Python socket 프로그래밍 실습 및 실습 환경 구축

컴퓨터네트워크

실습(1)

- server.py 파일 생성, “UDPServer” 코드 입력
- 터미널에서 “python server.py”와 같이 실행
- client.py 파일 생성, “UDPClient” 코드 입력
- 터미널에서 “python client.py”와 같이 실행
- 실행 후 각각 server.py, client.py 탭 터미널에서 출력 확인
- TCP, UDP 각각의 경우에 대해 실행해 보기
- serverName = '127.0.0.1’

- TCPServer, TCPClient 코드에 대해서도 위와 같이 진행

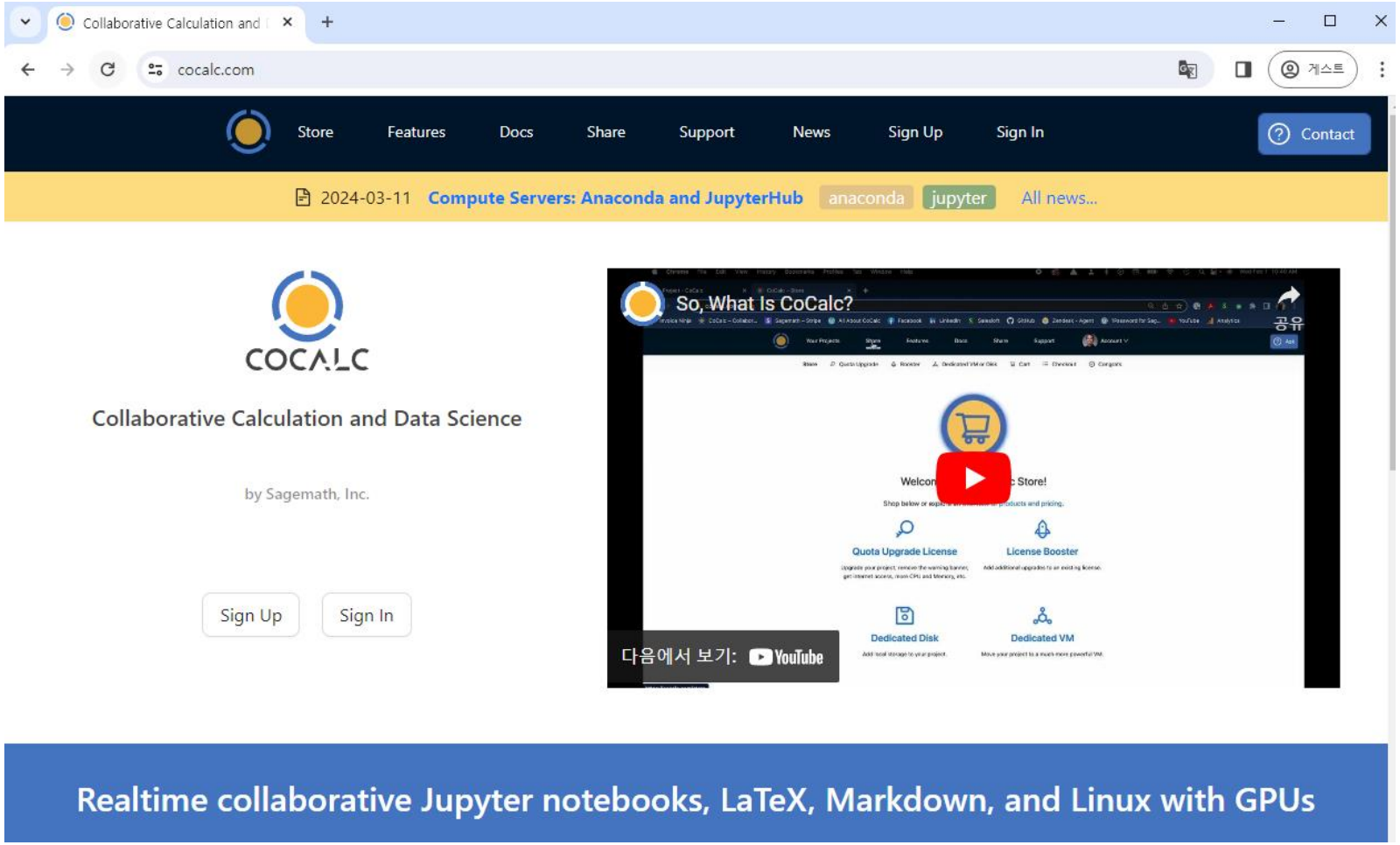
- python 개발 환경은 개인별 선택 (cocalc.com은 한가지 예)

실습(2) – 유사 HTTP 서버/클라이언트 프로그램

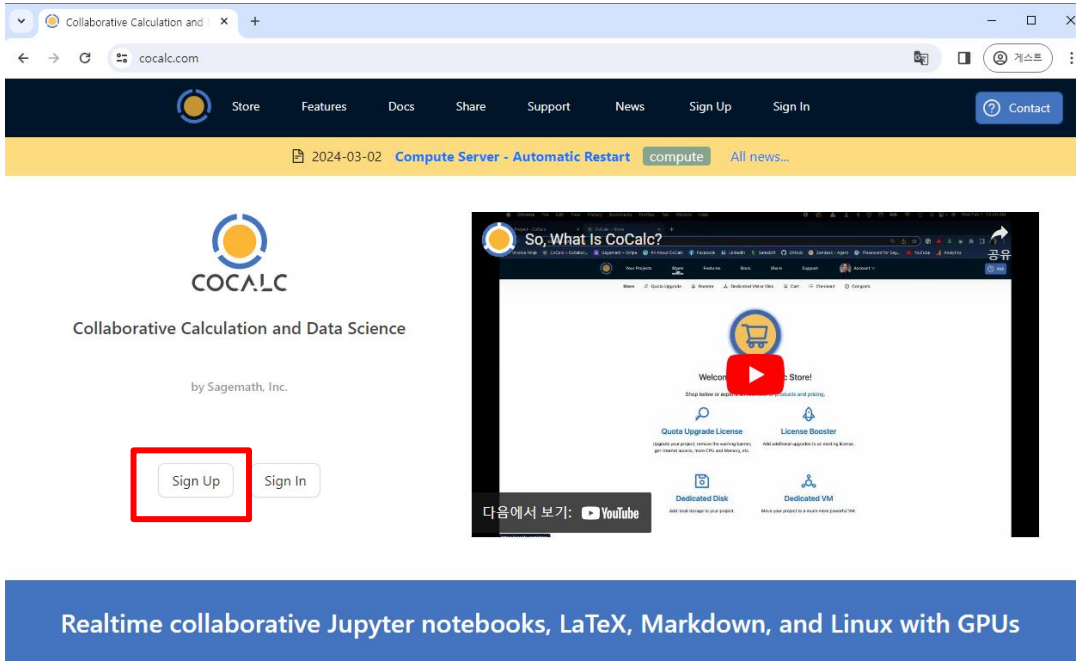
- 서버프로그램에서 수행할 작업
  - request message 수신 시, method 확인하여 GET 또는 POST 가 아닌 경우 "**`HTTP/0.1 405 Method Not Allowed`**"의 한 줄의 status line으로 (header는 0줄) 구성된 response message 클라이언트에게 전송
  - request message 수신 시, message의 형식에 부합한지 (request line이 있고 "end of header lines"가 표시 되었는지) 확인하여 부합하지 않는 경우 "**`HTTP/0.1 400 Bad Request`**"의 한 줄의 status line으로 구성된 response message 클라이언트에게 전송
  - GET request 수신 시, request message의 형식이 맞는지 확인하여 맞을 경우 request message의 크기(byte 단위)를 화면에 출력하고 "**`HTTP/0.1 200 OK`**"와 같은 한 줄의 status line으로 구성된 response message를 클라이언트에게 전송
  - POST request 수신 시 request message 내용(URL, header)과 관계없이 "**`HTTP/0.1 404 Not Found`**"와 같은 한 줄의 status line으로 구성된 response message를 클라이언트에게 전송
  - 서버는 request message 수신 시 수신한 message의 내용을 ASCII text 형식으로 message 크기(byte 단위)와 함께 화면에 출력
- 클라이언트 프로그램은 GET, POST, DELETE request message를 (각 request message는 HTTP 형식에 부합해야 함) 전송할 수 있도록 작성하여 서버프로그램에 각각의 request message를 전송함.
  - 클라이언트는 각 request를 전송 한 후 서버로부터 수신한 response message 내용을 ASCII text 형식으로 message 크기(byte 단위)와 함께 화면에 출력
- 서버/클라이언트 프로그램은 TCP 및 UDP를 이용하도록 각각 작성 후 테스트
- TCP 사용 시 non-persistent, persistent, pipelining 방식으로 동작하도록 작성 후 테스트
- 추가: POST method에서 **`Content-Length:`** 필드가 없는 경우 "411 Length Required" response 발생

