

LESSON

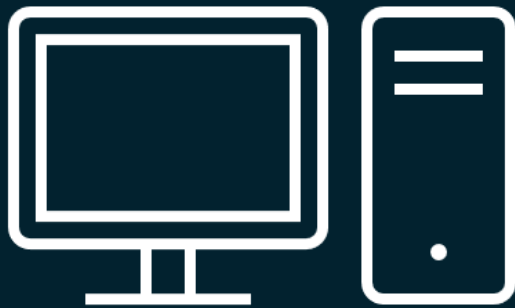
# 프로그래밍 언어

강사: 김은숙

# 프로그래밍 언어란?

---

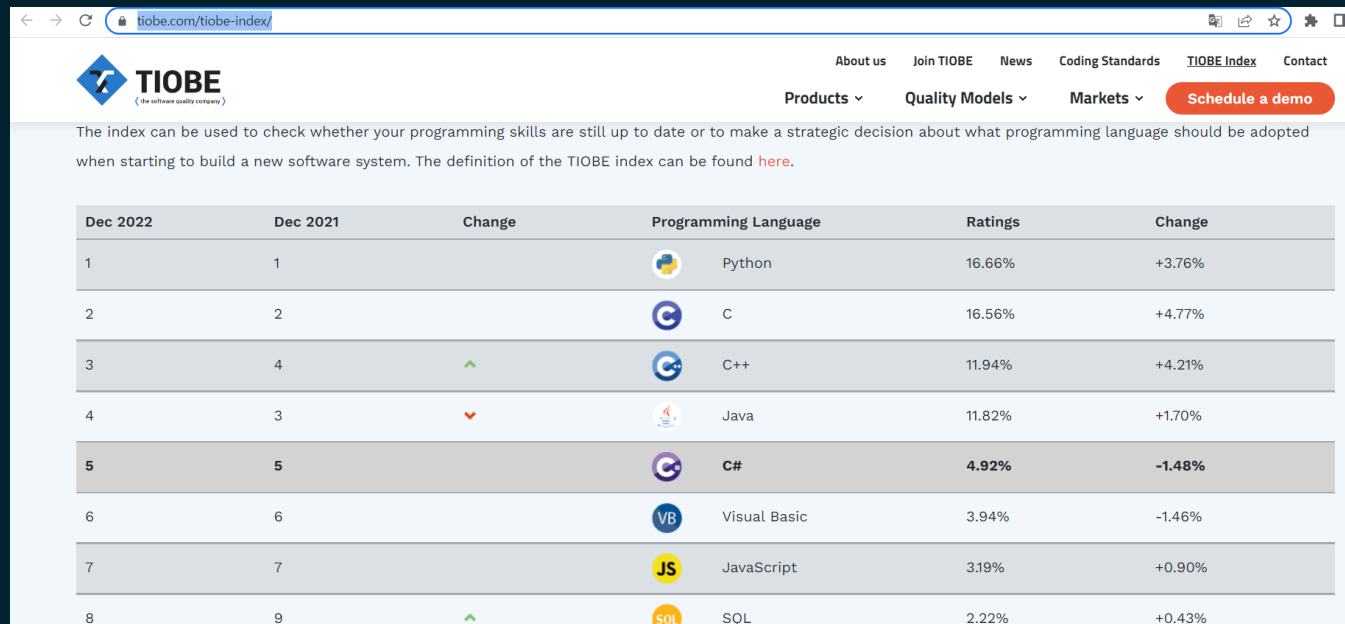
“컴퓨터를 작동하는 소프트웨어를 만드는 도구”











하드웨어

# 프로그래밍 언어 순위

<https://www.tiobe.com/tiobe-index>



The screenshot shows the TIOBE Index website. The header includes the TIOBE logo and navigation links: About us, Join TIOBE, News, Coding Standards, TIOBE Index, and Contact. Below the header, there are dropdown menus for Products, Quality Models, and Markets, along with a red button labeled 'Schedule a demo'. A paragraph explains the index's purpose: 'The index can be used to check whether your programming skills are still up to date or to make a strategic decision about what programming language should be adopted when starting to build a new software system. The definition of the TIOBE index can be found [here](#).'

Dec 2022	Dec 2021	Change	Programming Language	Ratings	Change
1	1		 Python	16.66%	+3.76%
2	2		 C	16.56%	+4.77%
3	4	▲	 C++	11.94%	+4.21%
4	3	▼	 Java	11.82%	+1.70%
5	5		 C#	4.92%	-1.48%
6	6		 Visual Basic	3.94%	-1.46%
7	7		 JavaScript	3.19%	+0.90%
8	9	▲	 SQL	2.22%	+0.43%

<https://pyp1.github.io/PYPL.html>

LESSON

# 프로그래밍 언어 종류

# 기계어(Machine Language)

- 2진수인 0과 1로 프로그래밍하는 언어

1000 1100 1010 0011 1111  
0000 1010 1110 0001 0001  
1000 1100 1010 0011 1111  
0000 1010 1110 0001 0001

+ : 11110  
- : 11111  
이라고 가정함.

3 + 4

작성할 코드 11110 0011 0100

5 - 6

작성할 코드 11111 0101 0110

# 어셈블리어(Assembly Language)

- 특정 기계어에 1:1 대응하는 단어를 지정해서 만든 언어.

ex) 1000 1010 -> ADD

1000 1011 -> SUB

+ : 11110 대신 ADD  
- : 11111 대신 SUB  
로 만든 상태로 가정함.

	3	+	4
작성할 코드	ADD a1, b1		

	5	-	6
작성할 코드	SUB a1, b1		

# 고급 언어(High-Level Language)

- 사람이 이해하기 쉽게 만들어진 언어.
- 기계어로 번역하는 **번역기(컴파일러, 인터프리터)** 프로그램이 필요함.

$$\boxed{3} + \boxed{4}$$

작성할 코드 **3+4**

$$\boxed{5} - \boxed{6}$$

작성할 코드 **5-6**

LESSON

# 언어 번역기

## (Language Translator)



# 어셈블러(Assembler)

컴퓨터는 기계어만 이해하므로 번역하는 어셈블러 프로그램이 필요함.  
어셈블리어 코드를 기계어로 번역하는 프로그램

```
push    ebp
mov     ebp, esp
sub     esp, 0E4h
push    ebx
push    esi
push    edi
lea     edi, [ebp+FFFFFF1Ch]
mov     ecx, 39h
mov     eax, 0CCCCCCCCh
```



어셈블러  
(Assembler)

```
1000 1100 1010 0011 1111
0000 1010 1110 0001 0001
0000 1010 1110 0001 0001
```

# 컴파일러(Compiler), 인터프리터(Interpreter)

# 고급 언어를 기계어로 번역하는 번역기 프로그램



```
a=10
b='hello'
c=10
print(a,b,c)
```

# 컴파일러 인터프리터

1000 1100 0011 1111  
0000 1010 1110 0001 0001  
0000 1010 1110 0001 0001

# 컴파일러 vs 인터프리터

Elsa?

엘사 언니?

Do you wanna build a  
snowman?

같이 눈사람 만들래?

Come on, let's go and play!

빨리 나가 놀자

Elsa?

Do you wanna build a  
snowman?

Come on, let's go and play!

엘사 언니?

같이 눈사람 만들래?

