

1주차

디자인 사고 및 컴퓨터 사고의 이해와 응용

Python으로 배우는 SW 디자인

주제 | 개요(Introduction) 및 개발 환경 구축

Contents

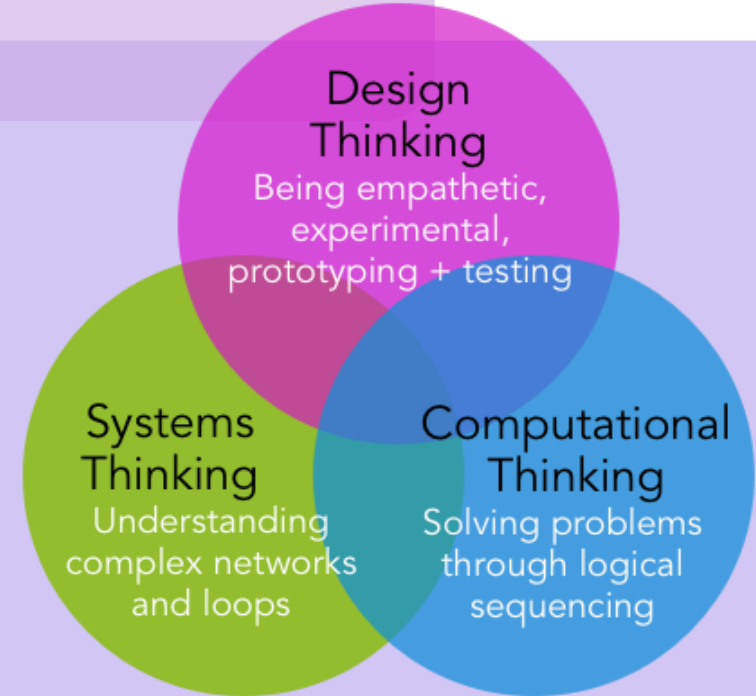
- 0. 커리큘럼 소개
- 1. 4차 산업혁명 시대에 갖추어야 할 사고
그리고 Python
 - 1) 디자인 사고
 - 2) 컴퓨팅 사고
 - 3) 파이썬(Python)
- 2. 개발환경 설정하기
 - 1) Anaconda 설치하기
 - 2) 가상환경 만들기
 - 3) 대화형 모드
 - 4) Pycharm
 - 5) Github에 프로젝트 올려보기

Python 기초반 커리큘럼 소개

- 수업 목표: python 기초 문법과 이를 활용한 간단한 프로젝트 진행
- 대상: Python을 처음 시작하는 중고등학생
- 커리큘럼
 - python 기초 문법 (turtle 모듈+Mission) [8주]
 - tkinter로 다양한 GUI 프로그램 만들기 [2주]
 - “갤러그“ 게임 만들기 [2주]

교과 목표

- 4차 산업혁명 시대를 이끌어갈 인재의 역량을 키우고자 함
 - **Design Thinking** (디자인 사고)
 - **Computational Thinking** (컴퓨팅 사고)
 - ***Systems Thinking** (시스템 사고)



- ➔ 문제해결능력을 향상시켜 다양한 문제 상황에 능동적으로 대처
- ➔ 디자인 사고라는 역동적인 **디자인적 방법론**에 익숙해지고 통합적 사고를 기르도록 함
- ➔ 문제 분해, 패턴 인식, 추상화, 알고리즘이라는 요소들을 가지는 컴퓨팅 사고
- ➔ 사회 현상의 단면만을 보지 않고 그 본질을 꿰뚫어볼 수 있는 **통찰력**

Next

4차 산업혁명 시대에 갖추어야 할 사고
그리고 **Python**

디자인 사고(Design Thinking)

- 깊이 공감하기(Empathize): 상황을 관찰하고 문제점을 파악/발견
- 문제 정의하기(Define): 문제가 생기는 이유를 생각하며(해석) 문제를 정의
- 아이디어 찾기(Ideate): 문제를 해결하기 위한 아이디어 모색
- 프로토타입(Prototype): 아이디어를 (빠르게) 구현
- 현장테스트(Test): 아이디어의 가치를 평가, 피드백 활동을 통한 최종 결과물 구현



컴퓨팅 사고 with 파이썬(Python)

- 파이썬(Python)

- 컴퓨터와 소통하기 위한 언어
- 비교적 배우기 쉬움
- 유용한 오픈소스 라이브러리를 제공 → 빠른 개발 속도 → 개발자 친화적
“Life is too short. You need Python”
- 프로그램 메모리 및 수행 속도 측면에서는 한계가 있음 (다른 언어와 비교했을 때)
- 초보자뿐만 아니라 인공지능, 금융, 생명공학, 물리, 화학, 수학 등 거의 모든 분야에서 전공자들도 사용



Python 추가 정보

- 네덜란드 출신의 프로그래머인 귀도 반 로섬(Guido van Rossum)이 1989년에 개발한 프로그래밍 언어이다
- 자신이 좋아하는 코미티 프로그램 “
- “인터프리터(interpreter) 언어”이다
- 인터프리터 언어가 뭐지..?



컴파일 언어 vs 인터프리터 언어

• 컴파일 언어

- like 통 번역기
- 대표적 예시: C, C++, Java, C#
- 특징
 1. 규모가 큰 프로그램은 컴파일 시 오래 걸릴 수 있다.
 2. 컴파일 후에는 모든 소스코드가 기계어로 변환되어 있기 때문에 실행시간이 빠르다

• 인터프리터 언어

- like 실시간 번역기
- 대표적 예시 - Python, R
- 특징
 1. 컴파일을 하지 않고 인터프리터가 직접 한 줄 씩 읽어서 실시간으로 실행
 - >> 컴파일 언어에 비해 속도가 느림
 2. 빌드 과정 없이 바로 실행이 가능

Next

Python 개발 환경 설정하기

Anaconda 설치

- Anaconda 란?
: 라이브러리들을 쉽게 설치하고 관리할 수 있는 도구.
- 왜 “굳이” 용량이 큰 Anaconda를 사용하는가?
 - 패키지를 효율적으로 관리하기 위해
 - 가상환경을 통해 개발 환경을 관리하기 위해(pip 명령은 패키지 충돌 시 무시)
 - 상당한 양의 수학 및 과학 관련 패키지들을 기본적으로 포함



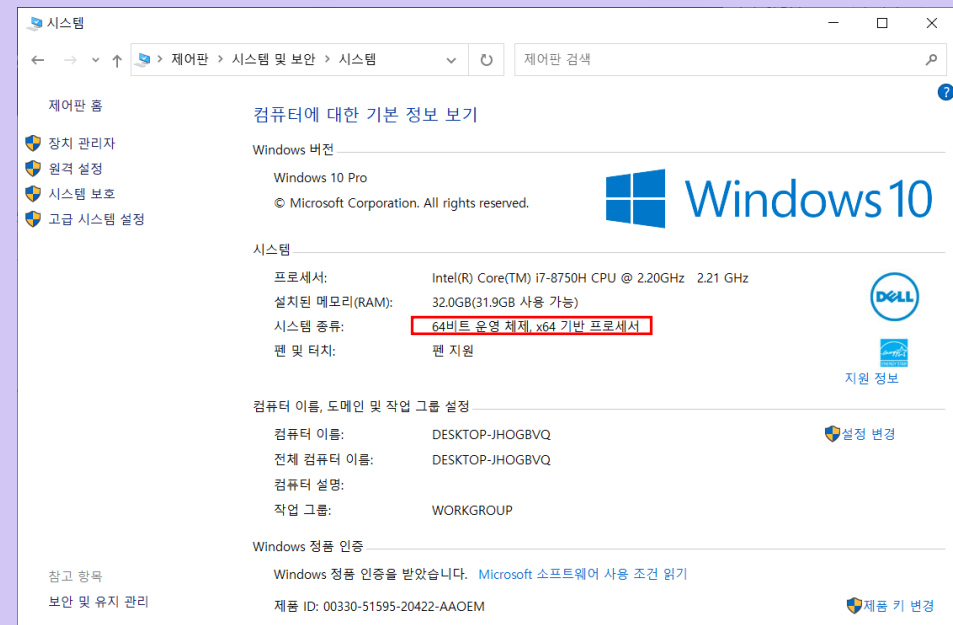
Anaconda 설치

- Anaconda 홈페이지 방문 & 프로그램 다운로드
- <https://www.anaconda.com/products/individual#Downloads>

Anaconda3-2021.05-Windows-x86_64

Anaconda Installers

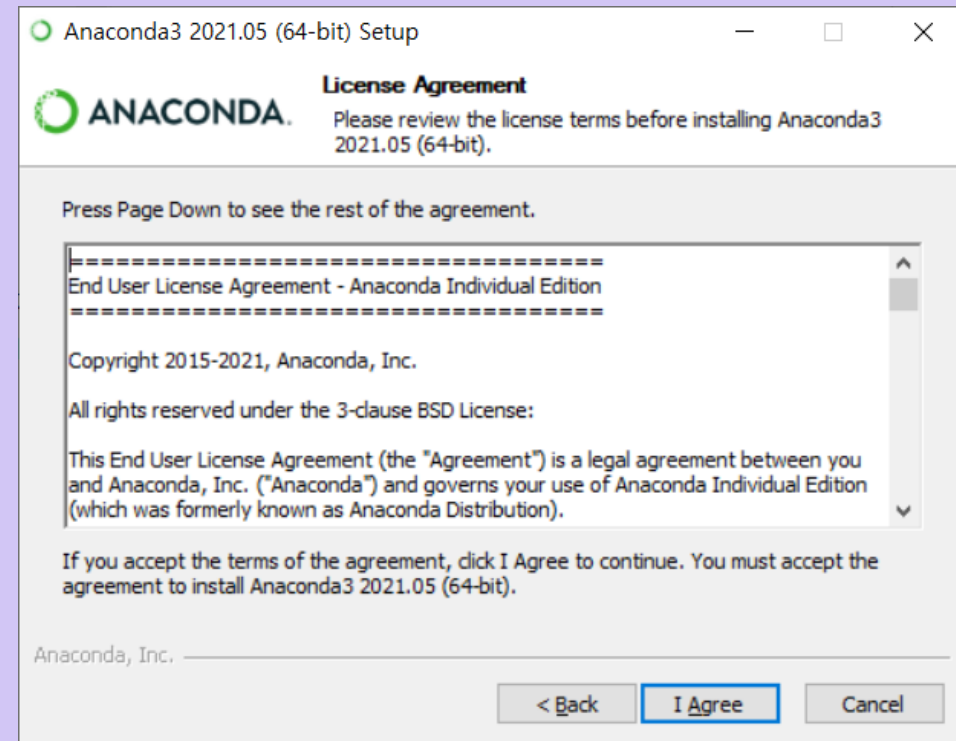
Windows	MacOS	Linux
Python 3.8 64-Bit Graphical Installer (477 MB) 32-Bit Graphical Installer (409 MB)	Python 3.8 64-Bit Graphical Installer (440 MB) 64-Bit Command Line Installer (433 MB)	Python 3.8 64-Bit (x86) Installer (544 MB) 64-Bit (Power8 and Power9) Installer (285 MB) 64-Bit (AWS Graviton2 / ARM64) Installer (413 M) 64-bit (Linux on IBM Z & LinuxONE) Installer (292 M)



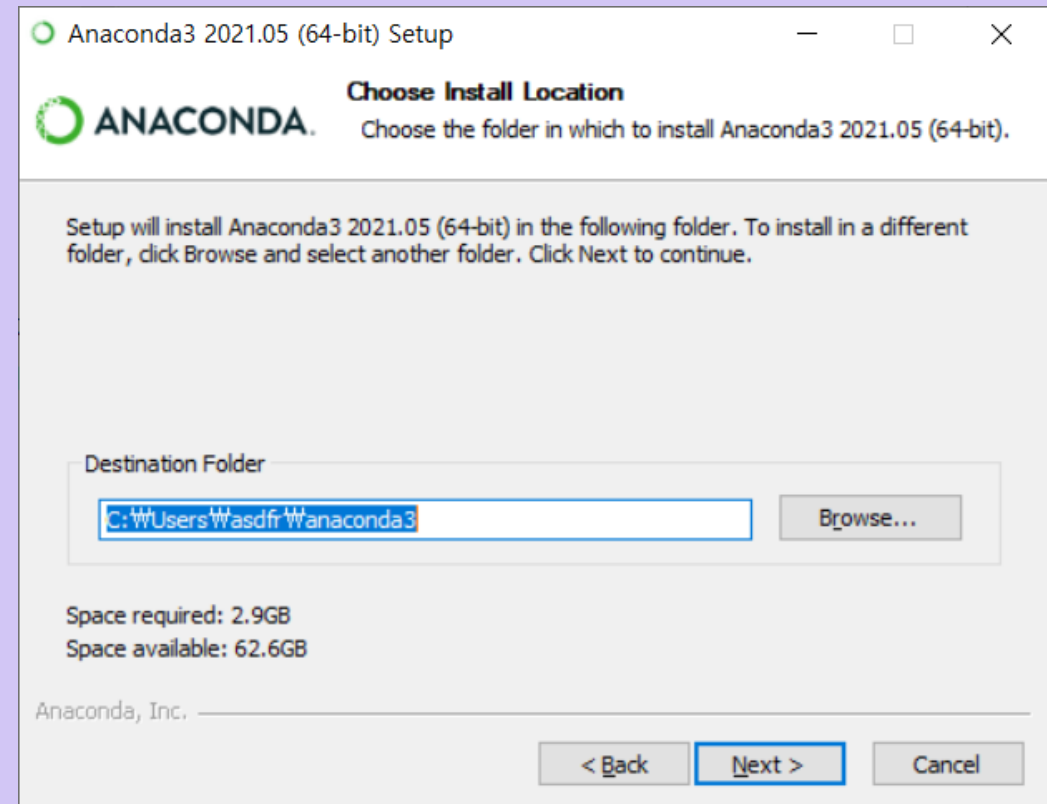
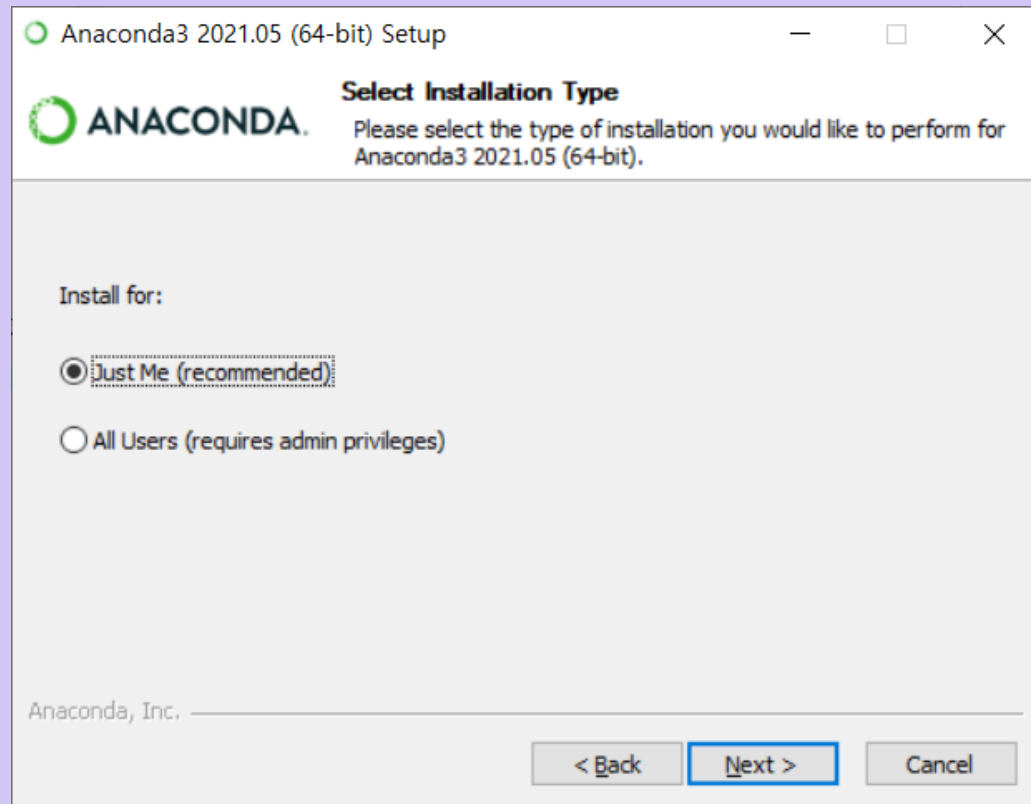
제어판
> 시스템 및 보안
> 시스템

