

Python socket 프로그래밍 실습 및 실습 환경 구축

컴퓨터네트워크

실습(1)

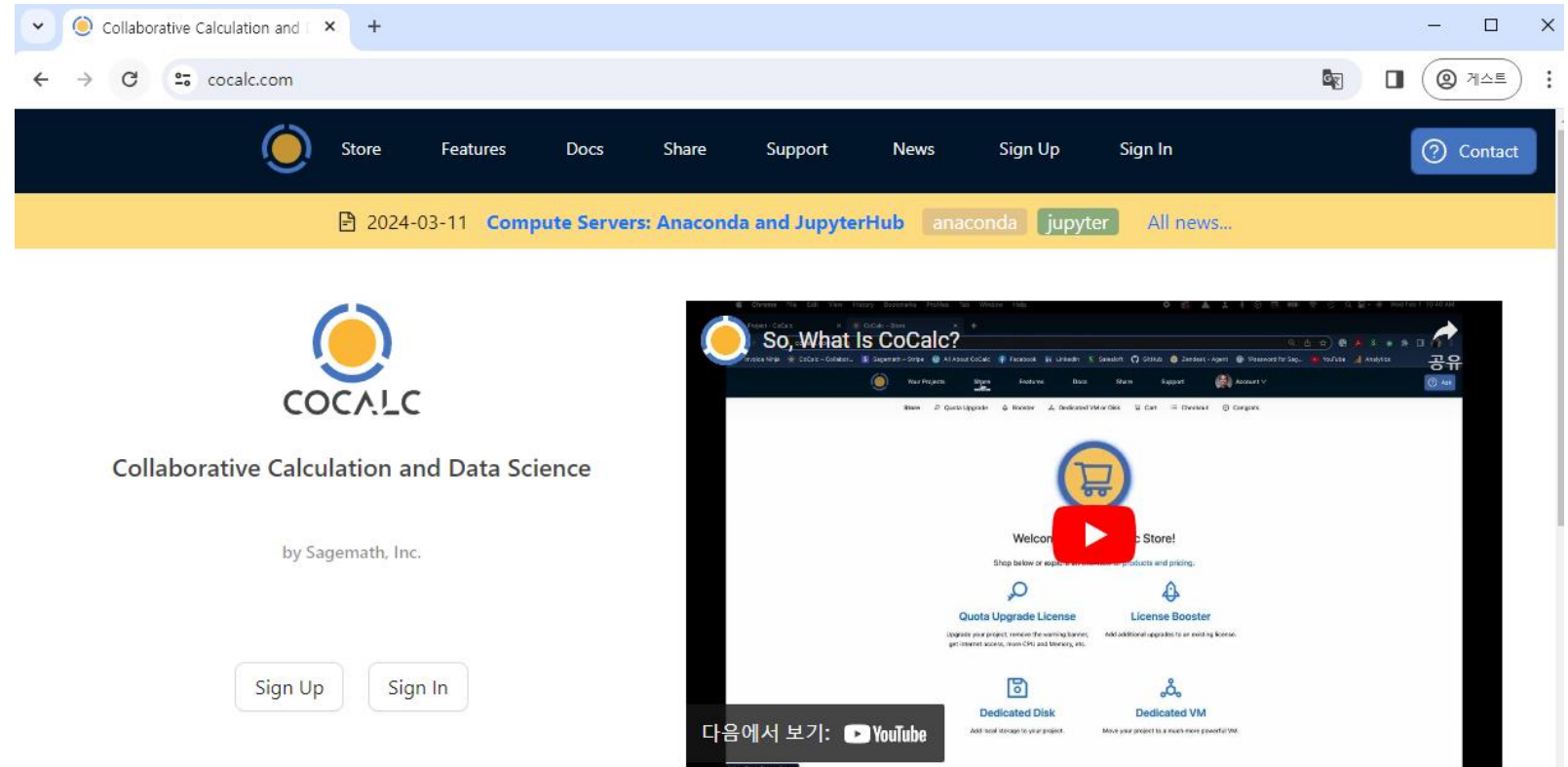
- server.py 파일 생성, "UDPServer" 코드 입력
 - 터미널에서 "python server.py"와 같이 실행
 - client.py 파일 생성, "UDPClient" 코드 입력
 - 터미널에서 "python client.py"와 같이 실행
 - 실행 후 각각 server.py, client.py 탭 터미널에서 출력 확인
 - TCP, UDP 각각의 경우에 대해 실행해 보기
 - serverName = '127.0.0.1'
-
- TCPServer, TCPClient 코드에 대해서도 위와 같이 진행
 - python 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

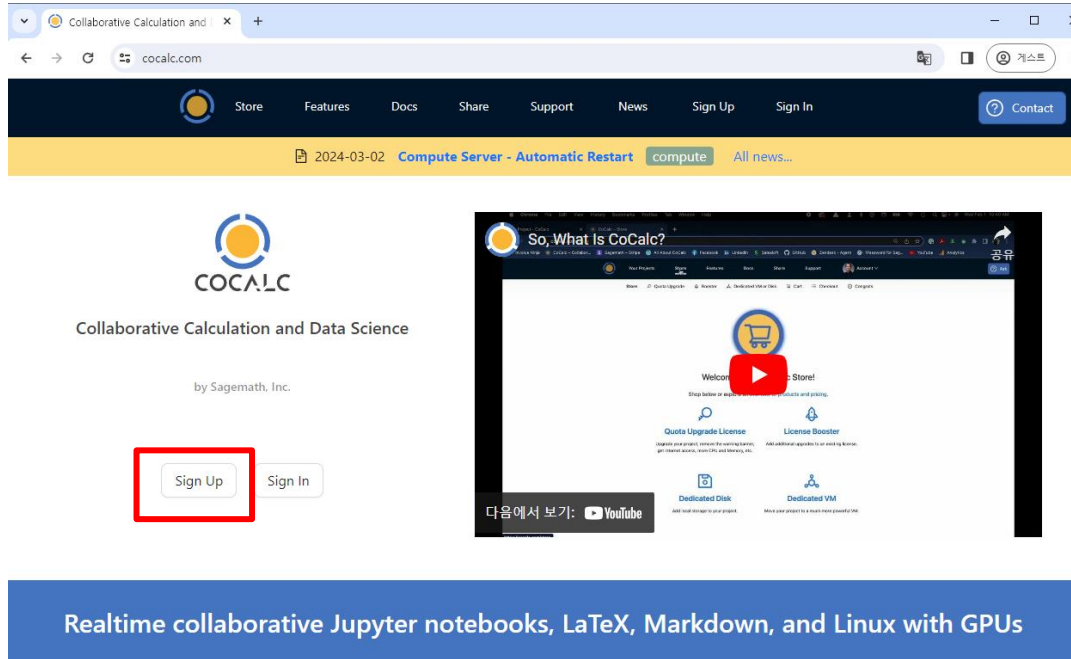
Python socket 프로그래밍 실습 환경 – cocalc.com

1. cocalc.com 접속



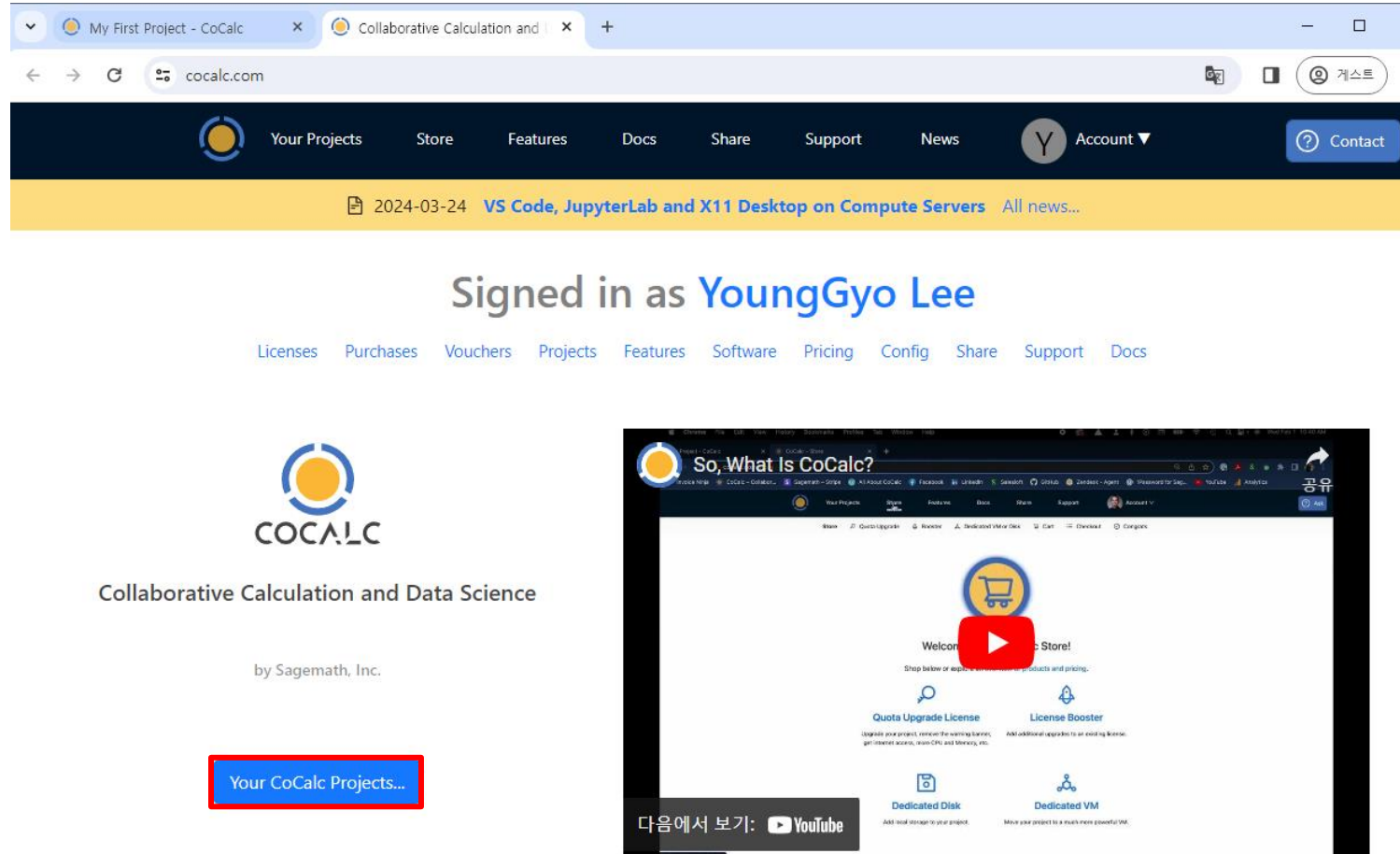
Realtime collaborative Jupyter notebooks, LaTeX, Markdown, and Linux with GPUs

