


MemoNote semi Project 3 230201

MyMEMO

프로젝트 눈누

눈누

상업용 무료한글폰트 사이트

 <https://noonnu.cc/>



작성 데모 리스트

2222 2023-01-30 23:41:03

test button

2222 2023-01-30 23:32:09

fdsafdsafdsa

fdsa

2222 2023-01-30 22:30:26

summernote

1. test

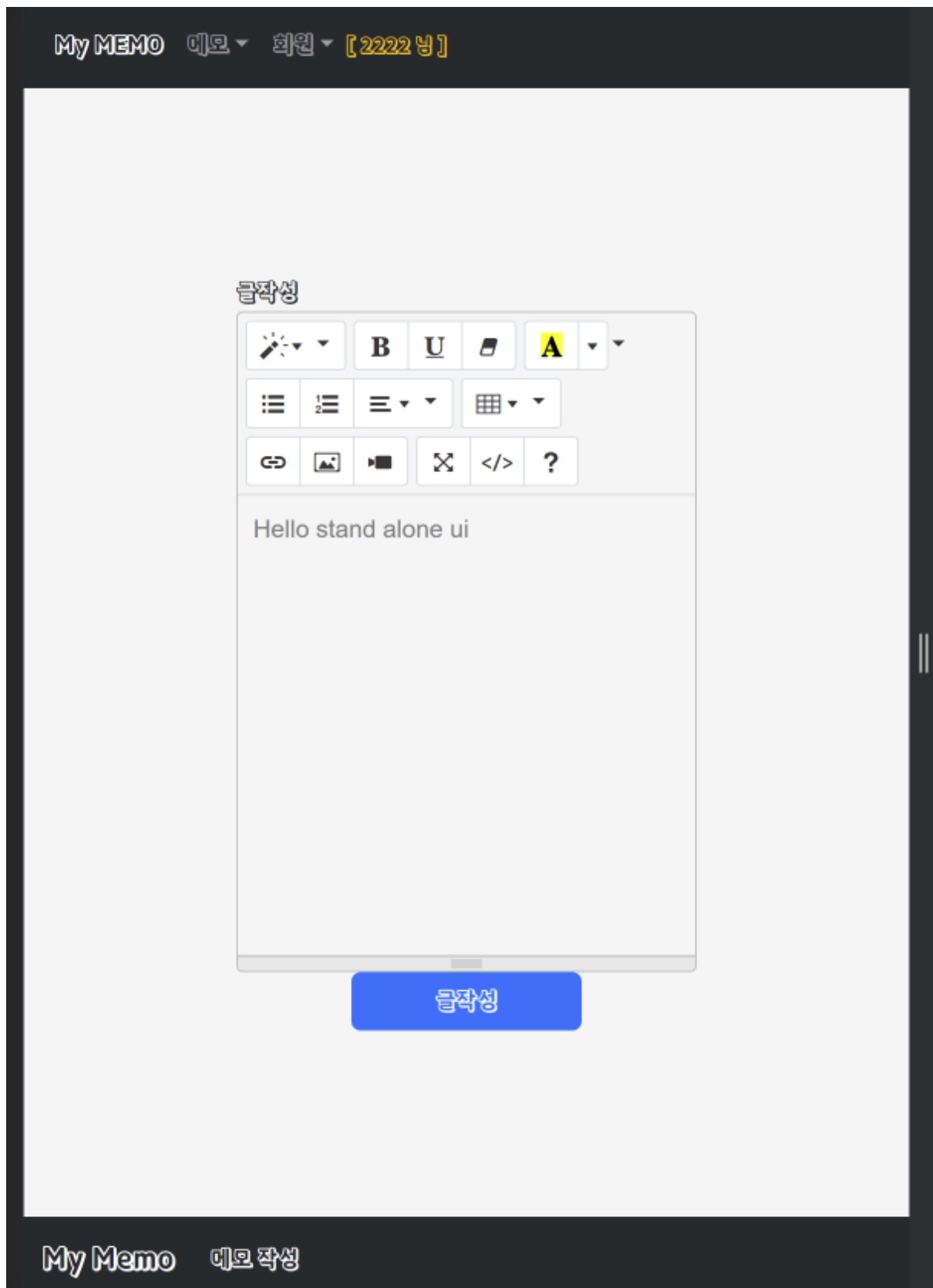
hello

2222 2023-01-30 22:19:57

jhfigdhgfd

2222 2023-01-30 22:19:08

test



맘에는 폰트를 찾아서 '웹폰트로 사용' 을 카피한다.

주의!!! 웹폰트가 없이 다운로드만 있는 글꼴도 있다. 웹폰트가 있는 것만 사용하자.

평창평화체

[다운로드 페이지로 이동](#)[★ 즐겨찾기](#)

제작 평창군 조희수 175K 형태 장식체 굵기 2가지

폰트 미리보기

Type something i 24px

예쁘고 독특한 별모양 폰트 300

예쁘고 독특한 별모양 폰트 700

START

웹폰트로 사용



```
@font-face {  
  font-family: 'PyeongCh  
  src: url('https://cdn.  
  font-weight: 700;  
  font-style: normal;  
}
```

같은 제작사

더보기 →

평창체

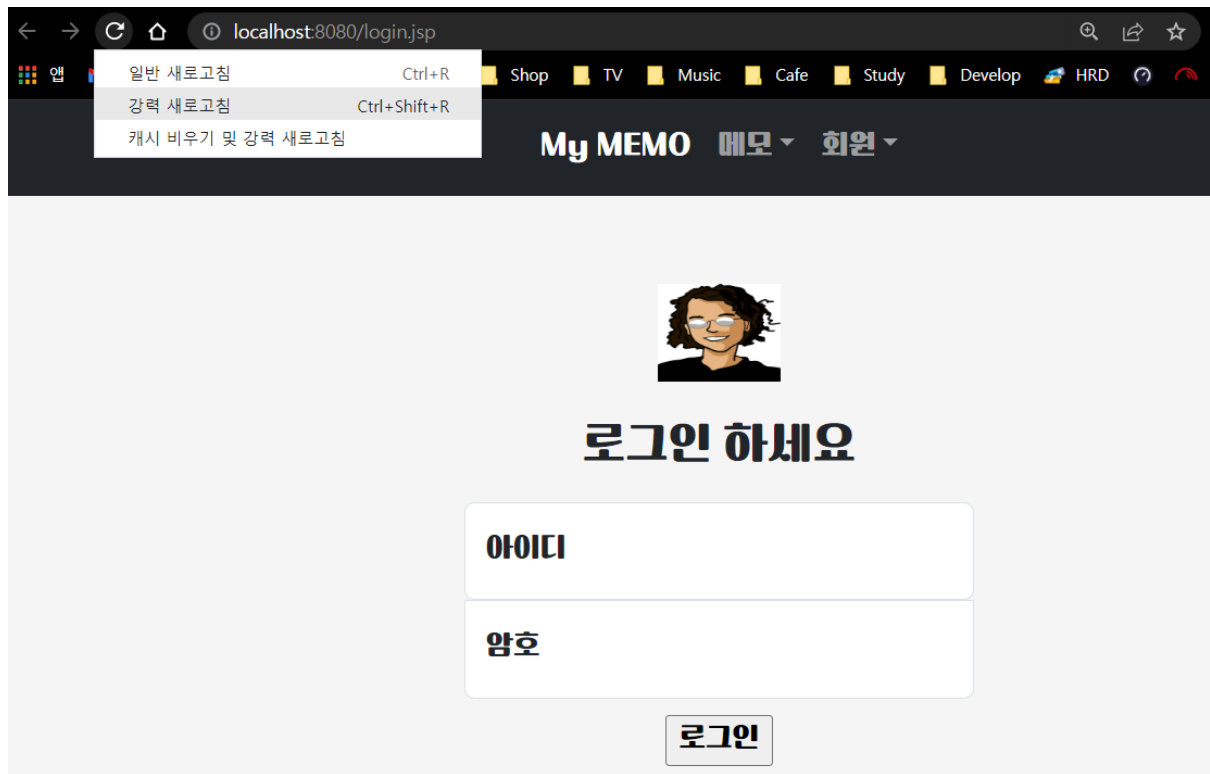
CSS 페이지의 제일 상단에 붙여 넣자

```
Insert title here  *sign-in.css
1 @font-face {
2     font-family: 'PyeongChangPeace-Bold';
3     src: url('https://cdn.jsdelivr.net/gh/projectnoonnu/noonfonts_2206-6');
4     font-weight: 700;
5     font-style: normal;
6 }
7
8
9
10 html,
11 body {
12     height: 100%;
13 }
14
15 body {
16     display: flex;
17     align-items: center;
18     padding-top: 40px;
19     padding-bottom: 40px;
20     background-color: #f5f5f5;
21 }
```