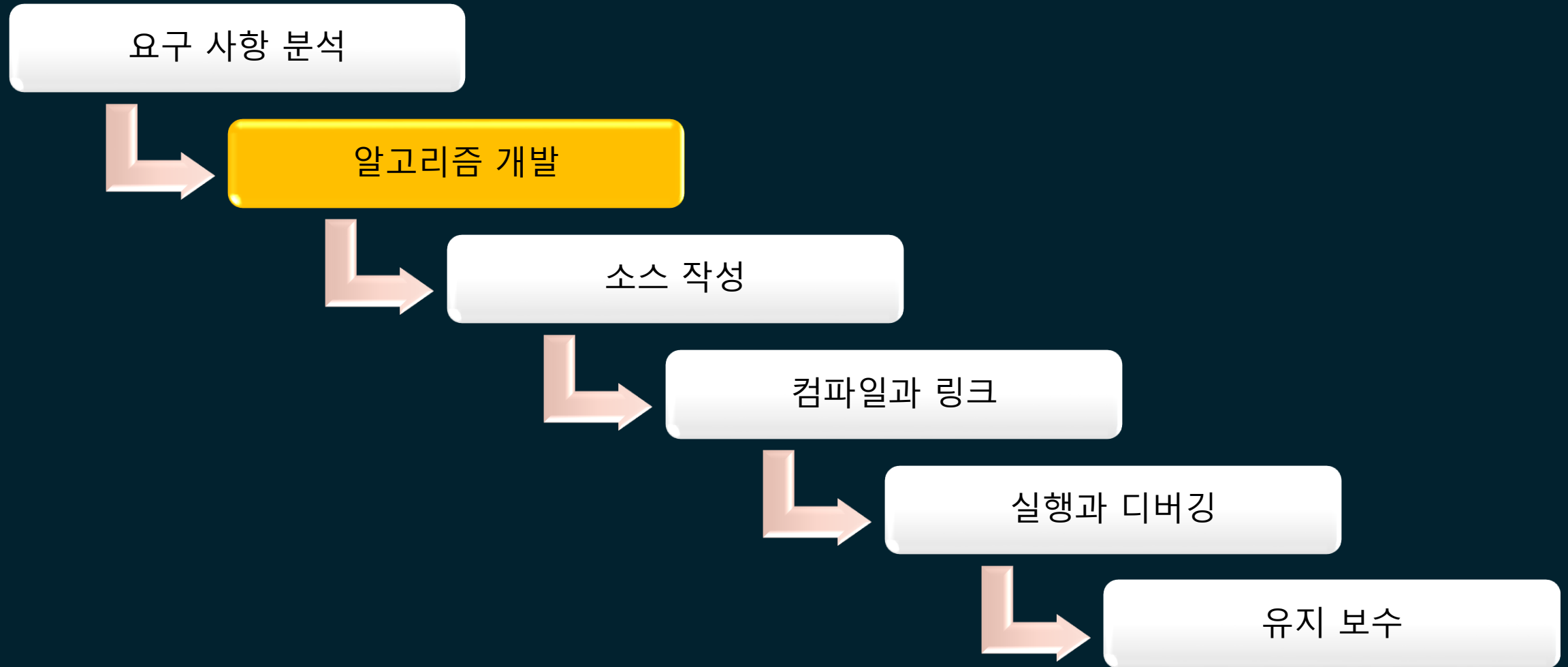
빨리 나가 놀자

LESSON

# 프로그램 개발 과정 알고리즘(Algorithm)

# 프로그램 개발 과정

---



# 알고리즘(Algorithm)이란?

요리를 예로 들어보자.

빵을 만들기 위해 오븐과 재료들을 준비하고 오븐 사용법도 배웠다.



그렇다면 누구나 빵을 만들 수 있는가?

빵을 **어떻게 만드는지** 알아야 빵을 만들 수 있다!

## Recipe

1. 빈 그릇을 준비한다.
2. 이스트를 밀가루, 우유에 넣고 저어준다.
3. 버터, 설탕, 계란을 추가로 넣고 섞는다.
4. 따뜻한 곳에 놓아두어 발효시킨다.
5. 170~180도의 오븐에서 굽는다.

프로그래밍 언어를 안다고 해서 프로그램을 작성할 수 있는 것은 아니다!!!

LESSON

# 파이썬(Python)

# 파이썬(Python)

1991년 네델란드의 수학자 귀도 반 로섬 (Guido van Rossum)이 C언어로 제작한 언어.

파이썬의 로고는 비단뱀 2마리가 꼬여 있는 모습을 형상화한 것.

그리스어 '피톤(python)'에서 유래했으며 '비단뱀'이라는 뜻을 가지고 있다.



파이썬은 C로 만든 CPython, 자바로 만든 JPython, C#으로 만든 IronPython 등이 있다. 우리가 일반적으로 사용하는 파이썬은 CPython이다.



# 파이썬(Python)의 특징

---

## 인터프리터 언어(Interpreter Language)

인터프리터에 의해 한 줄씩 읽어 바로 실행 결과를 확인할 수 있는 언어이다.

## 동적(Dynamic) 타입 언어

변수의 자료형을 지정하지 않고 선언만으로 값을 지정할 수 있다.

## 플랫폼에 독립적인 언어

운영체제(윈도우, 리눅스, 맥 등)에 관계없이 같은 코드를 사용할 수 있다.



플랫폼이란?

프로그램이 돌아가는 운영체제(OS: Operating System)환경이나 개발환경을 말한다.

# 파이썬(Python)의 장단점

---

## 장점

배우기 쉽다.

읽고 쓰기 쉬워서 유지 보수가 쉽고 개발 속도가 빠르다.

무료로 제공되는 수많은 라이브러리가 있다.

## 단점

컴파일 언어에 비해 속도가 느리다.



라이브러리(Library)?

어떠한 작업을 수행하는 로직을 재사용 가능하도록 모듈화 시켜서 저장한 프로그램.

# 파이썬으로 할 수 있는 것들

---

- 웹 데이터 수집
- 웹 애플리케이션 개발

파이썬으로 제작된 응용프로그램들>>>드롭박스, 인스타그램, 핀터레스트...

- 데이터 분석
- 데이터 시각화 <https://matplotlib.org/>
- AI(인공지능)
- 업무 자동화

LESSON

# 파이썬 설치

# 파이썬 인터프리터 설치

파이썬 언어로 작성된 소스

```
import time
st=time.time()
li=[]
for i in range(1000000):
    li.append(i)

et=time.time()
print('for문:',et-st)
```

인터프리터

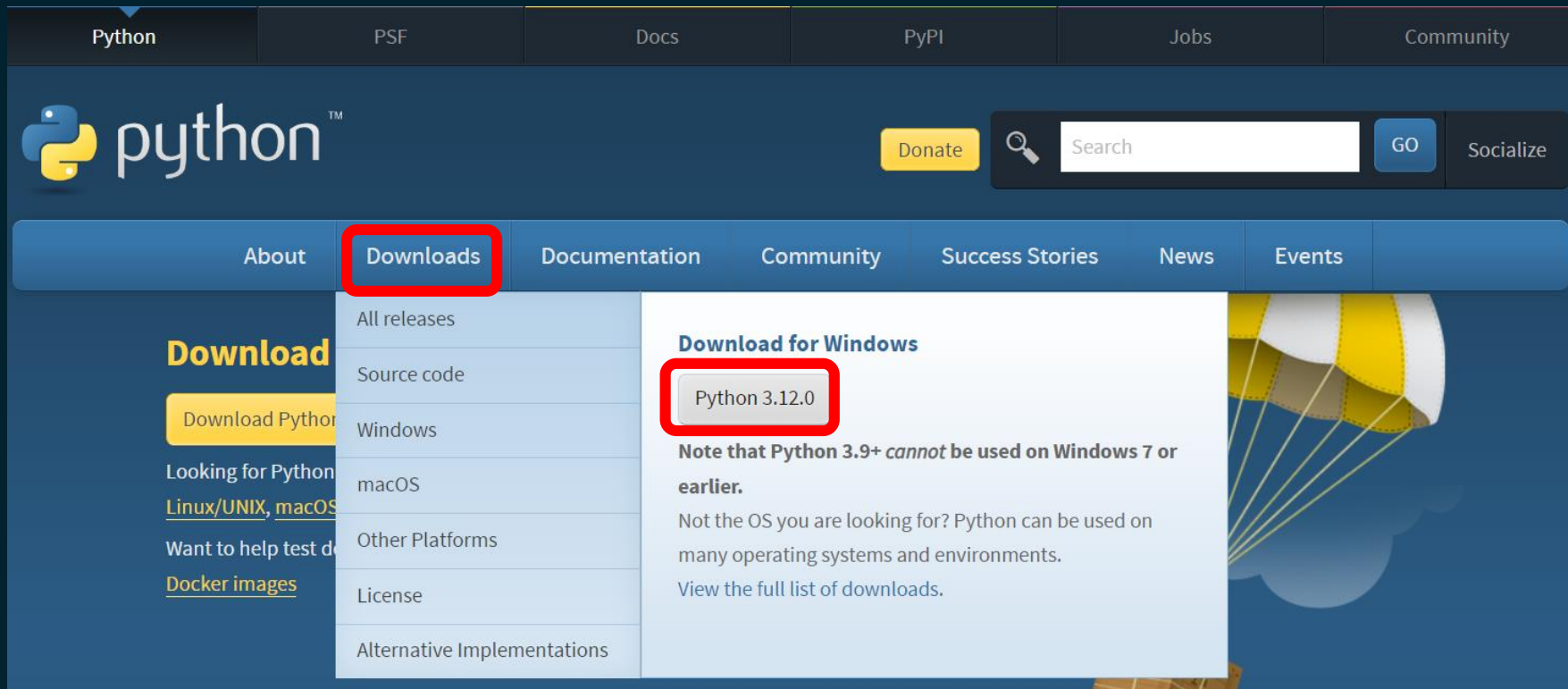
기계어

```
1000 1100 1010 0011 1111
0000 1010 1110 0001 0001
0000 1010 1110 0001 0001
1000 1100 1010 0011 1111
0000 1010 1110 0001 0001
0000 1010 1110 0001 0001
1000 1100 1010 0011 1111
```

# 파이썬 설치하기

1. <https://www.python.org> 에 접속한다.

2 [Downloads] 메뉴에 마우스를 올리면 사용중인 운영체제에 적합한 다운로드 버튼이 나타난다.