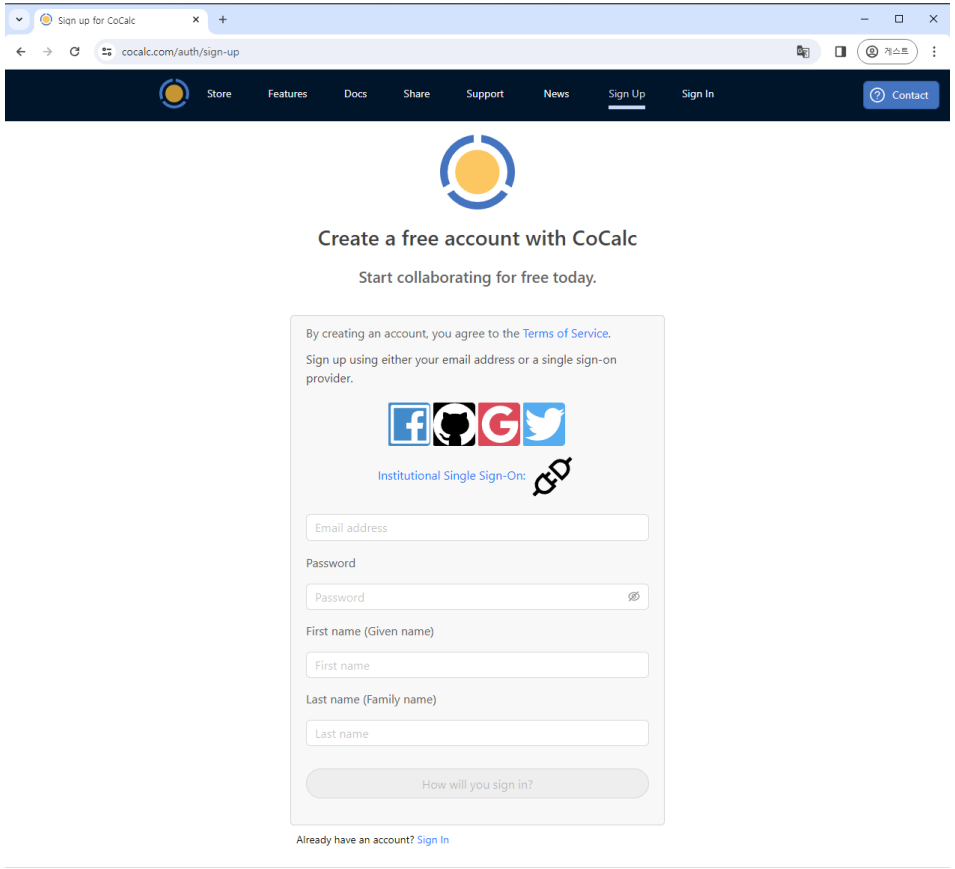
Python socket 프로그래밍 실습 환경 – cocalc.com

1. cocalc.com 접속

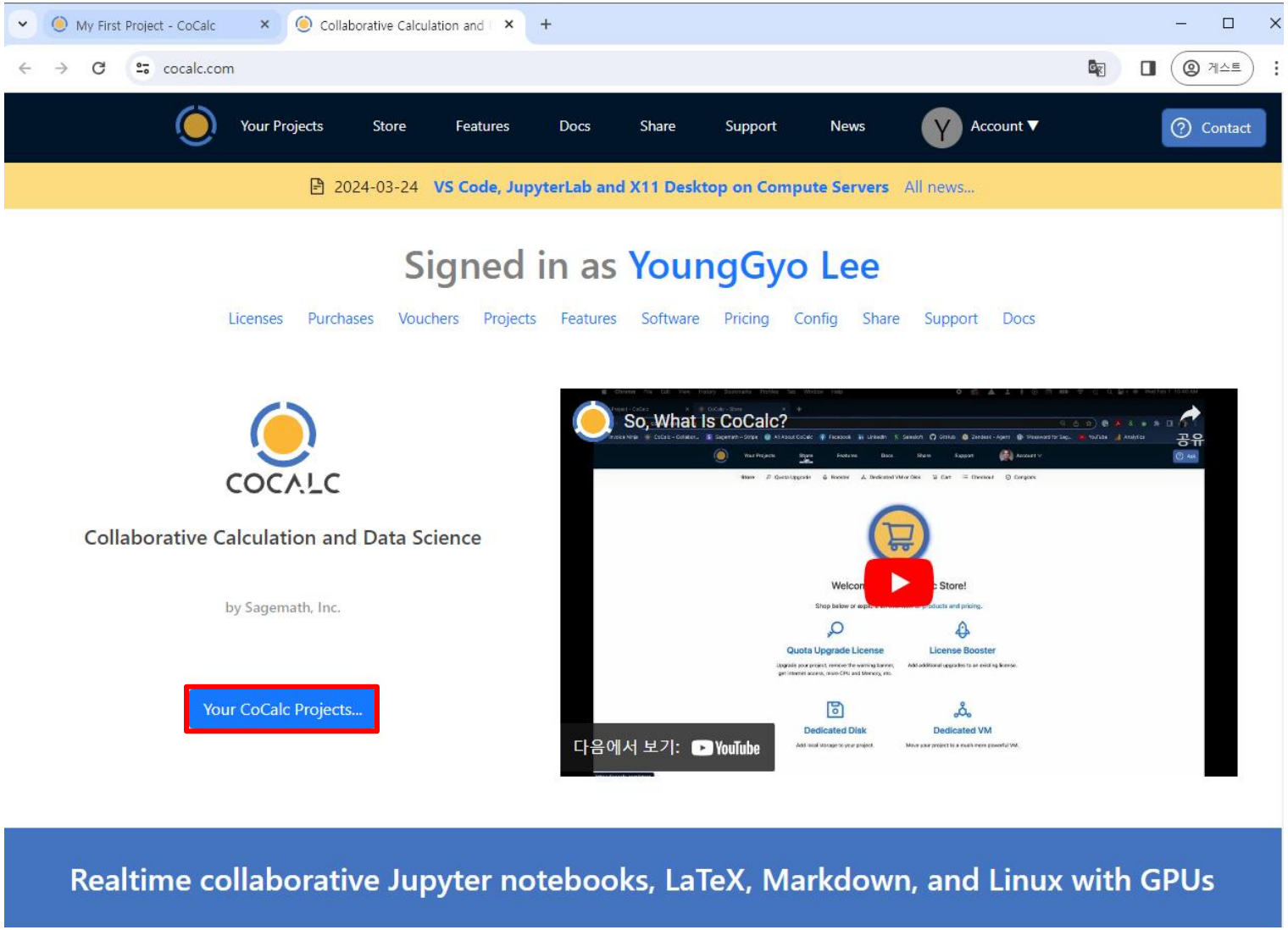


2. Sign up 클릭 후 회원가입 진행(무료)

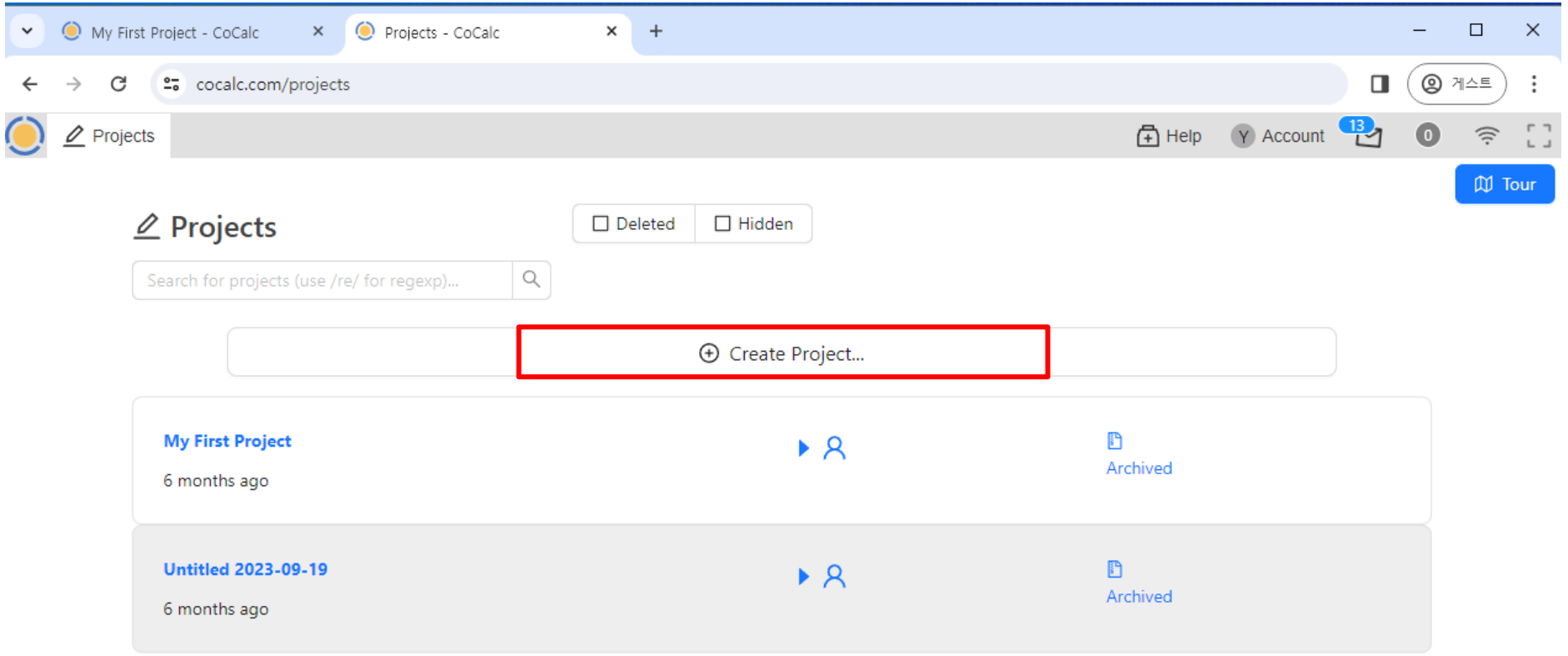




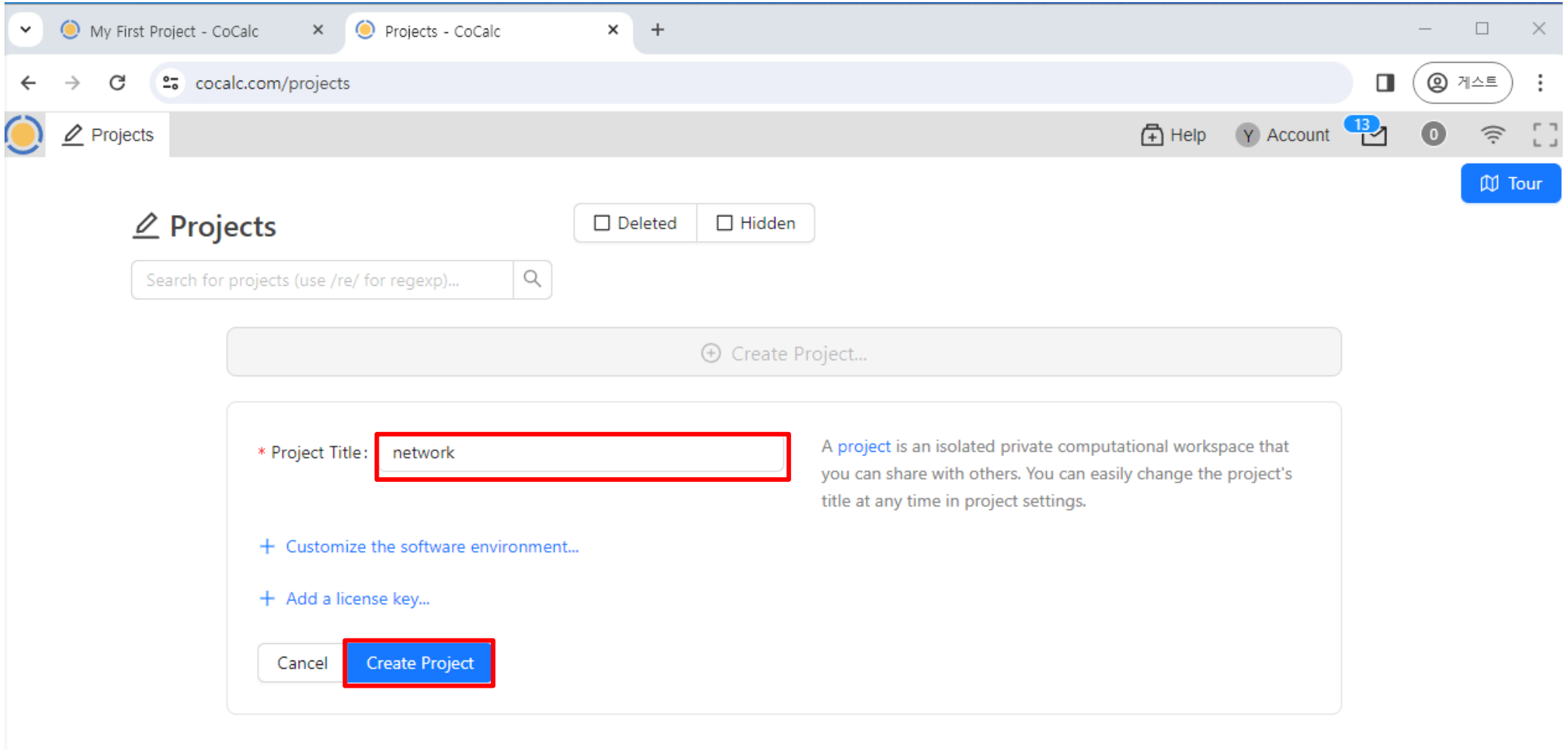