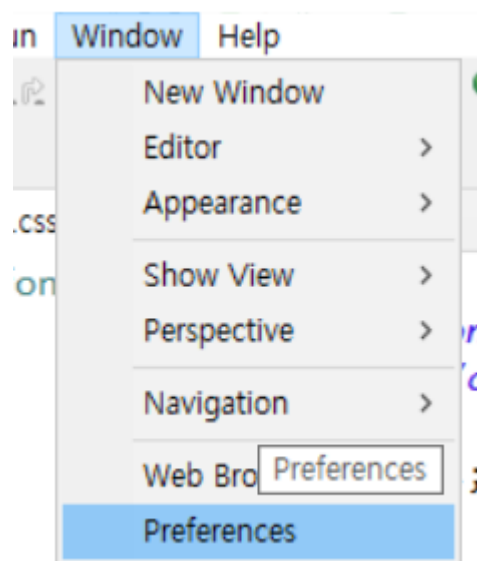
바디 부분에 폰트 패밀리 적용

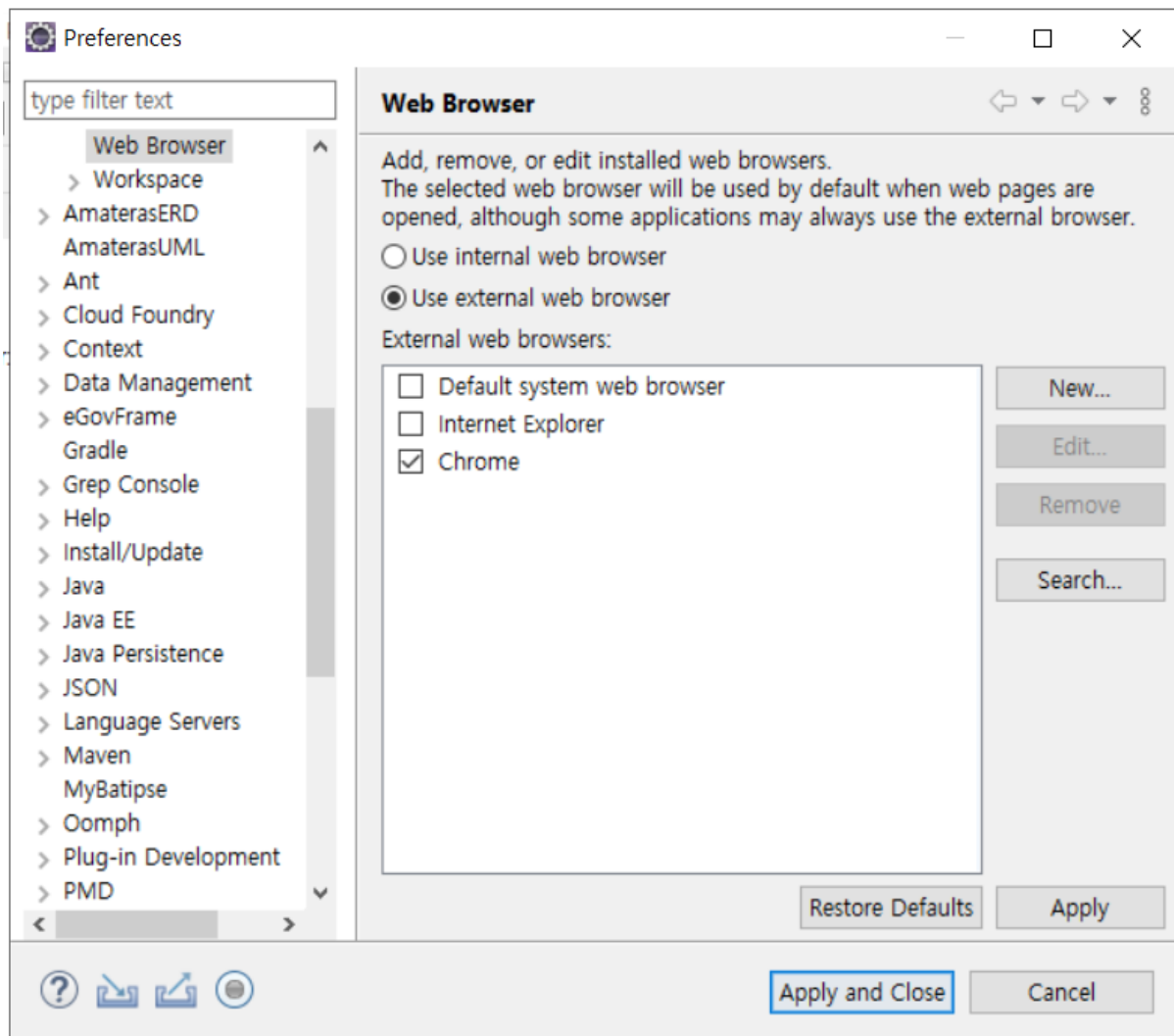
```
Insert title here  sign-in.css
1 @font-face {
2     font-family: 'PyeongChangPeace-Bold';
3     src: url('https://cdn.jsdelivr.net/gh/projectnoonnu/noonfonts_2206-6');
4     font-weight: 700;
5     font-style: normal;
6 }
7
8
9
10 html,
11 body {
12     height: 100%;
13     font-family: 'PyeongChangPeace-Bold';
14 }
15
```

**!!! 반드시 !!! 설정 후 폰트 변화가 보지 않으면
“강력 세로고침 할것 !!!**



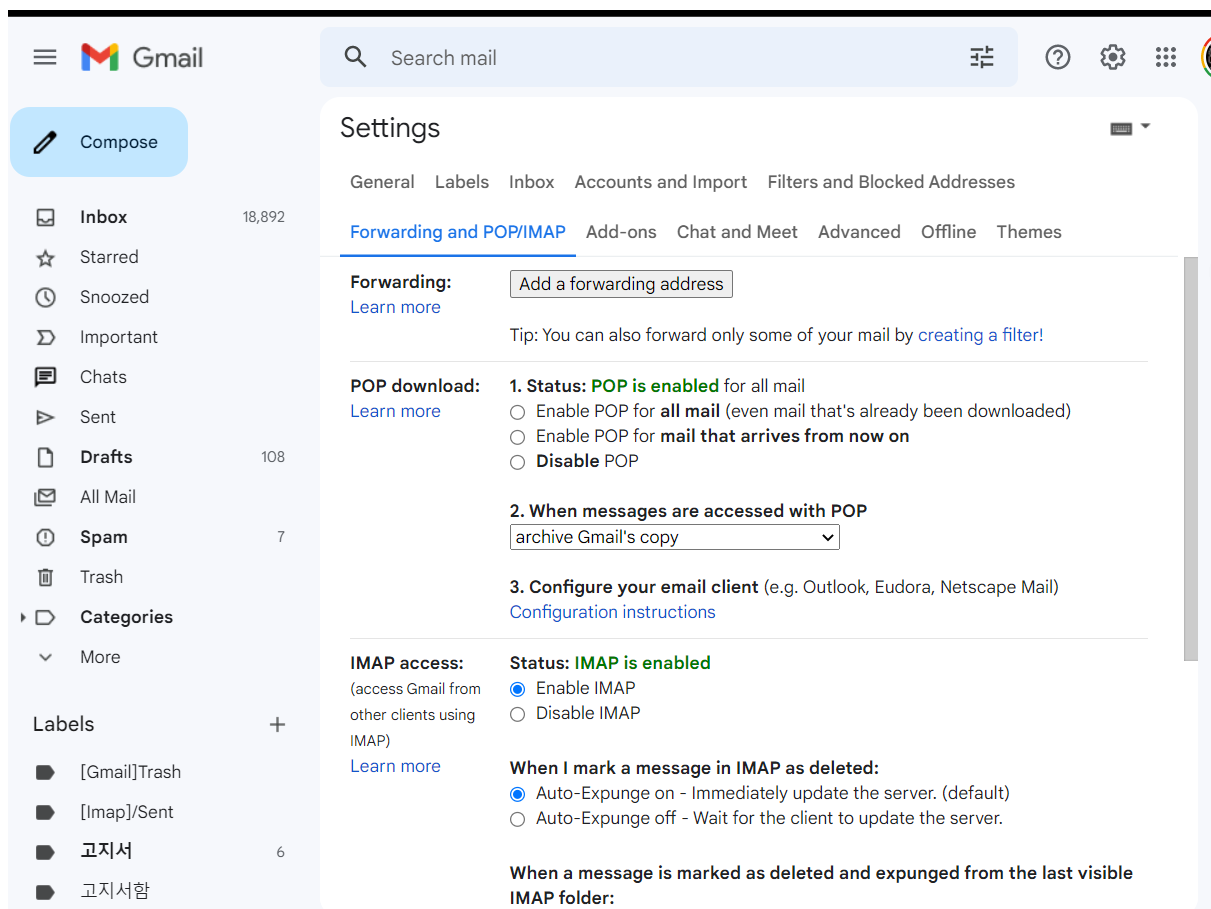
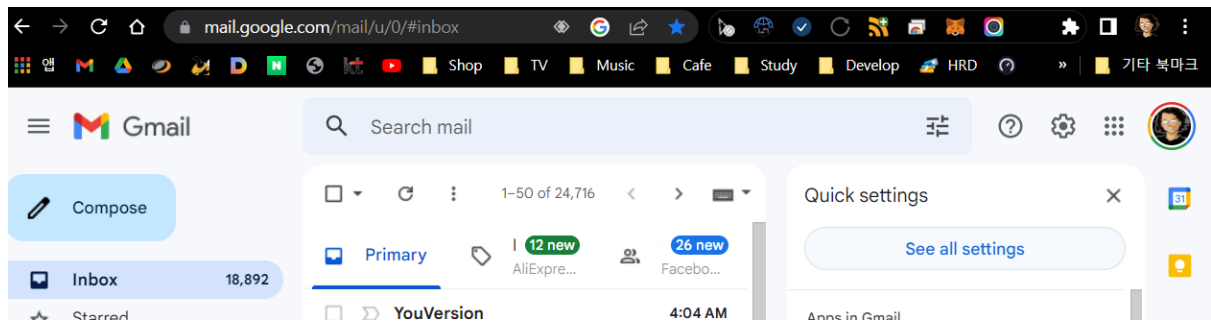
이클립스에서 기본 내장 브라우저가 아닌 크롬 브라우저로 바로 실행하는 방법



