

Anaconda 개요 및 설치

데이터 과학과 머신러닝을 위한 필수 도구인 Anaconda의 기본 개념과 설치 방법에 대해 알아보겠습니다. 이번 장에서는 Anaconda가 무엇인지, 왜 사용해야 하는지에 대한 개요를 살펴보고, 실제 설치 과정을 단계별로 안내해드릴 예정입니다.

1. Anaconda 개요

Anaconda는 데이터 과학과 머신러닝을 위한 가장 인기 있는 Python/R 배포판입니다. 이는 수천 개의 오픈소스 패키지와 라이브러리를 포함하는 올인원 플랫폼으로, 데이터 분석, 예측 분석, 과학 컴퓨팅을 위한 완벽한 환경을 제공합니다.

데이터 과학자, 연구원, 학생들에게 필수적인 도구인 Anaconda는 패키지 관리와 환경 관리를 단순화하여, 복잡한 데이터 과학 프로젝트를 더욱 효율적으로 수행할 수 있게 해줍니다.

Anaconda란 무엇인가?

Anaconda는 데이터 분석, 인공지능(AI), 기계 학습과 같은 작업을 쉽게 시작할 수 있도록 도와주는 **통합 소프트웨어 모음**입니다.

컴퓨터로 작업을 시작하기 위해서는 다양한 프로그램과 설정이 필요하지만, Anaconda는 이러한 과정에서 필요한 도구를 한 번에 제공하여 사용자에게 편리함을 제공합니다.

비유하자면, 요리를 하기 위해 재료와 도구를 별도로 준비하는 대신, 요리 키트처럼 필요한 모든 것을 한꺼번에 제공하는 제품이라고 볼 수 있습니다.

Anaconda를 사용하는 이유

Anaconda를 사용하는 주된 이유는 데이터 분석과 관련된 작업을 **더 쉽게 시작하고 관리**할 수 있기 때문입니다.

데이터 분석이나 프로그래밍 작업을 하려면 다양한 프로그램과 라이브러리(도구들)를 설치하고 설정해야 하는데, 이 과정은 복잡할 수 있습니다.

Anaconda는 이러한 작업을 간소화하여 한 번의 설치로 필요한 환경을 준비할 수 있게 합니다.

Anaconda가 제공하는 주요 장점

1. 모든 도구가 포함됨

2. 데이터를 분석하거나 머신러닝 작업을 수행하기 위한 프로그램(예: Python, Jupyter Notebook 등)이 함께 설치됩니다.

2. 쉬운 설치와 관리

3. 추가적인 설정 없이 필요한 도구를 한 번에 설치할 수 있으며, 추가적으로 필요한 프로그램도 간단히 설치 가능합니다.

3. 가상 환경 관리 기능

4. 서로 다른 프로젝트에서 사용하는 프로그램 버전이나 설정을 쉽게 구분하고 관리할 수 있습니다.

5. 예를 들어, 하나의 프로젝트에서는 Python 3.8을, 다른 프로젝트에서는 Python 3.9를 사용할 수 있습니다.

Anaconda의 주요 구성 요소

Anaconda는 다음과 같은 도구와 기능을 포함하고 있습니다:

1. Python

- 2. 데이터를 처리하고 분석하는 데 가장 널리 사용되는 프로그래밍 언어입니다.

2. Jupyter Notebook

- 3. 데이터를 분석하거나 시각화(그래프 그리기)하는 작업을 수행할 때 사용하는 도구입니다.
- 4. 코드를 작성하고 실행한 결과를 즉시 확인할 수 있어 편리합니다.
- 5. 이는 마치 수학 문제를 풀 때 문제와 풀이 과정을 함께 기록할 수 있는 연습장과 비슷합니다.

3. conda

- 4. 프로그램과 라이브러리(필요한 기능들)를 설치하고 관리할 수 있는 도구입니다.
- 5. 스마트폰의 앱스토어처럼 클릭 몇 번으로 프로그램을 추가하거나 제거할 수 있습니다.

4. 다양한 라이브러리

- 5. 데이터 분석이나 시각화를 위한 여러 도구가 포함되어 있습니다.
- 6. 예를 들어, Pandas(데이터 분석), Matplotlib(그래프 그리기), NumPy(수학 계산) 등이 포함됩니다.

Anaconda를 사용하는 실제 사례

1. 데이터 분석

- 2. Anaconda를 사용하여 데이터를 읽고 분석할 수 있습니다.
- 3. 예를 들어, 엑셀 파일의 판매 데이터를 읽어와 어떤 제품이 가장 많이 팔렸는지 분석할 수 있습니다.

2. 머신러닝

- 3. 데이터를 학습시켜 특정 결과를 예측할 수 있습니다.
- 4. 예를 들어, 고객의 구매 데이터를 분석하여 다음에 어떤 제품을 구매할지 예측하는 모델을 만들 수 있습니다.

3. 데이터 시각화

- 4. 데이터를 다양한 형태의 그래프로 표현할 수 있습니다.
- 5. 예를 들어, 월별 매출 증가 추이를 꺾은선 그래프로 나타낼 수 있습니다.

비유를 통한 이해

- **Anaconda**는 요리 키트와 같습니다.
 - 요리를 하기 위해 필요한 레시피(코드), 재료(데이터), 조리 도구(프로그램)가 모두 준비된 상태입니다.
 - 사용자는 복잡한 준비 과정 없이 바로 요리를 시작할 수 있듯, Anaconda를 설치하면 곧바로 데이터를 분석하거나 코딩을 시작할 수 있습니다.
-
- **Jupyter Notebook**은 연습장과 같습니다.
 - 데이터를 분석하는 코드를 작성하고 실행 결과를 확인할 수 있으며, 필요한 경우 메모를 추가하여 기록을 남길 수 있습니다.
- **Conda**는 앱스토어와 같습니다.
 - 필요한 추가 프로그램이나 도구를 간단히 설치하고 제거할 수 있습니다.