# 파이썬 설치하기

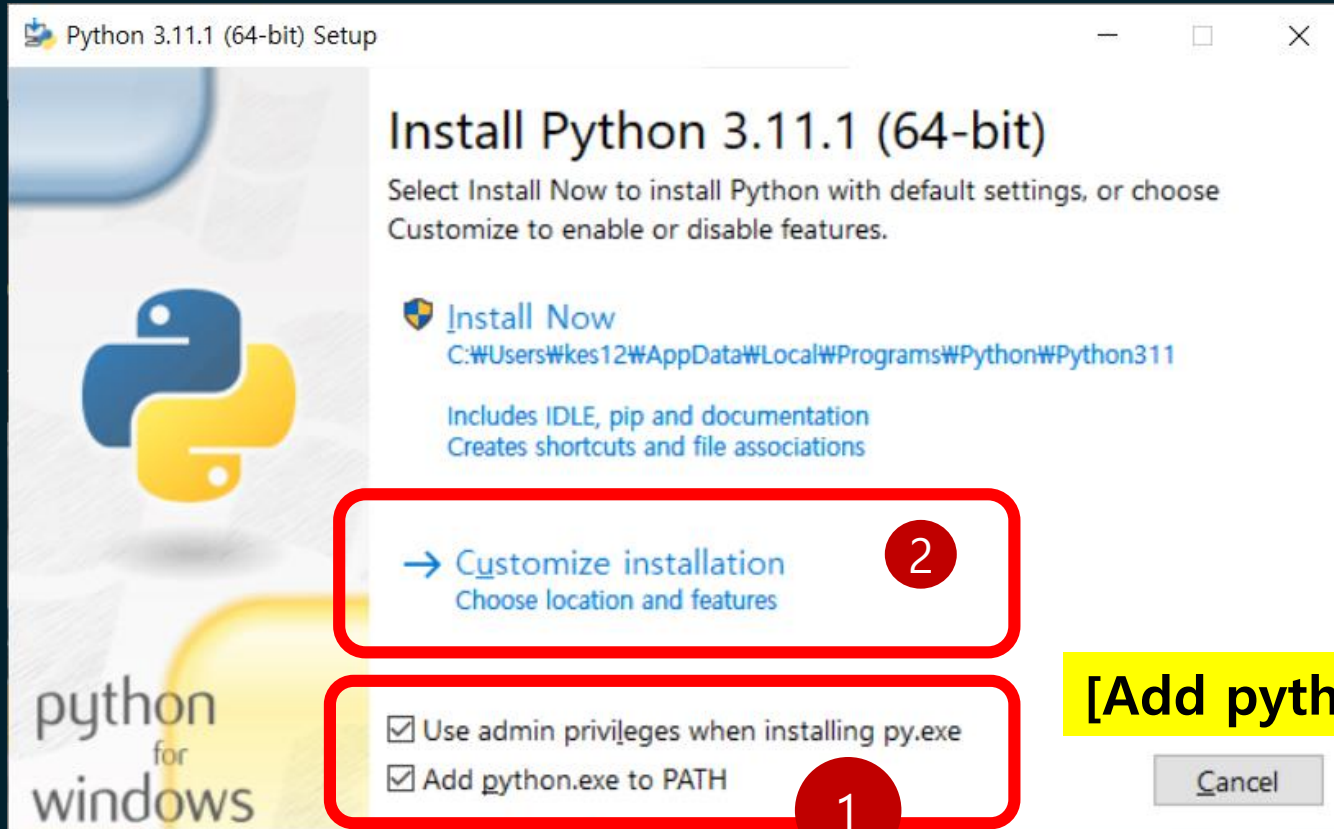
---

3. 다운로드 받은 파일을 실행한다.

설치 첫 단계에서 혹시 [이 앱이 디바이스를 변경할 수 있도록 허용하시겠습니까?] 라는 대화 상자가 나타난다면 [예]를 선택한다.

# 파이썬 설치하기

4. 다운로드 받은 파일을 실행하고 다음 화면과 같이 선택한 후 [**Customize installation**] 클릭한다.

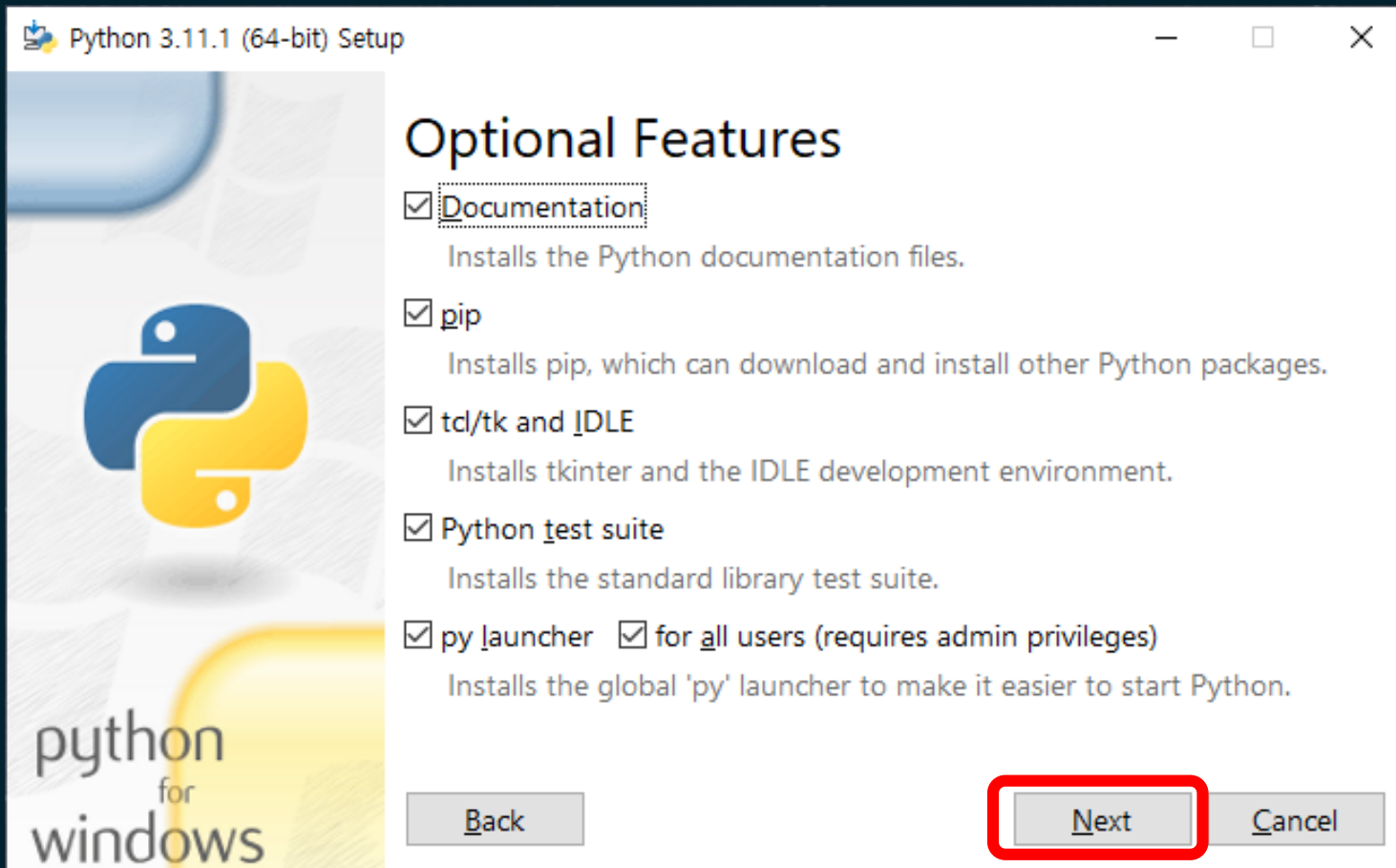


[Add python.exe to PATH]를 반드시 선택할 것



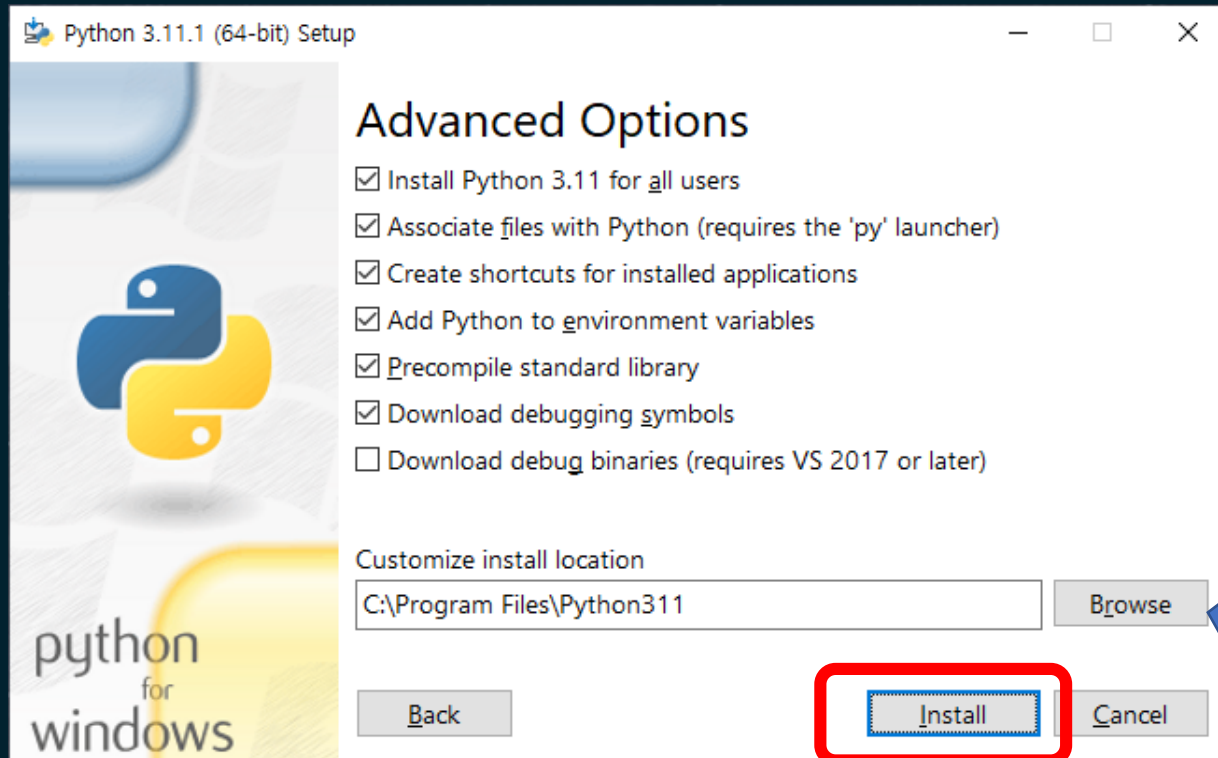
# 파이썬 설치하기

5. 다음 화면과 같이 모두 선택하고 [Next]를 클릭한다.