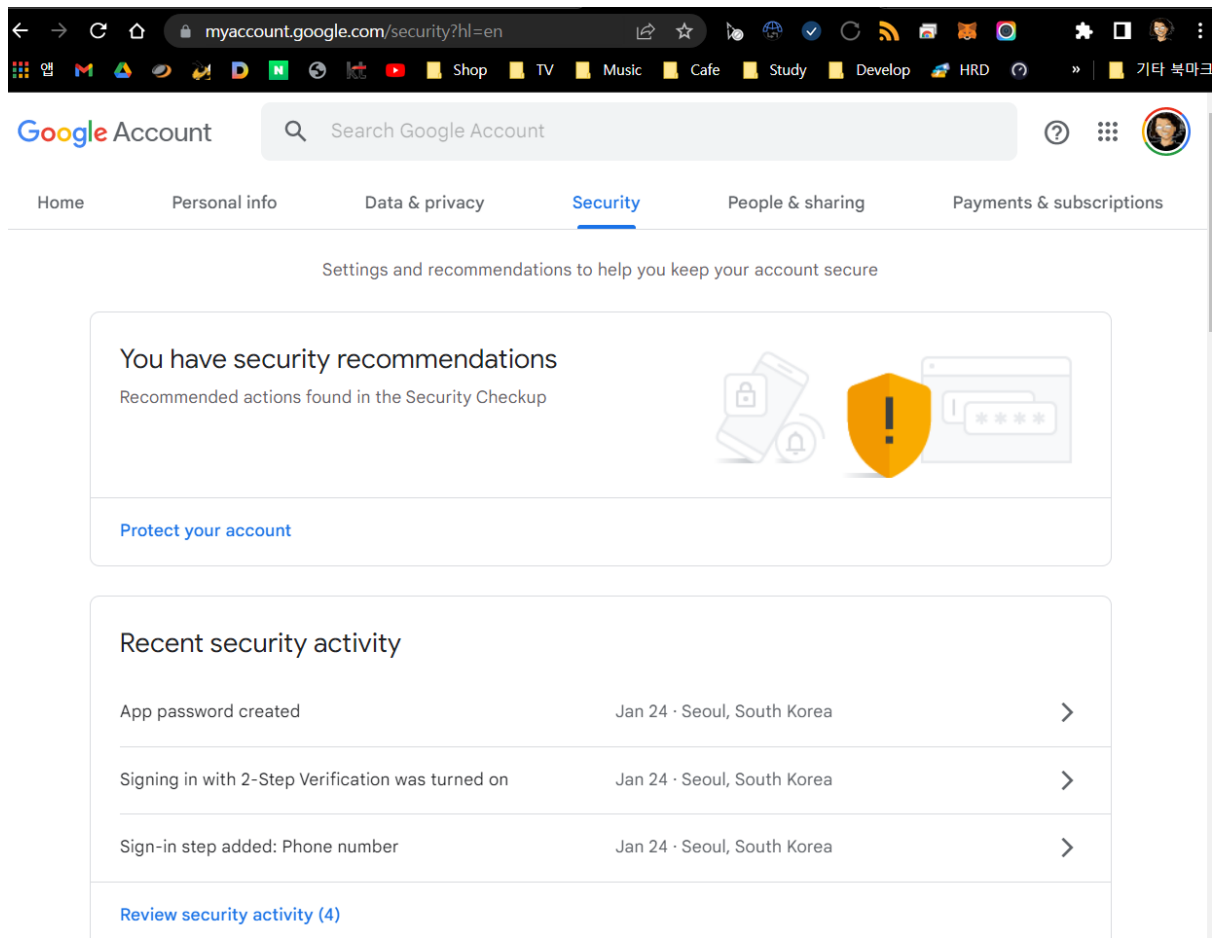


Gmail Setting

구글 메일 설정



구글 로그인 보안 설정에서 2단계 설정 할것 2-step verification
App password 발급 받기



기존 네이버 메일 서버에서 구글 서버 발송으로 변경을 위해 발신인의 이메일 주소도 구글 이메일 주소로 변경해야만 한다.

```
7 <title>SMTP 이메일 전송</title>
8 </head>
9 <body>
10 <h2>이메일 전송하기</h2>
11 <form method="post" action="SendProcess.jsp">
12 <table border=1>
13
14 <tr>
15 <td><input type="hidden" name="from" value="yoonkeunsoo@gmail.com" />
16 </td>
17 </tr>
```

SendProcess.jsp 에 서버명 변경

```

}

try {
    GoogleSMTP smtpServer = new GoogleSMTP(); // 메일 전송 클래스 생성
    smtpServer.emailSending(emailInfo);        // 전송
    out.print("이메일 전송 성공");
}
catch (Exception e) {
    out.print("이메일 전송 실패");
    e.printStackTrace();
}
%>

```

```

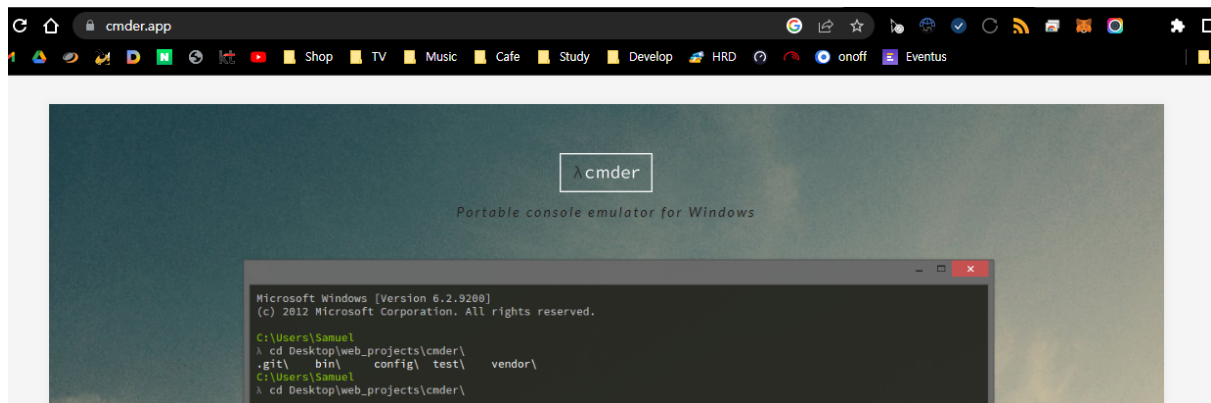
public GoogleSMTP() {
    // 네이버 SMTP 서버 접속 정보
    serverInfo = new Properties();
    serverInfo.put("mail.smtp.host", "smtp.gmail.com");
    serverInfo.put("mail.smtp.port", "465");
    serverInfo.put("mail.smtp.starttls.enable", "true");
    serverInfo.put("mail.smtp.auth", "true");
    serverInfo.put("mail.smtp.debug", "true");
    serverInfo.put("mail.smtp.socketFactory.port", "465");
    serverInfo.put("mail.smtp.socketFactory.class",
        "javax.net.ssl.SSLSocketFactory");
    serverInfo.put("mail.smtp.socketFactory.fallback", "false");

    // 사용자 인증 정보
    auth = new Authenticator() {
        @Override
        protected PasswordAuthentication getPasswordAuthentication() {
            return new PasswordAuthentication("yoonkeunsoo@gmail.com", "");
        }
    };
}