Anaconda의 장점 요약

1. 한 번의 설치로 모든 도구 준비

2. 데이터 분석에 필요한 Python, Jupyter Notebook, 다양한 라이브러리가 한꺼번에 설치됩니다.

2. 편리한 환경 관리

3. 프로젝트마다 서로 다른 설정이나 버전을 쉽게 관리할 수 있습니다.

3. 초보자 친화적인 인터페이스

4. 복잡한 명령어 없이 클릭만으로 필요한 작업을 시작할 수 있습니다.

2. Anaconda 설치

Anaconda는 운영체제별로 각각 다른 설치 방법을 제공하고 있습니다. Windows, macOS, Linux 모두 지원되며, 각 시스템에 맞는 설치 파일을 다운로드하여 설치할 수 있습니다.

Anaconda 설치

- <https://www.anaconda.com/download> 접속
- Skip registration 클릭

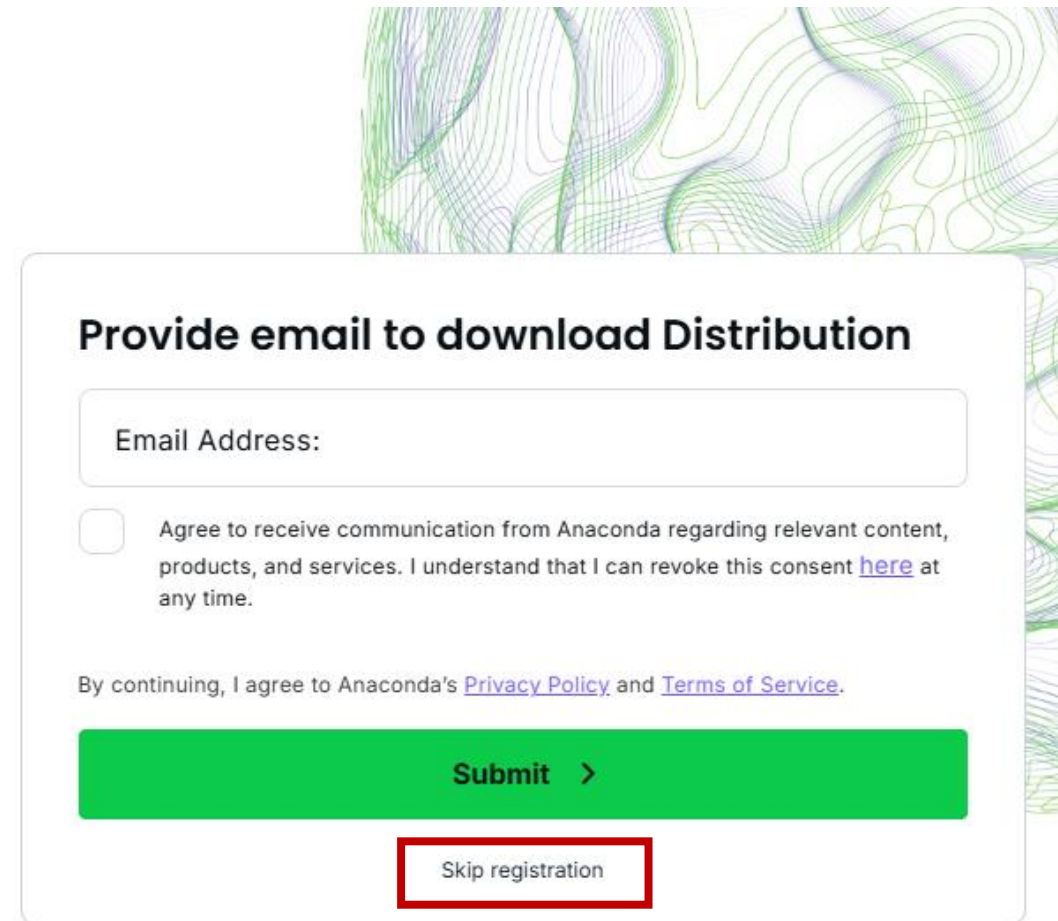
Distribution

Free Download*

Register to get everything you need to get started on your workstation including Cloud Notebooks, Navigator, AI Assistant, Learning and more.

- ✓ Easily search and install thousands of data science, machine learning, and AI packages
- ✓ Manage packages and environments from a desktop application or work from the command line
- ✓ Deploy across hardware and software platforms
- ✓ Distribution installation on Windows, MacOS, or Linux

*Use of Anaconda's Offerings at an organization of more than 200 employees requires a Business or Enterprise license. [See Pricing](#)



Provide email to download Distribution

Email Address:

☐ Agree to receive communication from Anaconda regarding relevant content, products, and services. I understand that I can revoke this consent [here](#) at any time.

By continuing, I agree to Anaconda's [Privacy Policy](#) and [Terms of Service](#).

Submit >

Skip registration

Anaconda 설치

■ Download 클릭

Download Now

For installation assistance, refer to [Troubleshooting](#).

Download Anaconda Distribution or [Miniconda](#) by choosing the proper installer for your machine. Learn the difference from our [Documentation](#).