2. Sign up 클릭 후 회원가입 진행(무료)



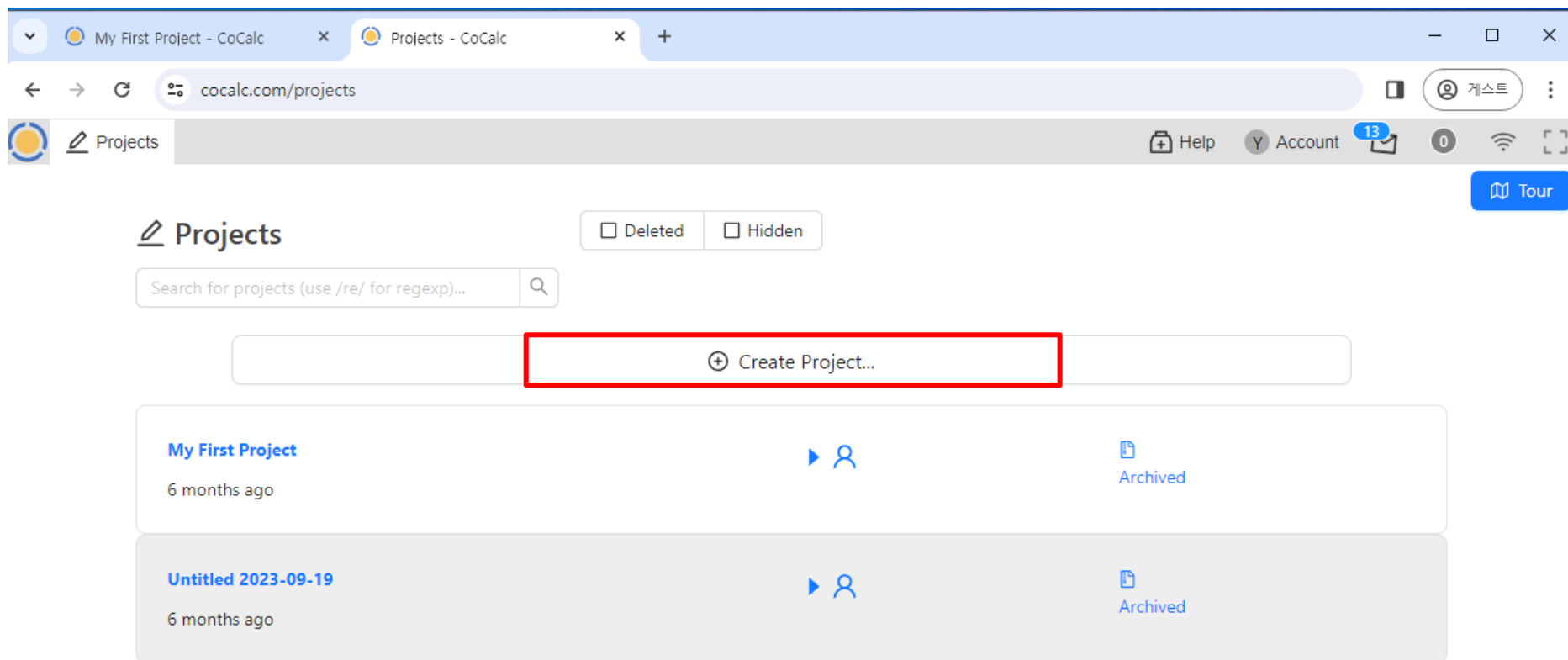
The screenshot shows the CoCalc sign-up form. The navigation bar is identical to the homepage. The main heading is 'Create a free account with CoCalc' with the subtext 'Start collaborating for free today.' Below this, a note states: 'By creating an account, you agree to the Terms of Service. Sign up using either your email address or a single sign-on provider.' There are icons for social media sign-in (Facebook, GitHub, Google, Twitter) and a link for 'Institutional Single Sign-On'. The form includes input fields for 'Email address', 'Password', 'First name (Given name)', and 'Last name (Family name)'. A 'How will you sign in?' button is at the bottom of the form. At the very bottom, a link says 'Already have an account? Sign In'.