# 파이썬 설치하기

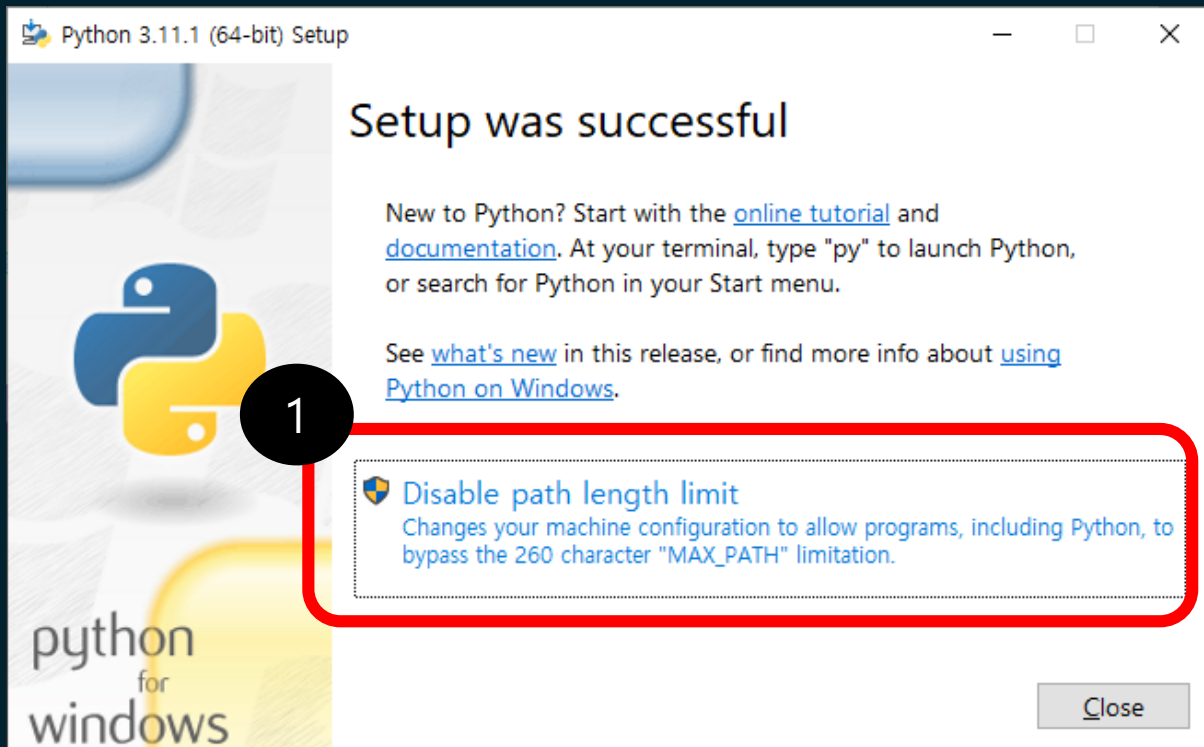
6. 다음 화면과 같이 선택하고 적절한 설치 경로를 설정한 후 [Install]을 클릭하면 설치가 진행된다.



설치를 원하는 폴더를 지정해도 된다.  
"Browse"를 클릭해서 설치할 폴더를 지정하면 된다.

# 파이썬 설치하기

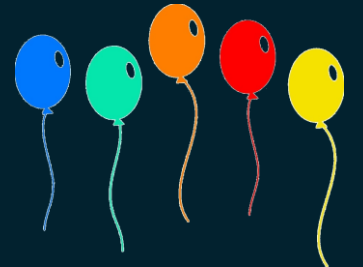
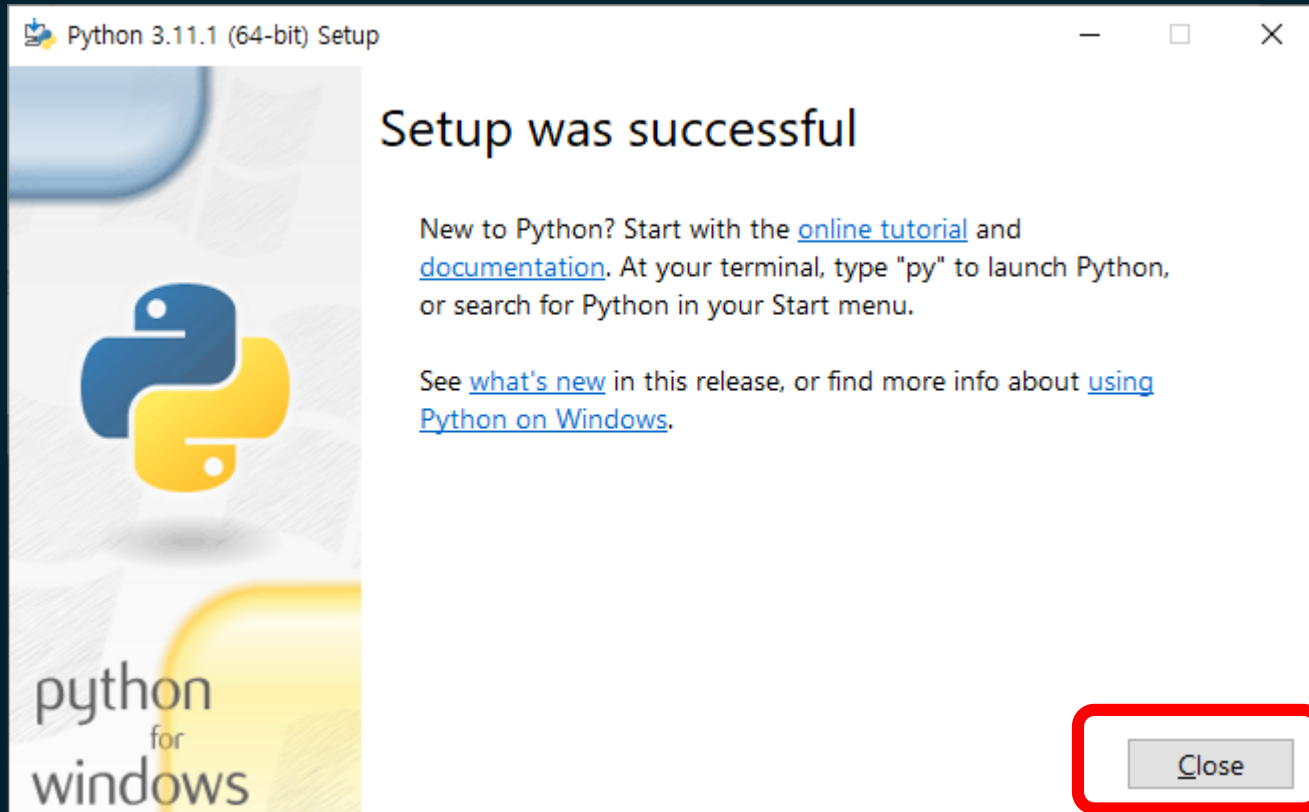
7. 다음 단계에서는 반드시 [Disable path length limit]를 클릭한다.  
Windows는 파일 이름과 경로의 길이가 260자로 제한되어 있다. 260자를 넘더라도 문제가 발생하지않도록 길이 제한을 해제하는 것이다.



(참고) 컴퓨터에 파이썬 설치한적이 있는 경우 이 화면은 나오지 않음

# 파이썬 설치하기

8. 다음 화면이 나타나면 [Close]를 클릭하여 설치를 완료한다.