3. Sign in 후 your CoCalc projects 클릭



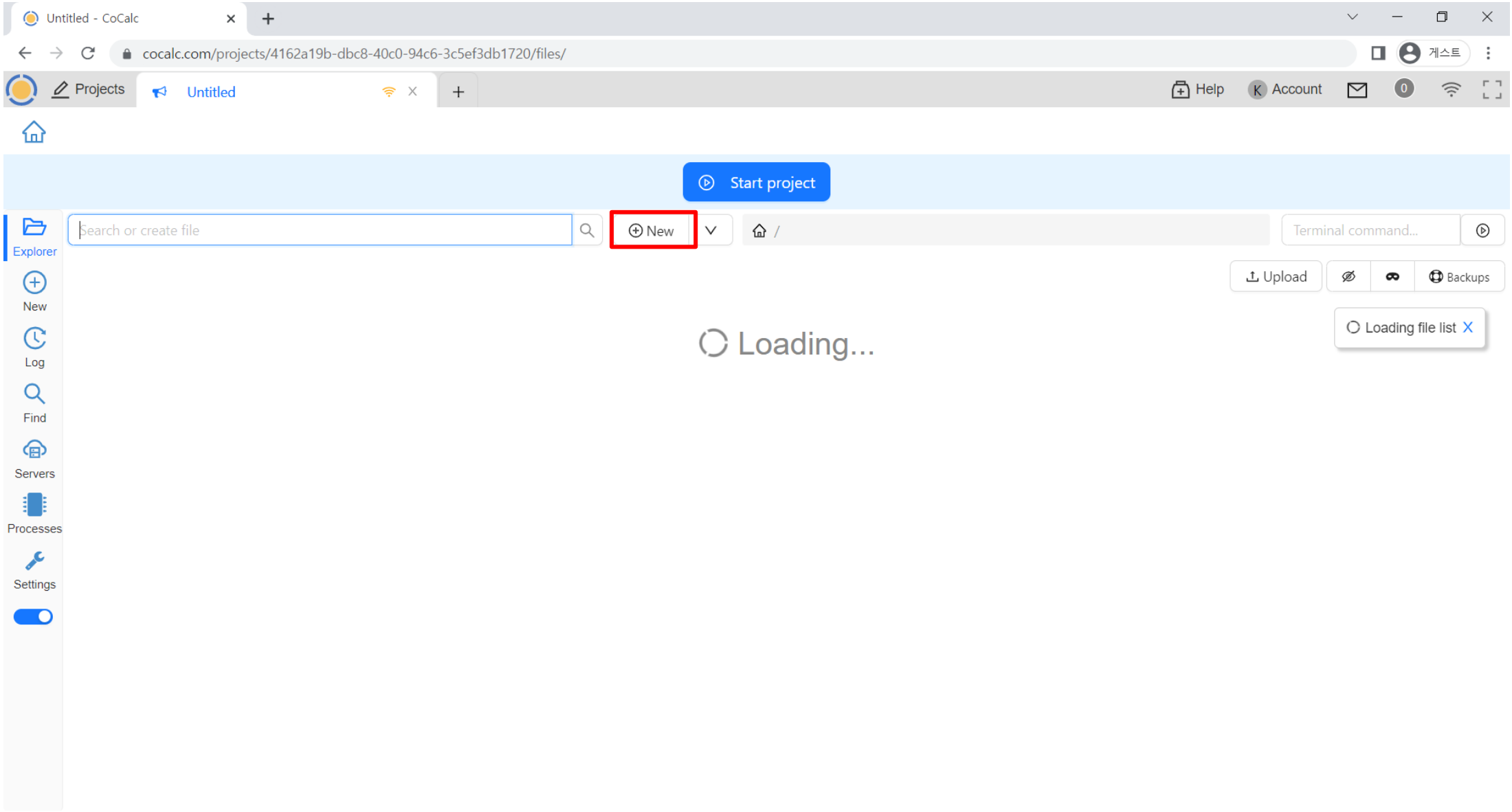
4. Create Project 클릭



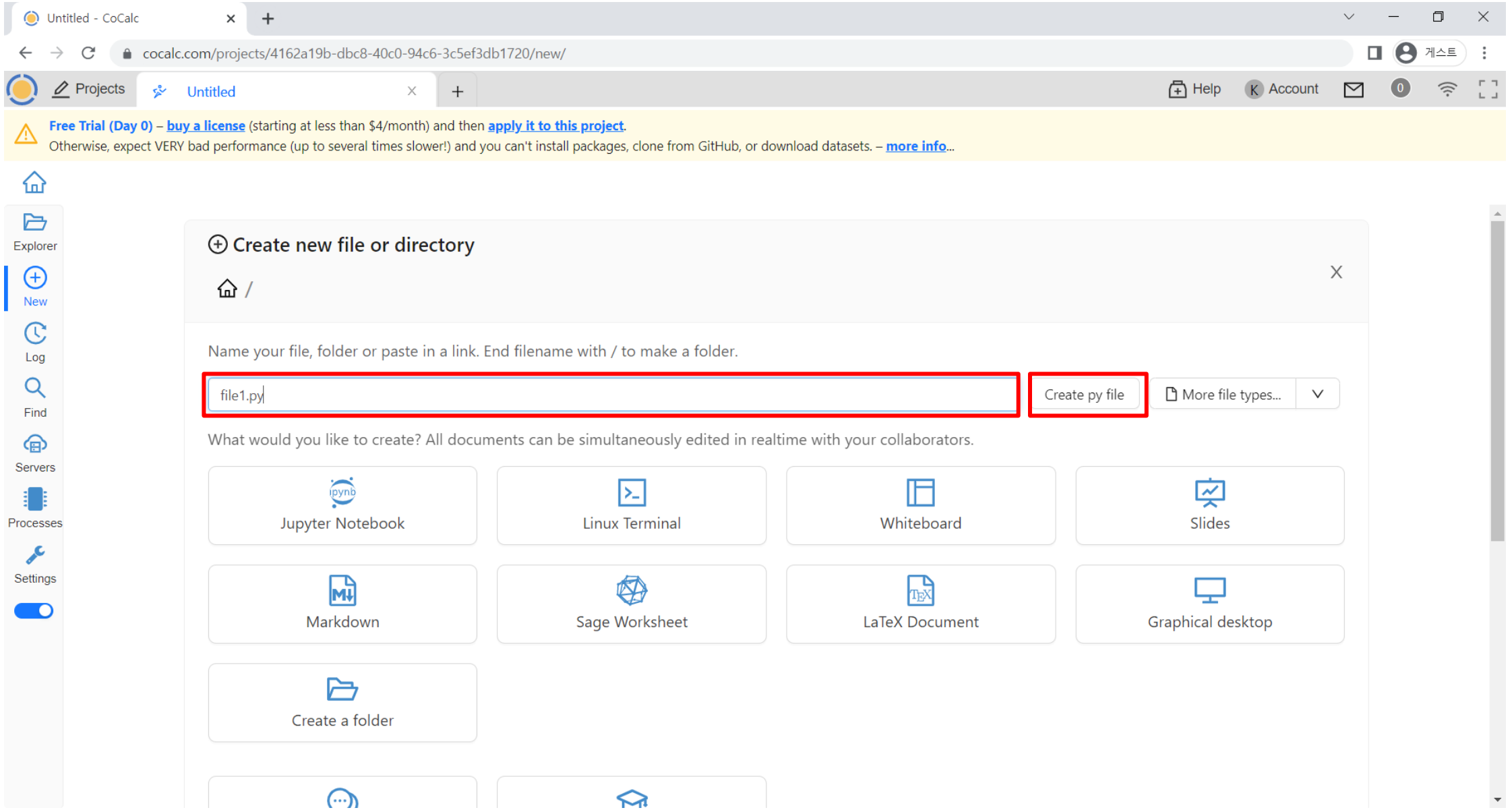
5. Project Title 작성 후 Create Project 클릭



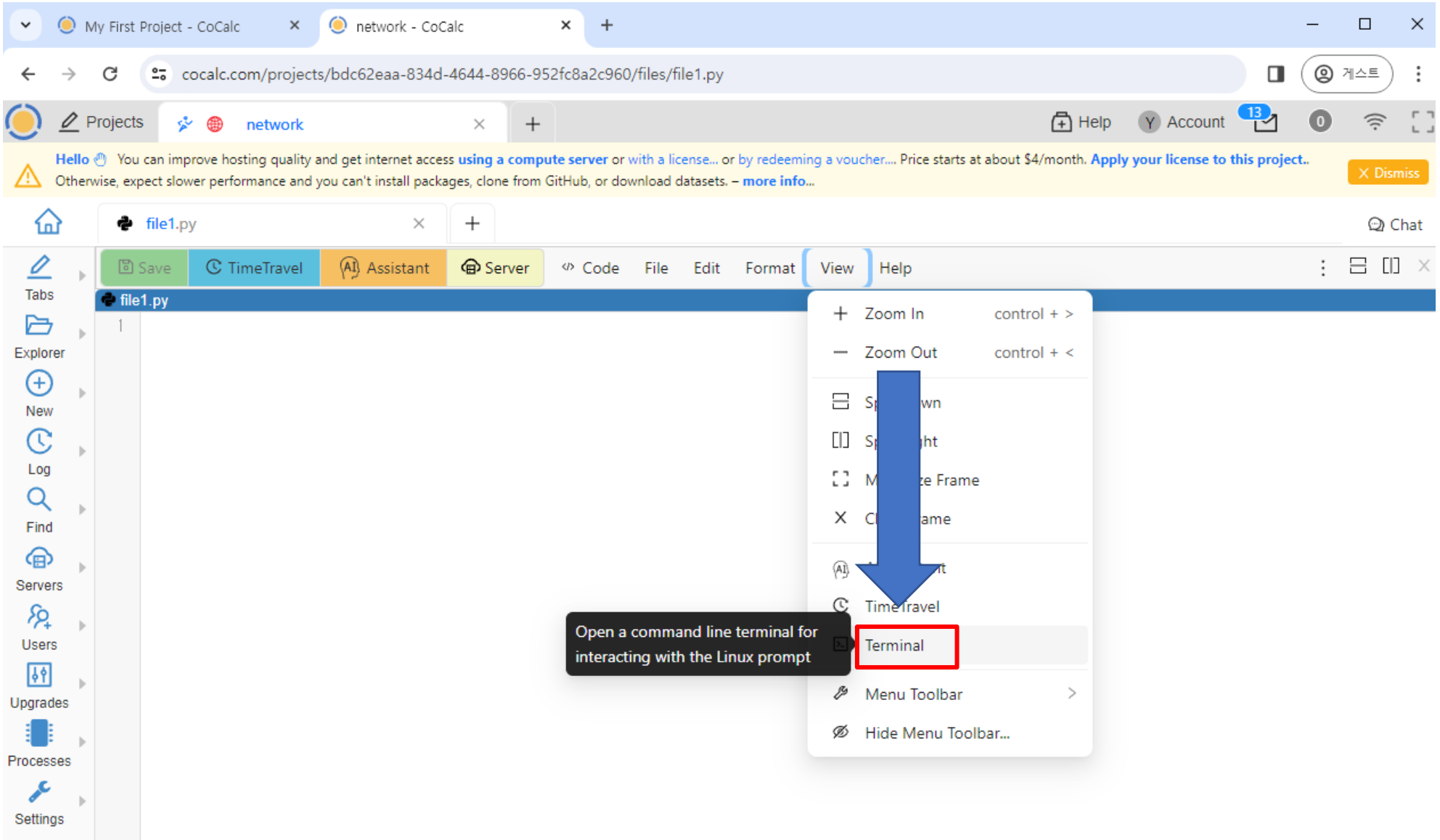
6. New 클릭


Untitled - CoCalc
cocalc.com/projects/4162a19b-dbc8-40c0-94c6-3c5ef3db1720/files/
게스트
Projects
Untitled
Help
Account
0
Start project
Search or create file
New
/
Terminal command...
Explorer
New
Log
Find
Servers
Processes
Settings
Upload
Backups
Loading file list
Loading...