3. Sign in 후 your CoCalc projects 클릭

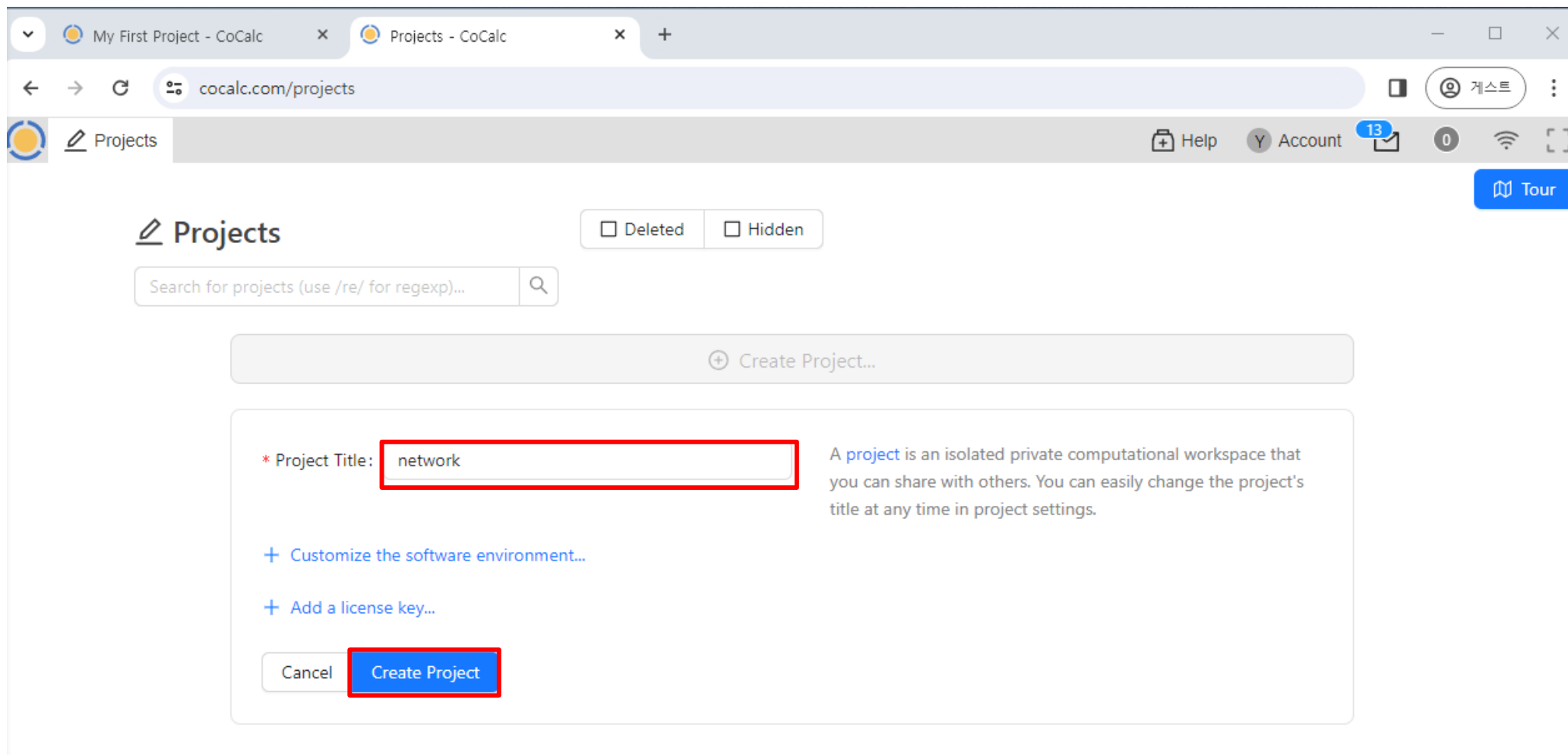


Realtime collaborative Jupyter notebooks, LaTeX, Markdown, and Linux with GPUs

4. Create Project 클릭



5. Project Title 작성 후 Create Project 클릭



The screenshot shows the CoCalc web interface. The browser tabs include 'My First Project - CoCalc' and 'Projects - CoCalc'. The address bar shows 'cocalc.com/projects'. The page header includes a 'Projects' link, 'Help', 'Account', and a 'Tour' button. The main content area is titled 'Projects' and includes a search bar and filters for 'Deleted' and 'Hidden' projects. A 'Create Project...' button is visible. Below it, a form for creating a new project is shown. The 'Project Title' field is filled with 'network' and is highlighted with a red box. To the right of the form, a description states: 'A project is an isolated private computational workspace that you can share with others. You can easily change the project's title at any time in project settings.' Below the form, there are two buttons: 'Cancel' and 'Create Project', with the 'Create Project' button highlighted by a red box.

My First Project - CoCalc Projects - CoCalc

cocalc.com/projects

Projects

Help Account 13 0

Tour

Projects

Deleted Hidden

Search for projects (use /re/ for regexp)...

Create Project...

* Project Title: network

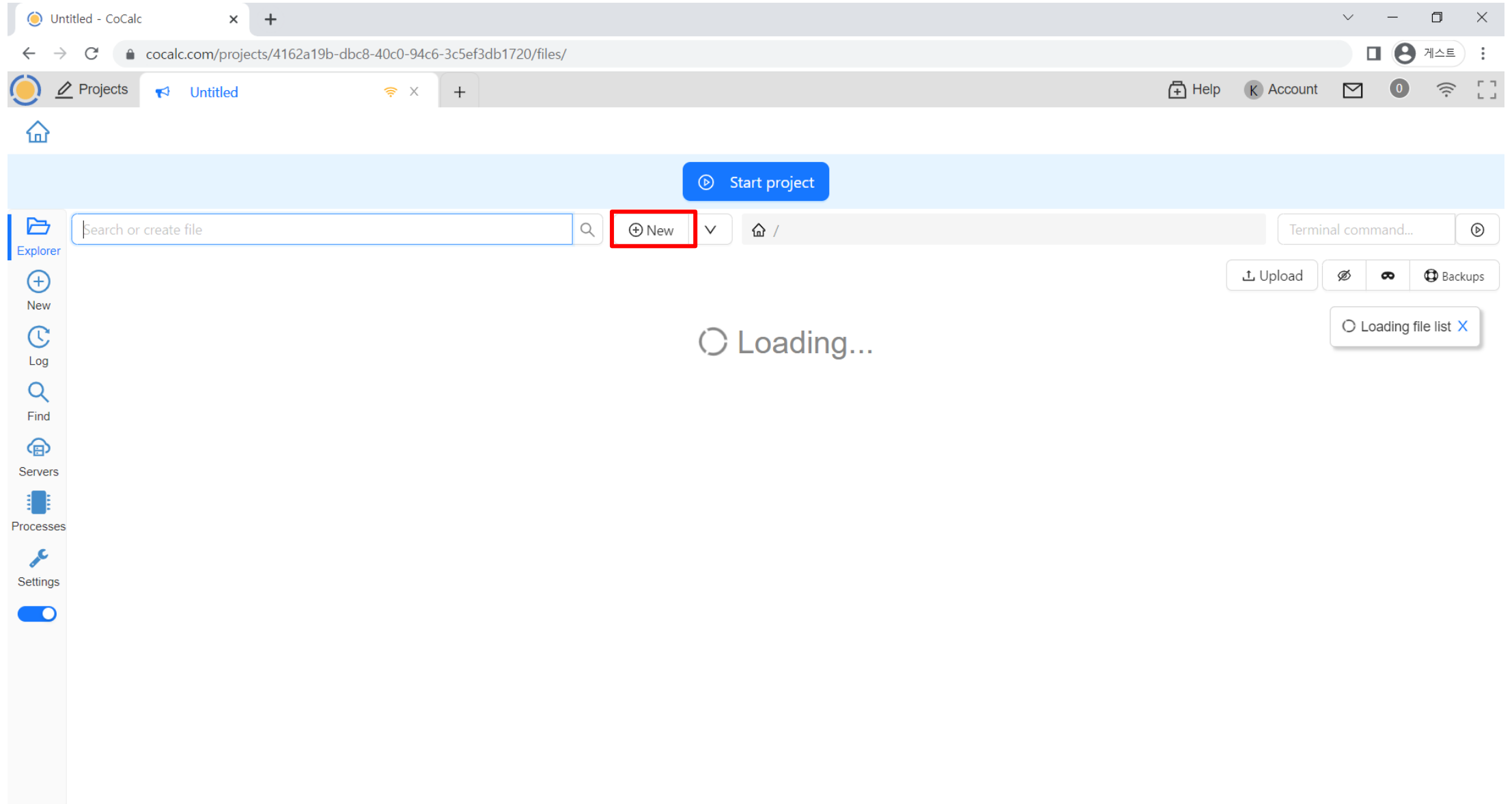
A project is an isolated private computational workspace that you can share with others. You can easily change the project's title at any time in project settings.

+ Customize the software environment...

+ Add a license key...

Cancel Create Project

6. New 클릭



7. 파일이름.py 작성 후 Create py file 클릭

The screenshot shows the CoCalc web interface. At the top, the browser tab is 'Untitled - CoCalc' and the address bar shows 'cocalc.com/projects/4162a19b-dbc8-40c0-94c6-3c5ef3db1720/new/'. Below the browser, the CoCalc interface has a top bar with 'Projects' and 'Untitled' tabs, and a right sidebar with 'Help', 'Account', and a notification icon. A yellow banner below the top bar reads: 'Free Trial (Day 0) - buy a license (starting at less than \$4/month) and then apply it to this project. Otherwise, expect VERY bad performance (up to several times slower!) and you can't install packages, clone from GitHub, or download datasets. - more info...'. On the left, a vertical sidebar contains icons for Explorer, New, Log, Find, Servers, Processes, and Settings. The main area displays a 'Create new file or directory' dialog. The dialog has a title bar with a close button. Below the title bar, it says 'Name your file, folder or paste in a link. End filename with / to make a folder.' The input field contains 'file1.py'. To the right of the input field is a 'Create py file' button, which is highlighted with a red box. Next to it is a 'More file types...' button with a dropdown arrow. Below the input field, it says 'What would you like to create? All documents can be simultaneously edited in realtime with your collaborators.' There are eight buttons arranged in a grid: 'Jupyter Notebook', 'Linux Terminal', 'Whiteboard', 'Slides', 'Markdown', 'Sage Worksheet', 'LaTeX Document', and 'Graphical desktop'. At the bottom, there is a 'Create a folder' button.

Untitled - CoCalc x +

cocalc.com/projects/4162a19b-dbc8-40c0-94c6-3c5ef3db1720/new/

Projects Untitled x +

Help Account 0

Free Trial (Day 0) - [buy a license](#) (starting at less than \$4/month) and then [apply it to this project](#). Otherwise, expect VERY bad performance (up to several times slower!) and you can't install packages, clone from GitHub, or download datasets. - [more info...](#)

Explorer

New

Log

Find

Servers

Processes

Settings

Create new file or directory X

/

Name your file, folder or paste in a link. End filename with / to make a folder.