```

create memo, sign up

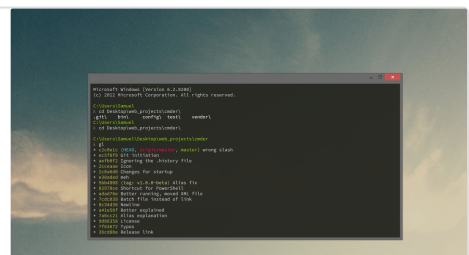
DOS 명령어 정리



Cmdr | Console Emulator

Portable console emulator for Windows Cmdr is a software package created out of pure frustration over the absence of nice console emulators on Windows. It is based on amazing

 <https://cmder.app/>



1. dir 디렉토리의 내용 보기 (맥 또는 리눅스에서는 ls)

```

dir
C 드라이브의 볼륨에는 이름이 없습니다.
볼륨 일련 번호: 5EA5-0099

C:\develop\cmder 디렉터리

2023. 02. 01. 오후 12:47 <DIR> .
2023. 02. 01. 오후 12:47 <DIR> ..
2023. 02. 01. 오후 12:47 <DIR> bin
2022. 12. 19. 오후 05:35 148,992 Cmdr.exe
2023. 02. 01. 오후 02:19 <DIR> config
2023. 02. 01. 오후 12:47 <DIR> icons
2022. 12. 19. 오후 05:34 1,090 LICENSE
2023. 02. 01. 오후 12:47 <DIR> opt
2023. 02. 01. 오후 12:48 <DIR> vendor
2023. 02. 01. 오후 12:46 0 Version 1.3.20.151
3개 파일 150,082 바이트
7개 디렉터리 390,785,175,552 바이트 남음
  
```

2. cd.. 디렉토리 한단계 위로 이동 change directory

```
C:\develop\cmdr
λcd..

C:\develop
λcd..

C:\
λ
```

3. cd 디렉토리이름 해당 디렉토리로 들어가기 change directory

```
C:\
λcd develop

C:\develop
λcd cmdr

C:\develop\cmdr
λ |
```

4. cd \ 루트 디렉토리로 한번에 이동

```
C:\develop\cmdr
λcd \

C:\
λ
```

5. md 디렉토리명 make directory 새로운 디렉토리 생성

```

2023. 02. 01. 오후 12:47 <DIR> .
2023. 02. 01. 오후 12:47 <DIR> ..
2023. 02. 01. 오후 12:47 <DIR> cmdr
2023. 01. 12. 오후 02:24 <DIR> eclipse
2023. 01. 30. 오전 10:24 <DIR> jsp
2023. 01. 12. 오후 05:41 <DIR> notebook
2023. 01. 12. 오후 05:11 <DIR> servlet
2023. 01. 12. 오후 02:16 <DIR> tomcat8
0개 파일 0 바이트
8개 디렉터리 390,788,648,960 바이트 남음

C:\develop
λmd temp

C:\develop
λdir
C 드라이브의 볼륨에는 이름이 없습니다.
볼륨 일련 번호: 5EA5-0099

C:\develop 디렉터리

2023. 02. 01. 오후 02:26 <DIR> .
2023. 02. 01. 오후 02:26 <DIR> ..
2023. 02. 01. 오후 12:47 <DIR> cmdr
2023. 01. 12. 오후 02:24 <DIR> eclipse
2023. 01. 30. 오전 10:24 <DIR> jsp
2023. 01. 12. 오후 05:41 <DIR> notebook
2023. 01. 12. 오후 05:11 <DIR> servlet
2023. 02. 01. 오후 02:26 <DIR> temp
2023. 01. 12. 오후 02:16 <DIR> tomcat8
0개 파일 0 바이트
9개 디렉터리 390,788,710,400 바이트 남음

C:\develop
λ

```

6 rd 디렉토리명

해당 디렉토리 삭제 remove directory

```

2023. 02. 01. 오후 02:26 <DIR> .
2023. 02. 01. 오후 02:26 <DIR> ..
2023. 02. 01. 오후 12:47 <DIR> cmdr
2023. 01. 12. 오후 02:24 <DIR> eclipse
2023. 01. 30. 오전 10:24 <DIR> jsp
2023. 01. 12. 오후 05:41 <DIR> notebook
2023. 01. 12. 오후 05:11 <DIR> servlet
2023. 02. 01. 오후 02:26 <DIR> temp
2023. 01. 12. 오후 02:16 <DIR> tomcat8
0개 파일 0 바이트
9개 디렉터리 390,788,710,400 바이트 남음

```

C:\develop

λrd temp

C:\develop

λdir

C 드라이브의 볼륨에는 이름이 없습니다.
볼륨 일련 번호: 5EA5-0099

C:\develop 디렉터리

```

2023. 02. 01. 오후 02:27 <DIR> .
2023. 02. 01. 오후 02:27 <DIR> ..
2023. 02. 01. 오후 12:47 <DIR> cmdr
2023. 01. 12. 오후 02:24 <DIR> eclipse
2023. 01. 30. 오전 10:24 <DIR> jsp
2023. 01. 12. 오후 05:41 <DIR> notebook
2023. 01. 12. 오후 05:11 <DIR> servlet
2023. 01. 12. 오후 02:16 <DIR> tomcat8
0개 파일 0 바이트
8개 디렉터리 390,784,663,552 바이트 남음

```

C:\develop

λ

7 copy con 파일명.확장자

파일 만들기

```
C:\develop
λcopy con temp.txt
test
1
2
3
4
^Z

1개 파일이 복사되었습니다.

C:\develop
λdir
C 드라이브의 볼륨에는 이름이 없습니다.
볼륨 일련 번호: 5EA5-0099

C:\develop 디렉터리

2023. 02. 01. 오후 02:29 <DIR> .
2023. 02. 01. 오후 02:29 <DIR> ..
2023. 02. 01. 오후 12:47 <DIR> cmdr
2023. 01. 12. 오후 02:24 <DIR> eclipse
2023. 01. 30. 오전 10:24 <DIR> jsp
2023. 01. 12. 오후 05:41 <DIR> notebook
2023. 01. 12. 오후 05:11 <DIR> servlet
2023. 02. 01. 오후 02:29 18 temp.txt
2023. 01. 12. 오후 02:16 <DIR> tomcat8
1개 파일 18 바이트
8개 디렉터리 390,782,889,984 바이트 남음

C:\develop
λ
```

8 type 파일명.확장자

파일 열어서 보기

```
C:\develop
λdir
C 드라이브의 볼륨에는 이름이 없습니다.
볼륨 일련 번호: 5EA5-0099

C:\develop 디렉터리

2023. 02. 01. 오후 02:29 <DIR>      .
2023. 02. 01. 오후 02:29 <DIR>      ..
2023. 02. 01. 오후 12:47 <DIR>      cmdr
2023. 01. 12. 오후 02:24 <DIR>      eclipse
2023. 01. 30. 오전 10:24 <DIR>      jsp
2023. 01. 12. 오후 05:41 <DIR>      notebook
2023. 01. 12. 오후 05:11 <DIR>      servlet
2023. 02. 01. 오후 02:29      18 temp.txt
2023. 01. 12. 오후 02:16 <DIR>      tomcat8
                        1개 파일      18 바이트
                        8개 디렉터리 390,782,889,984 바이트 남음

C:\develop
λtype temp.txt
test
1
2
3
4

C:\develop
λ
```