Anaconda 설치

- 프로그램 설치

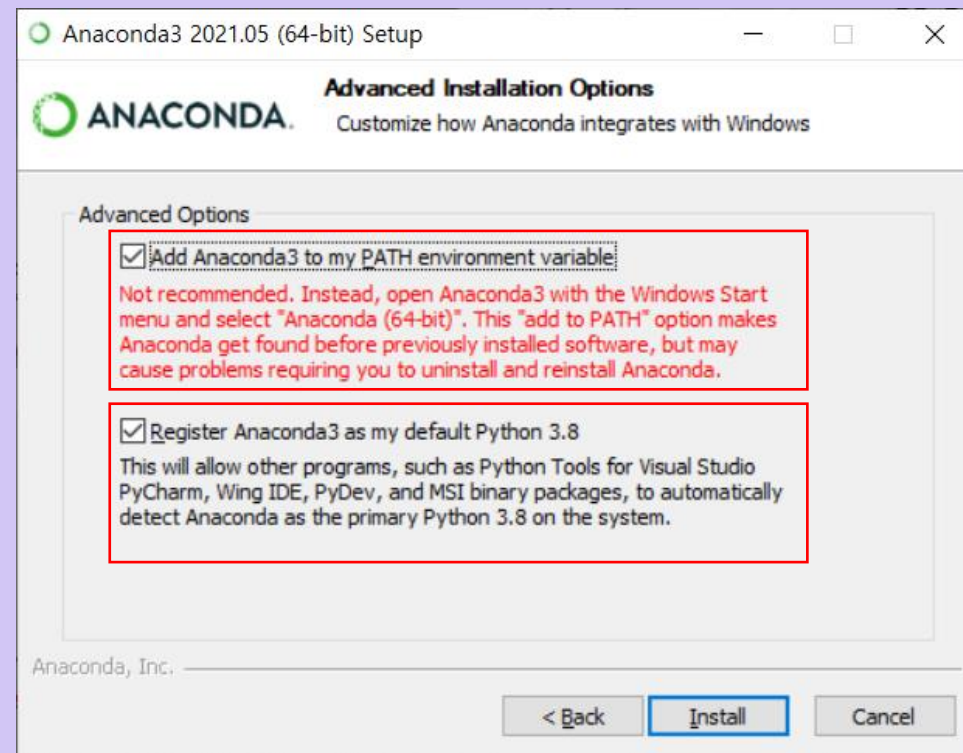


Anaconda 설치



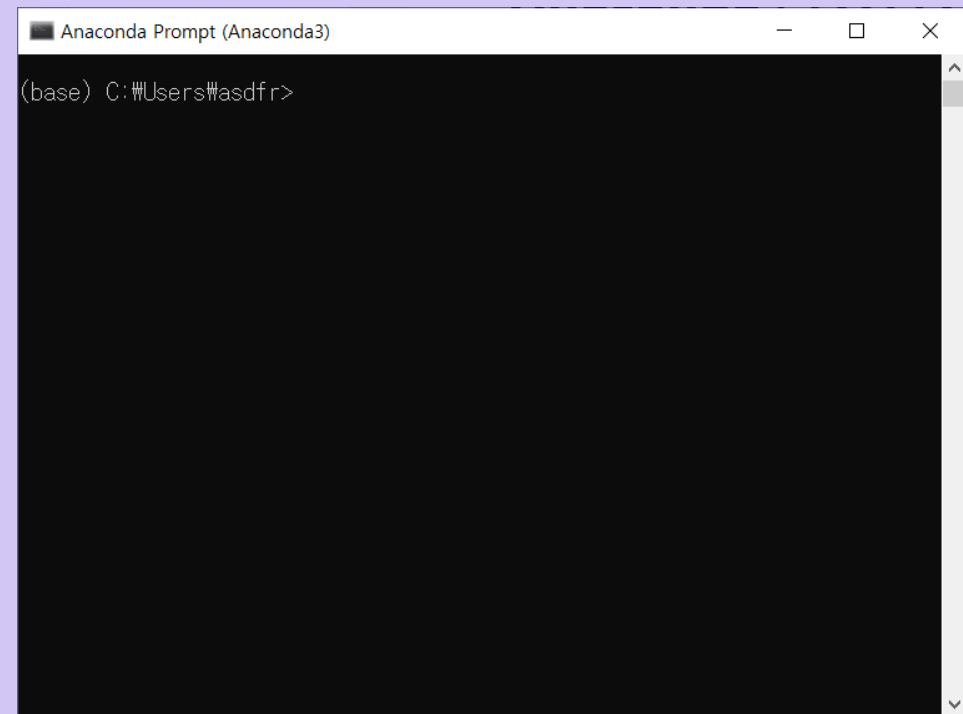
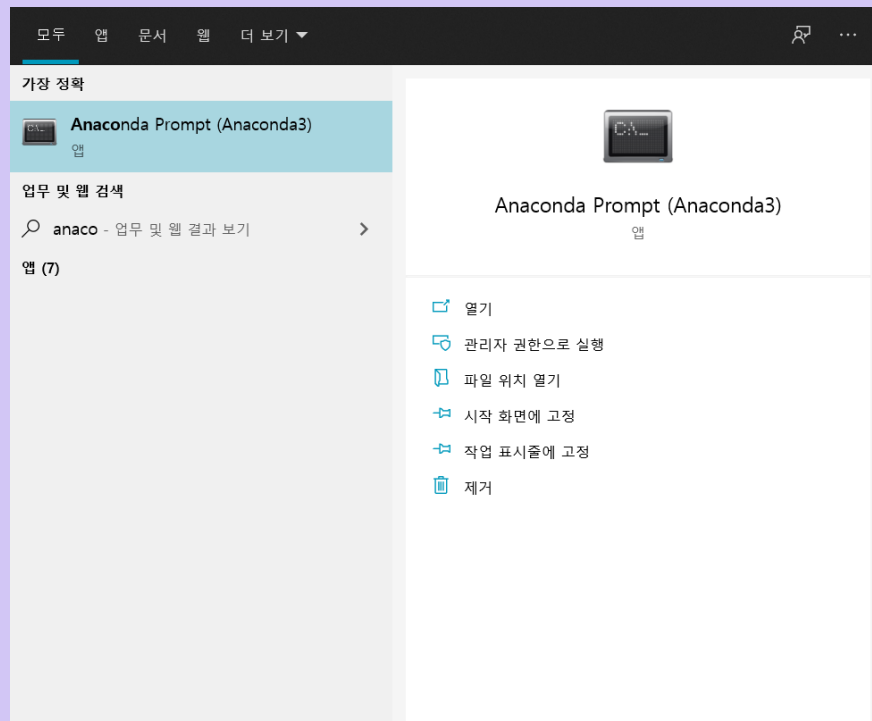
Anaconda 설치

- 주의! 기존에 python이 깔려 있으면 안됨.



Anaconda 가상환경 만들기

• Anaconda Prompte (Anaconda3) 실행하기



가상 환경 왜 쓰는 거지?



작업대

가상 환경 왜 쓰는 거지?



마을 바닥

Anaconda 가상환경 만들기

- Anaconda 버전 확인 & 최신 버전으로 업데이트
 - 버전 확인: `conda -version`
 - 업데이트: `conda update conda`
- Anaconda 가상환경 목록 확인
 - `conda env list`

Anaconda 가상환경 만들기

- 가상환경 만들기
 - [명령어] `conda create -n <붙이고 싶은 가상환경 이름> python = <버전>`
 - 예시) `conda create -n OSJ python=3.8`
- 가상환경 활성화 시키기 (가상환경에 패키지 설치할 경우 반드시 선행)
 - `conda activate <가상환경 이름>`

```
(base) C:\Users\masdfr>conda activate test  
(test) C:\Users\masdfr>
```

Anaconda 가상환경 만들기

- 가상환경 패키지 목록 확인
 - `conda list`
- 가상환경 패키지 설치하기
 - `conda install <원하는 패키지들의 이름, 구분은 띄어쓰기>`
 - `pip install <원하는 패키지 이름>` <- Anaconda 서버에 패키지가 없을 경우 사용
- 가상환경 비활성화
 - `conda deactivate`

대화형 모드(interpreter Mode)

- 수업 시간에 활용하는 일은 적겠지만 한 번 실행시켜 봅시다.
- Anaconda Prompt를 실행 시킨 후 다음을 실행
 - 가상환경 활성화
 - `python.exe -m idlelib` 입력

```
(base) C:\Users\wasdfr>conda activate test  
(test) C:\Users\wasdfr>python.exe -m idlelib
```

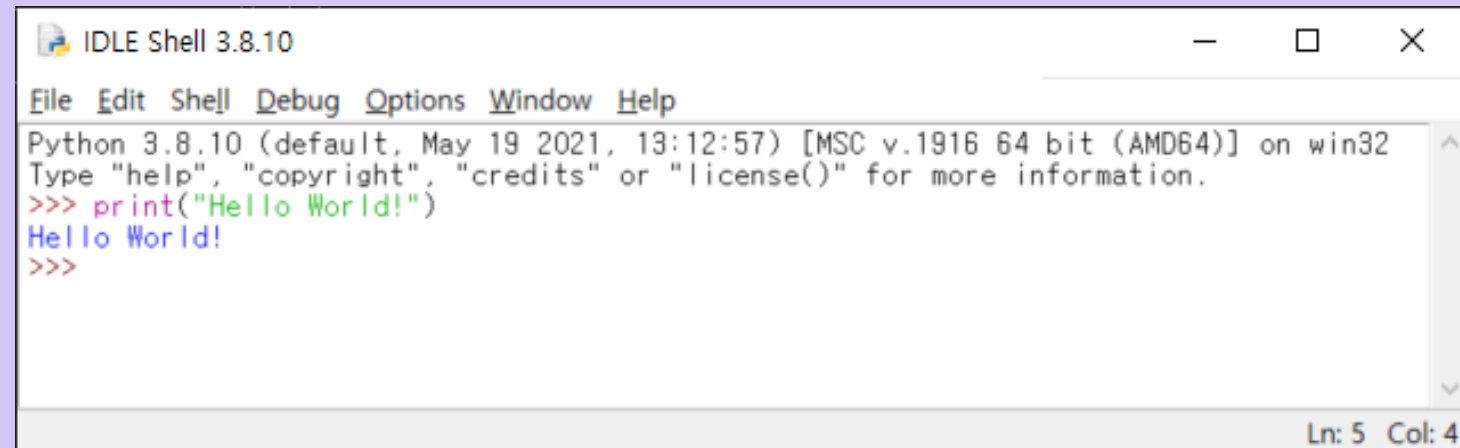
대화형 모드(interpreter Mode)

- python에서 자체적으로 제공하는 IDLE(Integrated DeveLopment Environment, 통합 개발 환경)
- 미션1: **import this**를 입력 후 실행해보자

```
Python 3.8.6 (tags/v3.8.6:db45529, Sep 23 2020, 15:52:53) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> import this
```


대화형 모드(interpreter Mode)

- python에서 자체적으로 제공하는 IDLE(Integrated DeveLopment Environment, 통합 개발 환경)
- 미션2: `print` 명령어를 통해 “Hello world”를 출력 해보자

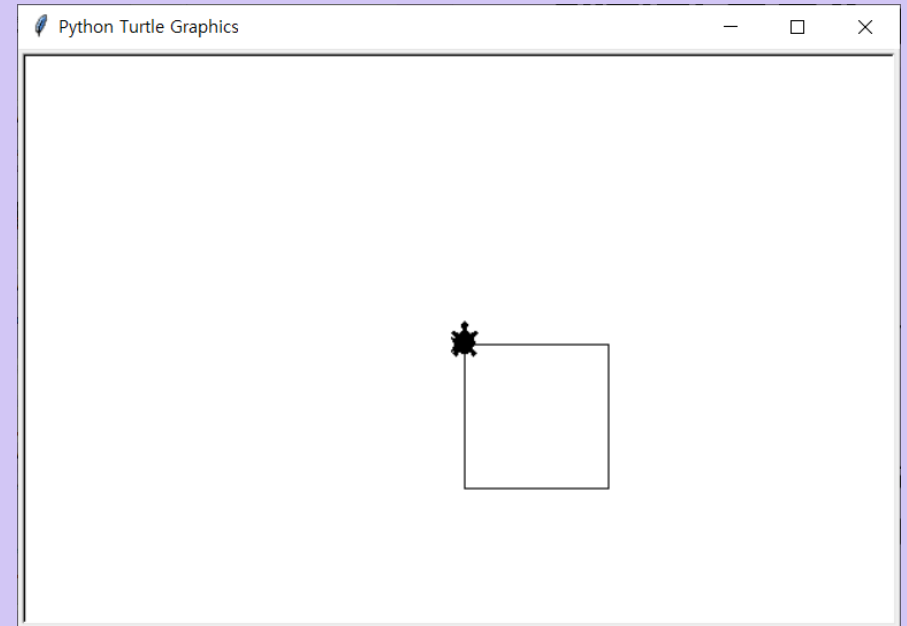


```
Python 3.8.10 (default, May 19 2021, 13:12:57) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> print("Hello World!")
Hello World!
>>>
```