file1.py Create py file More file types... v

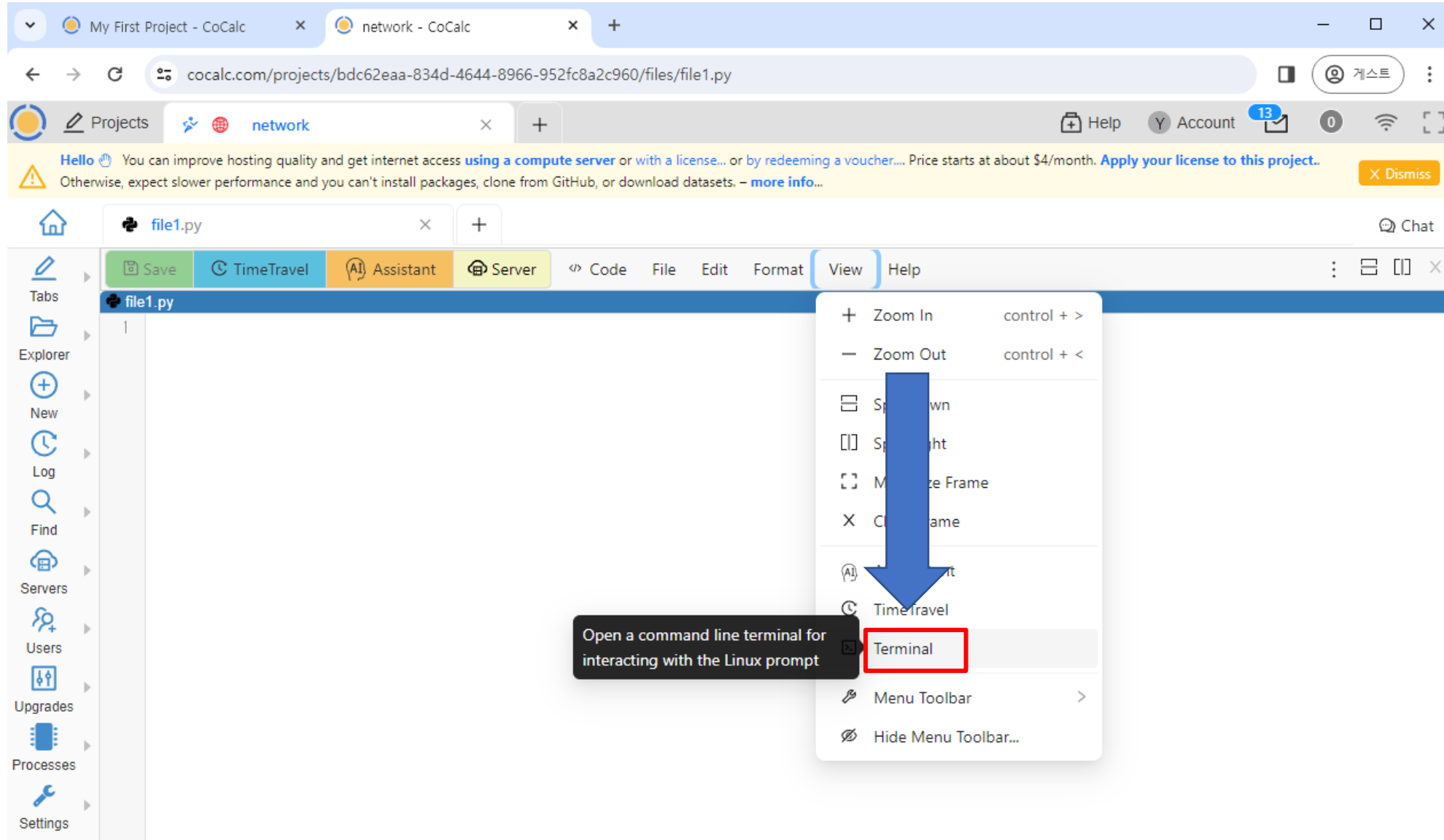
What would you like to create? All documents can be simultaneously edited in realtime with your collaborators.

Jupyter Notebook Linux Terminal Whiteboard Slides

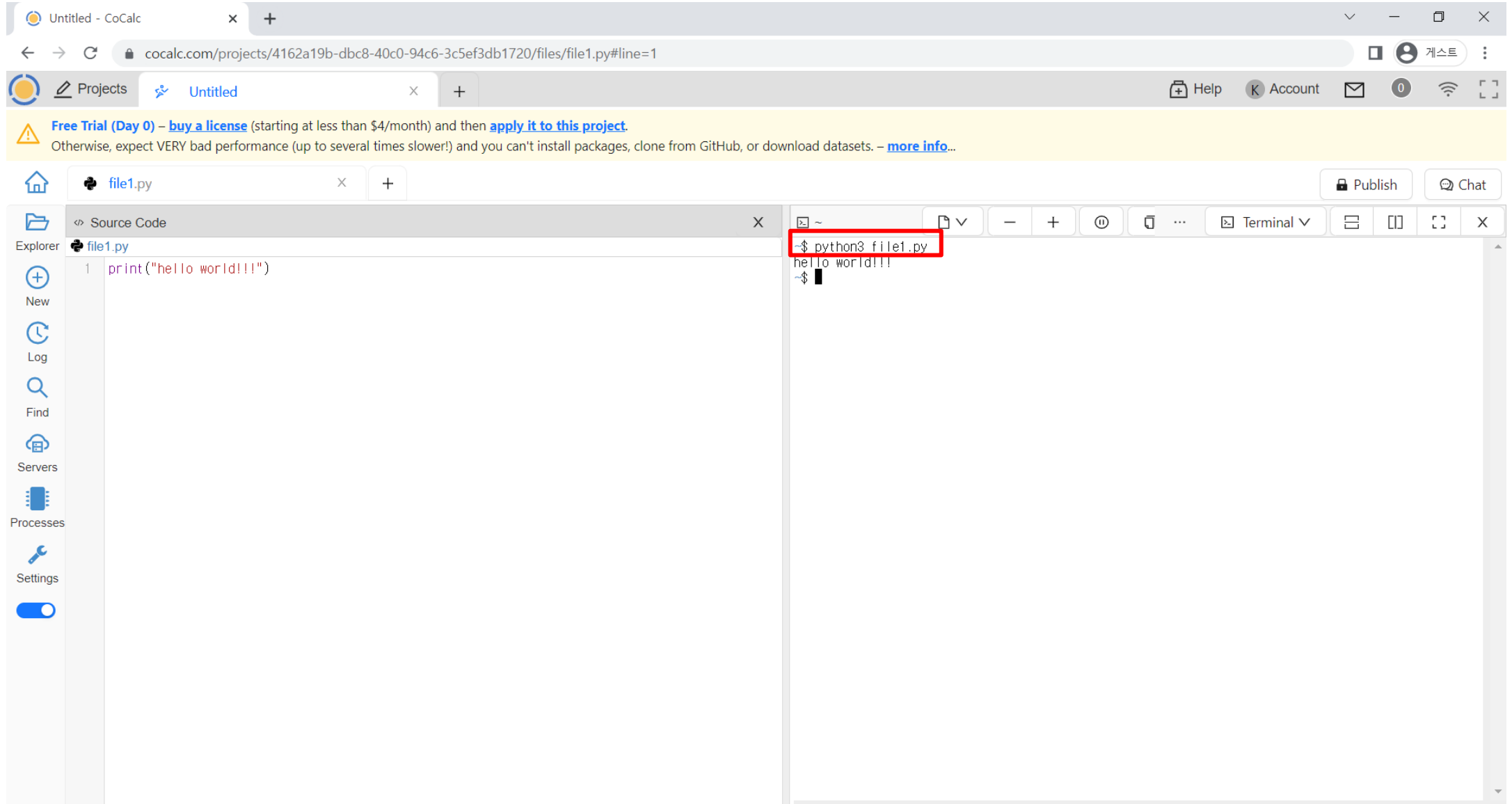
Markdown Sage Worksheet LaTeX Document Graphical desktop

Create a folder

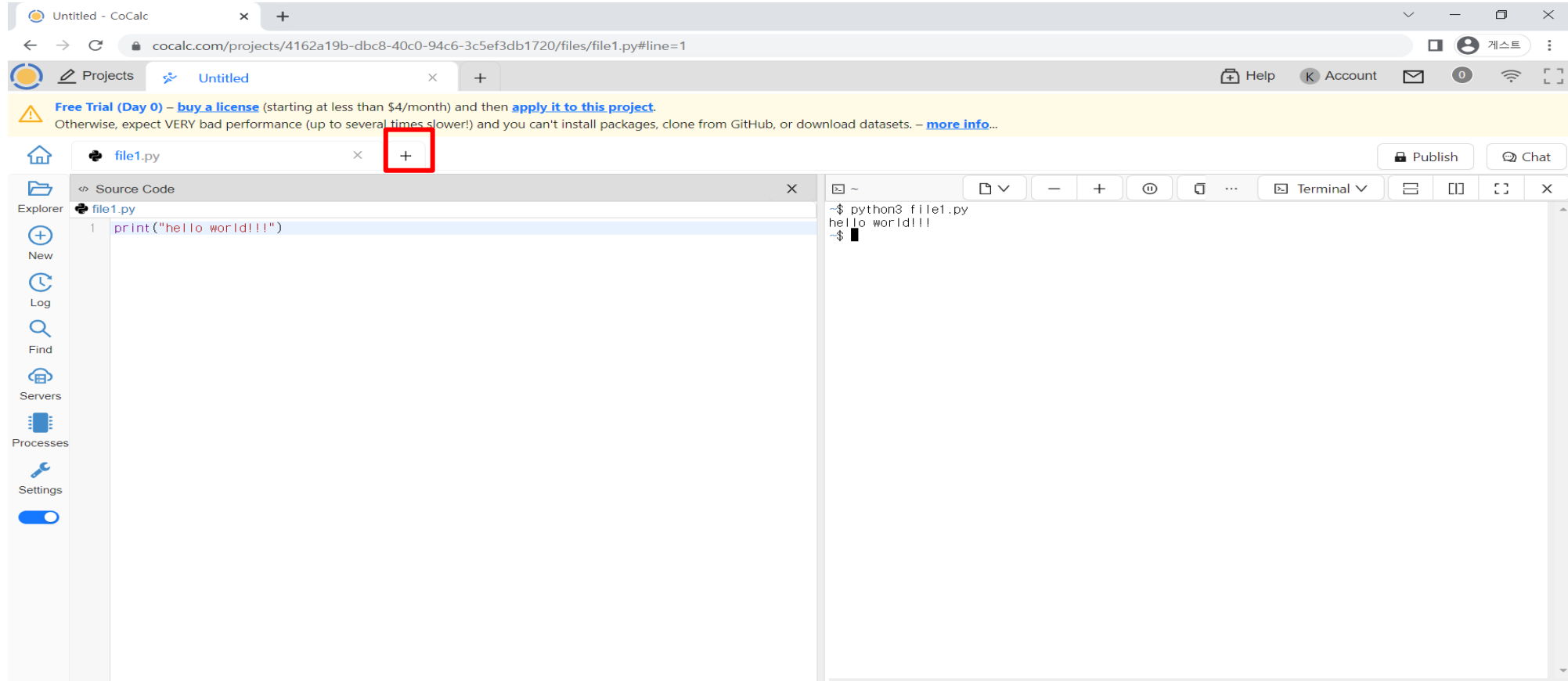
7-1. 파일 실행을 원하는 경우 Terminal 클릭하여 terminal 열기