9. del 파일명.확장자

파일 삭제


```
2023. 01. 12. 오후 02:24 <DIR> eclipse
2023. 01. 30. 오전 10:24 <DIR> jsp
2023. 01. 12. 오후 05:41 <DIR> notebook
2023. 01. 12. 오후 05:11 <DIR> servlet
2023. 02. 01. 오후 02:29 18 temp.txt
2023. 01. 12. 오후 02:16 <DIR> tomcat8
1개 파일 18 바이트
8개 디렉터리 390,782,889,984 바이트 남음 type 파일명.확장자

C:\develop
λtype temp.txt
test
1
2
3
4

C:\develop
λdel temp.txt

C:\develop
λdir
C 드라이브의 볼륨에는 이름이 없습니다.
볼륨 일련 번호: 5EA5-0099

C:\develop 디렉터리

2023. 02. 01. 오후 02:32 <DIR> .
2023. 02. 01. 오후 02:32 <DIR> ..
2023. 02. 01. 오후 12:47 <DIR> cmdr
2023. 01. 12. 오후 02:24 <DIR> eclipse
2023. 01. 30. 오전 10:24 <DIR> jsp
2023. 01. 12. 오후 05:41 <DIR> notebook
2023. 01. 12. 오후 05:11 <DIR> servlet
2023. 01. 12. 오후 02:16 <DIR> tomcat8
0개 파일 0 바이트
8개 디렉터리 390,773,288,960 바이트 남음

C:\develop
λ
```

github 병목현상으로 업로드 문제 발생 시 처리 방법

먼저 기존의 git history 를 삭제한다.

```
C:\develop\jsp (main -> origin)
λ rm -rf .git

C:\develop\jsp
λ
```

Python Basic

새로운 git 설정

```
C:\develop\jsp
λ git init
hint: Using 'master' as the name for the initial branch. This default branch name
hint: is subject to change. To configure the initial branch name to use in all
hint: of your new repositories, which will suppress this warning, call:
hint:
hint:     git config --global init.defaultBranch <name>
hint:
hint: Names commonly chosen instead of 'master' are 'main', 'trunk' and
hint: 'development'. The just-created branch can be renamed via this command:
hint:
hint:     git branch -m <name>
Initialized empty Git repository in C:/develop/jsp/.git/

C:\develop\jsp (master)
λ
```

모든 파일을 새롭게 git 에 업로드 하기 위한 명령어

```
C:\develop\jsp (master)
λ git add .
```

새로운 모든 파일에 붙이기

```
C:\develop\jsp (master)
λ git commit -m "new commit"
```

주의!!!

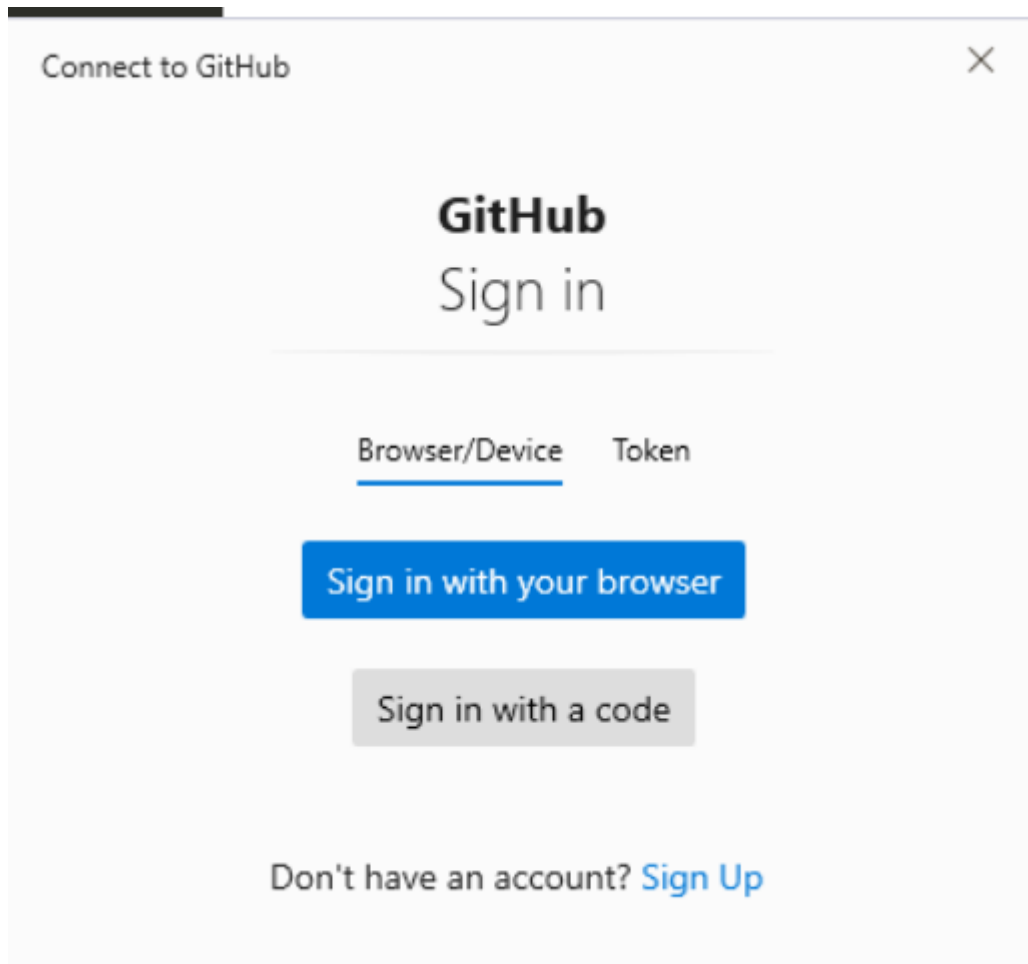
```
C:\develop\jsp (master)  
λ git remote rm origin
```

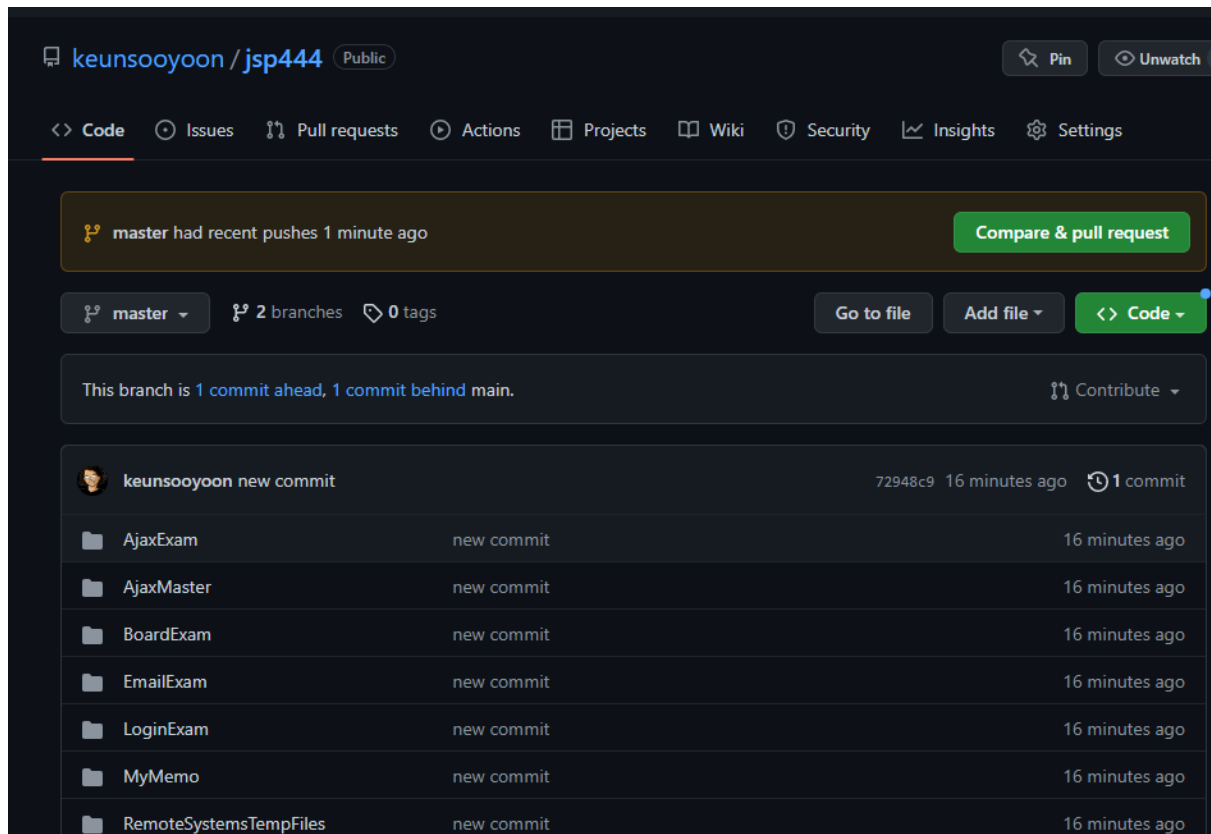
새로운 리파지토리와 연결 (깃 가서 새로 만들어 놓을 것)

```
C:\develop\jsp (master)  
λ gitremote add origin https://github.com/keunsooyoon/jsp444
```