Ln: 5 Col: 4

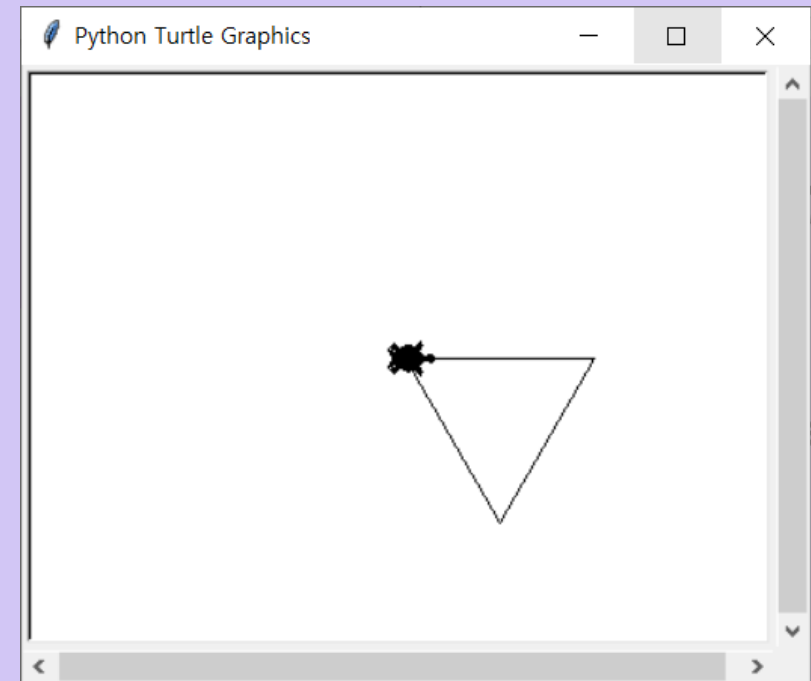
대화형 모드(interpreter Mode)

- 미션3: 대화형 모드에서 turtle 모듈을 활용하여 사각형을 그려보자
- 필요 명령어 list
 - `import turtle`: turtle 모듈 가져오기
 - `turtle.Screen()`
 - `turtle.Turtle()`
 - `turtle.forward(이동하고 싶은 거리)`
 - `turtle.right(현재 방향에서 오른쪽으로 꺾을 각도)`



스크립트 모드(Script Mode)

- 대화형 모드에서 'File>New File' 클릭
- 스크립트 모드 실행
- 미션4 : 스크립트 모드를 실행하여 정삼각형 그리기
(실행 단축키: F5)



Pycharm 설치하기

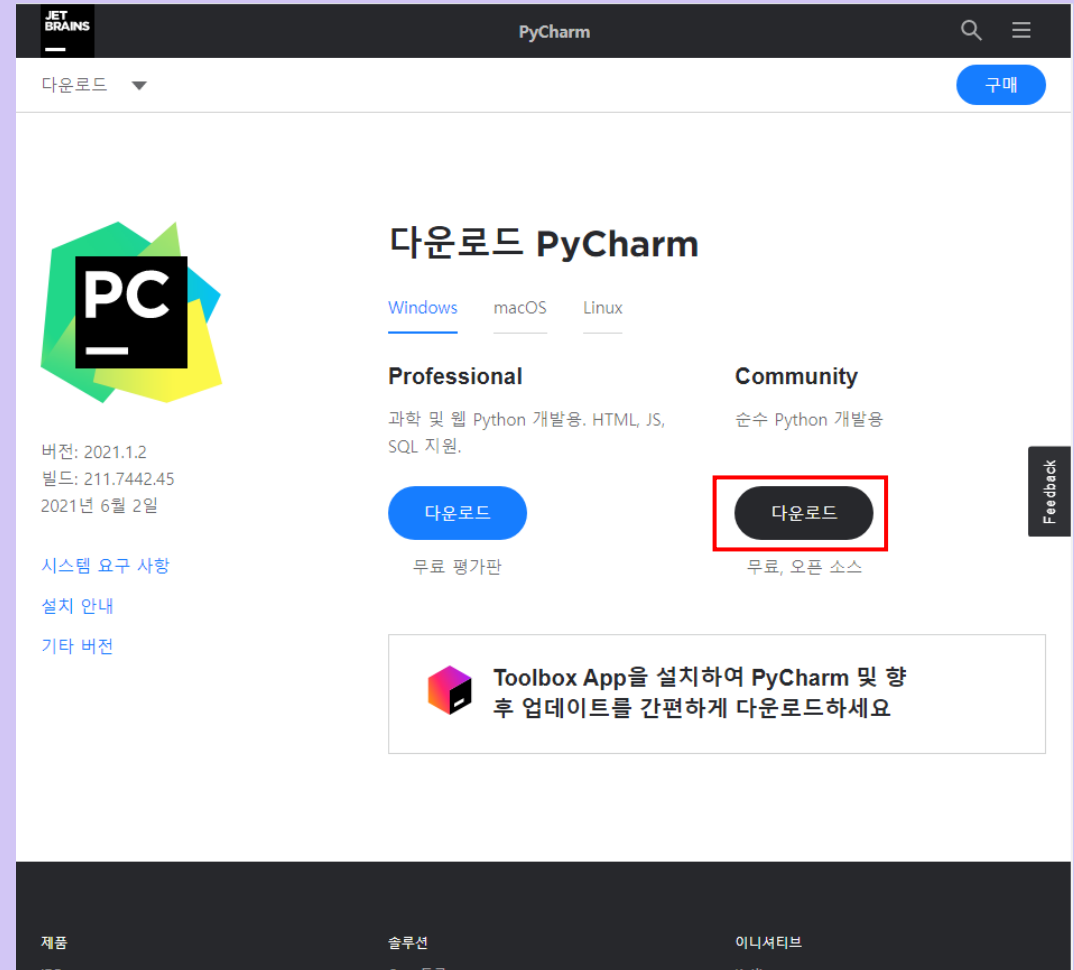
- python을 위한 IDE 중 하나
- 활용이 편리하며 디버깅 가능
- 가상환경 연결을 쉽게 할 수 있음



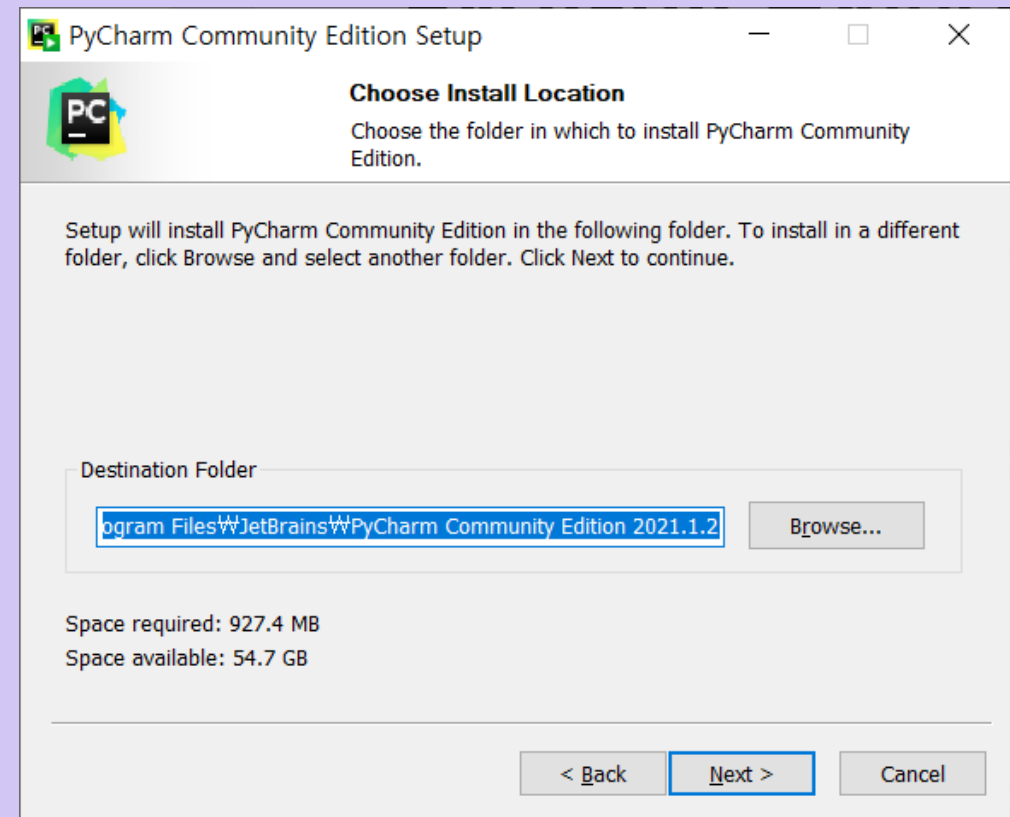
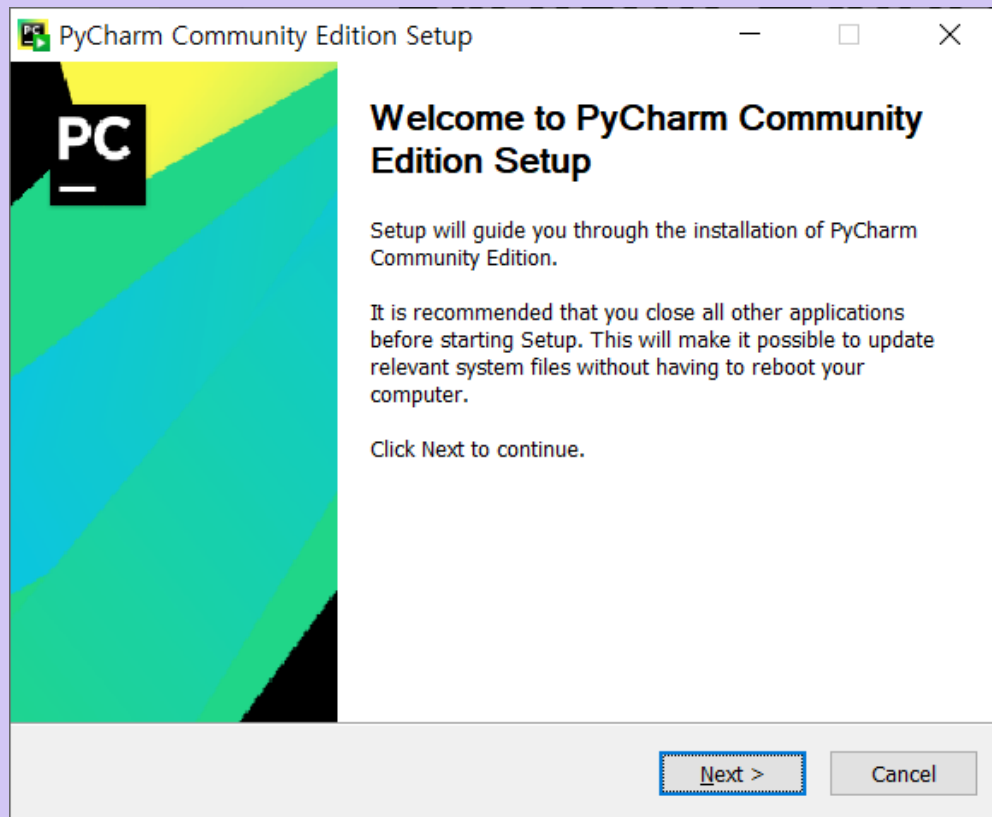
Pycharm 설치하기

- Pycharm 설치 파일 다운로드
- <https://www.jetbrains.com/ko-kr/pycharm/download/#section=windows>
- 자신의 OS에 맞게 Community 버전의 설치 프로그램 다운로드 & 실행