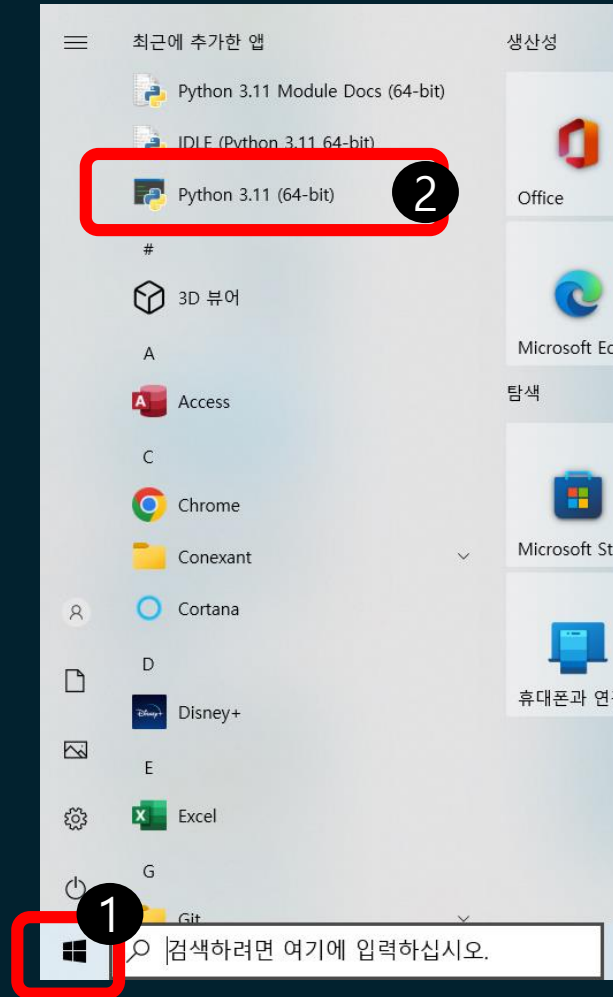
축하합니다!!!

LESSON

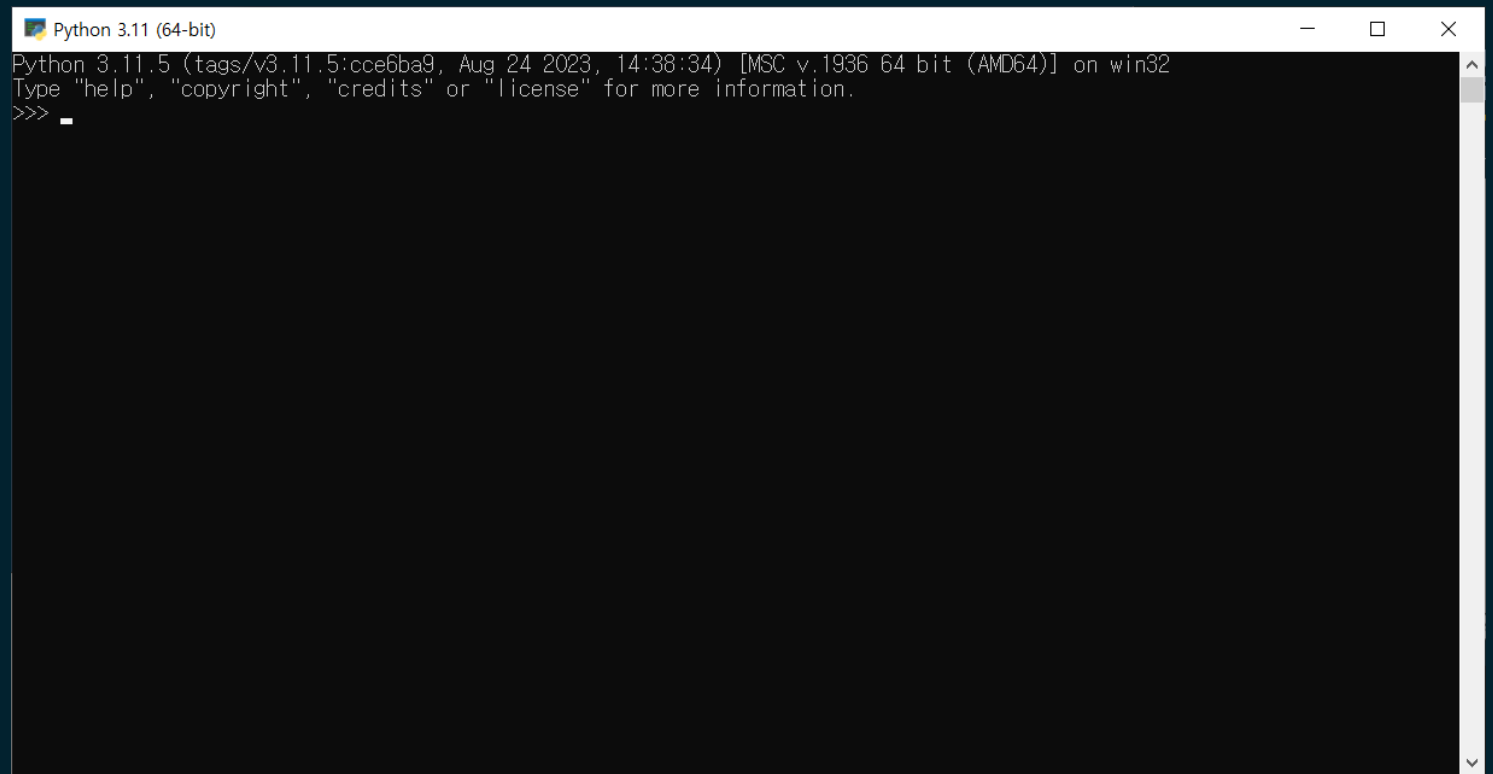
# 파이썬 실행하기

# 파이썬을 실행하는 방법1


1. [시작]버튼을 누르면 방금 설치했던 Python 3.xx이 표시된다.

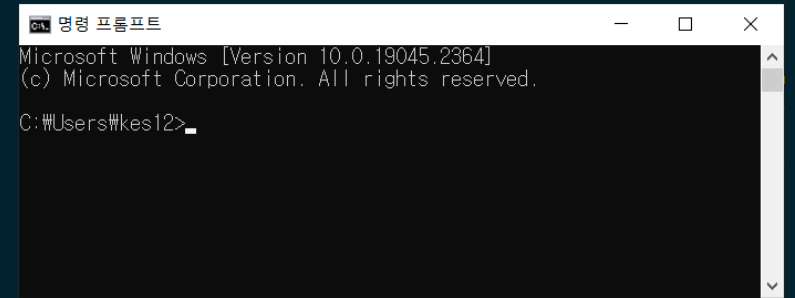
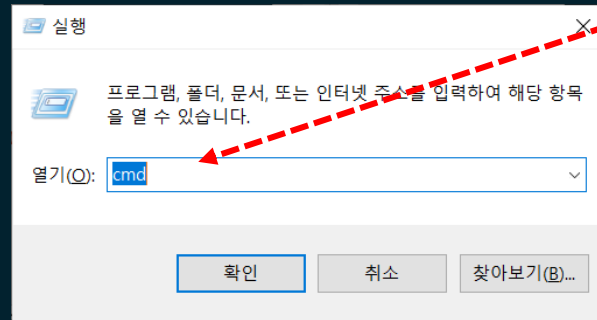


클릭하면 파이썬 인터프리터(셸)가 아래와 같이 나타난다.

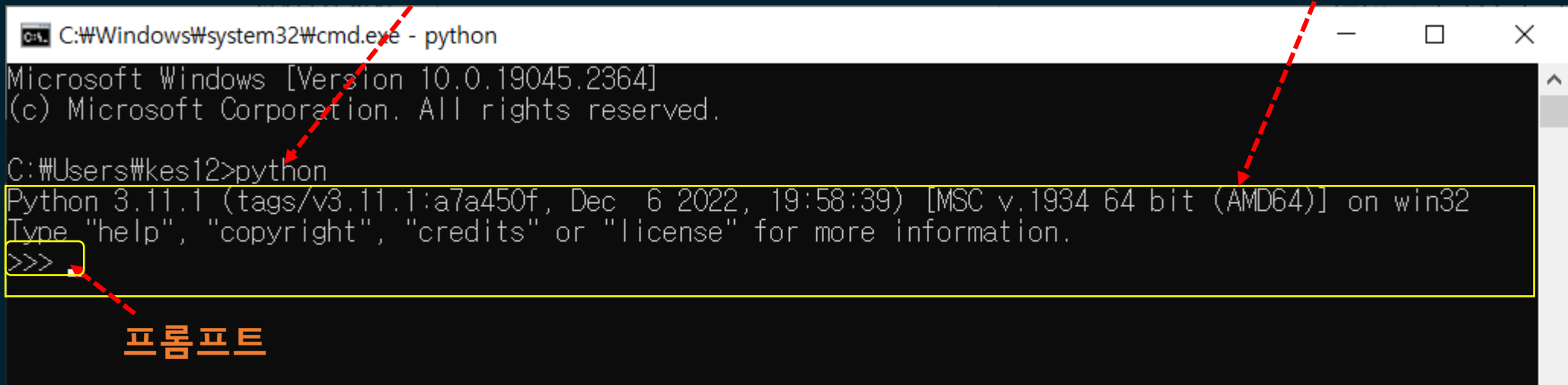


# 파이썬 실행하는 방법2

1. 키보드에서  + R 을 눌러 [실행]창이 나타나면, 'cmd'를 입력하고 엔터키를 누르면 명령 프롬프트 창이 표시 된다.



2. 명령 프롬프트에 "python"을 입력하고 엔터를 누르면 아래와 같이 파이썬 셸이 표시됨.



# 파이썬 실행하는 방법3

---

통합개발환경(IDE)에서 실행

컴퓨터로 문서를 작성하려면  
메모장?  
워드?



LESSON

# 통합개발환경(IDE)

Integrated Development Environment

효율적으로 소프트웨어를 개발하기 위한 프로그램.