푸시 후 로그인 창으로 로그인

```
C:\develop\jsp (master)  
λ gitpush -u --force origin master
```

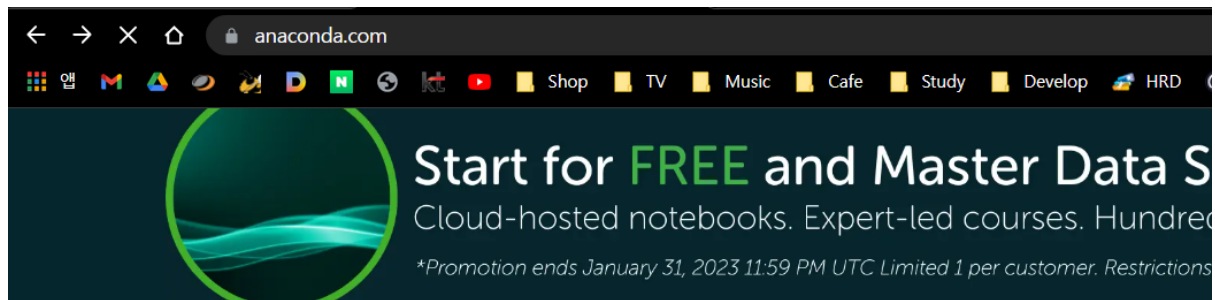




Python Basic


- about python - colab, anaconda, vs code

anaconda 다운로드



Data science technology a competitive edge

Anaconda offers the easiest way to perform Python machine learning on a single machine. Start working with open-source packages and libraries today.

Download 

For Windows

Python 3.9 • 64-Bit Graphical Installer • 621 MB

파이썬은 1990 귀도 반 로섬 이 개발한 인터프리터 언어(≠ 컴파일러 언어, 자바는 일반적으로는 컴파일러 언어로 구분하지만 엄밀히 따지면 인터프리터 언어의 특징도 가지고 있기 때문에 하이브리드 언어라는 표현도 사용 하기도 한다.) 이다.



귀도가 파이썬 이라는 이름을 자신이 좋아하는 코미디 프로 '파이썬의 날라다니는 서커스' 에서 따왔다고 한다. 파이썬의 사전적인 의미는 고대 신화에 나오는 파르나소스 산의 동굴에 살던 큰 뱀을 뜻한다.



파이썬은 컴퓨터 프로그래밍 교육을 위해 많이 사용되기도 하며 동시에 기업의 실무에서도 많이 사용되는 언어이다. 대표적으로 구글에서 만든 소프트웨어의 50% 이상이 파이썬으로 작성되었다고 한다. 예를 들면 인스타그램 드롭박스 등등이 파이썬으로 작성되었다.

파이썬의 특징

- 파이썬은 쉽다.
- 파이썬은 빠르다.
- 파이썬은 강력하다.
- 파이썬은 간결하다.
- 파이썬은 즐겁다.