Anaconda Installers

 Download



Windows

Python 3.12

📄 64-Bit Graphical Installer (912.3M)



Mac

Python 3.12

📄 64-Bit (Apple silicon) Graphical Installer (704.7M)

📄 64-Bit (Apple silicon) Command Line Installer (707.3M)

📄 64-Bit (Intel chip) Graphical Installer (734.7M)

📄 64-Bit (Intel chip) Command Line Installer (731.2M)



Linux

Python 3.12

📄 64-Bit (x86) Installer (1007.9M)

📄 64-Bit (AWS Graviton2 / ARM64) Installer (800.6M)

📄 64-bit (Linux on IBM Z & LinuxONE) Installer (425.8M)

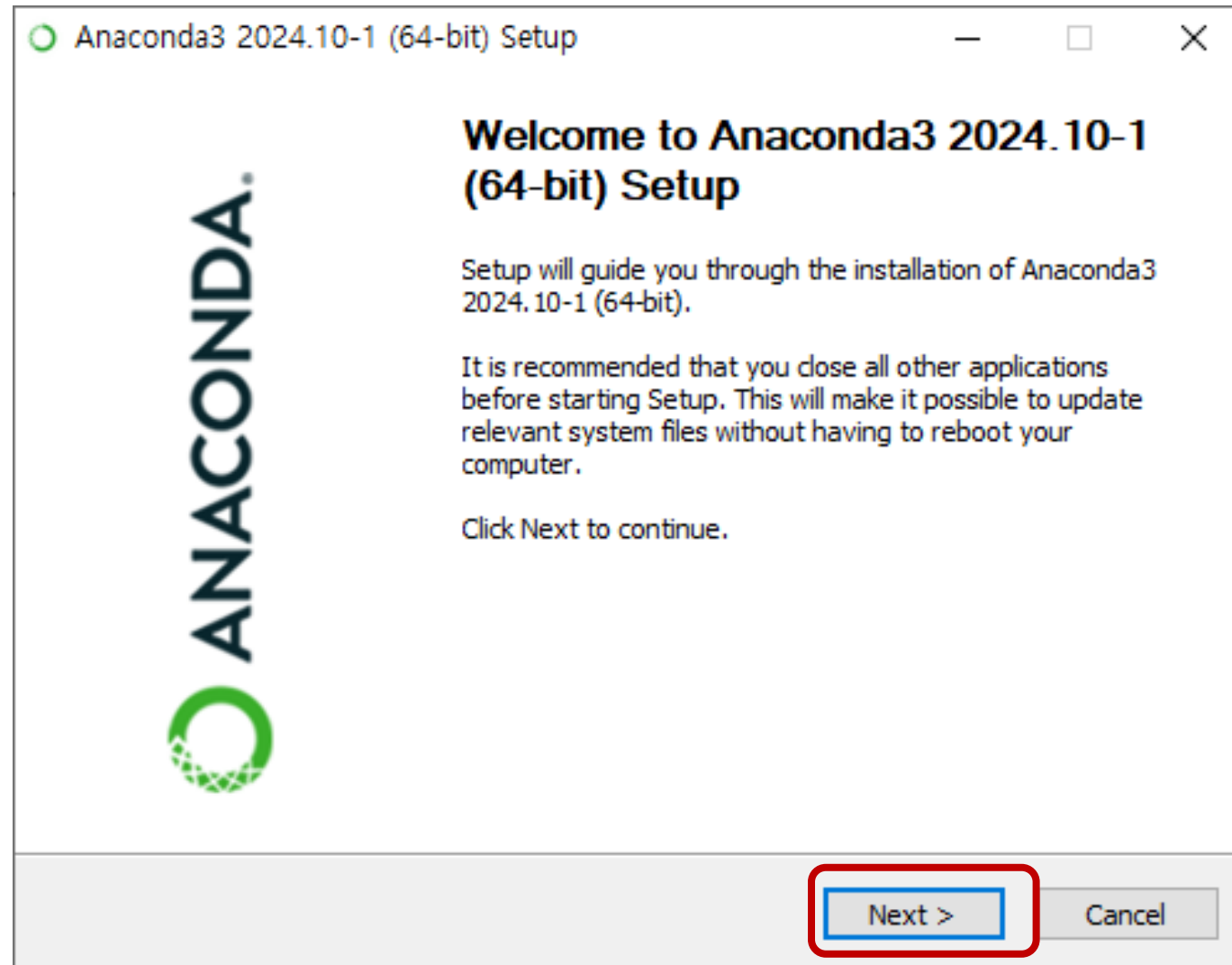
Anaconda 설치

- 다운받은 파일 더블클릭