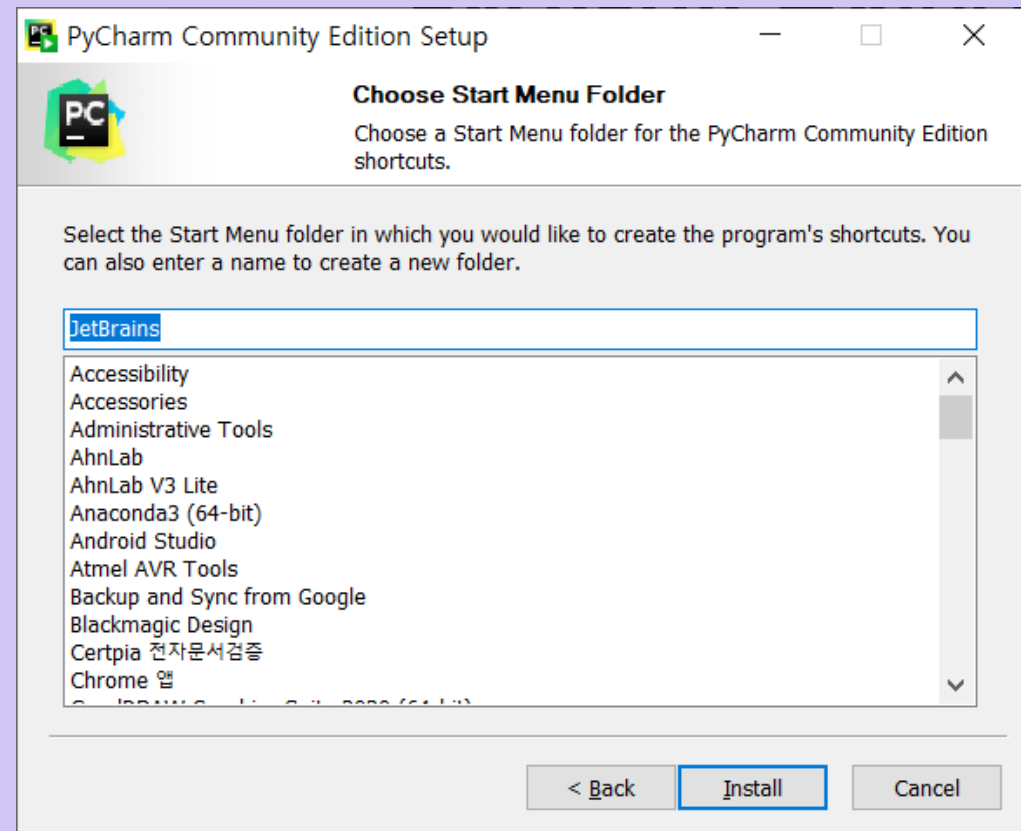
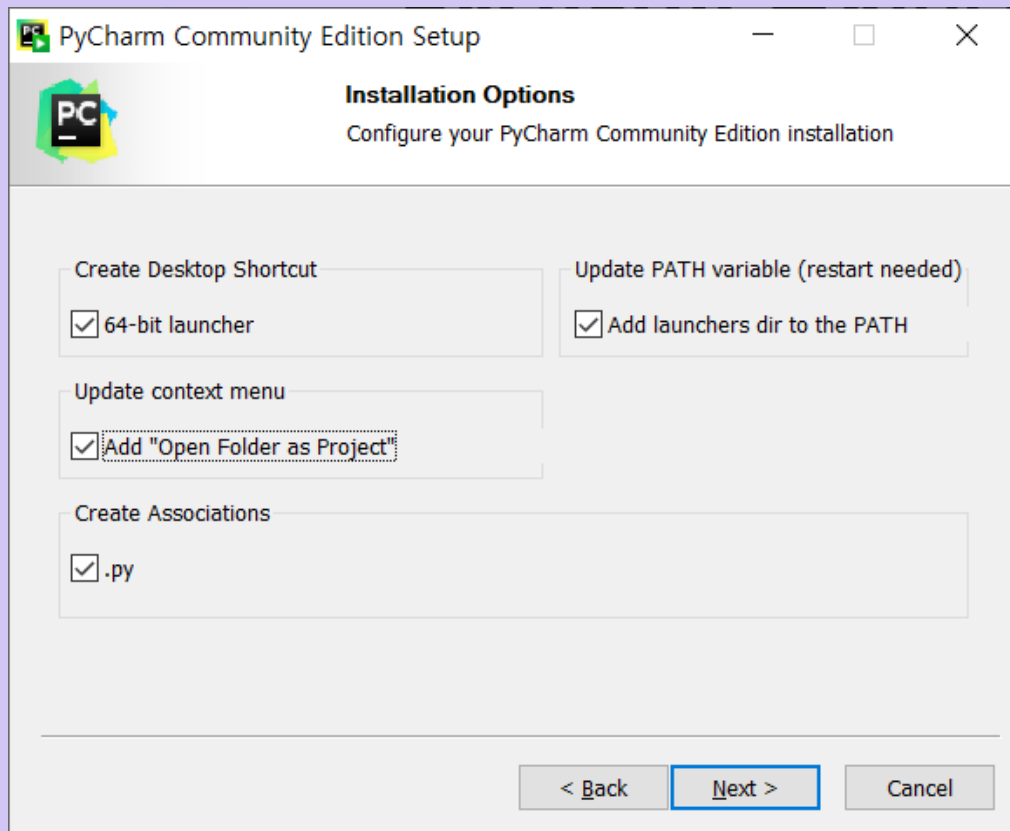
 pycharm-community-2021.1.2



Pycharm 설치하기

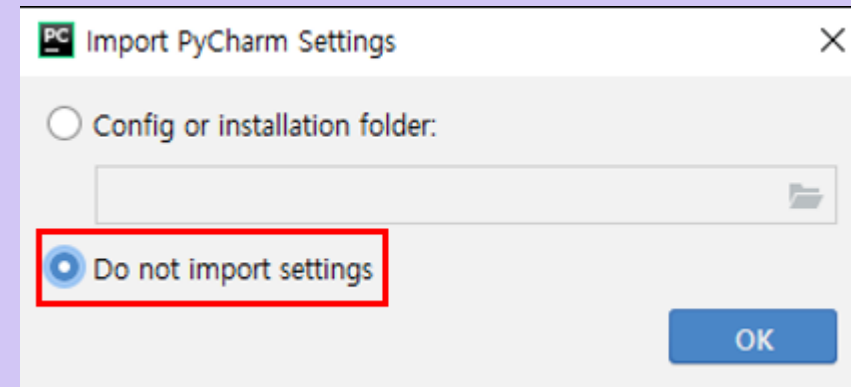
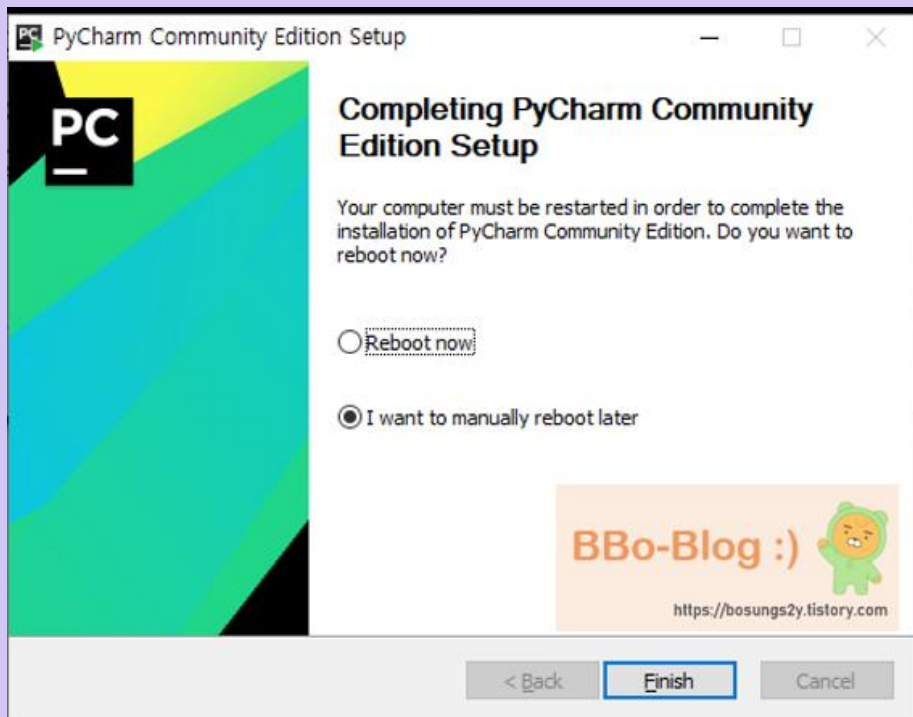


Pycharm 설치하기

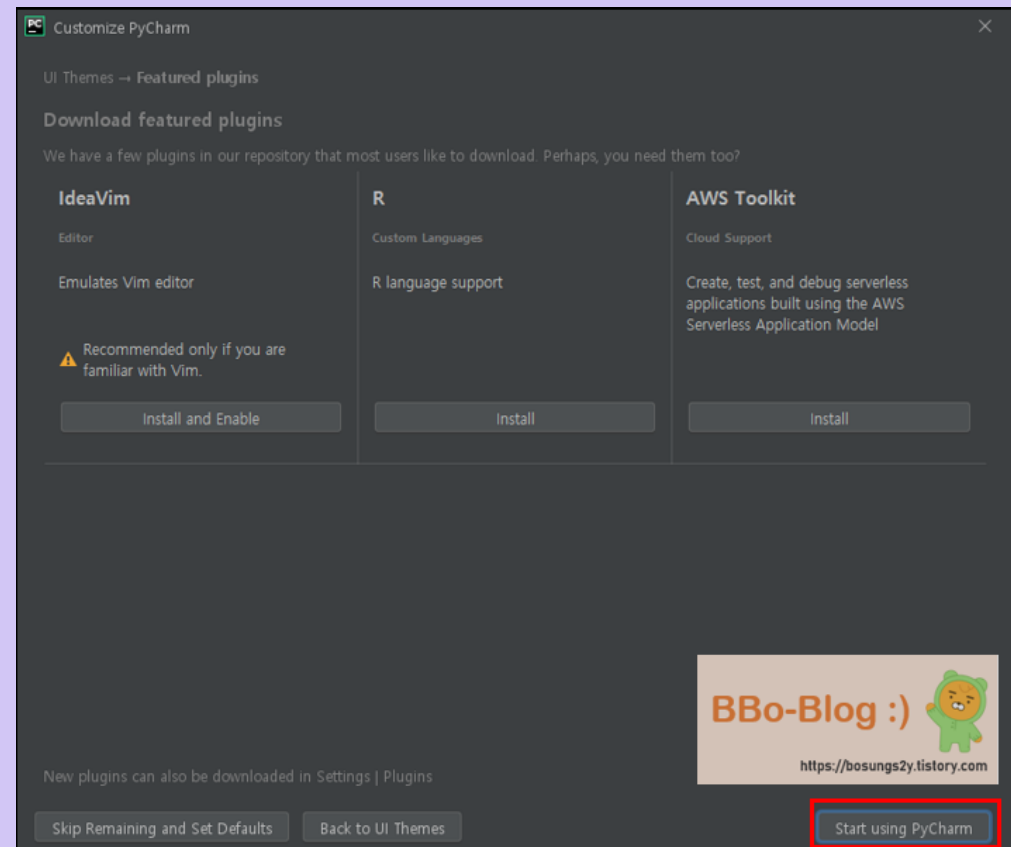
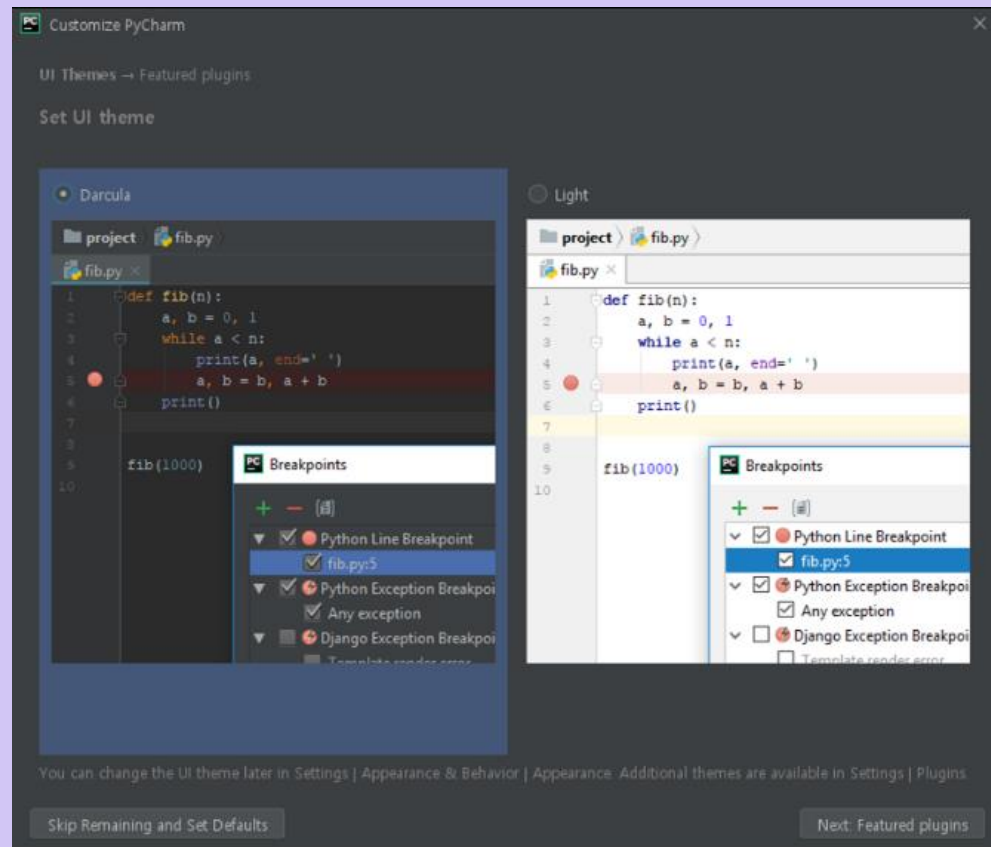


Pycharm 설치하기

- 설치 완료 후 실행



Pycharm 설치하기



Github에 프로젝트 올리기

• Github 가입 & 계정 유형 선택

Join GitHub

Create your account

Username *

Email address *

Password *

Make sure it's at least 15 characters OR at least 8 characters including a number and a lowercase letter. [Learn more](#)

Email preferences



☐ Send me occasional product updates, announcements, and offers.

Verify your account

이 과정을 통해서 귀하가 인간이라는 것을 알 수 있게 해주십시오

확인

Create account

Individual	Team
Pick the plan that's right for you, personally.	Choose a plan to help your team grow and collaborate.
 <p>Free \$0 USD</p> <p>The basics of GitHub for every developer</p> <p>Choose Free</p>	 <p>Pro \$7 USD</p> <p>Per month</p> <p>Pro tools for developers with advanced requirements</p> <p>Choose Pro</p>
<ul style="list-style-type: none"> ✓ Unlimited public repositories ✓ Unlimited private repositories ✓ Limited to 3 collaborators for private repositories ✓ 2,000 total Action minutes/month ✓ 500MB of GitHub Packages storage ✓ Advanced vulnerability scanning for public repositories ✓ Automated security updates ✓ GitHub Security Advisories ✓ Issues and bug tracking ✓ Project management 	<p>← Includes everything in Free</p> <ul style="list-style-type: none"> ✓ Unlimited collaborators ↑ 3,000 total Action minutes/month ↑ 1GB of GitHub Packages storage ✓ Private GitHub Pages and Wikis ✓ Private protected branches ✓ Code owners ✓ Repository insights

Github에 프로젝트 올리기

- 설문 응답 & 이메일 인증

Welcome to GitHub

Woohoo! You've joined millions of developers who are doing their best work on GitHub. Tell us what you're interested in. We'll help you get there.

How much programming experience do you have?

None

I don't program at all

A little

I'm new to programming

A moderate amount

I'm somewhat experienced

A lot


I'm very experienced

What do you plan to use GitHub for?