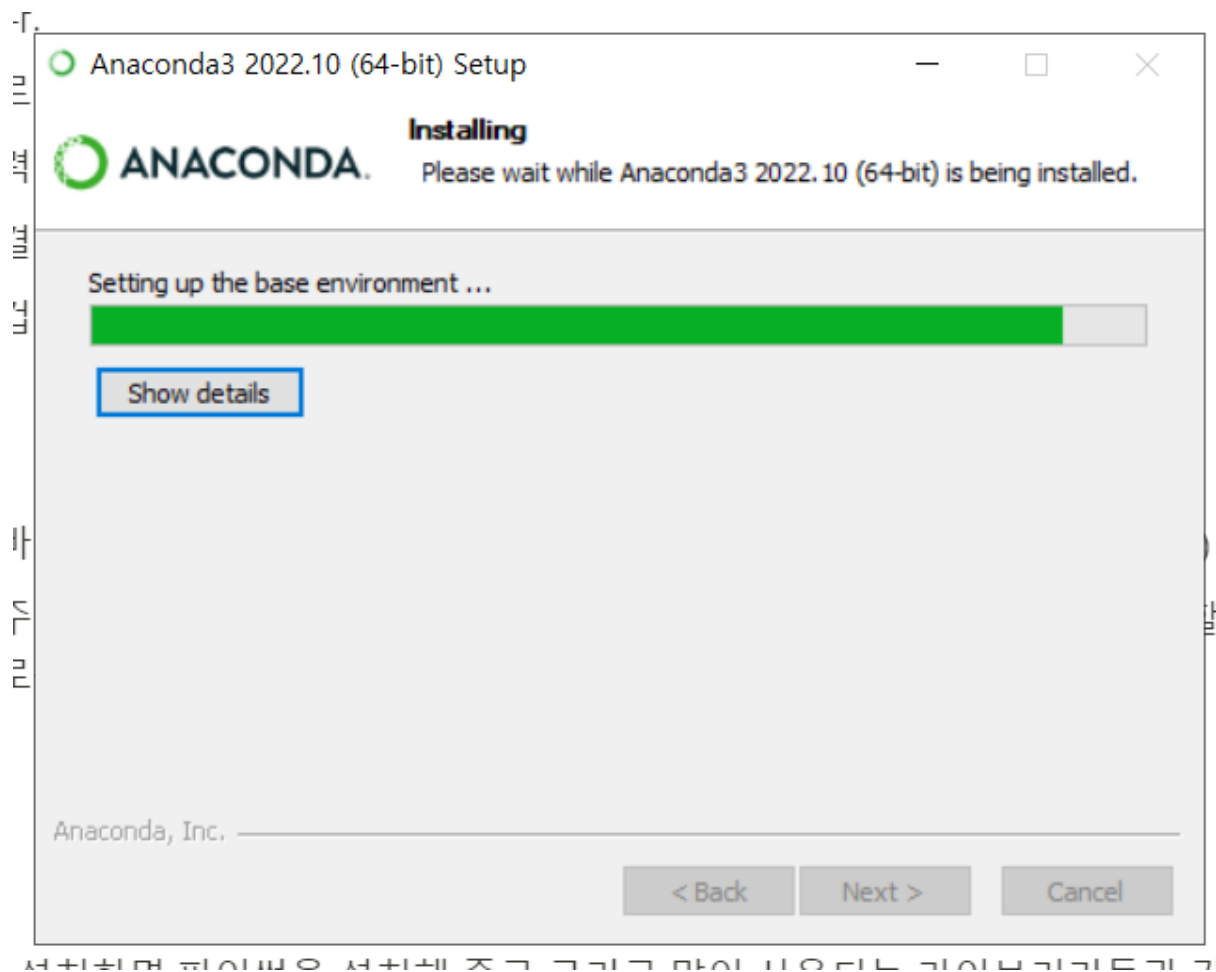
파이썬의 약점

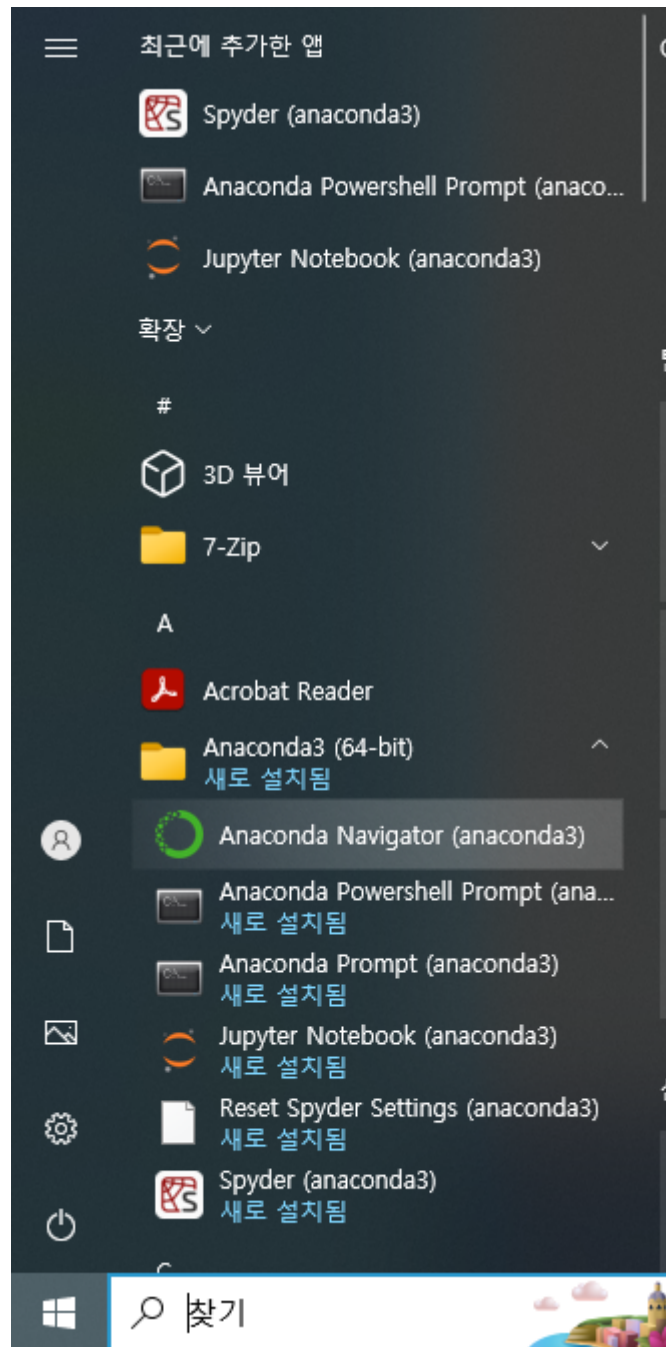
- 파이썬은 모바일 앱을 만들수는 없다. (아이폰 앱 안드로이드 폰 앱 등은 만들수 없다.)
- 시스템과 아주 밀접한 프로그래밍 영역의 접근은 조금 힘들다. (C 라이브러리를 사용할 수 있기 때문에 앞으로는 접근 가능한 영역이 될것이다.)

파이썬 설치

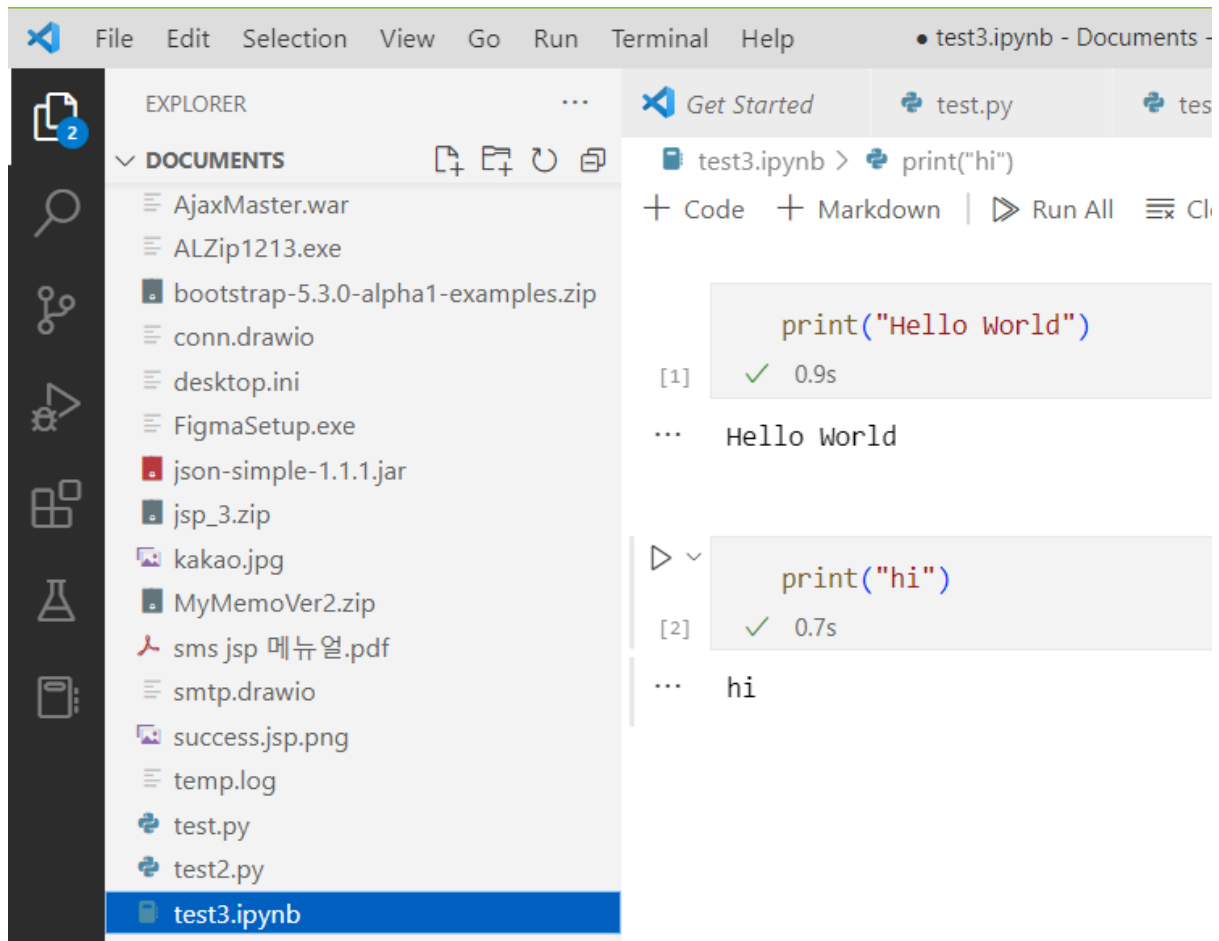
- anaconda 를 설치하면 파이썬을 설치해 주고 Jupyter Notebook 이라는 개발환경 그 리고 많이 사용되는 라이브러리들과 가상환경 설정까지 한 번에 끝낼수 있다.
- 웹 개발환경인 Colab은 설치의 과정이 필요없이 바로 파이썬 개발이 가능하다. 다만 파 이썬 으로 윈도우 앱을 개발하거나 웹 개발 등은 성격상 코랩에서는 진행 할수 없다.

애나콘다 설치시 모든 옵션은 디폴트로 진행한다.

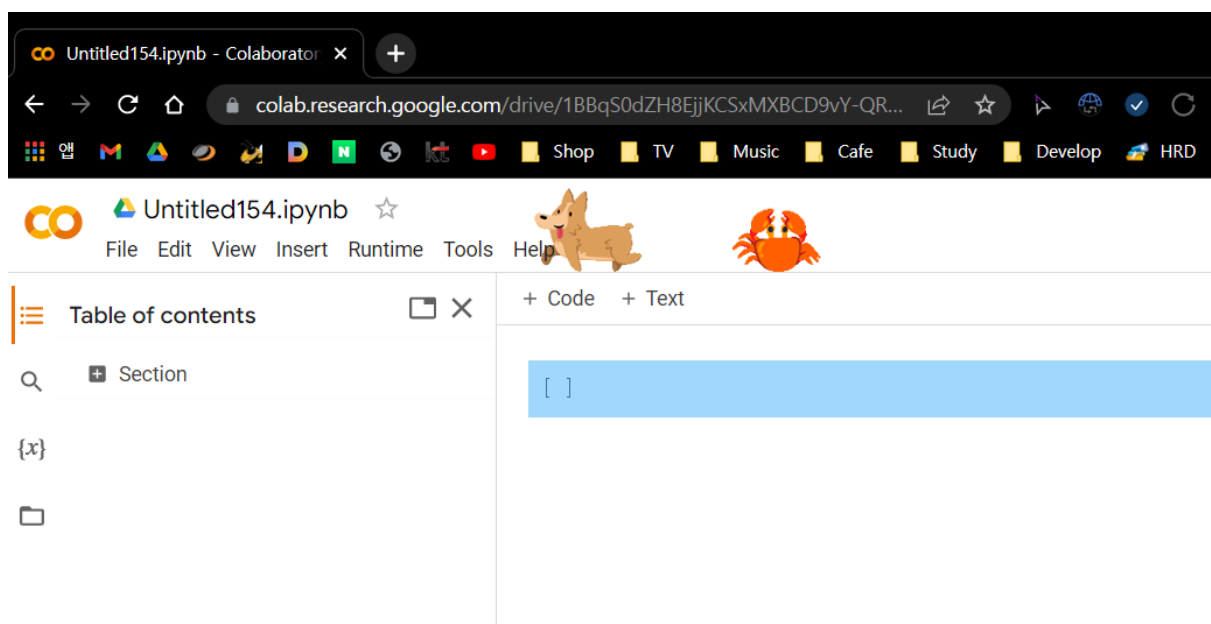




파일 만들때 확장자를 파일명.ipynb 로 적으면 vs 코드에서도 라인바이라인 실행 가능하다.