“소스 코드 작성, 디버깅(debugging), 실행”



LESSON

# 파이참(PyCharm) 설치



LESSON

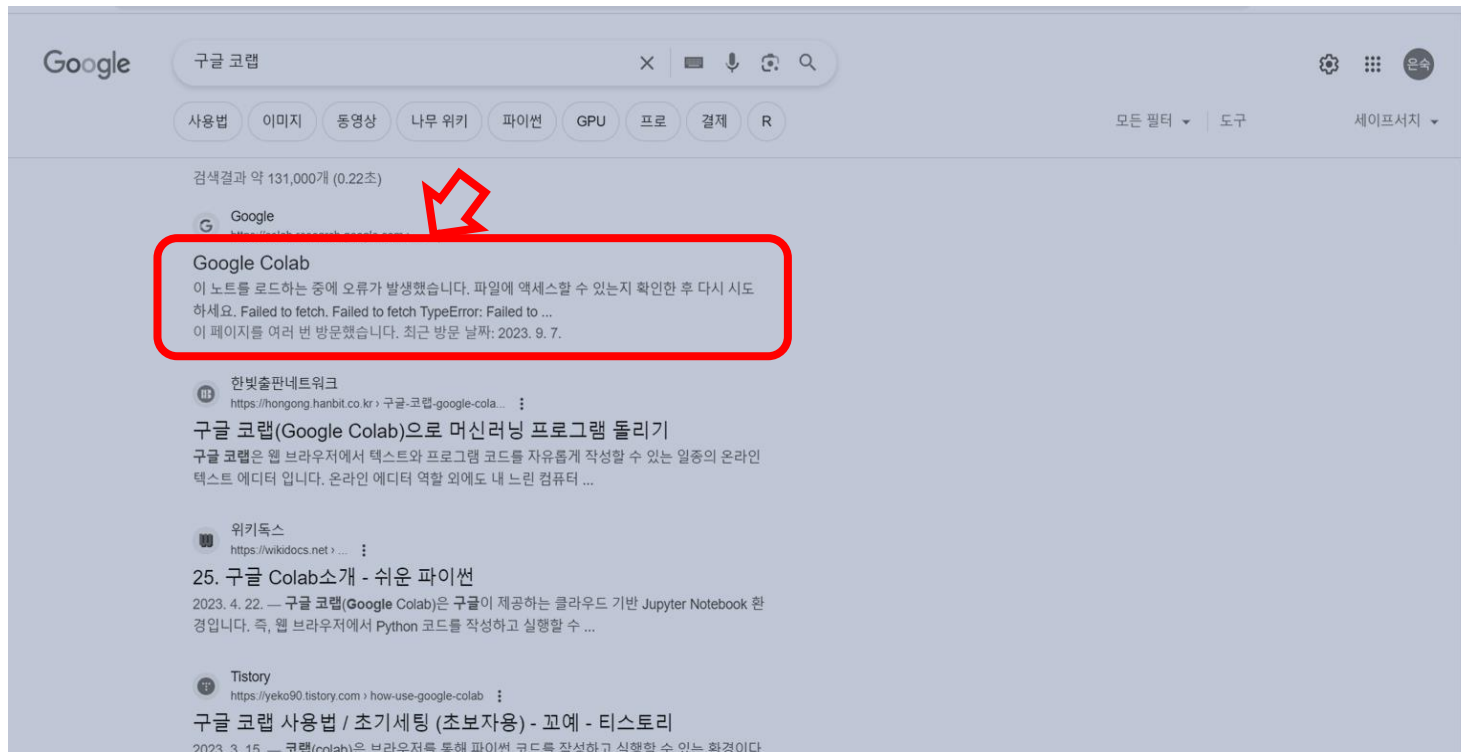
# Google의 Colab

# 구글 로그인

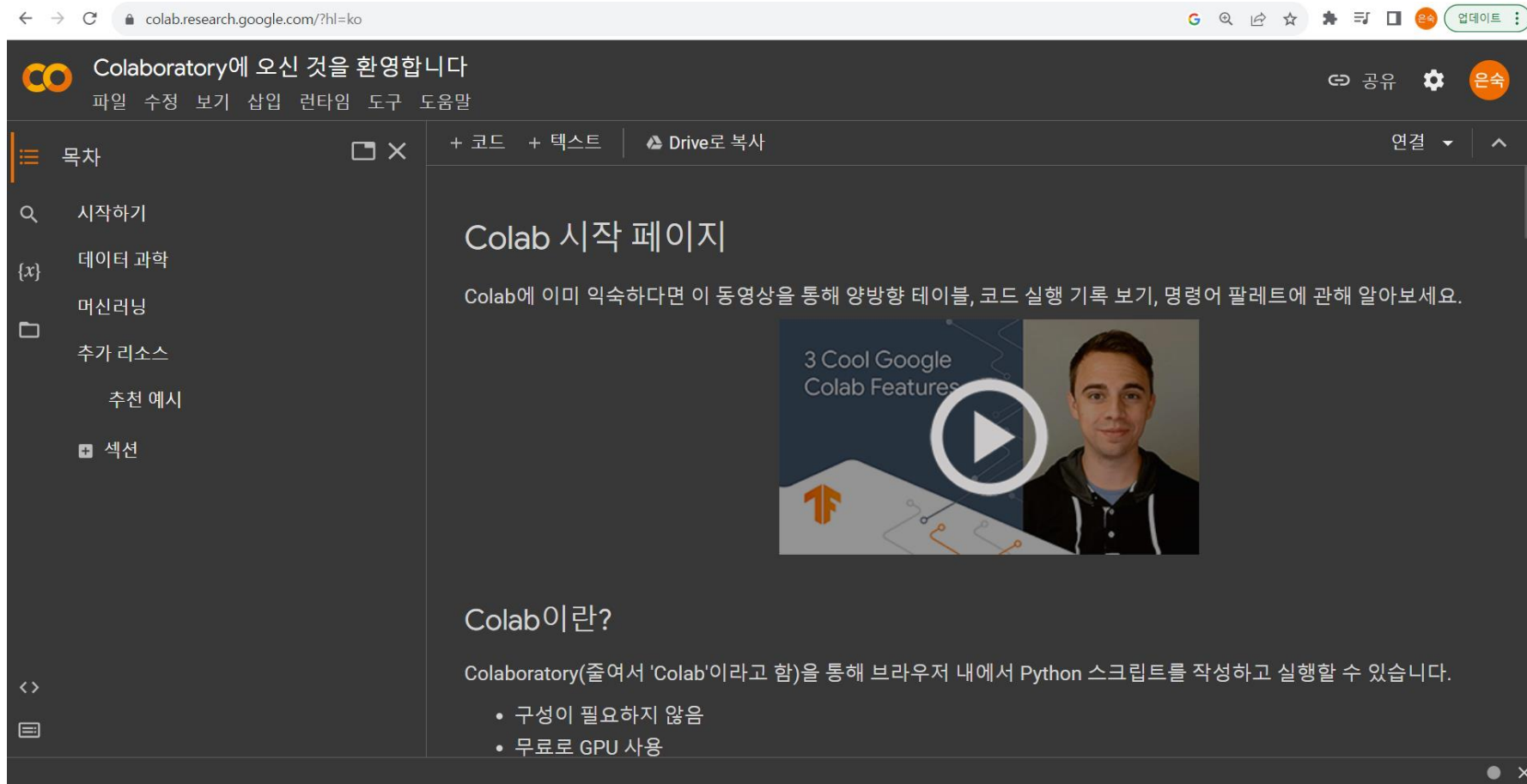
- 각자 자신의 계정으로 로그인한다.