(Select up to 3)

Complete setup

[Skip this step](#)



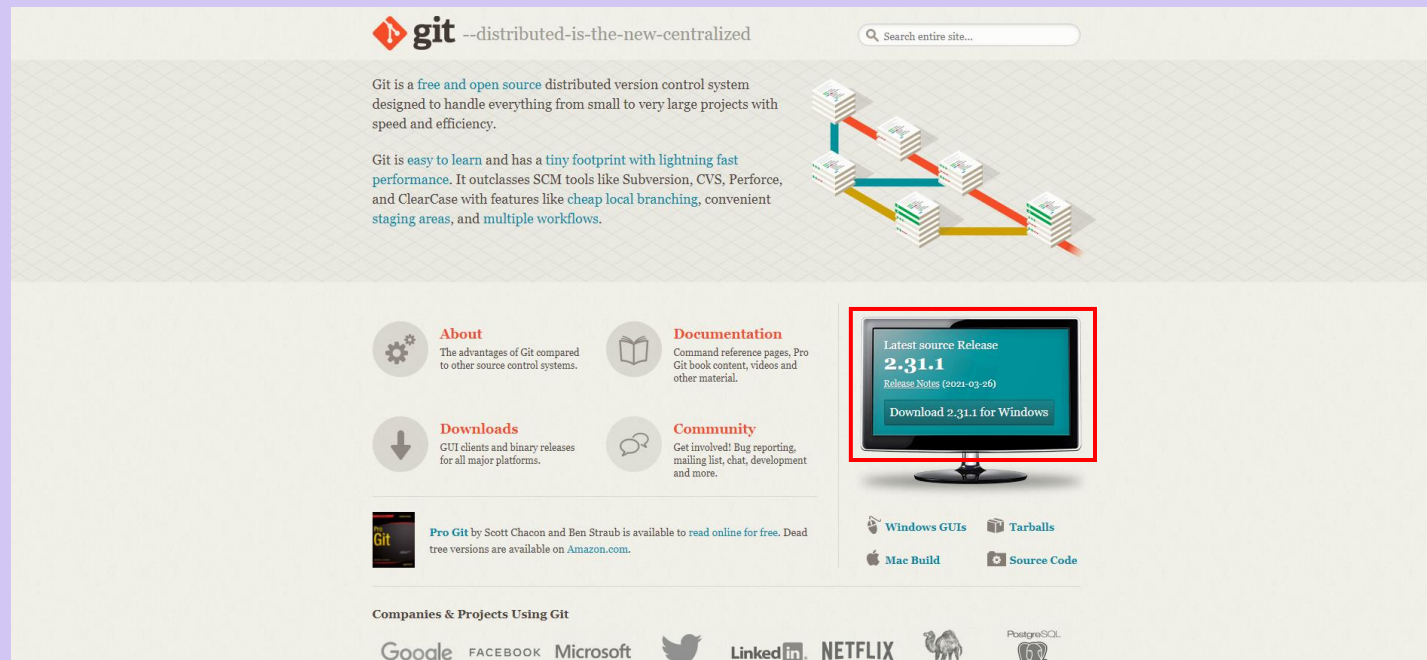
Please verify your email address

Before you can contribute on GitHub, we need you to verify your email address. An email containing verification instructions was sent to **이메일 주소**

Didn't get the email? [Resend verification email](#) or [change your email settings](#).

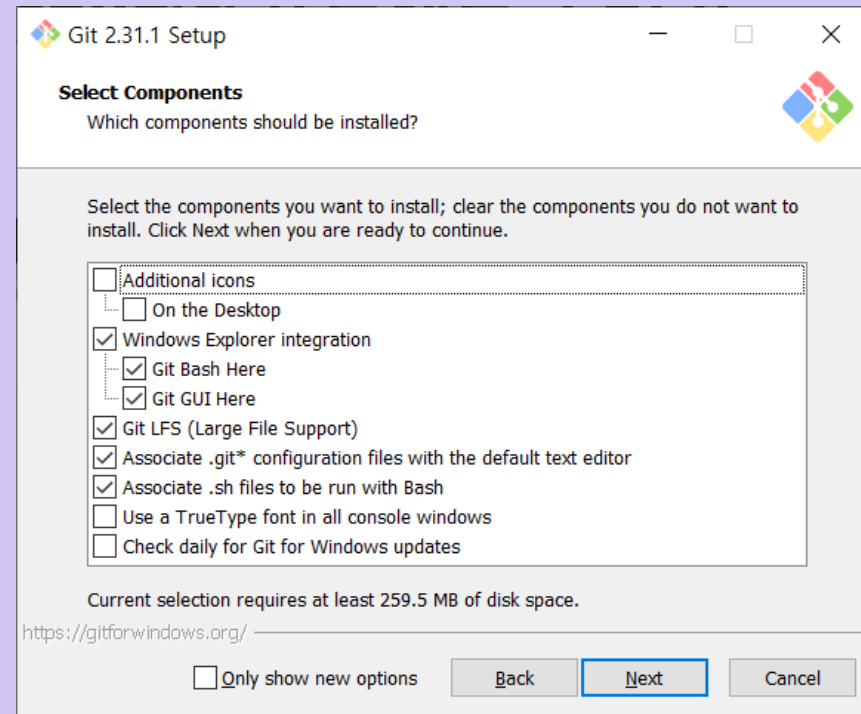
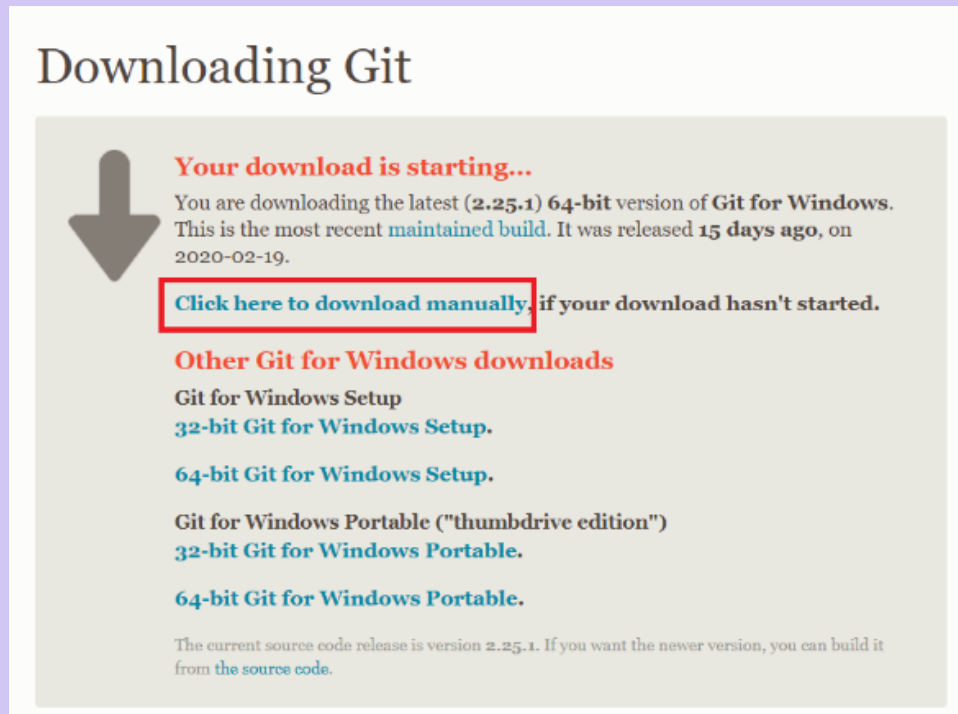
Github에 프로젝트 올리기

- Git Bash 설치
 - 다운로드 링크: <https://git-scm.com/>



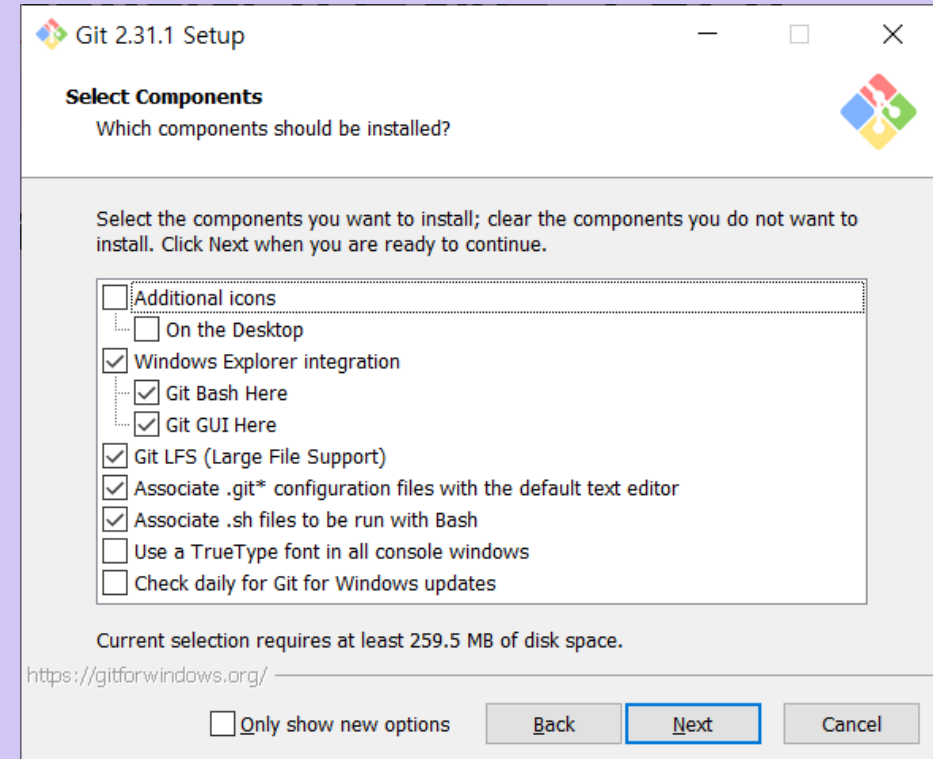
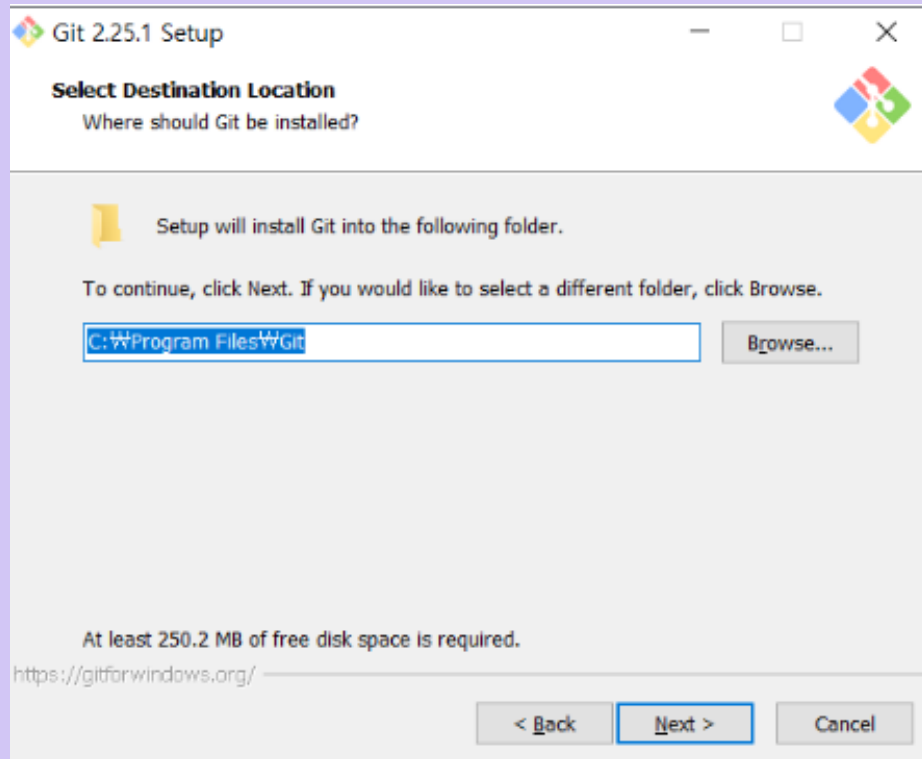
Github에 프로젝트 올리기

- 자동으로 설치가 안된다면... & 다운로드 후 설치 파일 실행



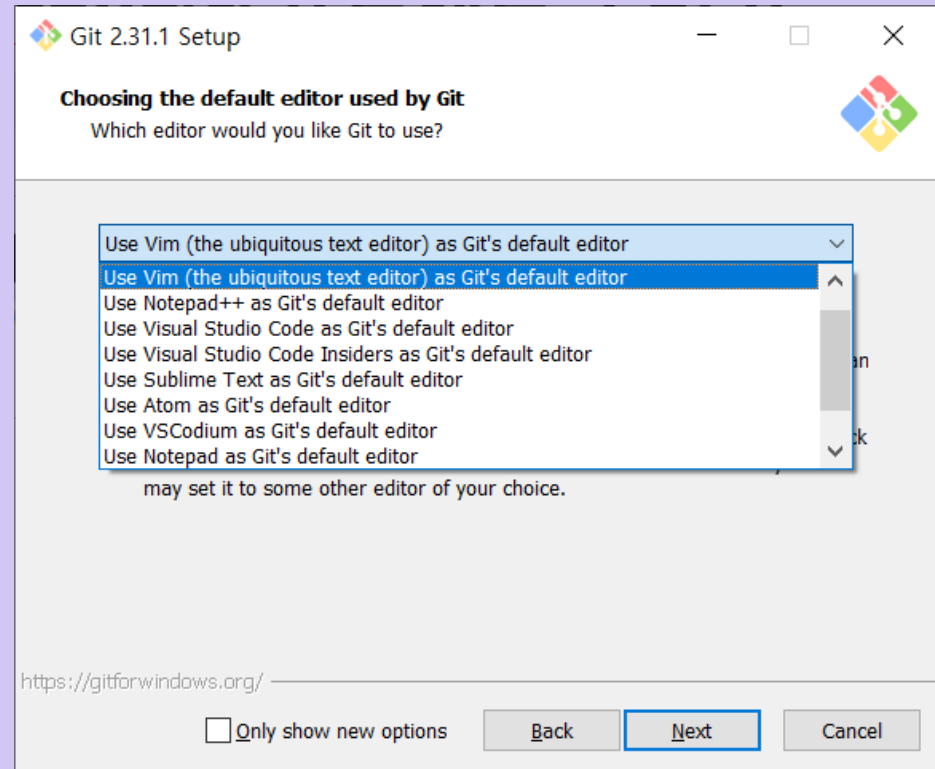
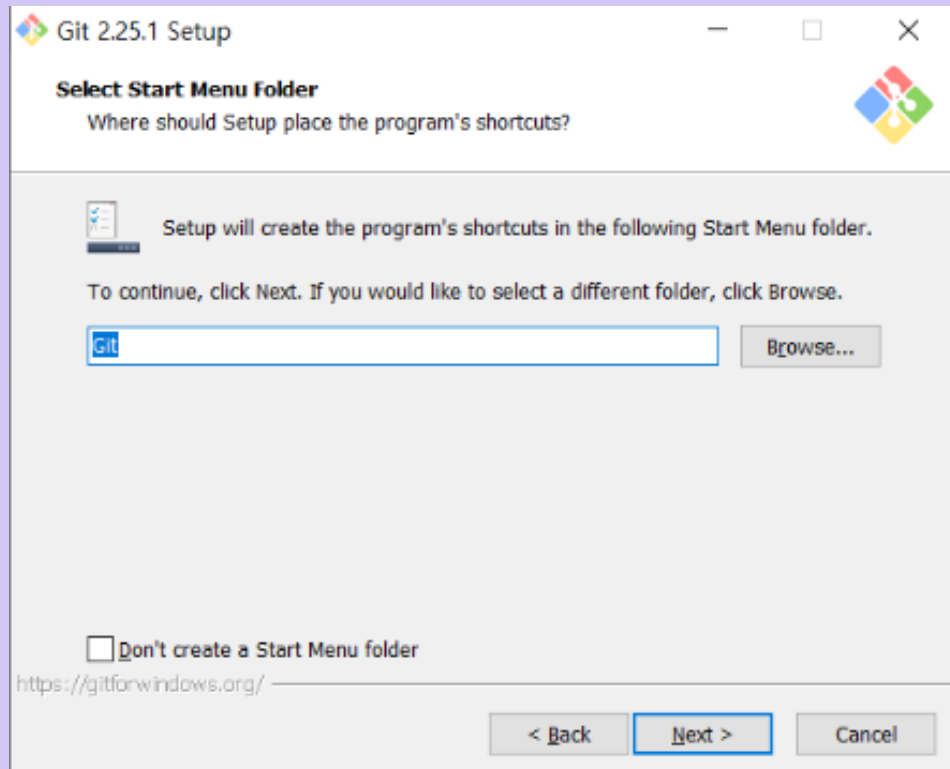
Github에 프로젝트 올리기