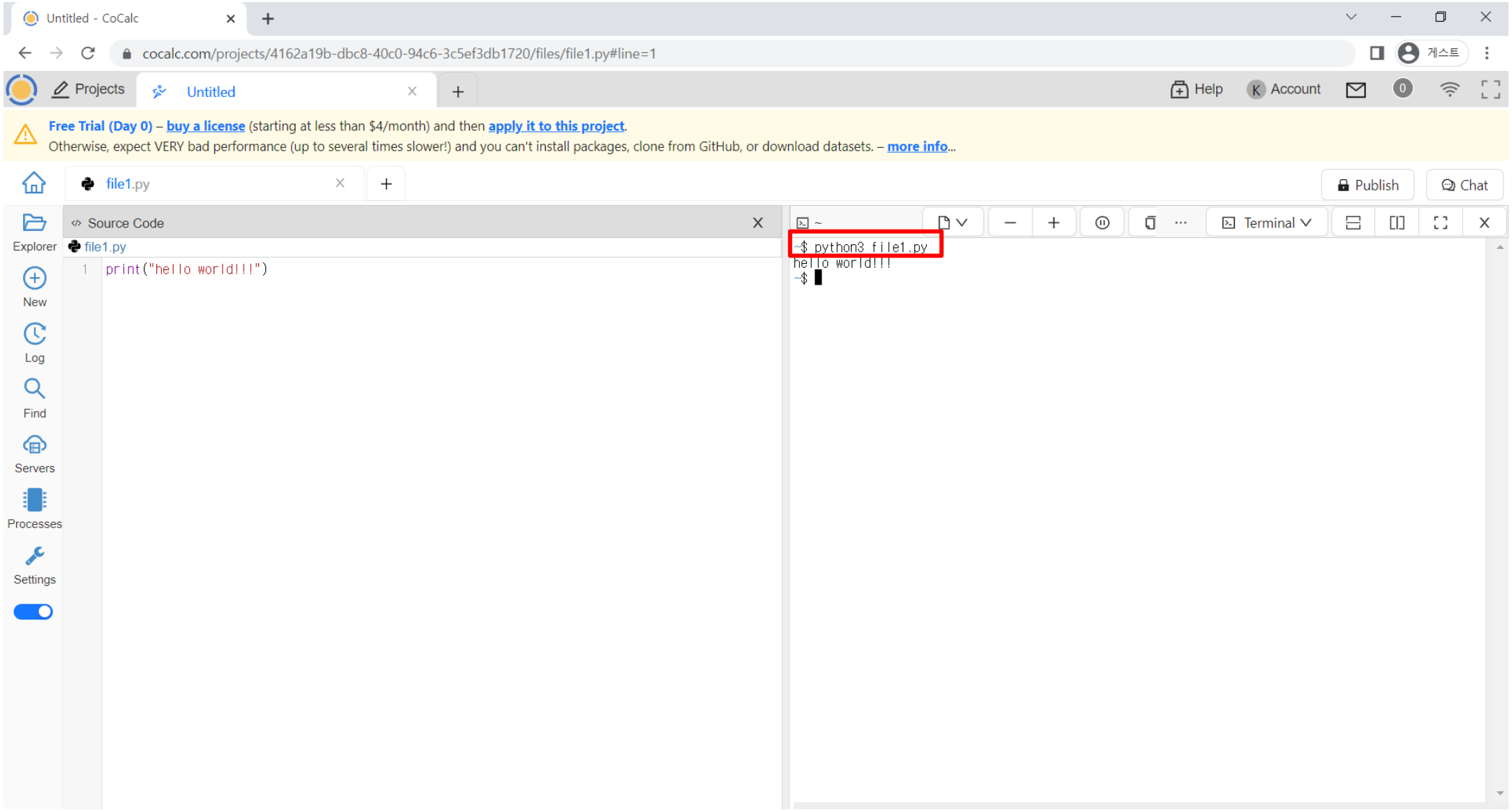
7. 파일이름.py 작성 후 Create py file 클릭



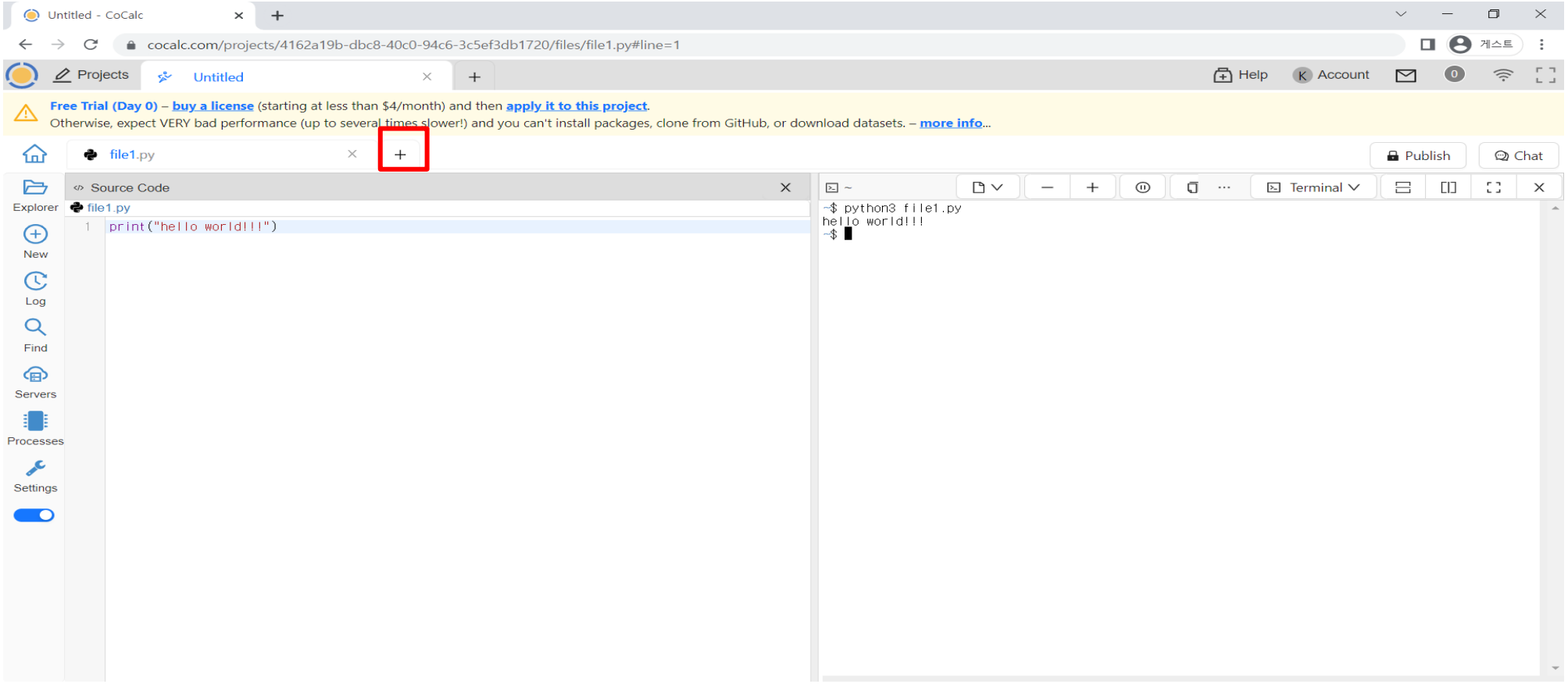
7-1. 파일 실행을 원하는 경우 Terminal 클릭하여 terminal 열기



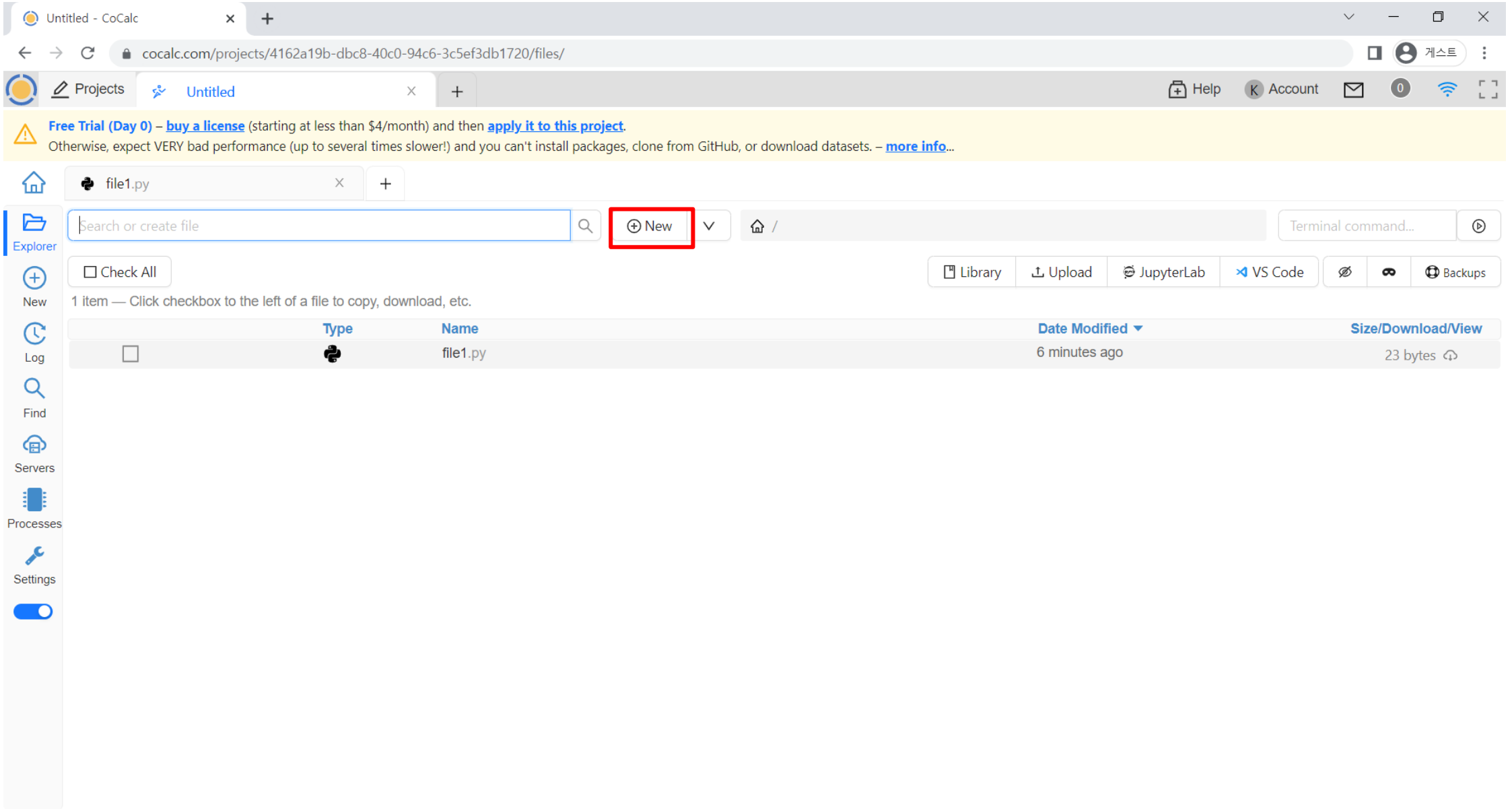
7-2. 터미널에 python3 파일이름.py 입력하여 파일 실행 가능



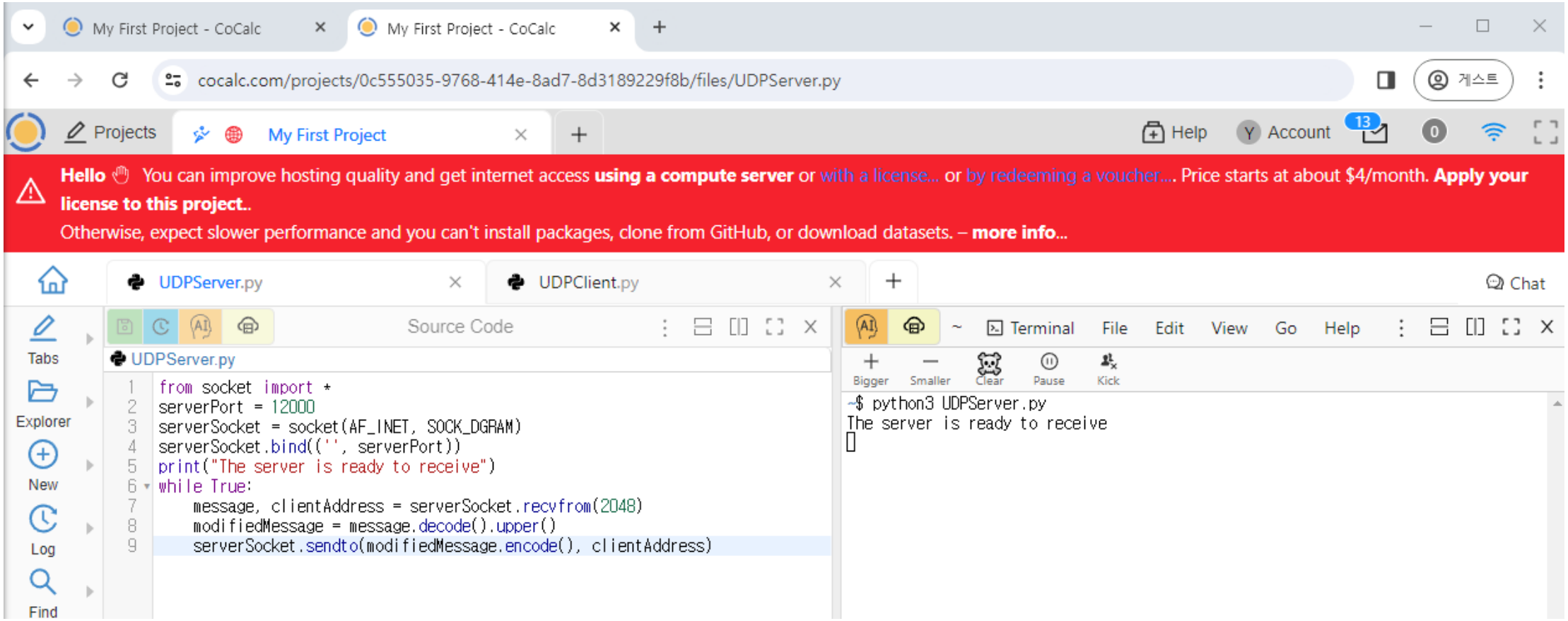
8. + 클릭하여 파일 추가: 서버와 클라이언트와 같이 두 프로세스의 실행을 위해