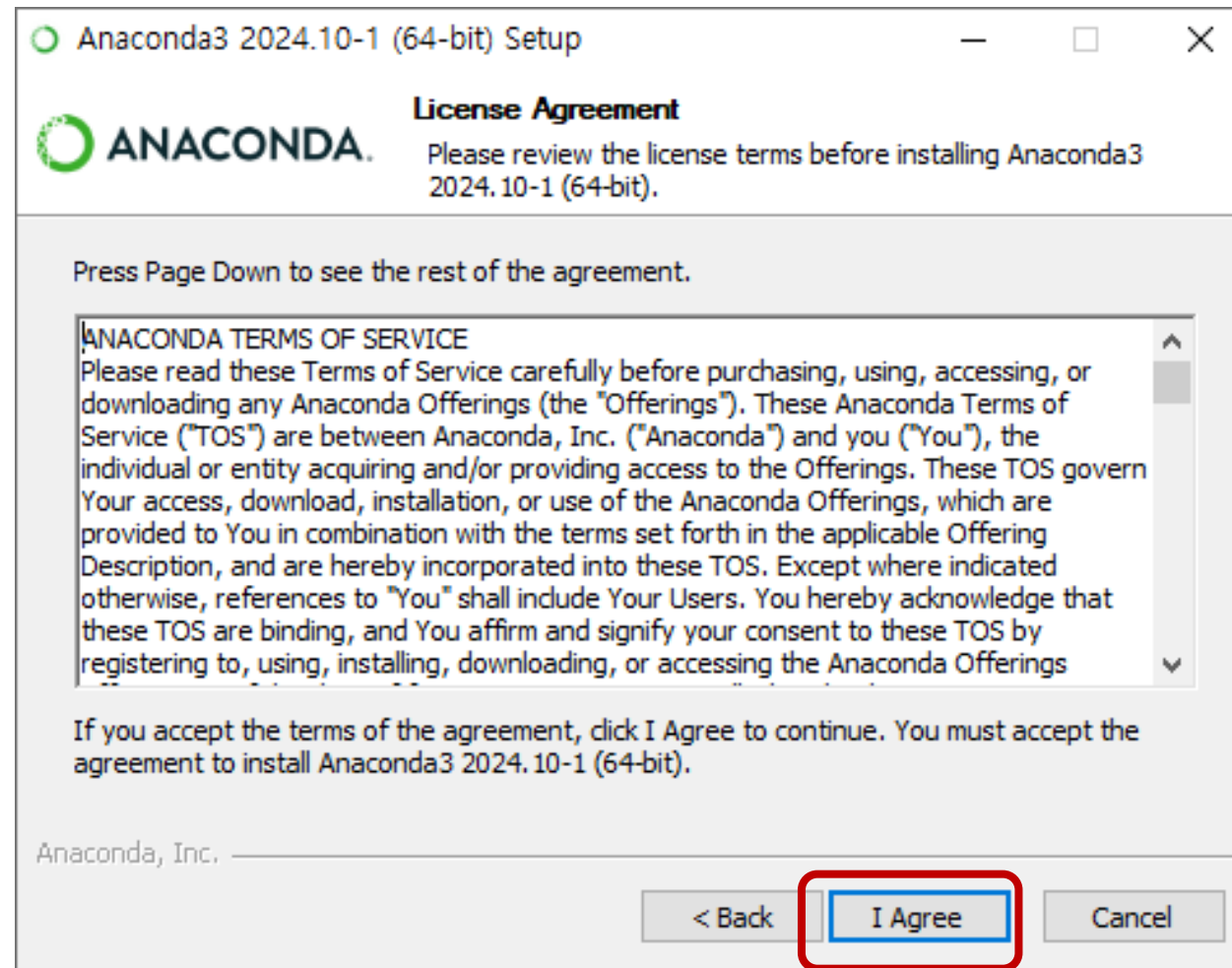
Anaconda 설치

- Next 버튼 클릭



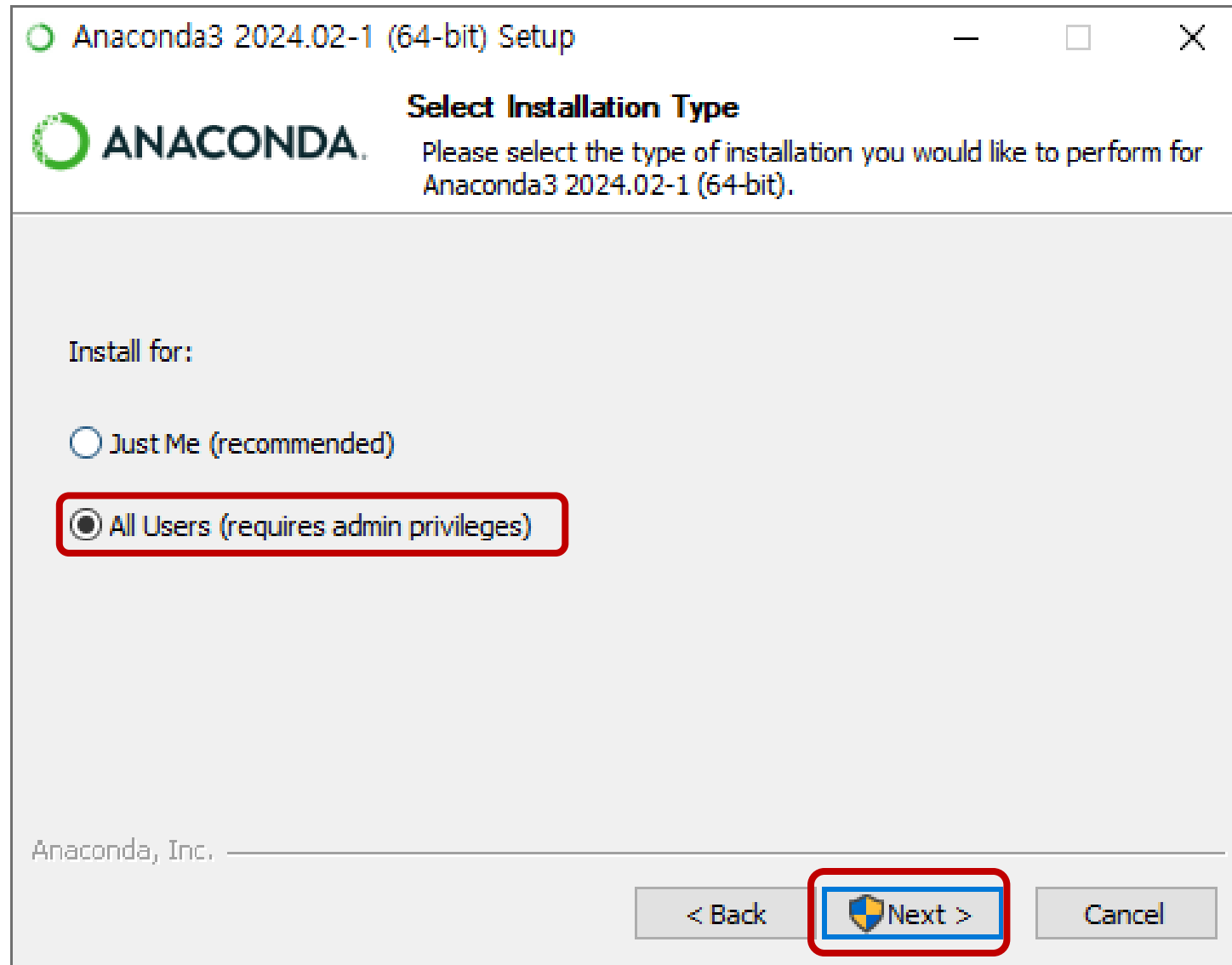
Anaconda 설치

■ I Agree 클릭



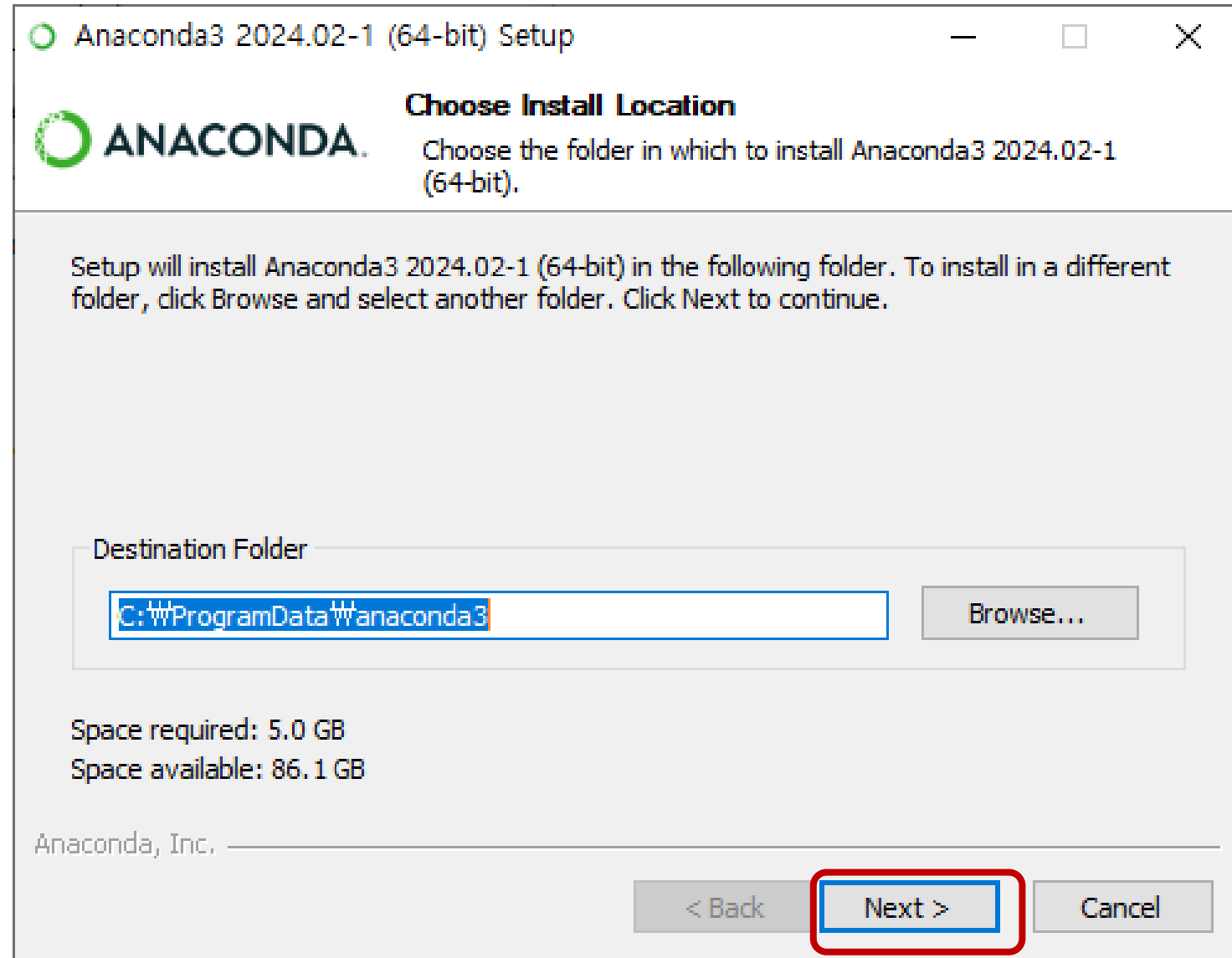
Anaconda 설치