# Colab 접속

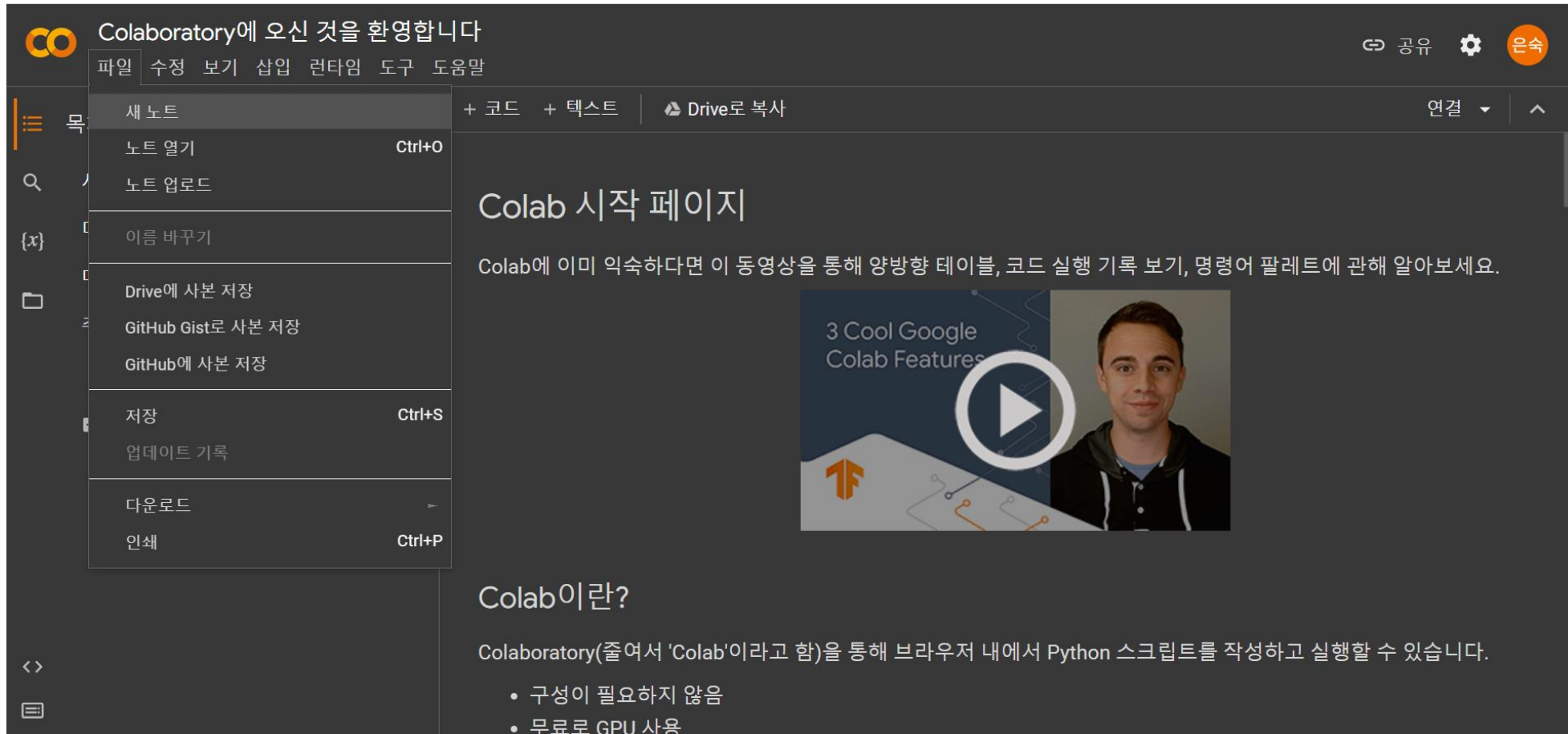
- 구글 검색창에 "구글 Colab" 입력 후 아래 클릭



# Colab 접속후 화면



# 새로운 파일 만들기



Colaboratory에 오신 것을 환영합니다

파일 수정 보기 삽입 런타임 도구 도움말


새 노트  
노트 열기 Ctrl+O  
노트 업로드  
이름 바꾸기  
Drive에 사본 저장  
GitHub Gist로 사본 저장  
GitHub에 사본 저장  
저장 Ctrl+S  
업데이트 기록  
다운로드  
인쇄 Ctrl+P

+ 코드 + 텍스트 | Drive로 복사

연결 ^

## Colab 시작 페이지

Colab에 이미 익숙하다면 이 동영상상을 통해 양방향 테이블, 코드 실행 기록 보기, 명령어 팔레트에 관해 알아보세요.



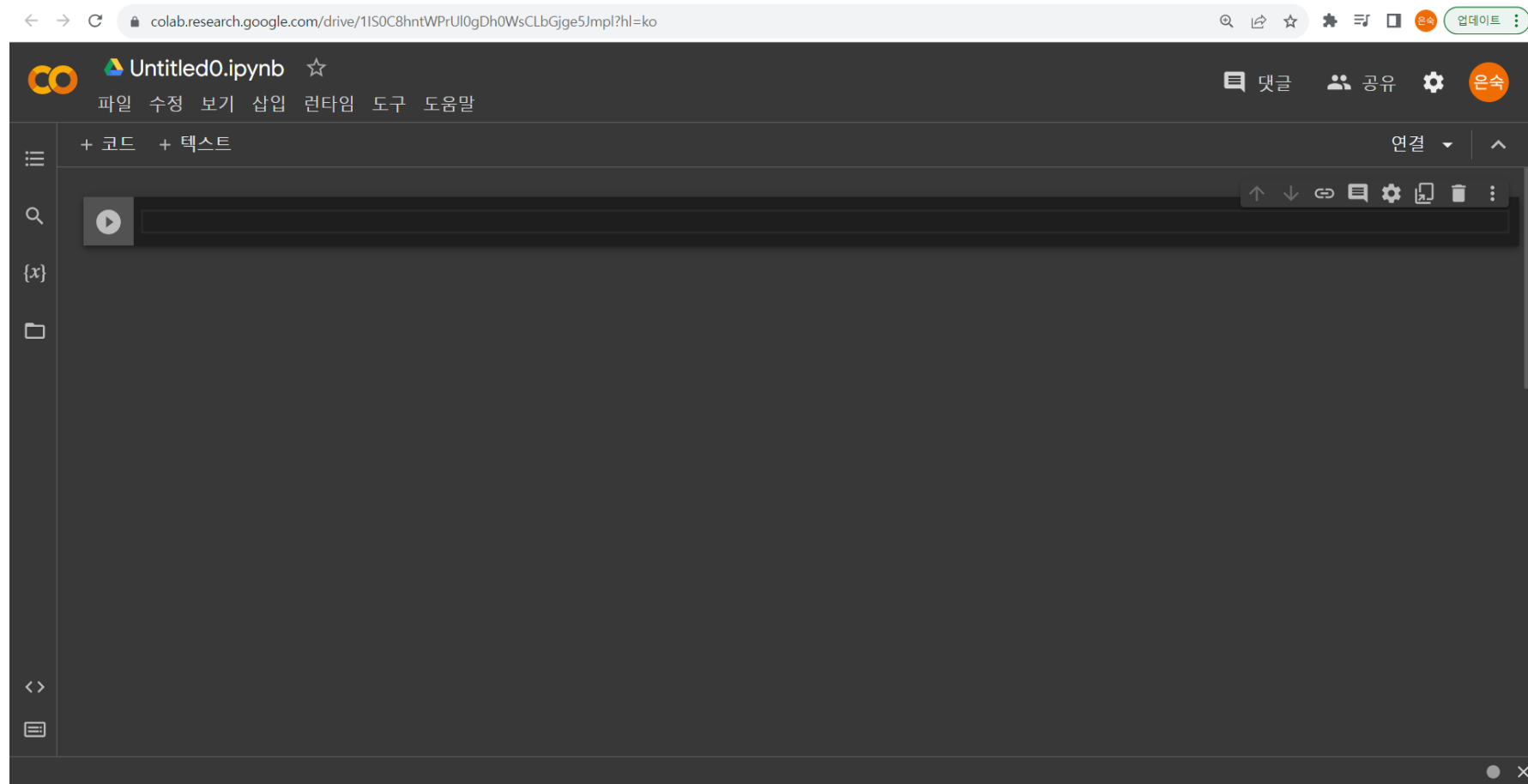
## Colab이란?

Colaboratory(줄여서 'Colab'이라고 함)을 통해 브라우저 내에서 Python 스크립트를 작성하고 실행할 수 있습니다.