• Git 경로 지정 & 설치할 구성요소 선택



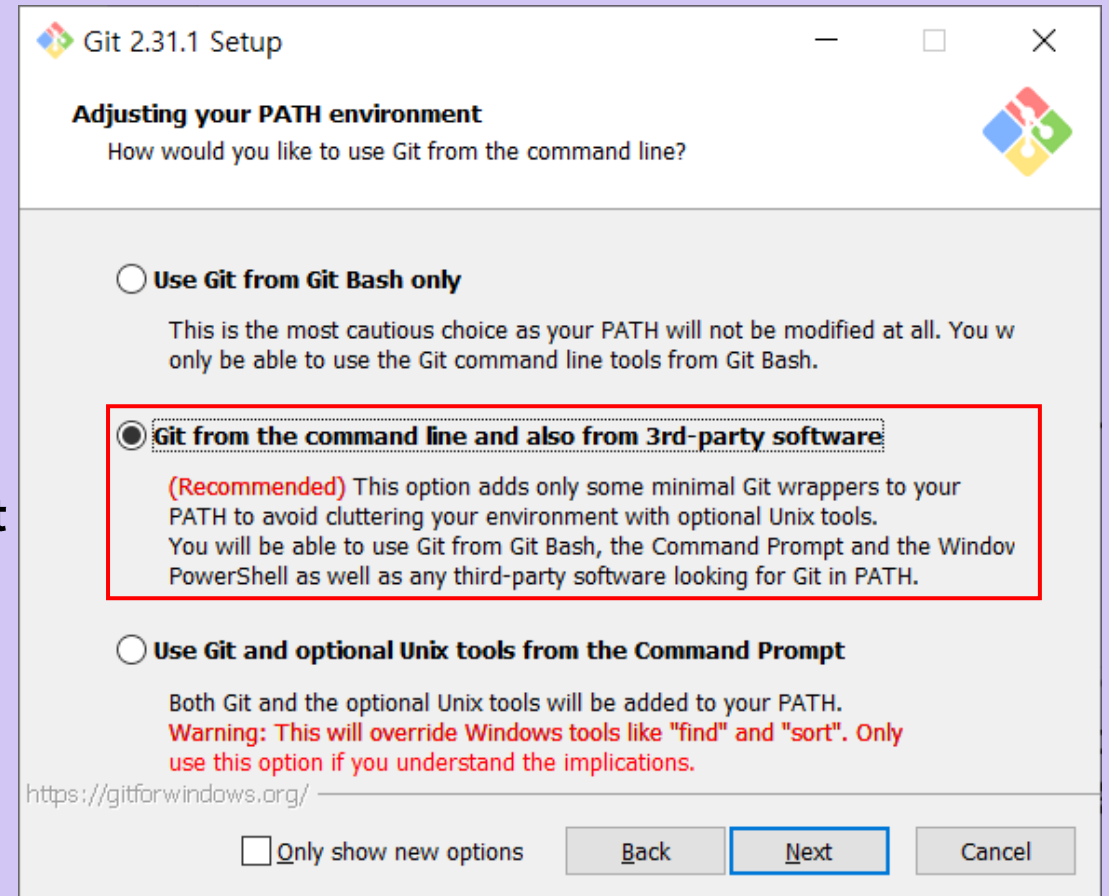
Github에 프로젝트 올리기

- 시작 폴더 경로 선택 & Git을 사용할 기본 에디터 선택



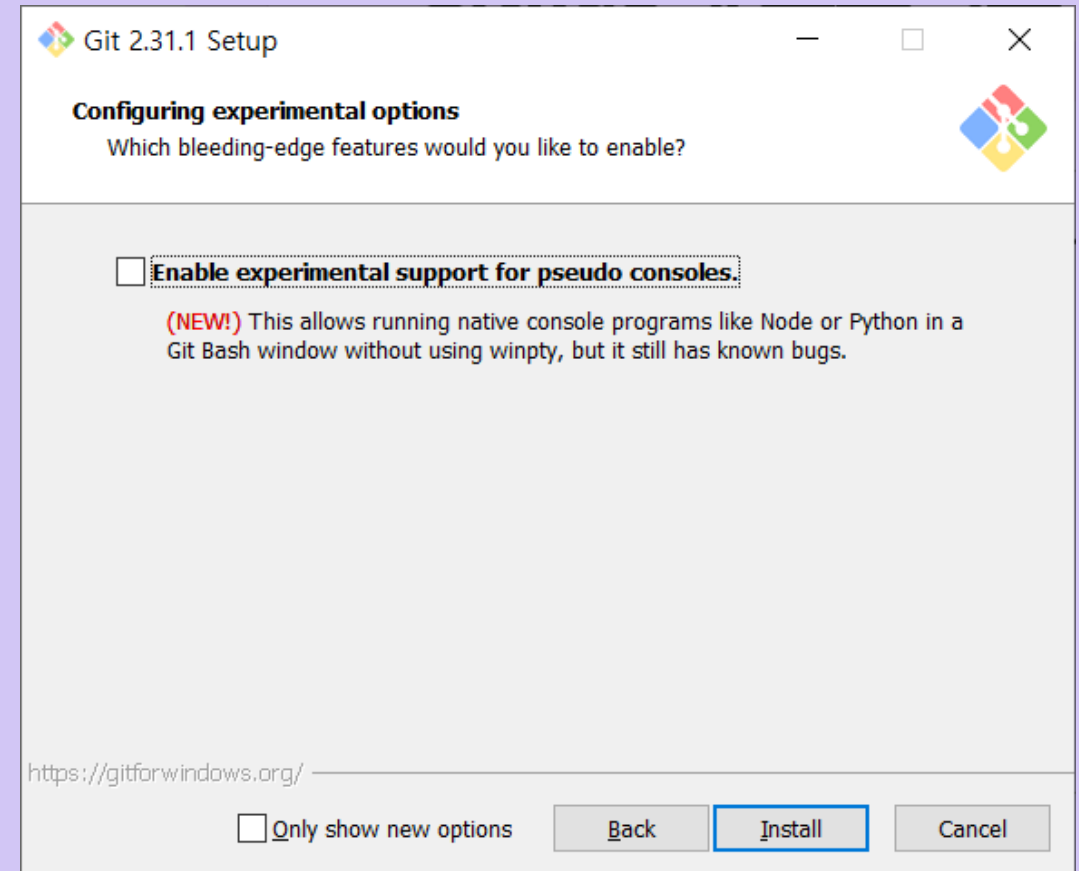
Github에 프로젝트 올리기

- 환경변수 옵션 설정
 - Use Git from Git Bash only
: Git bas에서만 Git Command 수행
 - Use Git from the Windows Command Prompt
: Git을 환경변수에 등록하고 윈도우 창 등 에서도 Git 사용 가능
 - Use Git and optional Unix tools from the Windows Command Prompt
: 윈도우 cmd에서 Git과 유닉스도구를 사용할 경우 환경변수에 추가



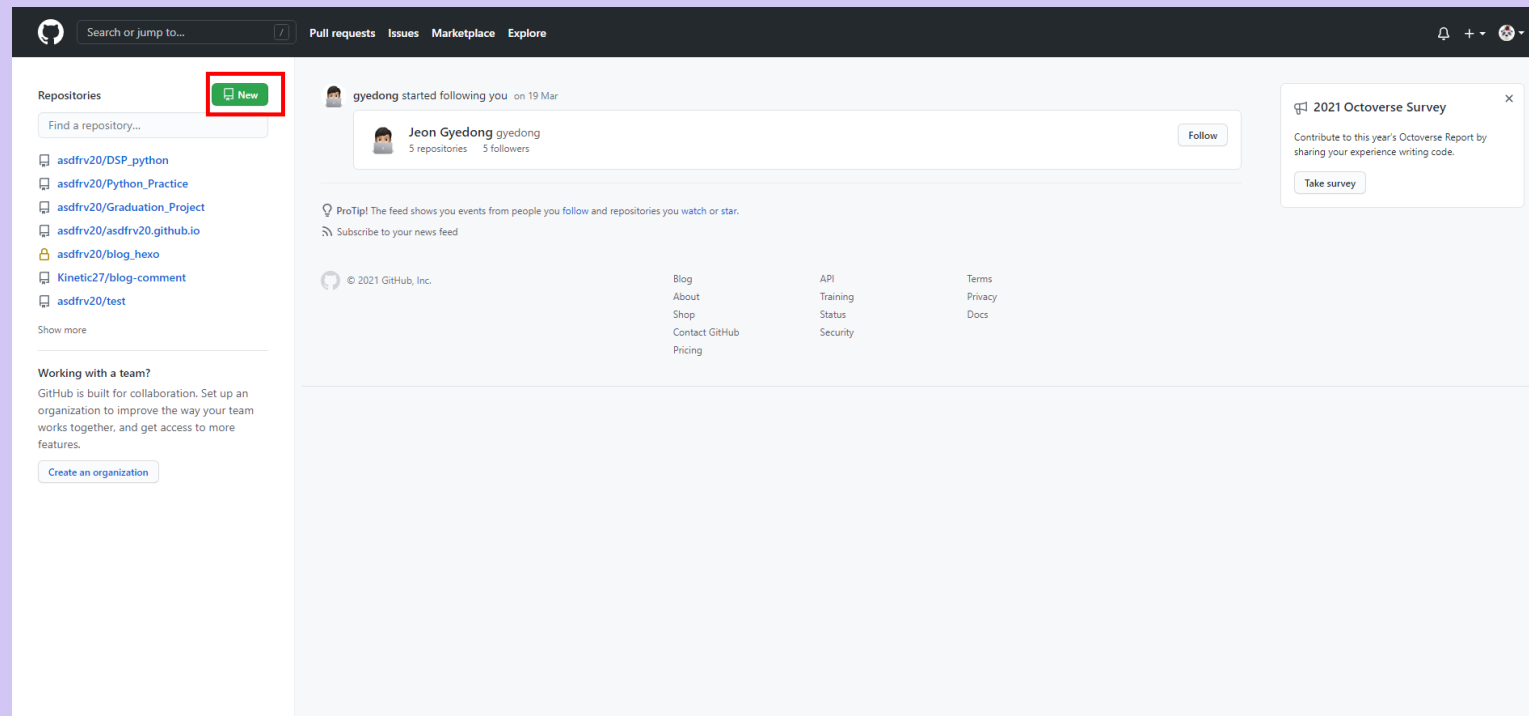
Github에 프로젝트 올리기

- 그 뒤에 언급되는 내용들은 모두 Next
- “Install”
- local에 Git 설치 완료



Github에 프로젝트 올리기

- Github(remote)에 repository 만들기



Github에 프로젝트 올리기

• Github(remote)에 repository 만들기

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Owner * asdfvr20 Repository name *

Great repository names are short and memorable. Need inspiration? How about [silver-cto-spork?](#)

Description (optional)

☒ Public
Anyone on the internet can see this repository. You choose who can commit.

☐ Private
You choose who can see and commit to this repository.

Initialize this repository with:
Skip this step if you're importing an existing repository.

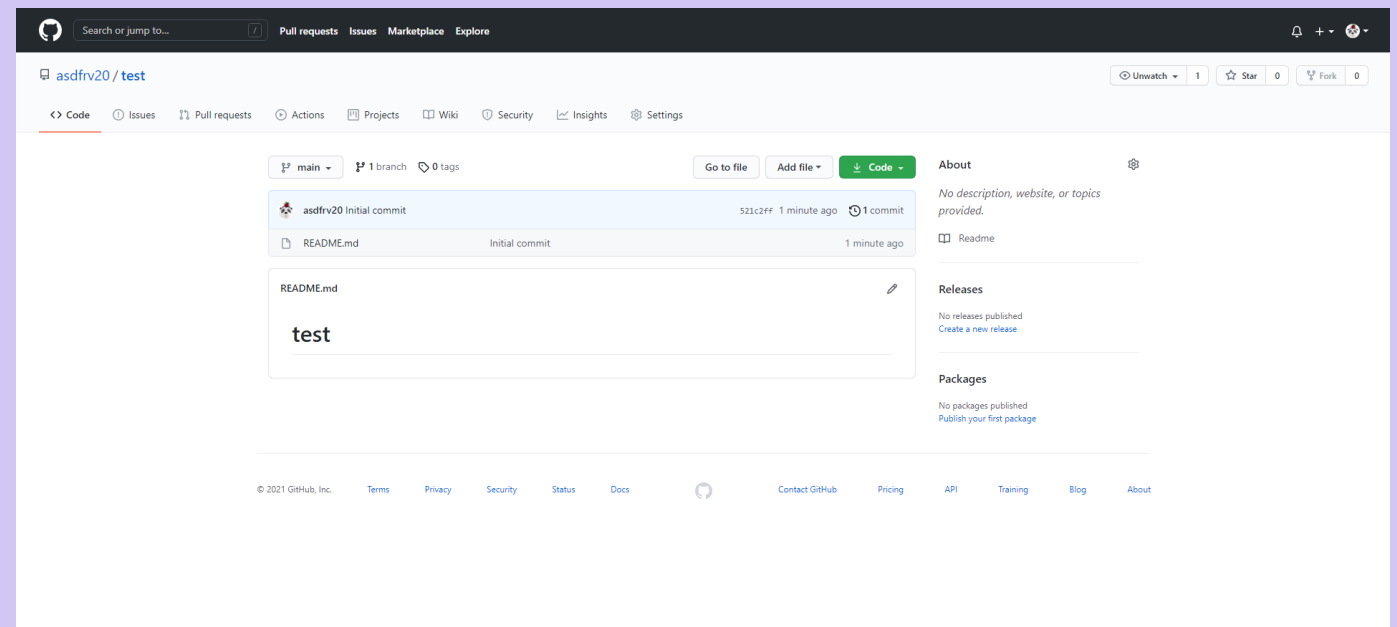
☒ Add a README file
This is where you can write a long description for your project. [Learn more.](#)

☐ Add .gitignore
Choose which files not to track from a list of templates. [Learn more.](#)

☐ Choose a license
A license tells others what they can and can't do with your code. [Learn more.](#)

This will set `main` as the default branch. Change the default name in your [settings](#).