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

The screenshot shows the CoCalc web interface. At the top, there's a browser tab titled 'Untitled - CoCalc' and a URL bar showing 'cocalc.com/projects/4162a19b-dbc8-40c0-94c6-3c5ef3db1720/files/'. Below the browser bar, there's a navigation bar with 'Projects' and 'Untitled' tabs. A yellow banner at the top of the workspace area contains a warning icon and text about the free trial. The main workspace area has a search bar, a '+ New' button (highlighted with a red box), and a file list. The file list shows one item, 'file1.py', with a checkbox, type icon, name, date modified, and size/download/view options. The left sidebar contains navigation icons for Explorer, New, Log, Find, Servers, Processes, and Settings.

Untitled - CoCalc

cocalc.com/projects/4162a19b-dbc8-40c0-94c6-3c5ef3db1720/files/

Projects Untitled

Help Account

Free Trial (Day 0) – [buy a license](#) (starting at less than \$4/month) and then [apply it to this project](#).
Otherwise, expect VERY bad performance (up to several times slower!) and you can't install packages, clone from GitHub, or download datasets. – [more info...](#)

file1.py

Search or create file

+ New

Check All

1 item — Click checkbox to the left of a file to copy, download, etc.

	Type	Name	Date Modified	Size/Download/View
<input type="checkbox"/>		file1.py	6 minutes ago	23 bytes

Library Upload JupyterLab VS Code Backups

Explorer

New

Log

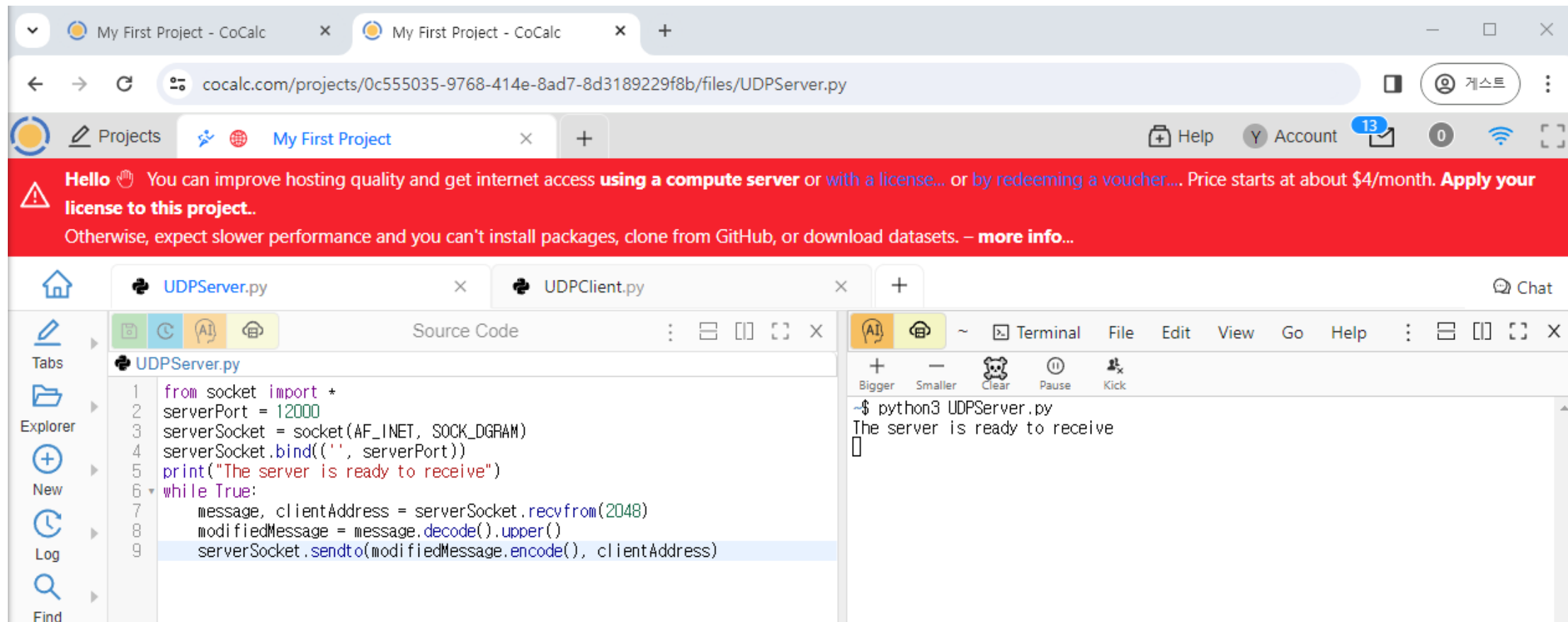
Find

Servers

Processes

Settings

실습(1) UDPServer 실행 결과 예시



The screenshot displays a web browser window with two tabs, both titled "My First Project - CoCalc". The address bar shows the URL: `cocalc.com/projects/0c555035-9768-414e-8ad7-8d3189229f8b/files/UDPServer.py`. Below the browser window, a red banner contains a message: "Hello 🖐️ You can improve hosting quality and get internet access using a compute server or with a license... or by redeeming a voucher... Price starts at about \$4/month. Apply your license to this project.. Otherwise, expect slower performance and you can't install packages, clone from GitHub, or download datasets. – more info...".

The main interface is a code editor with two tabs: "UDPServer.py" and "UDPCClient.py". The "UDPServer.py" tab is active, showing the following Python code:

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

On the right side of the editor, there is a terminal window. The terminal shows the command `python3 UDPServer.py` being executed, followed by the output `The server is ready to receive` and a blank line.

실습(1) UDPClient 실행 결과 예시

The screenshot displays a web-based IDE interface with two tabs: 'UDPServer.py' and 'UDPClient.py'. The 'UDPClient.py' tab is active, showing the following Python code:

```
1 from socket import *
2 serverName = 'localhost'
3 serverPort = 12000
4 clientSocket = socket(AF_INET, SOCK_DGRAM)
5 message = input('Input lowercase sentence:')
6 clientSocket.sendto(message.encode(), (serverName, serverPort))
7 modifiedMessage, serverAddress = clientSocket.recvfrom(2048)
8 print(modifiedMessage.decode())
9 clientSocket.close()
```