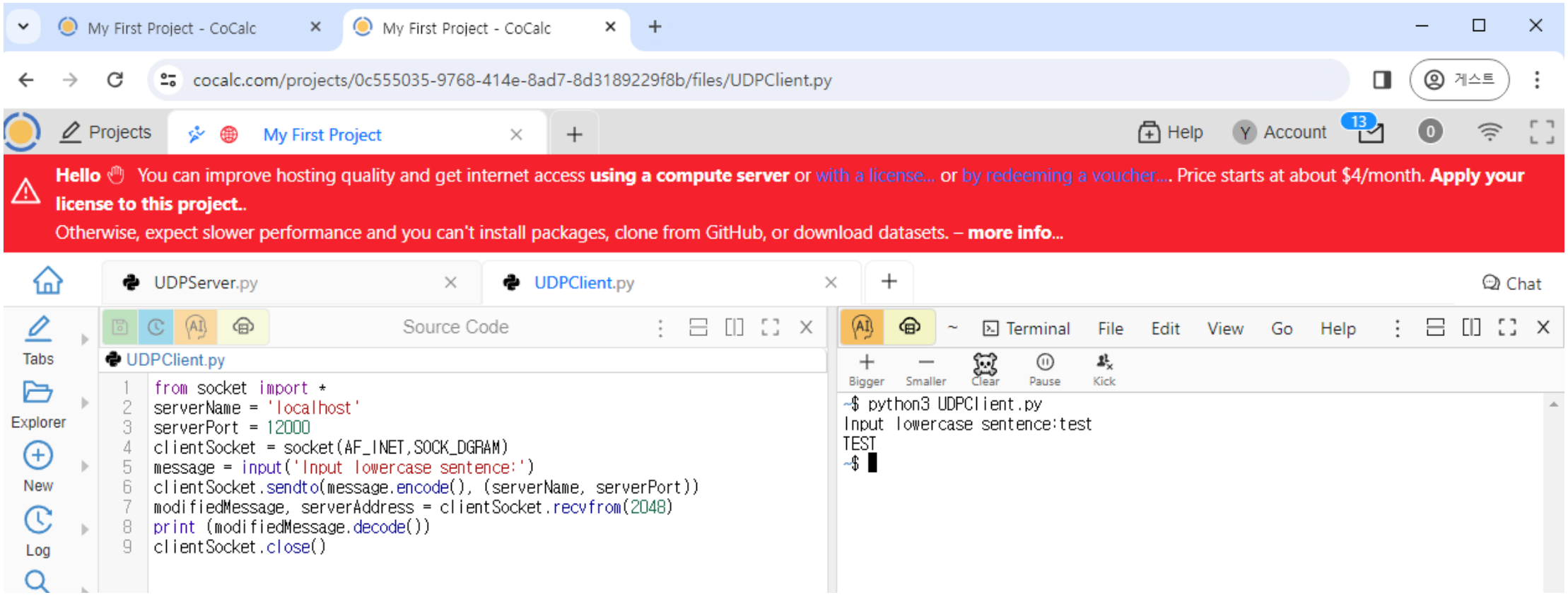
8-1. New 클릭하여 6과정 반복



실습(1) UDPServer 실행 결과 예시

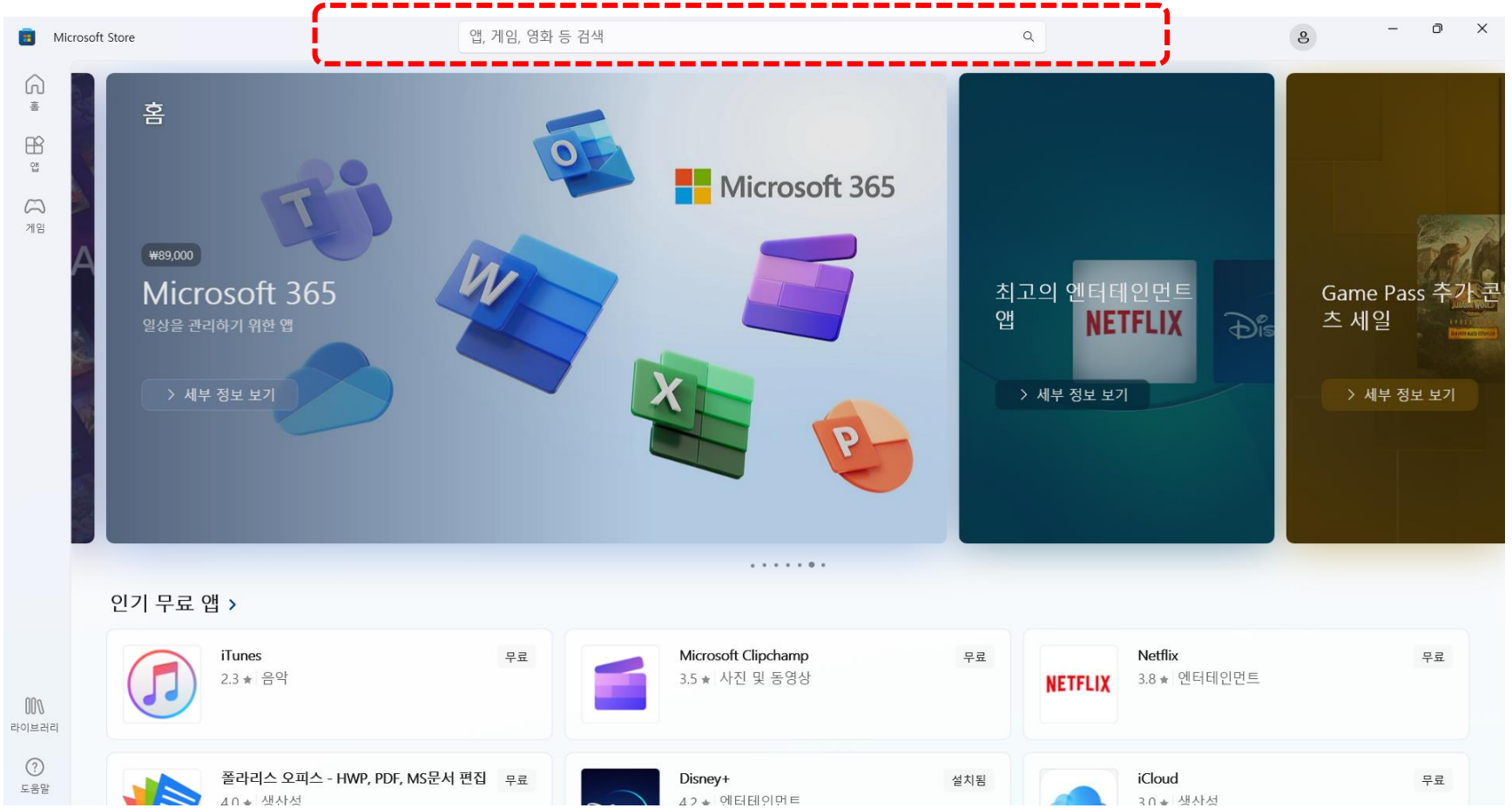


실습(1) UDPClient 실행 결과 예시

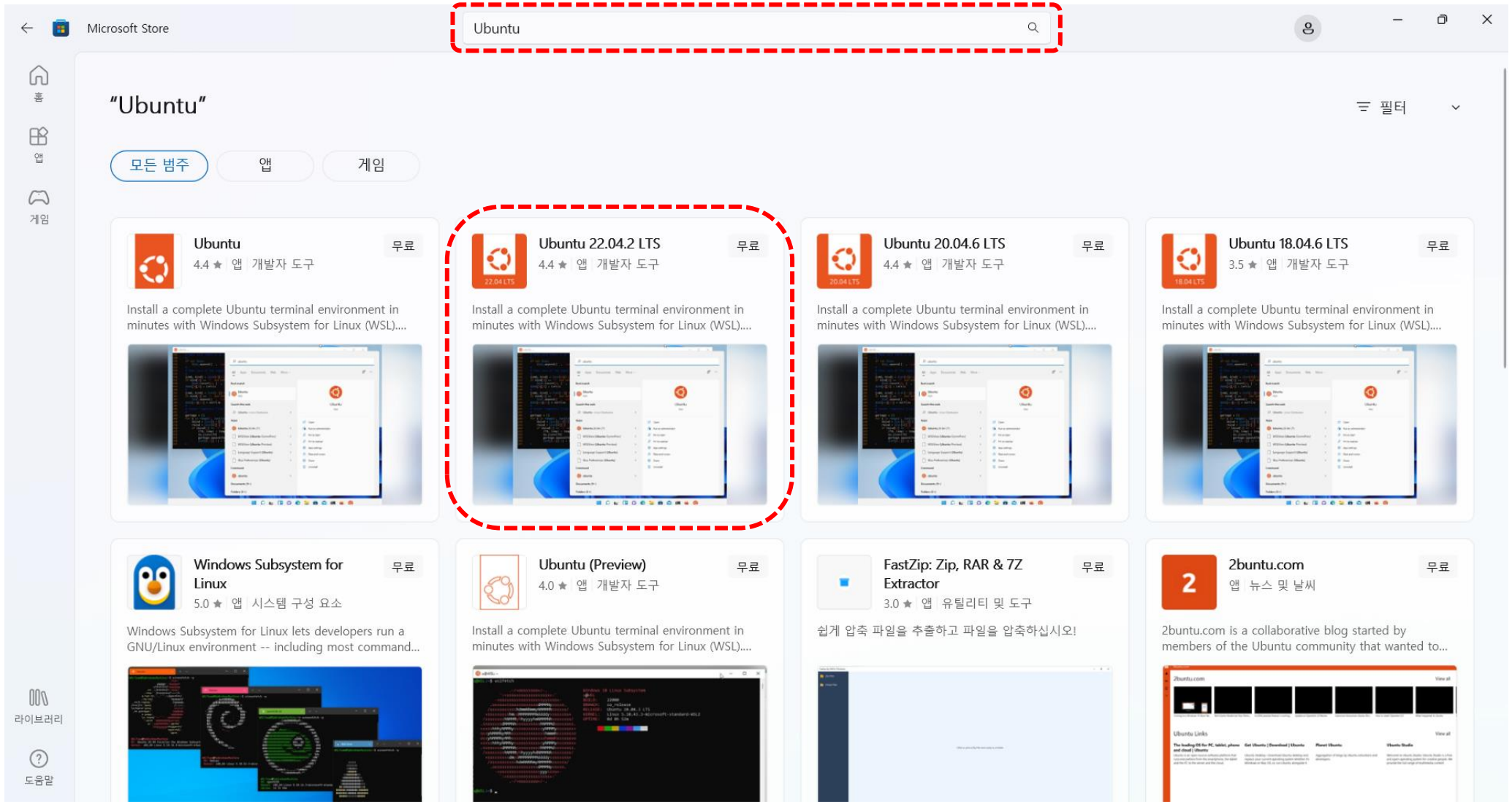


Python socket 프로그래밍 실습 환경 – WSL/Ubuntu

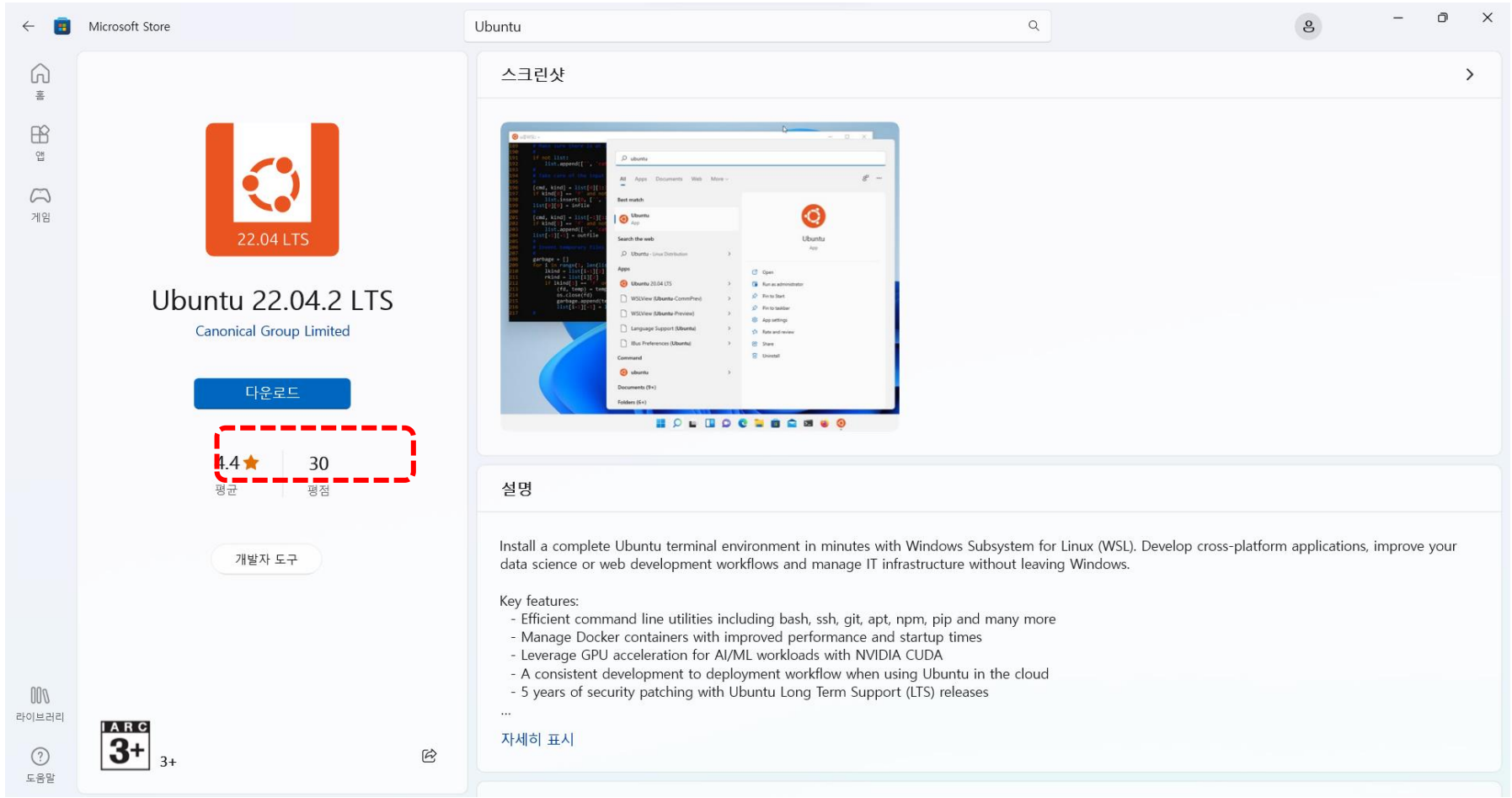
1. Microsoft Store 접속



2. Ubuntu 검색


Microsoft Store
Ubuntu
홈
앱
게임
라이브러리
도움말
"Ubuntu"
필터
모든 범주
앱
게임
Ubuntu
4.4 ★ 앱 개발자 도구
무료
Install a complete Ubuntu terminal environment in minutes with Windows Subsystem for Linux (WSL)....
Ubuntu 22.04.2 LTS
4.4 ★ 앱 개발자 도구
무료
Install a complete Ubuntu terminal environment in minutes with Windows Subsystem for Linux (WSL)....
Ubuntu 20.04.6 LTS
4.4 ★ 앱 개발자 도구
무료
Install a complete Ubuntu terminal environment in minutes with Windows Subsystem for Linux (WSL)....
Ubuntu 18.04.6 LTS
3.5 ★ 앱 개발자 도구
무료
Install a complete Ubuntu terminal environment in minutes with Windows Subsystem for Linux (WSL)....
Windows Subsystem for Linux
5.0 ★ 앱 시스템 구성 요소
무료
Windows Subsystem for Linux lets developers run a GNU/Linux environment -- including most command...
Ubuntu (Preview)
4.0 ★ 앱 개발자 도구
무료
Install a complete Ubuntu terminal environment in minutes with Windows Subsystem for Linux (WSL)....
FastZip: Zip, RAR & 7Z Extractor
3.0 ★ 앱 유틸리티 및 도구
무료
쉽게 압축 파일을 추출하고 파일을 압축하십시오!
2buntu.com
앱 뉴스 및 날씨
무료
2buntu.com is a collaborative blog started by members of the Ubuntu community that wanted to...