- All Users->Next 클릭



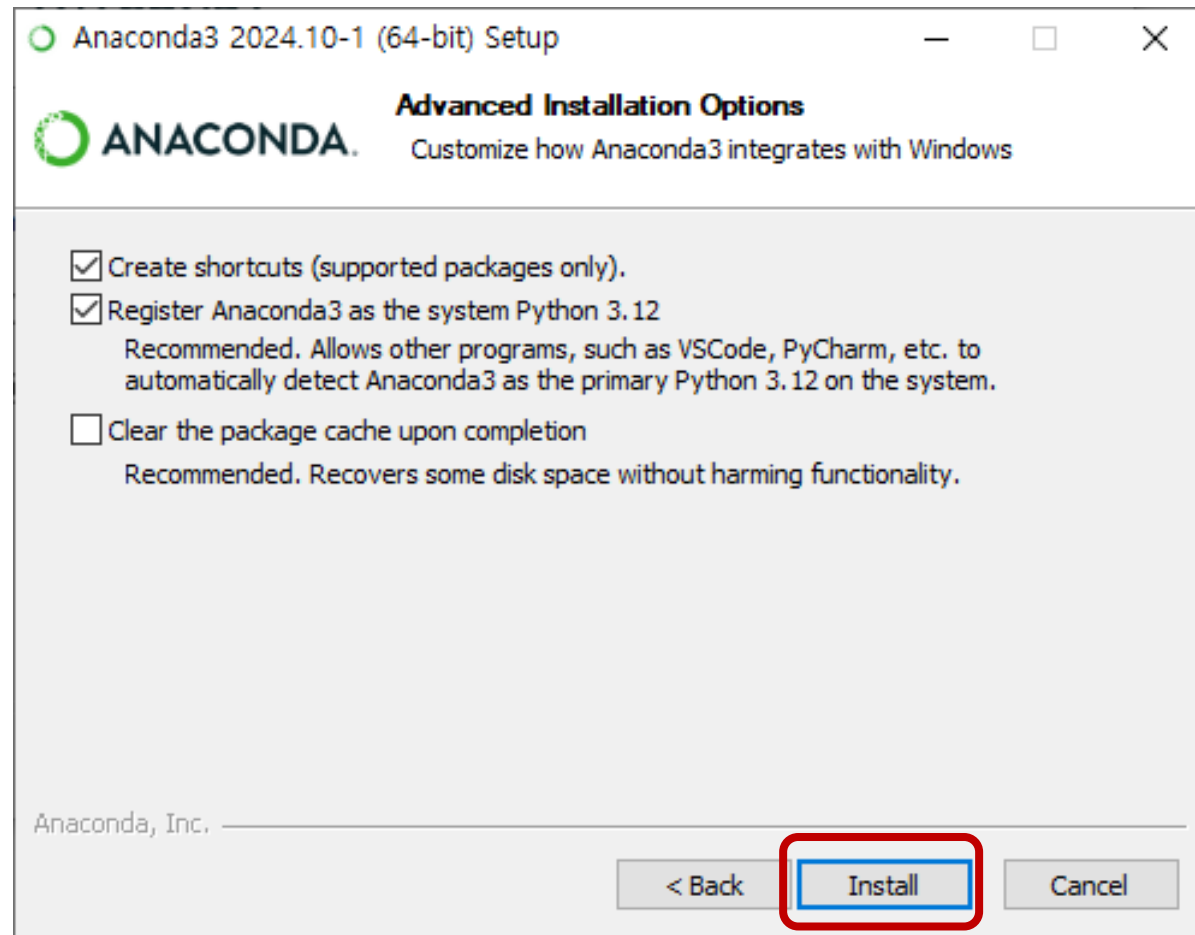
Anaconda 설치

- Next 클릭

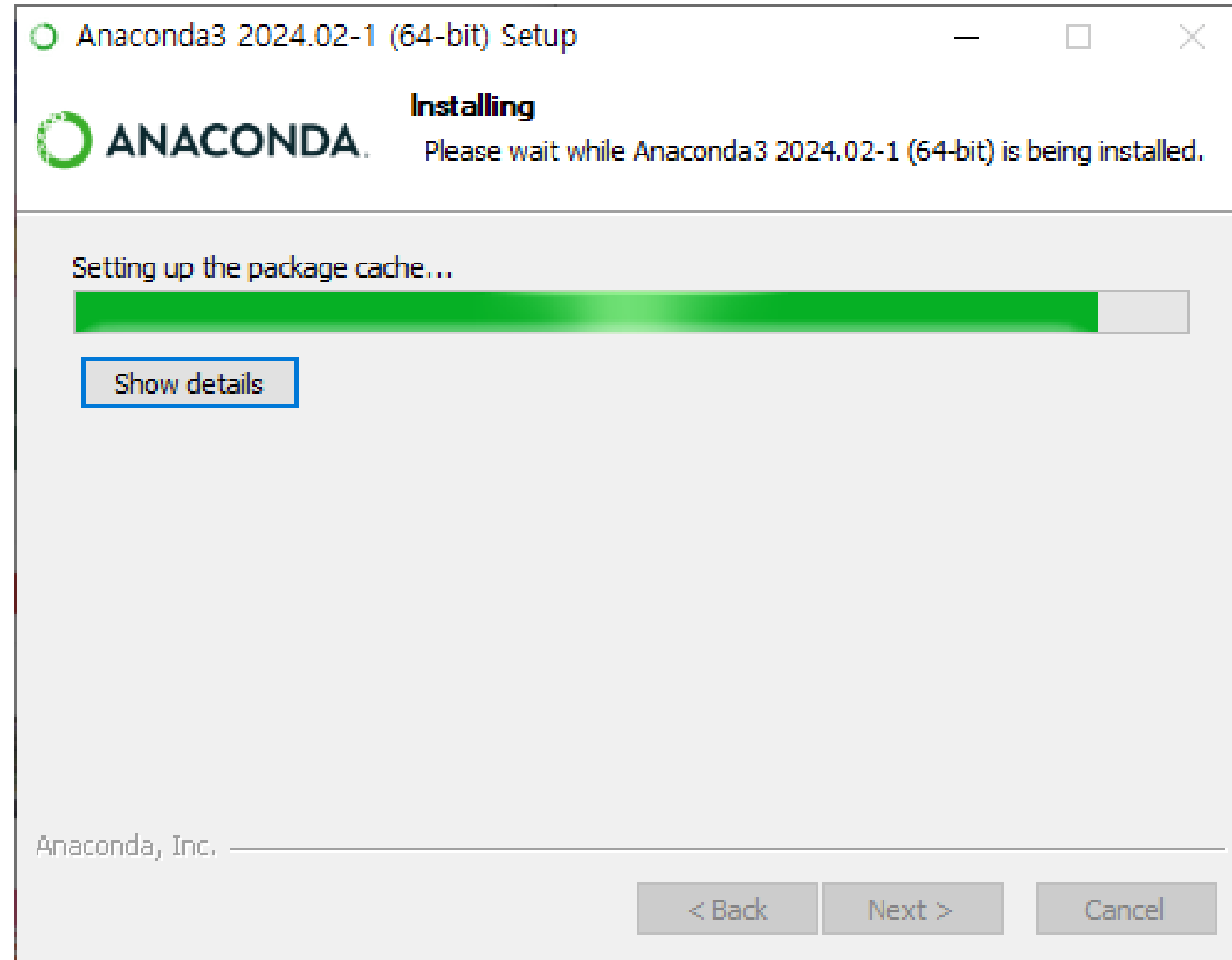


Anaconda 설치

■ Install 클릭

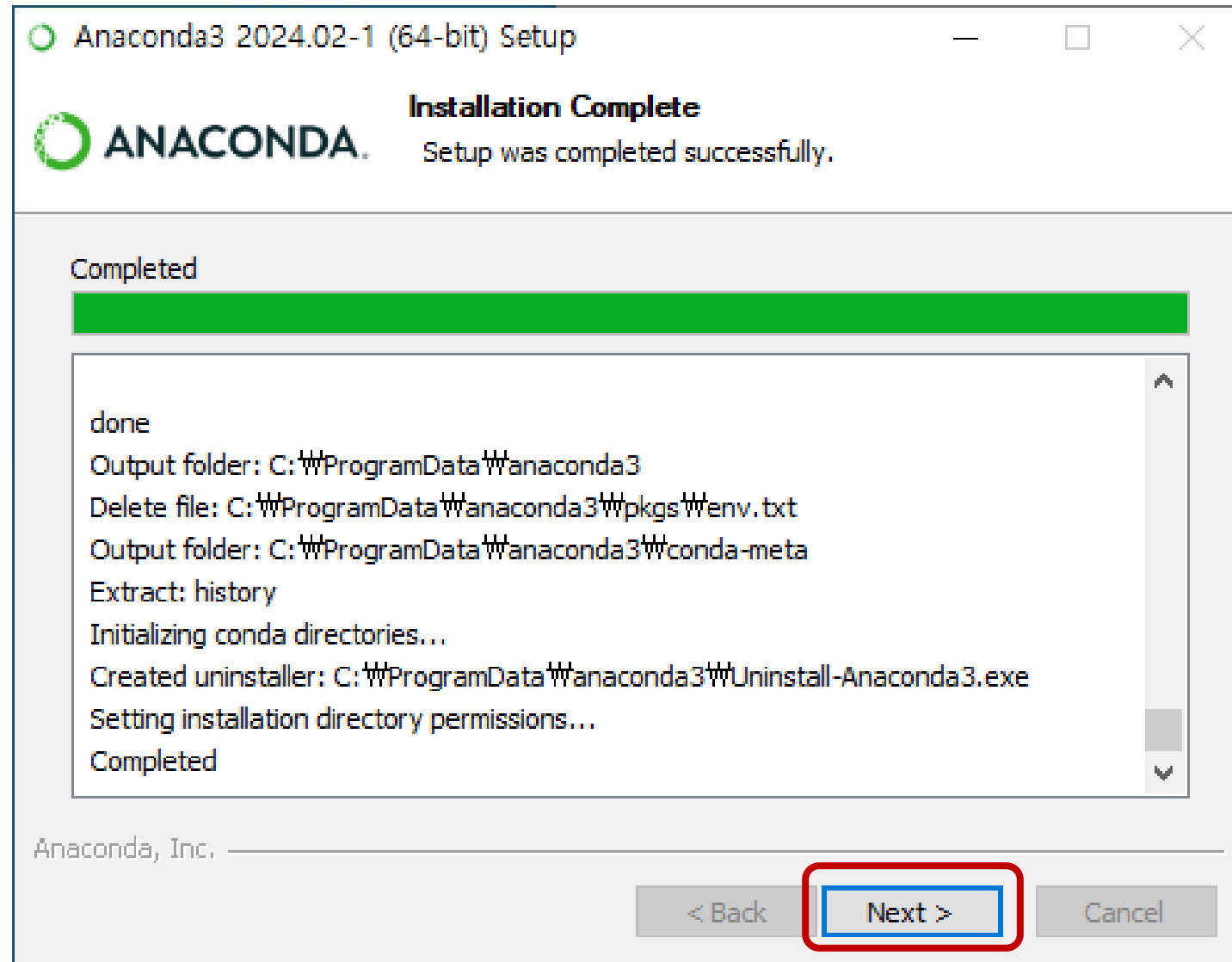