Create repository



Github에 프로젝트 올리기

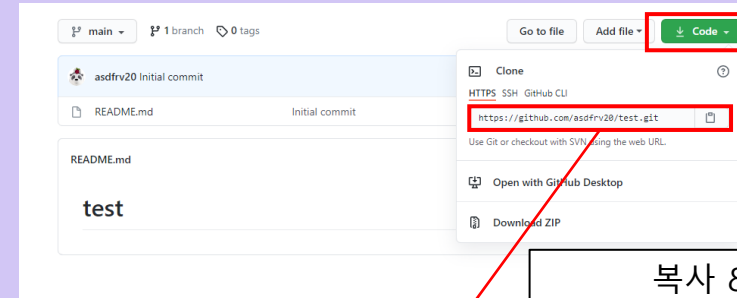
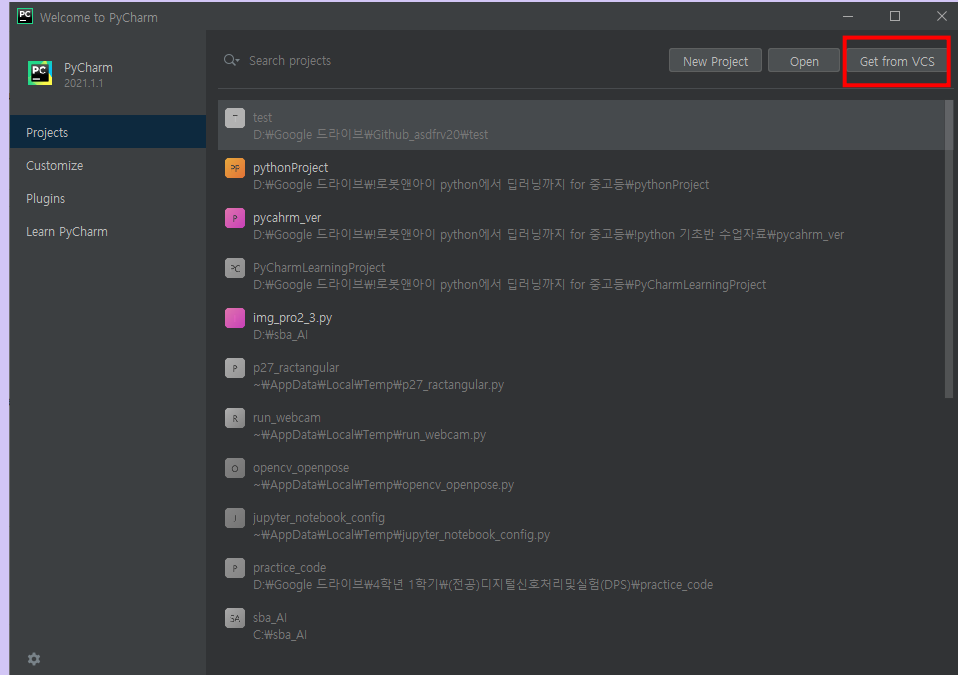
- Git 튜토리얼 (<https://evan-moon.github.io/2019/07/25/git-tutorial/>)
- Git의 역할: 감시자, 프로젝트의 버전 관리
- Git에서 쓰이는 용어들
 - Remote: 리모트 서버 자체(Github, Bitbucket, GitLab 등)
 - Origin: 내가 지금 업로드할 리모트 서버 하나를 지칭하는 말
 - Repository(Repo): 저장소, 리모트 서버 내에서 구분되는 프로젝트 단위
 - Branch: 버전 관리를 위해 독자적인 작업을 진행하기 위한 작업 공간

Github에 프로젝트 올리기

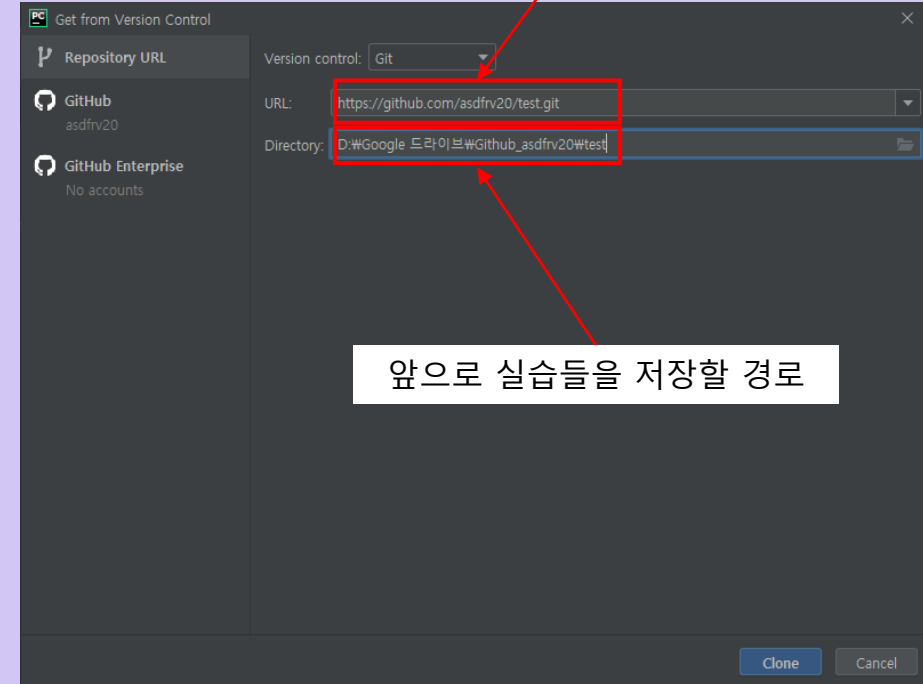
- 우리가 활용할 Git 명령어 설명
 - clone: remote 서버의 repo에서 클라이언트로 파일을 복사&붙여넣기 하는 명령
 - add: origin에 보낼 변경된 파일들 선택하기
 - commit: add한 내용들을 상자에 넣어 포장하기(origin에 보낼 준비)
 - push: origin에 commit한 내용 보내기

Github에 프로젝트 올리기

- Pycharm 실행 및 Github 연동
(여기에서 github 로그인 과정을 거침)



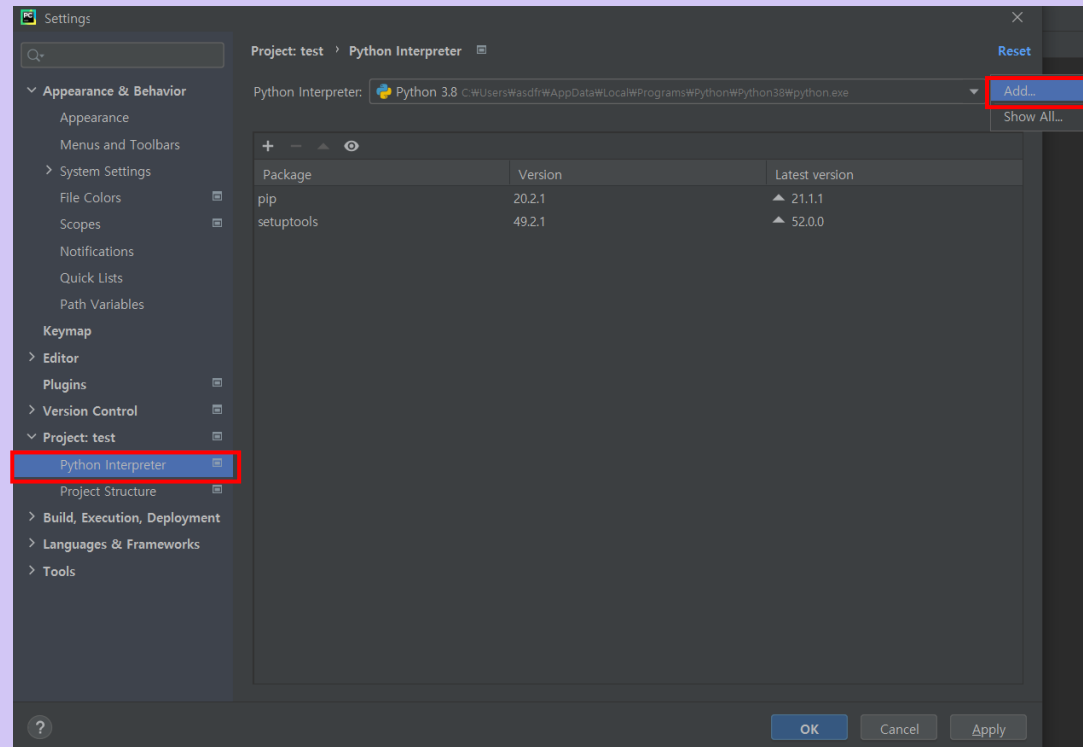
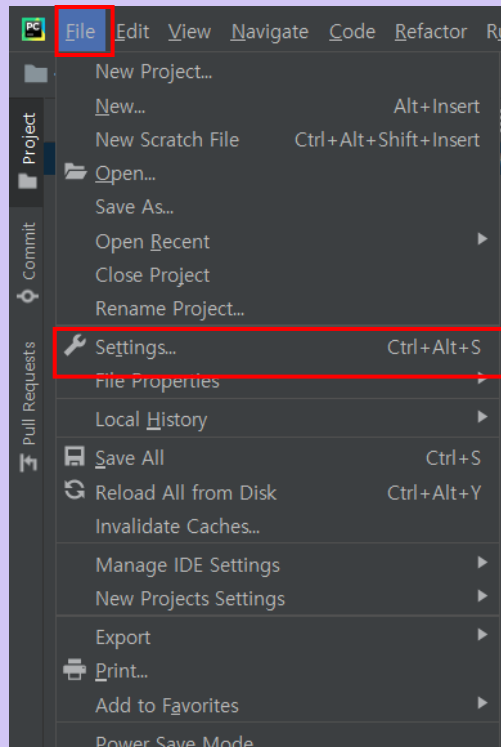
복사 & 붙여넣기