파이썬 웹 개발환경 Colab



Data Type

자바

```
int a = 20;
```

```
String b = "Hello";
```

```
a= "hi"; (X)
```

파이썬

```
a=20
```

```
b="Hello"
```

```
a="hi" (O)
```

파이썬과 자바스크립트 등은 타입을 강제하지는 않지만 이점이 반드시 장점으로만 작용하지는 않는다.

어떤 프로그래밍 언어든지 그 언어의 데이터 타입을 잘 알고 이해한다면 이미 그언어의 절반을 터득한 것이나 마찬가지다.

특히 파이썬에서는 List, Tuple, Dict, Set 네가지 자료형이 아주 중요한 의미를 갖는다.

숫자형 자료형

- 정수형 Integer 양의 정수 음의 정수 0

```
# 정수형 자료형

a = 100
b = -123
c = 0
```

- 실수형 float 소수점이 포함된 숫자

```
# 실수형 자료형

a = 3.14
b = -5.16
```

- 숫자형 자료형에서의 연산

▶ # 숫자형 자료형에서의 연산

```
a = 3  
b = 4
```

```
# 코랩에서는 print 를 생략해도 알아서 print() 넣어준다. 마지막 예만 넣어준다..  
a + b  
a - b
```

↗ -1

▶
a + b
a - b

```
print(a+b)  
print(a-b)
```

```
# print() 구문을 생략하면 마지막 구문만 출력해주기 때문에 전부 출력하고 싶으면  
# 각각 print() 구문을 다 적어야 한다. 이것은 파이썬의 특징이 아닌 코랩의 특징이다.
```

```
7  
-1
```

▶ # 나머지 연산자 - 홀수 짝수 배수 확인

```
5 / 2 # 일반적인 나머지 구하기
```

↗ 2.5

[8] 5 % 2

```
# 나눈 결과의 몫은 버리고 나머지만 취한다.  
# 따라서 어떤수를 2로 나눈 나머지가 0이 나오면 짝수 1이 나오면 홀수
```

```
1
```

[9] 8 % 2

```
# 어떤수 8를 어떤수 2로 나눈 나머지가 0이면 배수이다.
```

```
0
```

문자열 자료형

- 문자열 String 자료형은 문자 또는 단어로 구성된 글자들의 집합이다.

[10] # 문자열 자료형 - 파이썬에서는 "" 와 ''가 완전히 동일하게 사용된다.

```
# 1 큰 따옴표
"Hello World"

# 2 작은 따옴표
'Hello World'

# 3 3중 따옴표 Doc String
"""Hello World"""
'''Hello World'''
```

```
'Hello World'
```



파이썬에서는 큰따옴표와 작은 따옴표에 전혀 차이가 없지만
아래의 용도로 분리하여 사용된다.

```
"Python's"    # 작은 따옴표 하나를 출력하기 위한 용도
'say "Hello"' # 작은 따옴표 안에서 큰 따옴표를 출력하기 위한 용도
```

```
'say "Hello"'
```



```
# \n 줄바꿈
a = "I love you\nYou love me?"
print(a)
```

```
b = """I love you
You love me?"""
print(b)
```

```
# 예러 발생
# c = "I love you
# You love me "
```



```
I love you
You love me?
I love you
You love me?
```

문자열 연산



문자열 연산

파이썬에서는 문자열도 연산이 가능하다. 특히 다른 언어에서는 발견하기 어려운 파이썬만의
독특한 특징이다. 이런 점들이 바로 파이썬은 우리의 생각을 그대로 반영할수 있다는 파이썬 만의
장점들이다.

```
a = "I "  
b = "love "  
c = "you"
```

```
print(a+b+c)
```

```
d = "love "  
print(d*3)
```

```
e = "="  
print(e *30)
```

```
I love you  
love love love  
=====
```



문자열 연산

파이썬에서는 문자열도 연산이 가능하다. 특히 다른 언어에서는 발견하기 어려운 파이썬만의
독특한 특징이다. 이런 점들이 바로 파이썬은 우리의 생각을 그대로 반영할수 있다는 파이썬 만의
장점들이다.

```
a = "I "  
b = "love "  
c = "you"
```

```
print(a+b+c)
```

```
d = "love "  
print(d*3)
```

```
e = "="  
print(e *30)
```

```
I love you  
love love love  
=====
```

```
# 인덱싱과 슬라이싱 indexing & slicing
# 참고로 인덱싱과 슬라이싱은 리스트와 튜플에서도 완전히 동일하게 사용된다.

# 인덱싱 - 한 개의 글자 추출
a = 'I love You'
print(a[0])
print(a[9])
print(a[-1]) # 마지막 글자 추출은 -1 을 사용한다.
print(a[-3])

# 슬라이싱 - 여러개의 글자 추출
a = 'I love you'
print(a[2] + a[3] + a[4] + a[5])
print(a[2:6]) # 끝자리를 나타내는 수는 미만을 의미한다.
print(a[7:]) # 끝자리를 생략하면 끝까지 출력한다.
print(a[:2]) # 시작 자리를 생략하면 처음 즉 0 부터 시작한다.
print(a[:]) # 모든 자리 생략하면 처음 부터 끝까지 출력한다.
print(a[7:-1]) # 슬라이싱에서도 - 표기를 사용한다. 역시 미만으로 사용된다.
```

```
I
u
u
Y
love
love
you
I
I love you
yo
```

문자열 출력 방법 - 총 4가지 방법 중 2가지를 가장 많이 사용한다.



파이썬에서 출력 방법 총 4가지

1 문자열 포매팅 (중요도 하)

```
print("I %s you" % "love")
print("I ate %d apples" % 5)
print("I ate %s apples" % 5)

print("I %s you, I ate %s apples" % ("love", 5))
```

2 format() 포맷 함수 (중요도 하)

```
print("I {} you".format("love"))
print("I ate {} apples".format(5))
print("I ate {} apples".format(5))

print("I {} you, I ate {} apples".format("love", 5))
```



```
I love you
I ate 5 apples
I ate 5 apples
I love you, I ate 5 apples
I love you
I ate 5 apples
I ate 5 apples
I love you, I ate 5 apples
```



3 f스트링 (중요도 상)

```
a = "love"
b = 5

print(f"I {a} you")
print(f"I ate {b} apples")
print(f"I ate {b} apples")

print(f"I {a} you, I ate {b} apples")
```



```
I love you
I ate 5 apples
I ate 5 apples
I love you, I ate 5 apples
```





4 연산자 사용

```
a = "love"  
b = 5  
  
print("I " + a + " you")  
print("I ate" , b , "apples")  
print("I ate" , b , "apples")  
print("I " + a + " you, I ate" , b , "apples")  
print("I" , a , " you")  
print("I ate" , b , "apples")  
print("I ate" , b , "apples")  
print("I" , a , " you, I ate" , b , "apples")
```



```
I love you  
I ate 5 apples  
I ate 5 apples  
I love you, I ate 5 apples  
I love you  
I ate 5 apples  
I ate 5 apples  
I love you, I ate 5 apples
```

Google Colaboratory

 <https://colab.research.google.com/drive/1BBqS0dZH8EjjKC/SxMXBCD9vY-QR2A8be?usp=sharing>



```
C:\  
λ cd develop  
  
C:\develop  
λ cd cmder  
  
C:\develop\cmder  
λ |
```