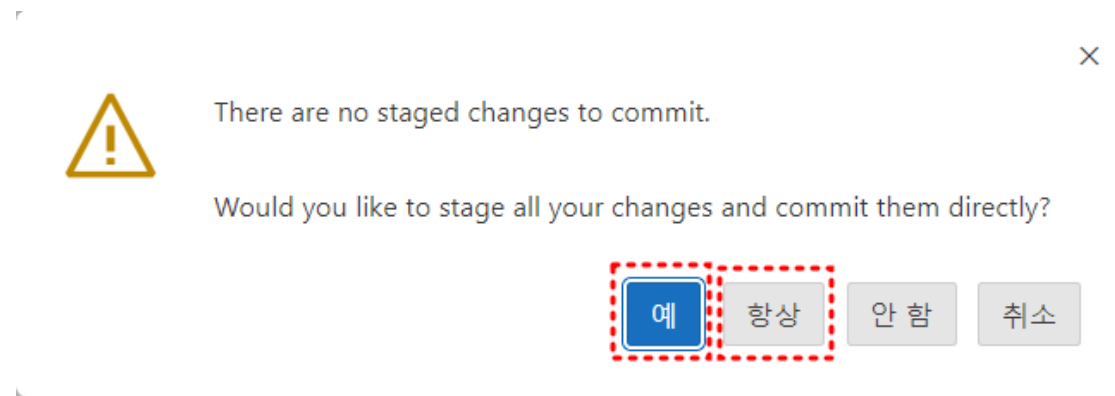
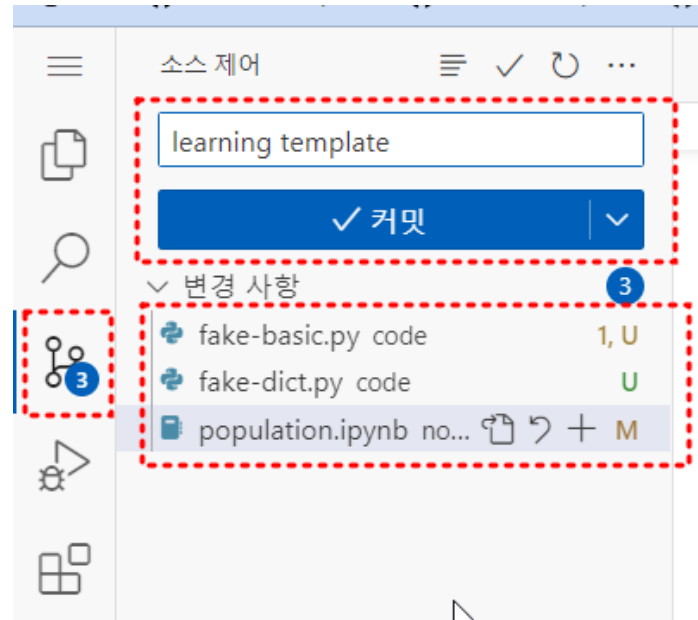


AI Experts  
Who Lead  
The Future

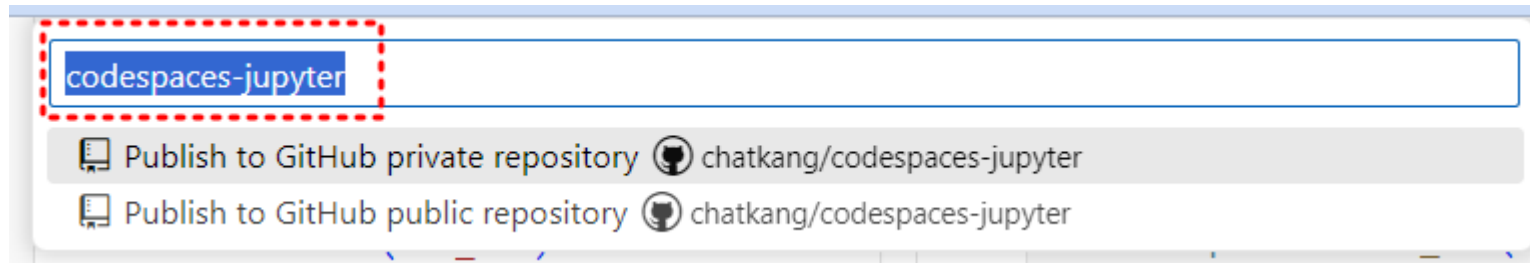
## 03

### 코드스페이스 저장소 깃허브 게시

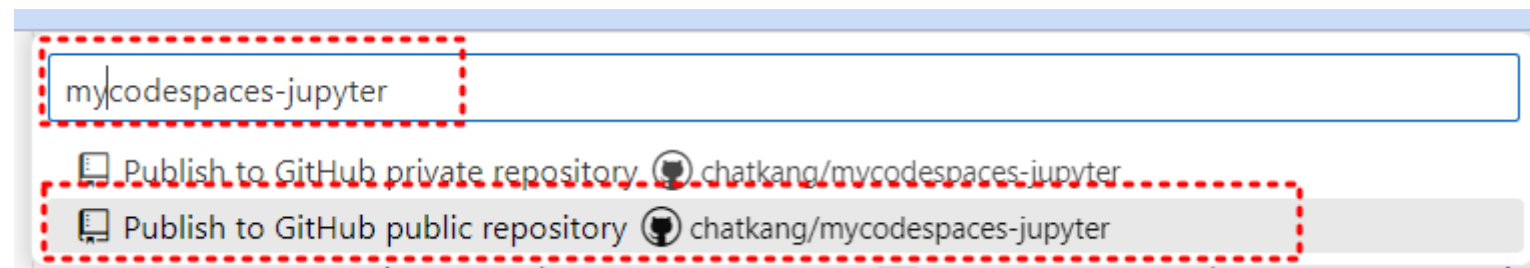
- 메시지 작성



- 자신의 깃허브 저장소에 저장
  - 기본 이름은 현재의 폴더 이름으로



- 이름 수정 가능



- Public 으로 클릭
  - 간혹 오류가 나기도



# 자신의 깃허브 저장소 확인

오픈소스 소프트웨어를 위한 깃과 깃허브 Python language