앞으로 실습들을 저장할 경로

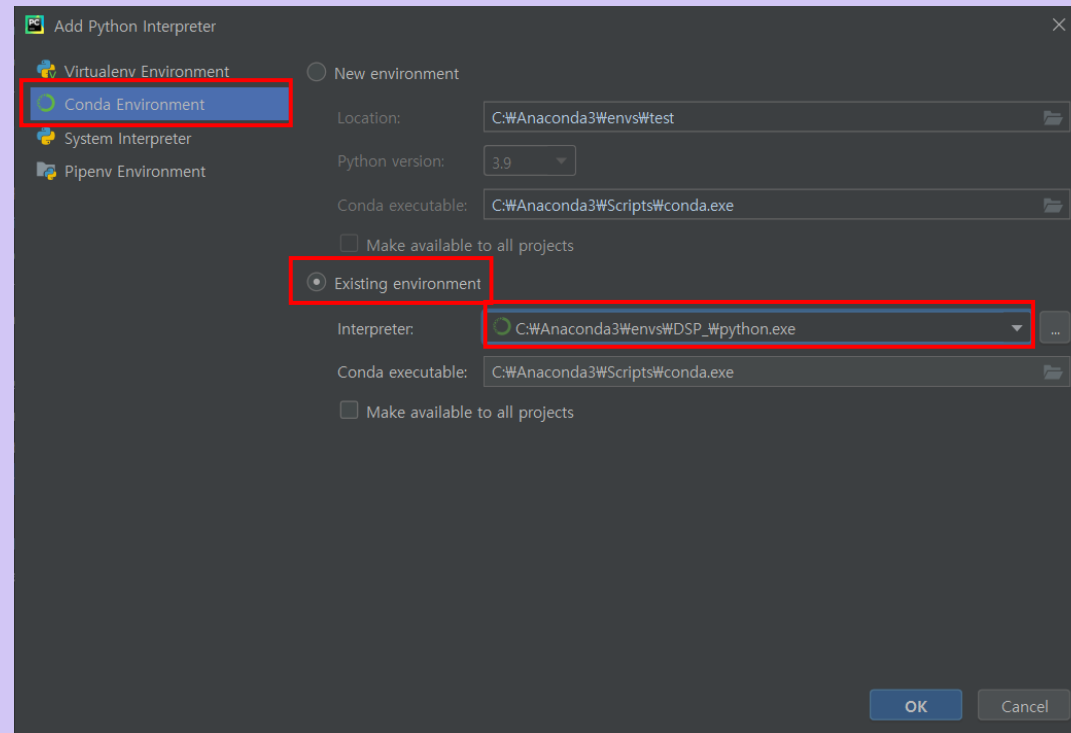
Github에 프로젝트 올리기

• 생성한 가상환경 연동



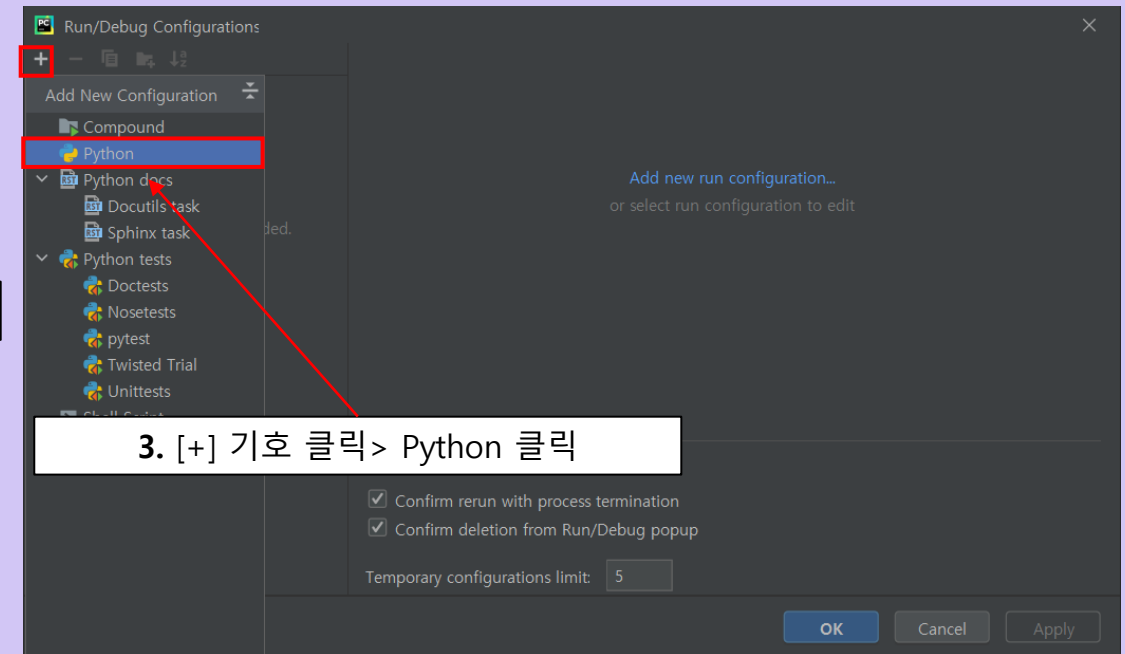
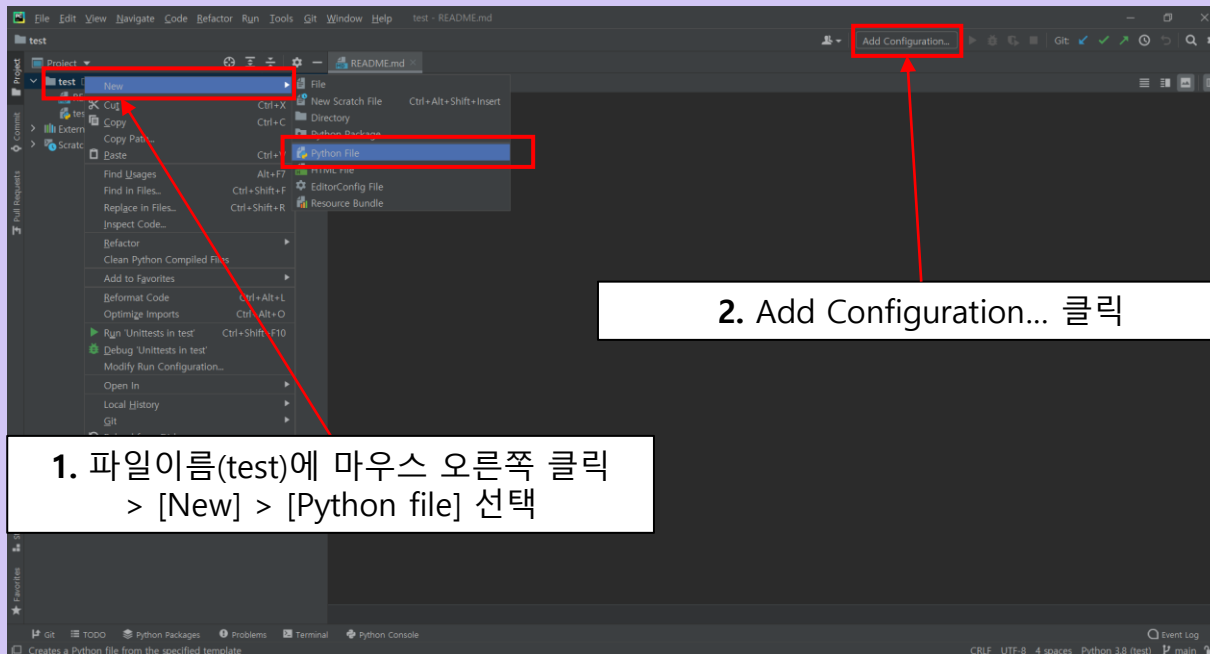
Github에 프로젝트 올리기

- 생성한 가상환경 연동



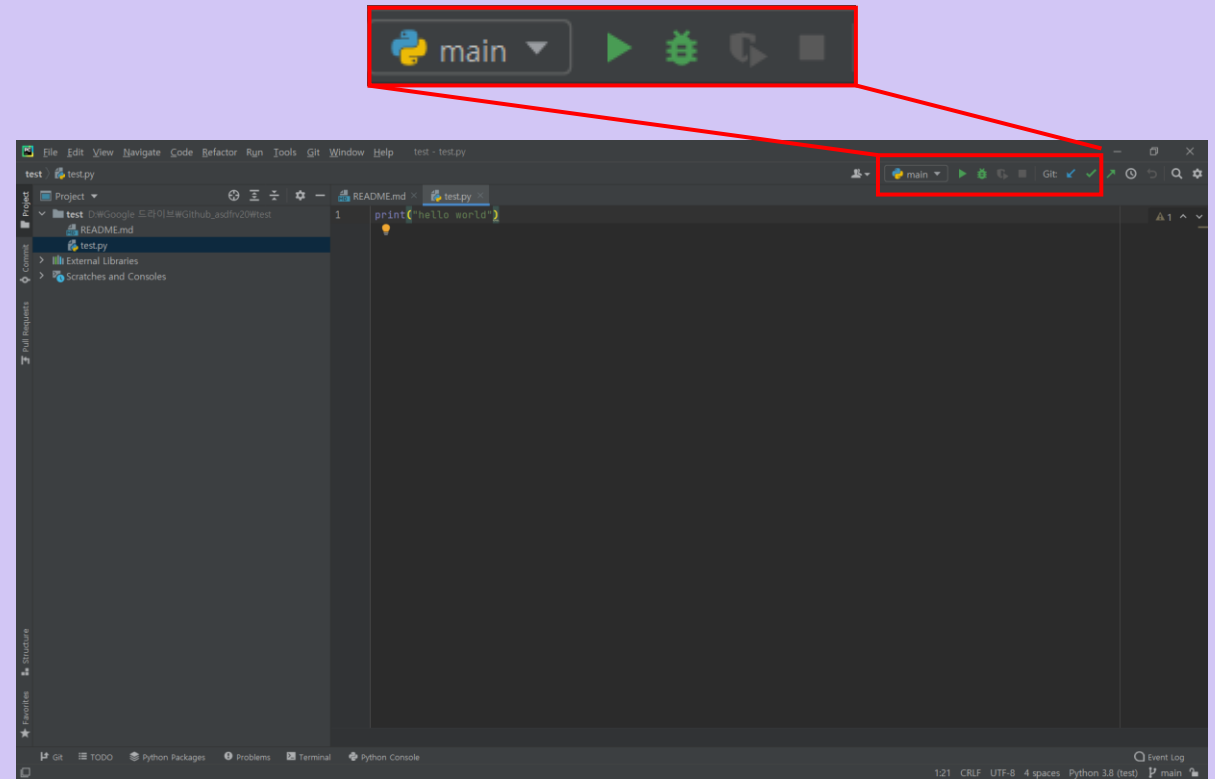
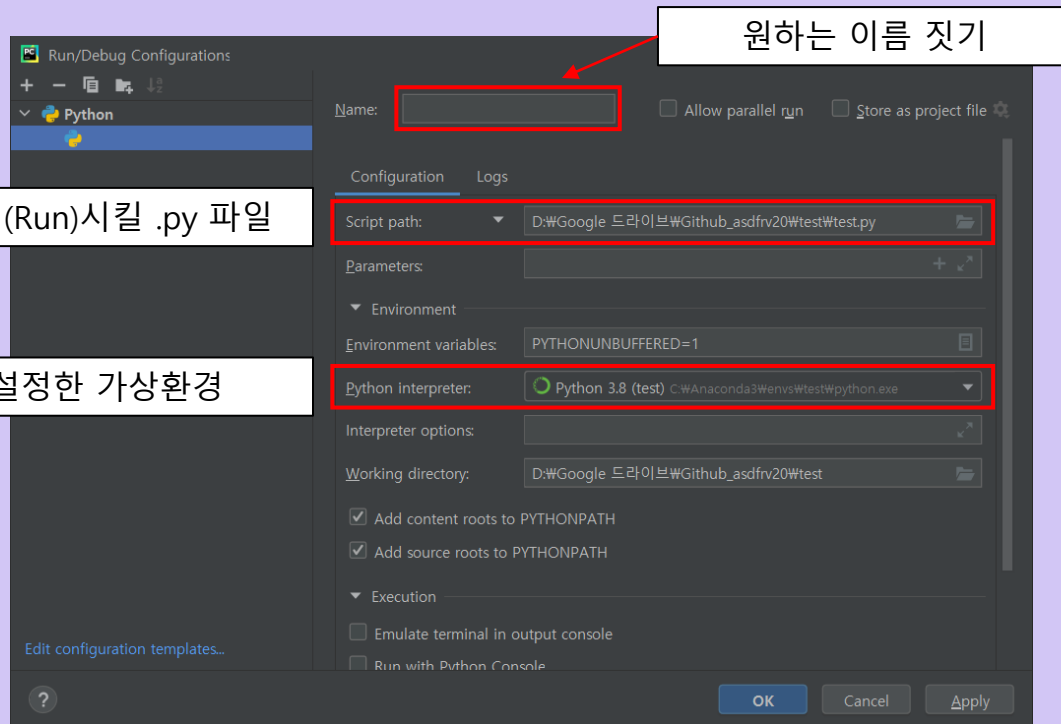
Github에 프로젝트 올리기

• 파일 생성 & Run Configuration 설정



Github에 프로젝트 올리기

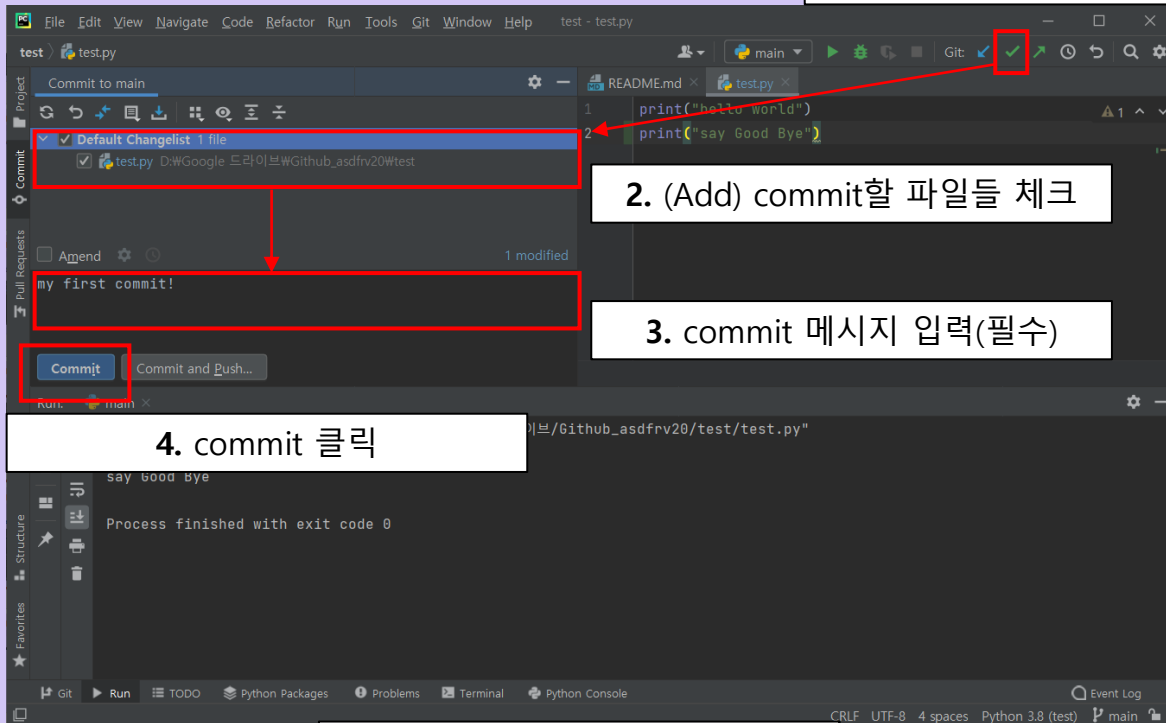
• Run Configuration 설정



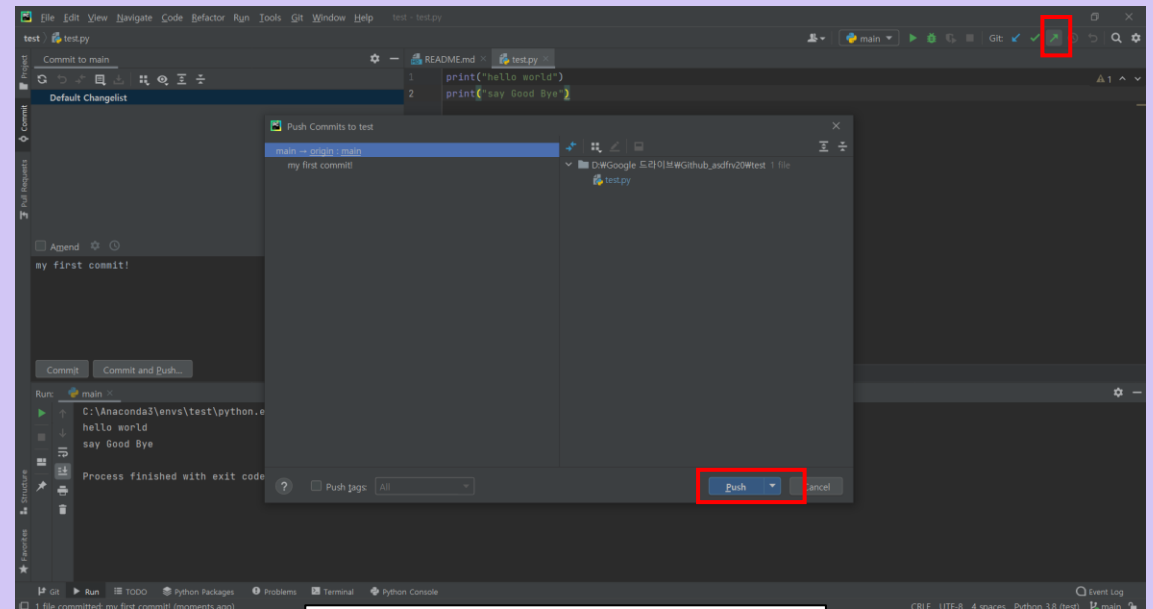
Github에 프로젝트 올리기

• commit & push

1. commit 클릭



commit



push

Github에 프로젝트 올리기

- Github repository에 업로드 되었는지 확인