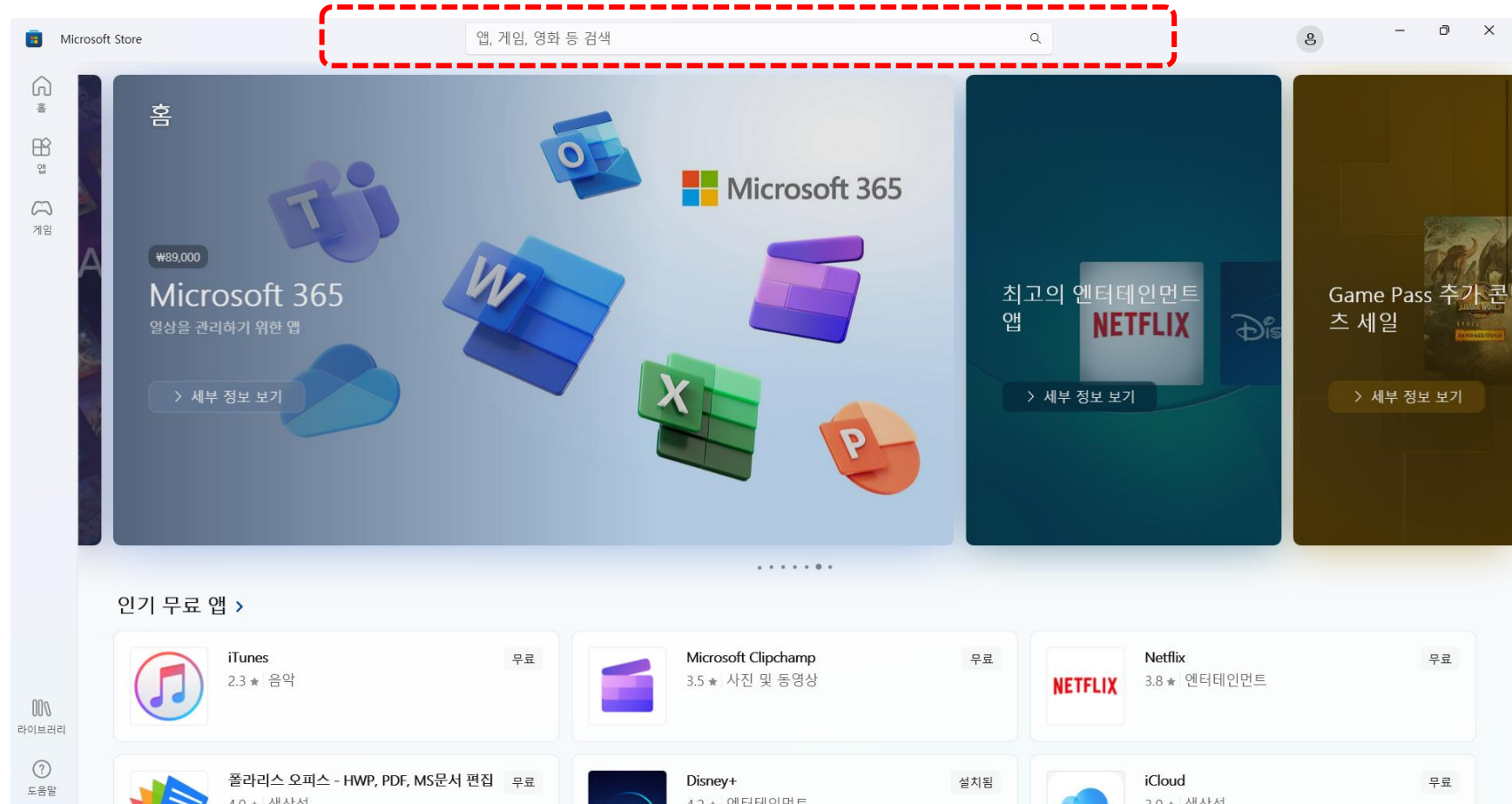
Below the code editor is a terminal window showing the execution of the script:

```
~$ python3 UDPClient.py
Input lowercase sentence:test
TEST
~$
```

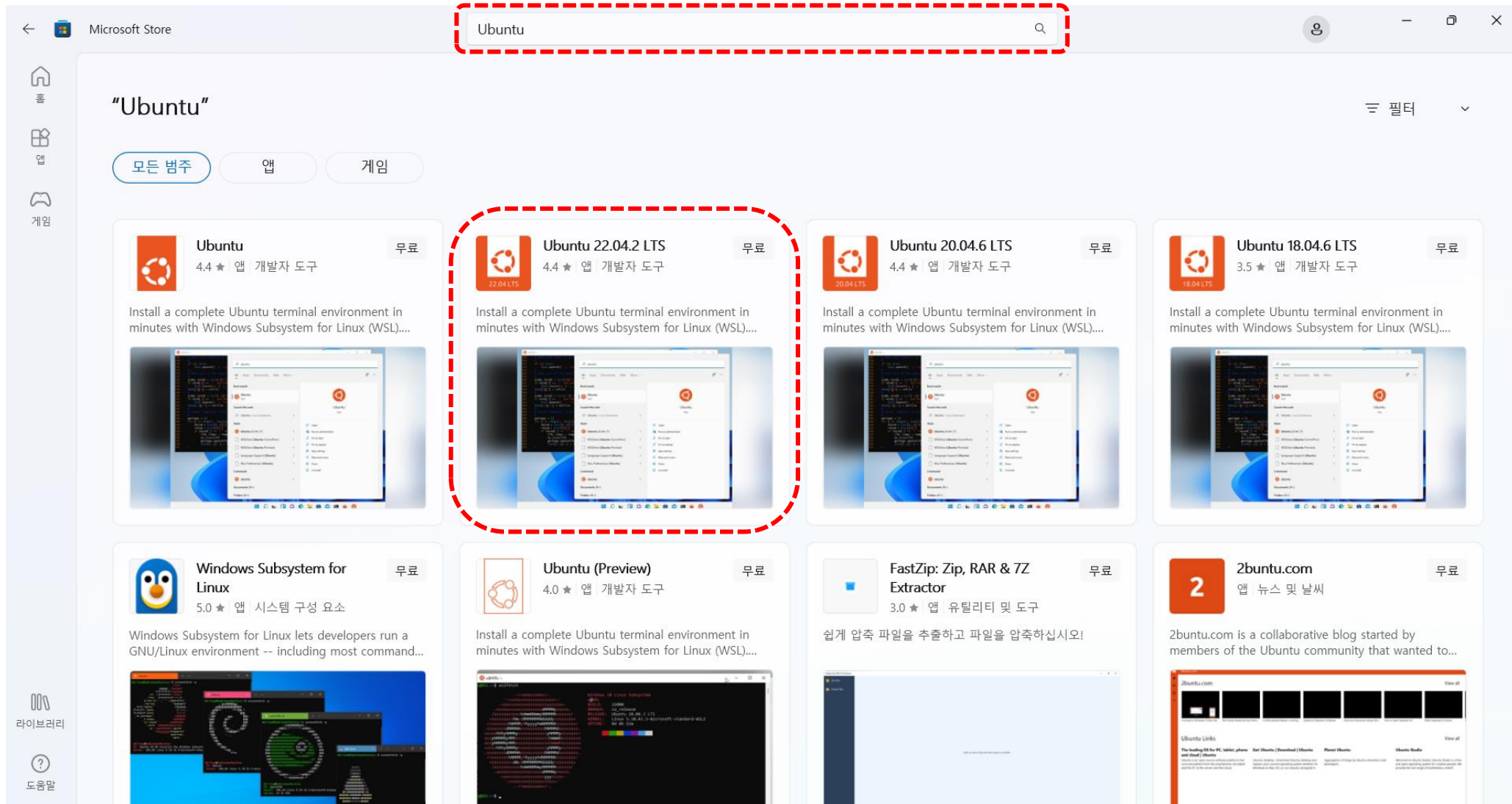
The interface also includes a sidebar with 'Explorer' and 'Log' views, and a top navigation bar with 'Projects' and 'My First Project' tabs. A red banner at the top of the IDE area contains a warning message about hosting quality and license requirements.