- 구성이 필요하지 않음
- 무료로 GPU 사용



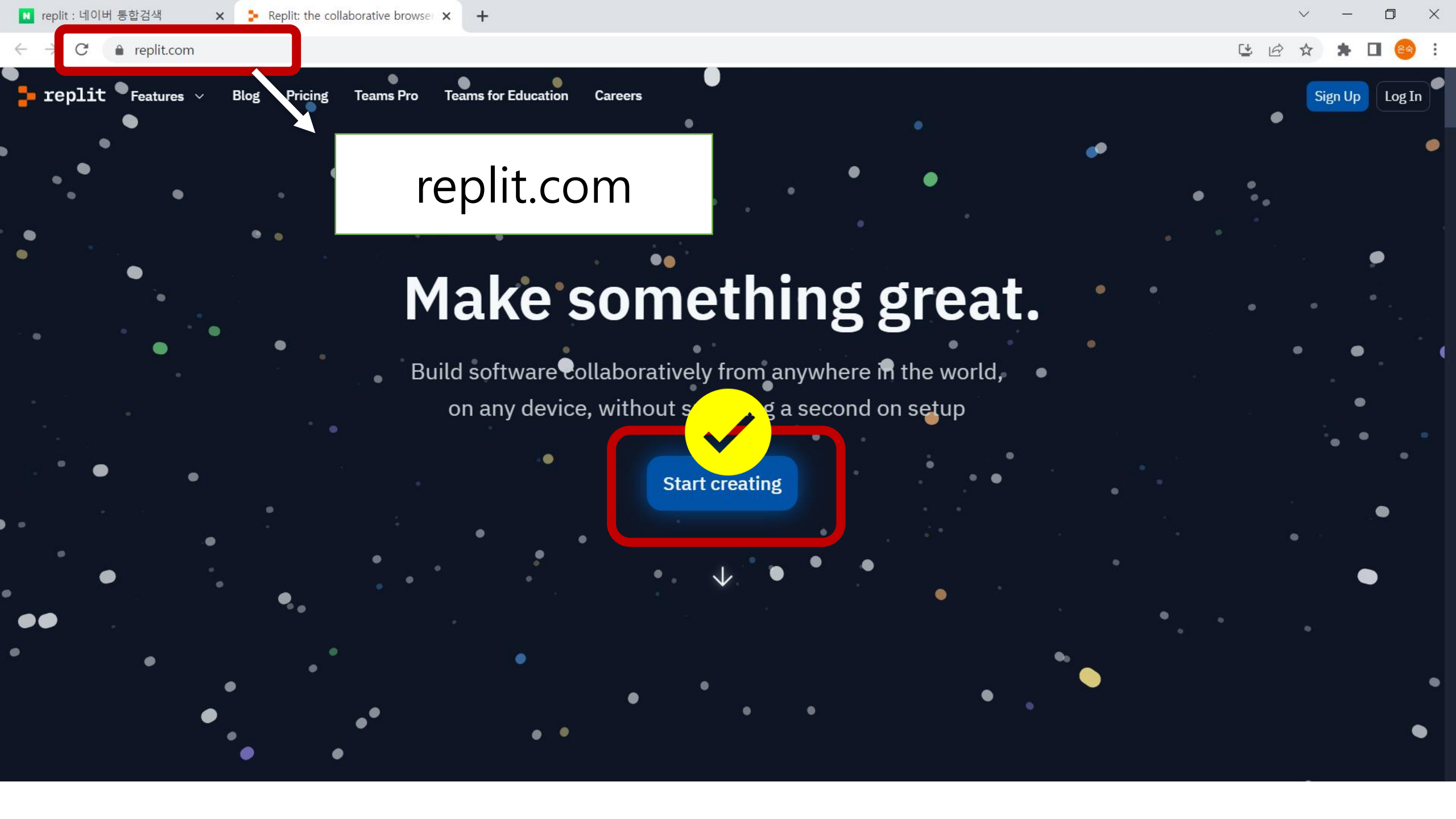
# 새로 만들기후 화면



LESSON

# 파이썬 웹 IDE

[replit.com](https://replit.com)



replit.com

replit.com

# Make something great.

Build software collaboratively from anywhere in the world,  
on any device, without setting up a second machine

Start creating



## Create a Replit account

Teacher? Sign up for our education product





+ Create Account



Have an account? [Log In](#)

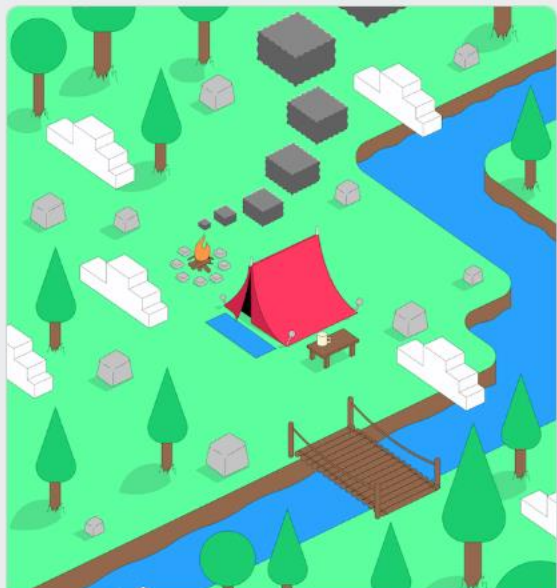
Trouble signing up? [Get help](#)

By continuing, you agree to Replit's [Terms of Service](#) and [Privacy Policy](#), and to receiving emails with updates.

-  Continue with Google
-  Continue with Github
-  Continue with Facebook
-  Continue with Apple



## How familiar are you with programming?



### Not at all

Never written any code before



### Beginner

Written some lines of code before



### Intermediate

Built a working product, like an app, game, website

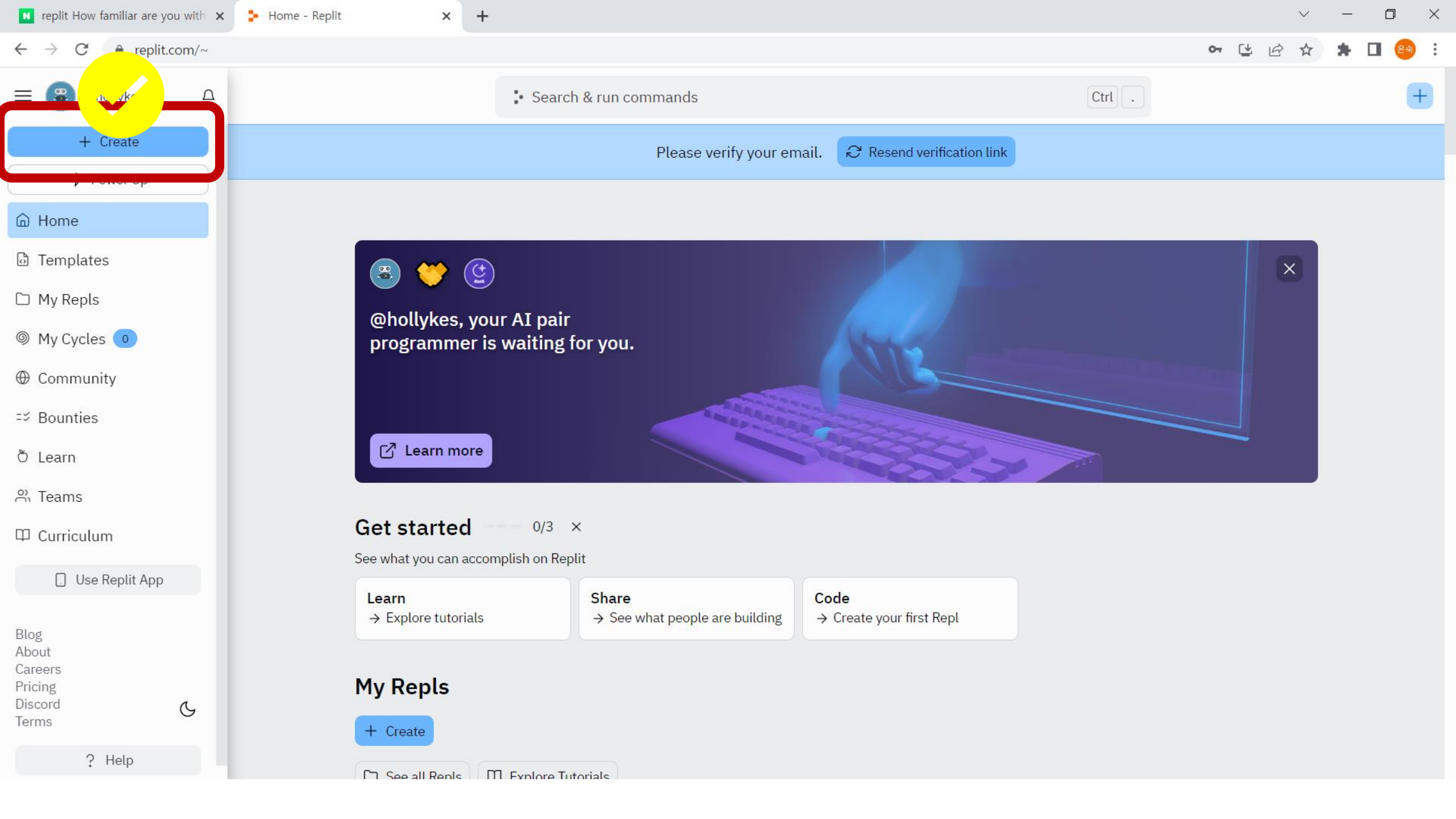


### Expert

Built and launched multiple projects



✓ Skip



repl: 네이버 통합검색

Home - Replit

replit.com/~

@hollykes

+ Create

Power Up

Home

Templates

My Repls

My Cycles 0

Community

Bounties

Learn

Teams

Curriculum

Use Replit App

Blog

About

Careers

Pricing

Discord

Terms

Help

Search & run commands

Ctrl .

Please verify your email. [Resend verification link](#)

Create a Repl

Import from GitHub

Template

Python

Python

Python is a high-level, interpreted, general-purpose programming language.

replit 2K + 16.6M

Title

ClassicCandidDowngrade

Public

Anyone can view and fork this Repl.

Power Up to make private

+ Create Repl

원하는 이름 입력



Q Search

Files

- main.py

Packager files

- poetry.lock
- pyproject.toml

Tools

- Docs
- Chat
- Threads
- Packages
- Git
- Debugger
- Shell
- Help

Try Ghostwriter

CPU RAM Storage

main.py x +

```
1 Not sure what to do? Run some examples or generate code with Ghostwriter (start typing to dismiss)
```

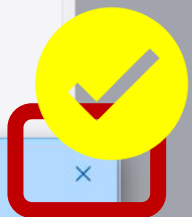
Line 1 : Col 1

\_ Console x Shell x +

Welcome to the Workspace

The Workspace is where you create projects on Replit. These are called **Repls**.  
Let's explore around!

> Next





main.py - ClassicCandidDowngrade

replit.com/@hollykes/ClassicCandidDowngrade#main.py

ClassicCandidDowngrade  
hollykes

Run

Invite

Search

Files

main.py

introduce.txt

Package files

poetry.lock

pyproject.toml

Tools

Docs

Chat

Threads

Packages

Git

Debugger

Try Ghostwriter

CPU

RAM

Storage

Help

main.py

1 f = open('introduce.txt', 'w')

2 f.write('안녕하세요.\n')

3 f.write('python을 공부중입니다.')

4 f.close()

5

Console

Shell

탭을 드래그하면 탭을 분리할 수 있다.

Line 5 : Col 1

History

LESSON

# 웹 저장소 Github

# github 가입하기

## github.com