3. Ubuntu 22.04 LTS 버전 설치



Ubuntu 설치 (또는 실행) 시 다양한 에러 발생 가능. 대부분 검색으로 문제 해결 가능

[WSL] Error: 0x80370102 해결 (velog.io)

4) 메모장 이용 python 코드 파일 생성 (UDP version)

UDPClient

파일 편집 보기

```
from socket import *
serverName = 'localhost'
serverPort = 12000
clientSocket = socket(AF_INET,SOCK_DGRAM)
message = input('Input lowercase sentence:')
clientSocket.sendto(message.encode(), (serverName, serverPort))
modifiedMessage, serverAddress = clientSocket.recvfrom(2048)
print (modifiedMessage.decode())
clientSocket.close()
```

줄 1, 열 1 | 100% | Windows (CRLF) | UTF-8

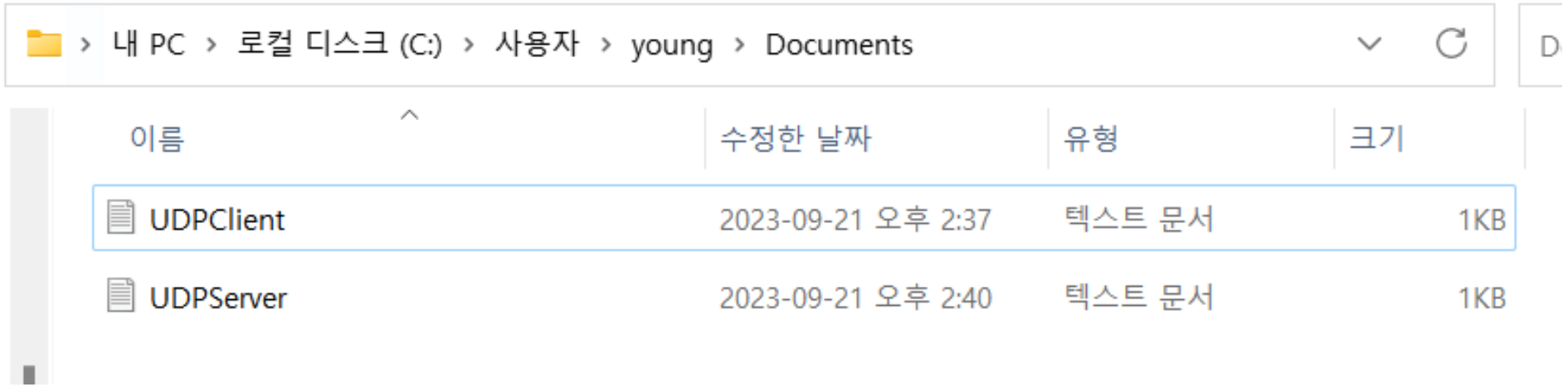
UDPServer

파일 편집 보기

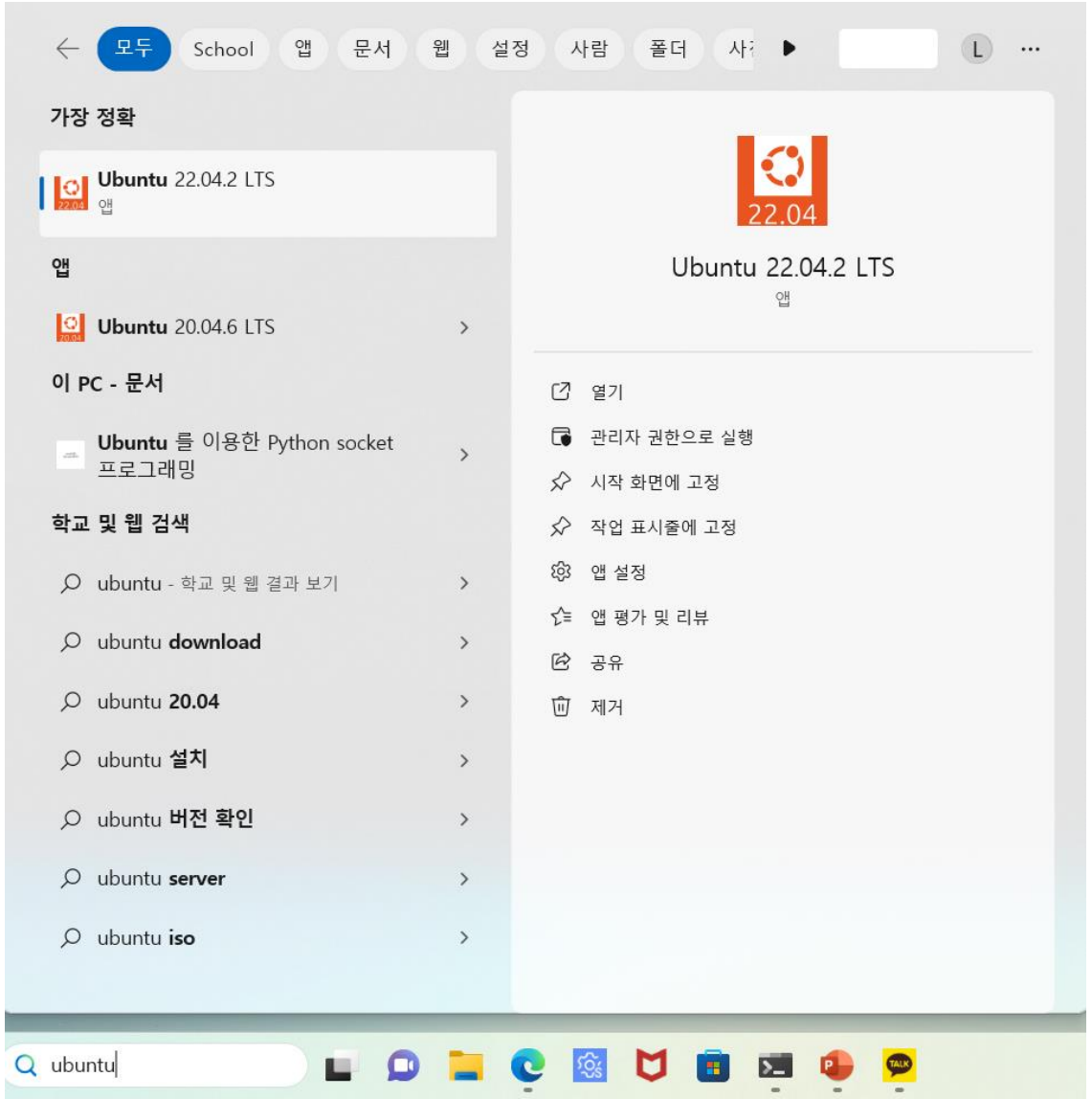
```
from socket import *
serverPort = 12000
serverSocket = socket(AF_INET, SOCK_DGRAM)
serverSocket.bind(('', serverPort))
print("The server is ready to receive")
while True:
    message, clientAddress = serverSocket.recvfrom(2048)
    modifiedMessage = message.decode().upper()
    serverSocket.sendto(modifiedMessage.encode(), clientAddress)
```

줄 9, 열 44 | 100% | Windows (CRLF) | UTF-8

5) File 저장 위치 Documents

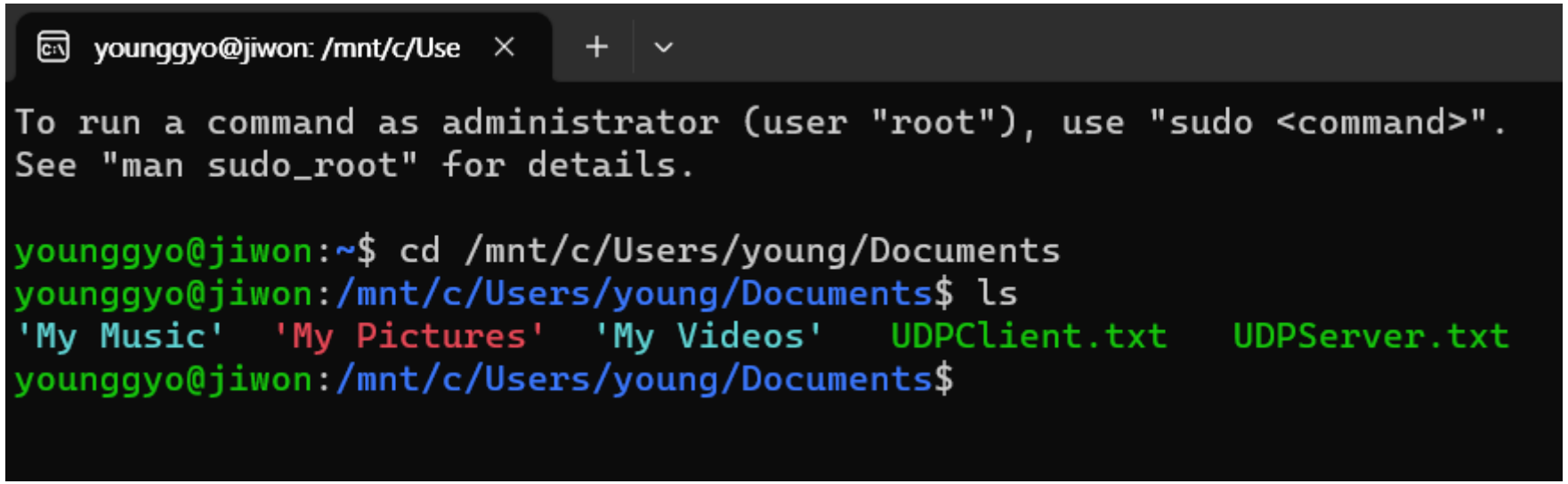


6) Ubuntu 실행


모두
School
앱
문서
웹
설정
사람
폴더
가장 정확
Ubuntu 22.04.2 LTS
앱
앱
Ubuntu 20.04.6 LTS
이 PC - 문서
Ubuntu 를 이용한 Python socket 프로그래밍
학교 및 웹 검색
ubuntu - 학교 및 웹 결과 보기
ubuntu download
ubuntu 20.04
ubuntu 설치
ubuntu 버전 확인
ubuntu server
ubuntu iso
22.04
Ubuntu 22.04.2 LTS
앱
열기
관리자 권한으로 실행
시작 화면에 고정
작업 표시줄에 고정
앱 설정
앱 평가 및 리뷰
공유
제거
ubuntu

7) 파일 위치 접근 : cd 명령어

cd /mnt/c/Users/{사용자명}/Documents



8) Ubuntu에서 코드 실행

!!! Server와 Client 코드에 대해 각각 Ubuntu (terminal/shell) instance 실행해야 함 !!!