Python socket 프로그래밍 실습 환경 – WSL/Ubuntu

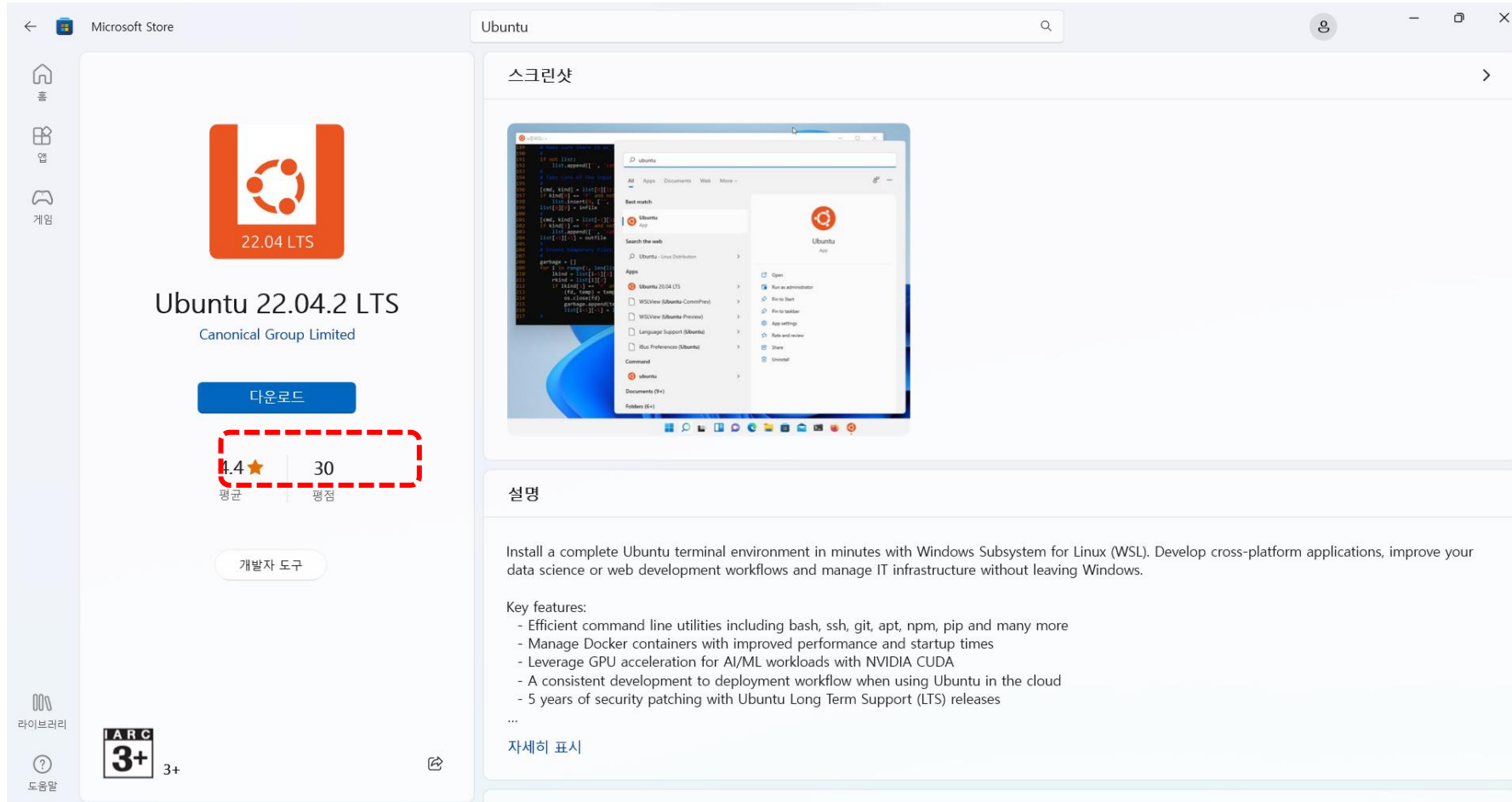
1. Microsoft Store 접속



2. Ubuntu 검색



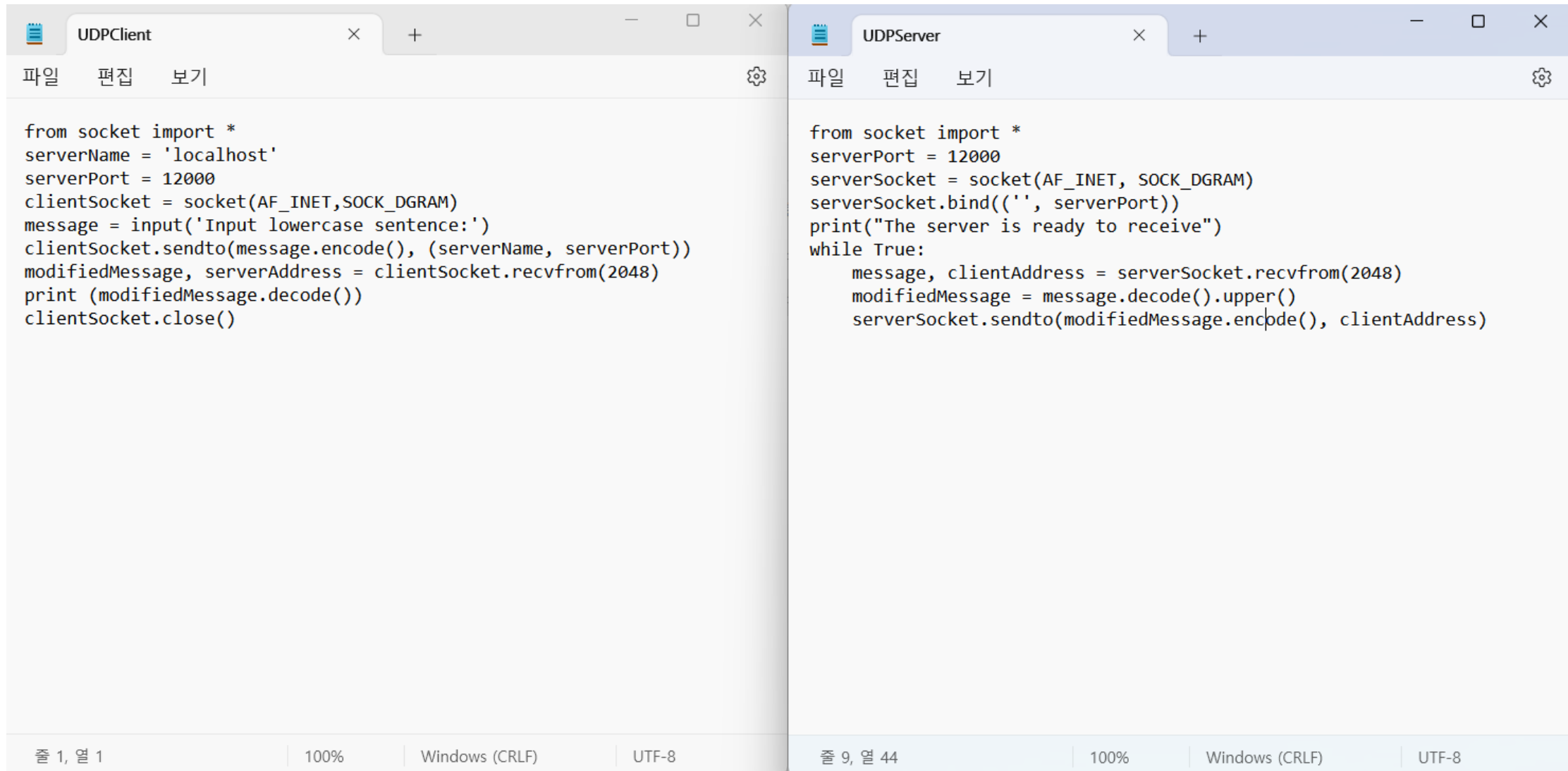
3. Ubuntu 22.04 LTS 버전 설치



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

[\[WSL\] Error: 0x80370102 해결 \(velog.io\)](https://velog.io)

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



The image displays two side-by-side Notepad++ windows. The left window, titled 'UDPClient', contains Python code for a client that connects to a server at localhost:12000, sends a lowercase sentence, and receives a response. The right window, titled 'UDPServer', contains Python code for a server that binds to port 12000, receives a message, converts it to uppercase, and sends it back. Both windows show standard menu bars (File, Edit, View) and status bars at the bottom indicating line numbers, zoom level, and encoding.

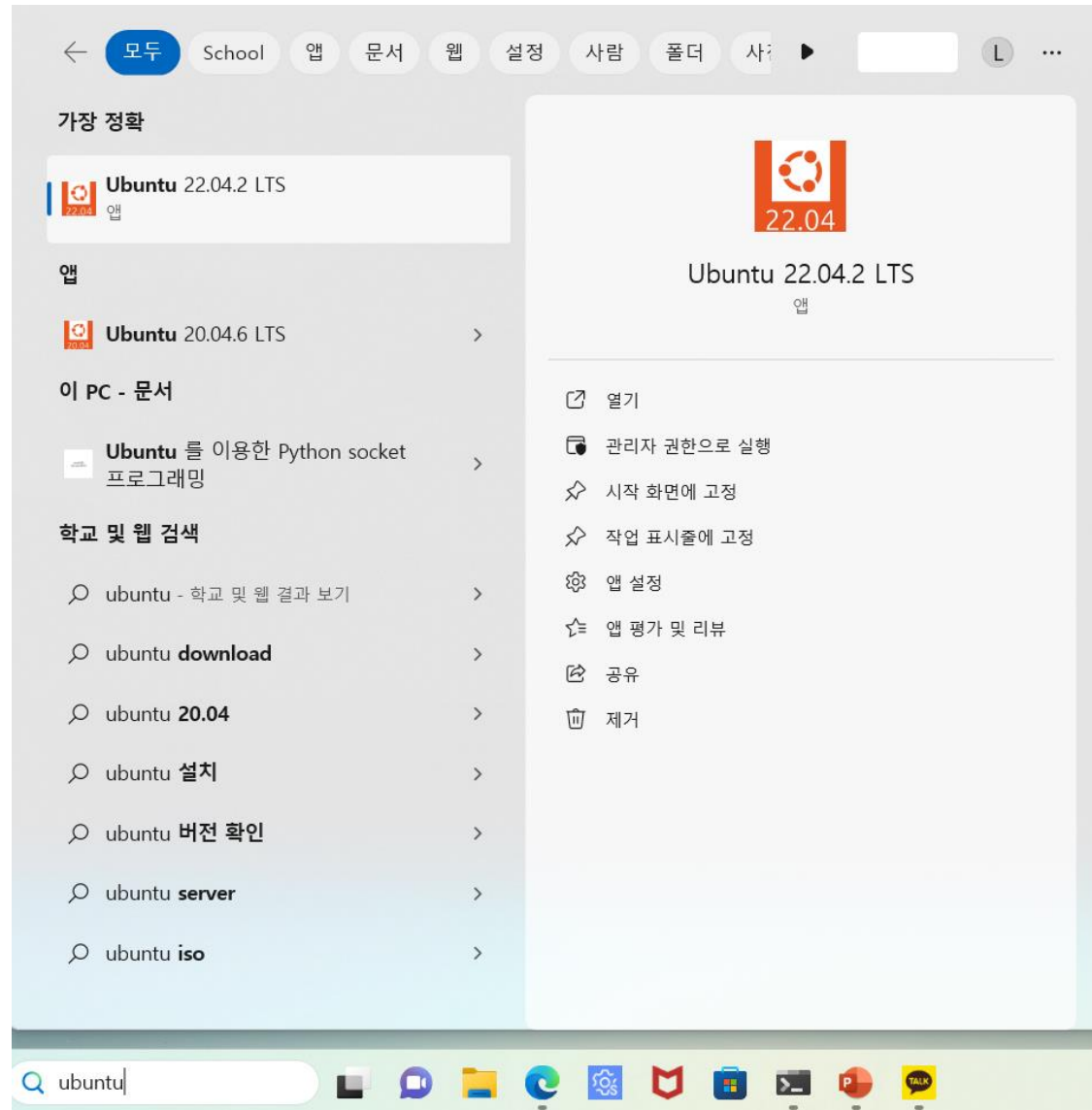
```
from socket import *
serverName = 'localhost'
serverPort = 12000
clientSocket = socket(AF_INET, SOCK_DGRAM)
message = input('Input lowercase sentence:')
clientSocket.sendto(message.encode(), (serverName, serverPort))
modifiedMessage, serverAddress = clientSocket.recvfrom(2048)
print (modifiedMessage.decode())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

```
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)
```

5) File 저장 위치 Documents

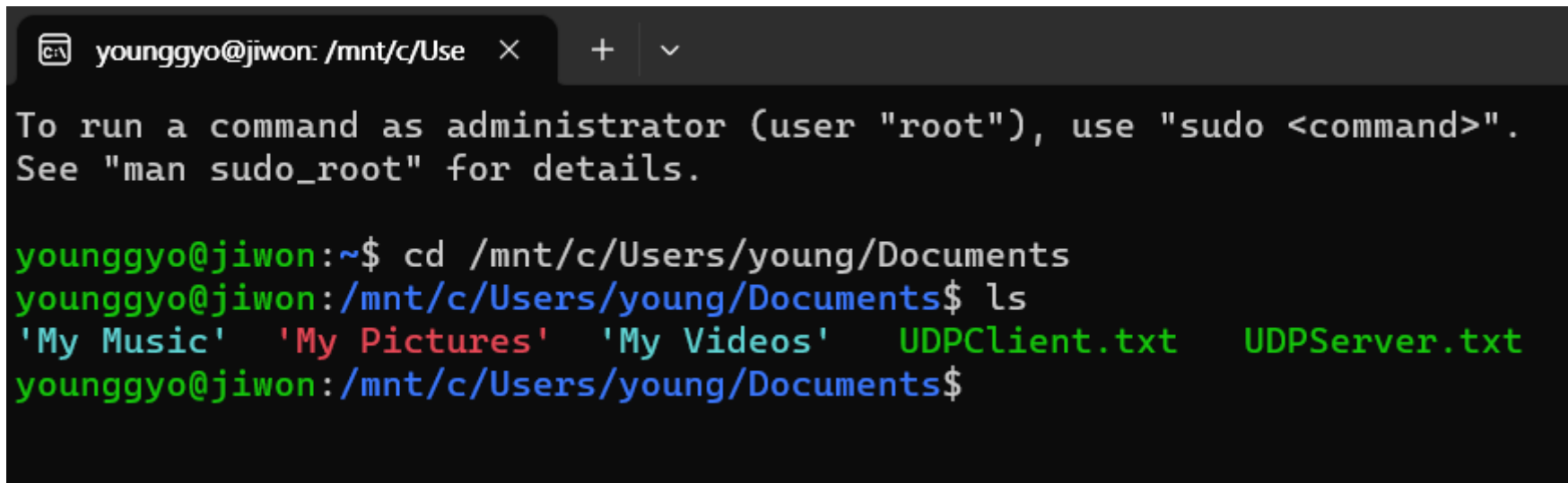
📁 > 내 PC > 로컬 디스크 (C:) > 사용자 > young > Documents					▼	🔄	D
이름	수정한 날짜	유형	크기				
📄 UDPCClient	2023-09-21 오후 2:37	텍스트 문서	1KB				
📄 UDPServer	2023-09-21 오후 2:40	텍스트 문서	1KB				

6) Ubuntu 실행



7) 파일 위치 접근 : cd 명령어

`cd /mnt/c/Users/{사용자명}/Documents`

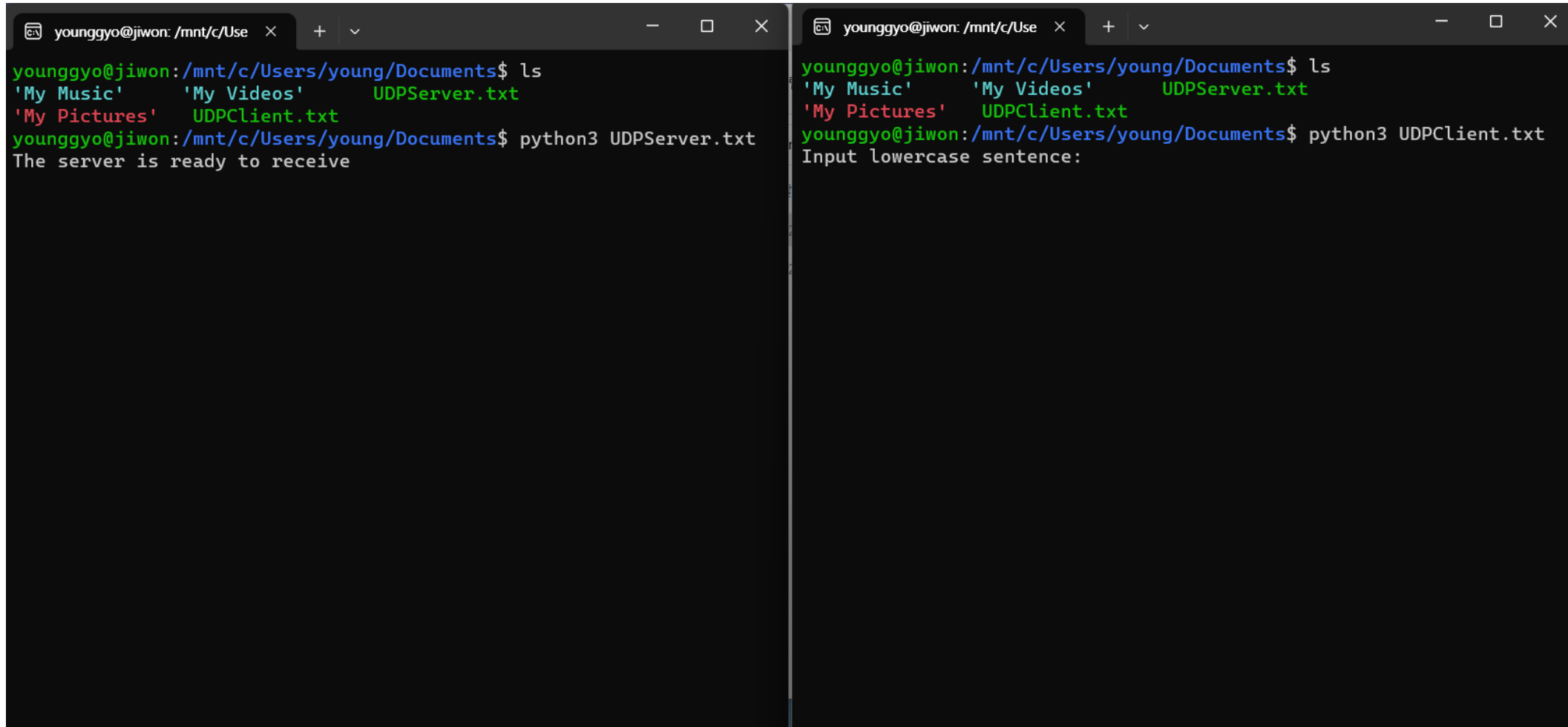


```
younggyo@jiwon: /mnt/c/Use × + v
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

younggyo@jiwon:~$ cd /mnt/c/Users/young/Documents
younggyo@jiwon:/mnt/c/Users/young/Documents$ ls
'My Music'  'My Pictures'  'My Videos'  UDPClient.txt  UDPServer.txt
younggyo@jiwon:/mnt/c/Users/young/Documents$
```

8) Ubuntu에서 코드 실행

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



The image shows two terminal windows side-by-side. Both windows have a title bar that reads 'younggyo@jiwon: /mnt/c/Use'. The left window shows the execution of a UDP server program. The right window shows the execution of a UDP client program.

```
younggyo@jiwon:/mnt/c/Users/young/Documents$ ls
'My Music'      'My Videos'    UDPServer.txt
'My Pictures'   UDPClient.txt
younggyo@jiwon:/mnt/c/Users/young/Documents$ python3 UDPServer.txt
The server is ready to receive
```

```
younggyo@jiwon:/mnt/c/Users/young/Documents$ ls
'My Music'      'My Videos'    UDPServer.txt
'My Pictures'   UDPClient.txt
younggyo@jiwon:/mnt/c/Users/young/Documents$ python3 UDPClient.txt
Input lowercase sentence:
```