Anaconda 설치



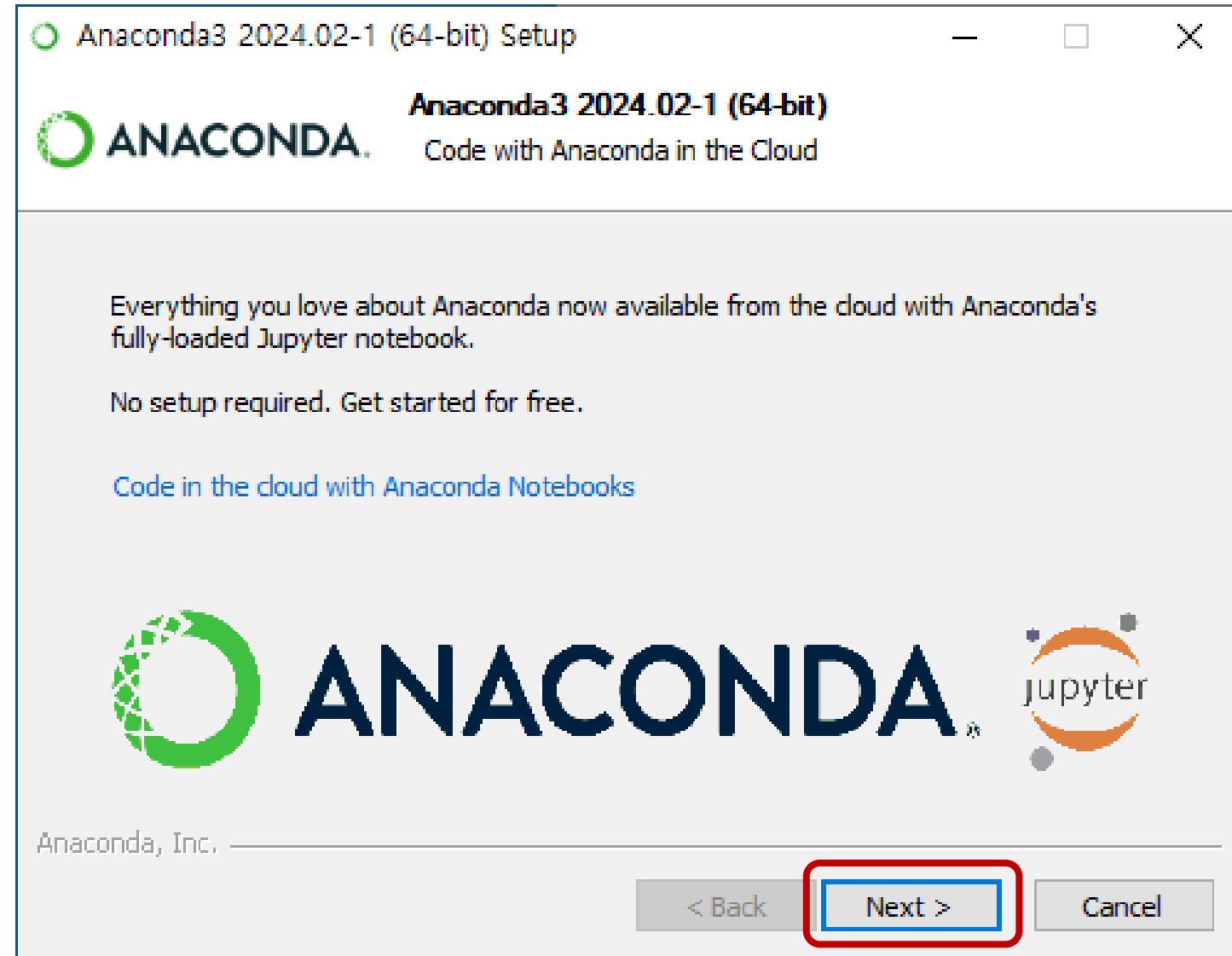
Anaconda 설치

- Next 버튼 클릭



Anaconda 설치

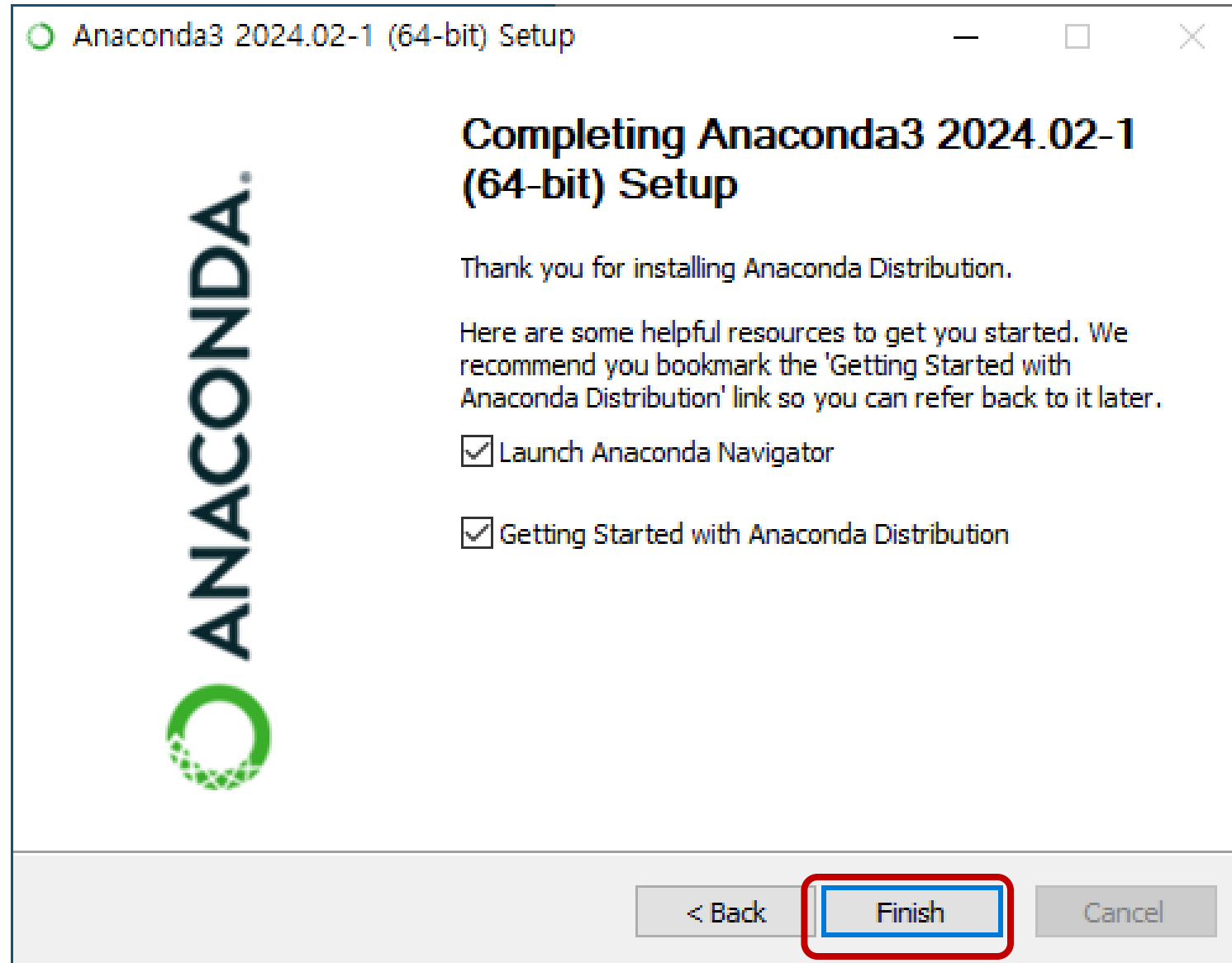
- Next 클릭



20

Anaconda 설치

■ Finish 클릭



21

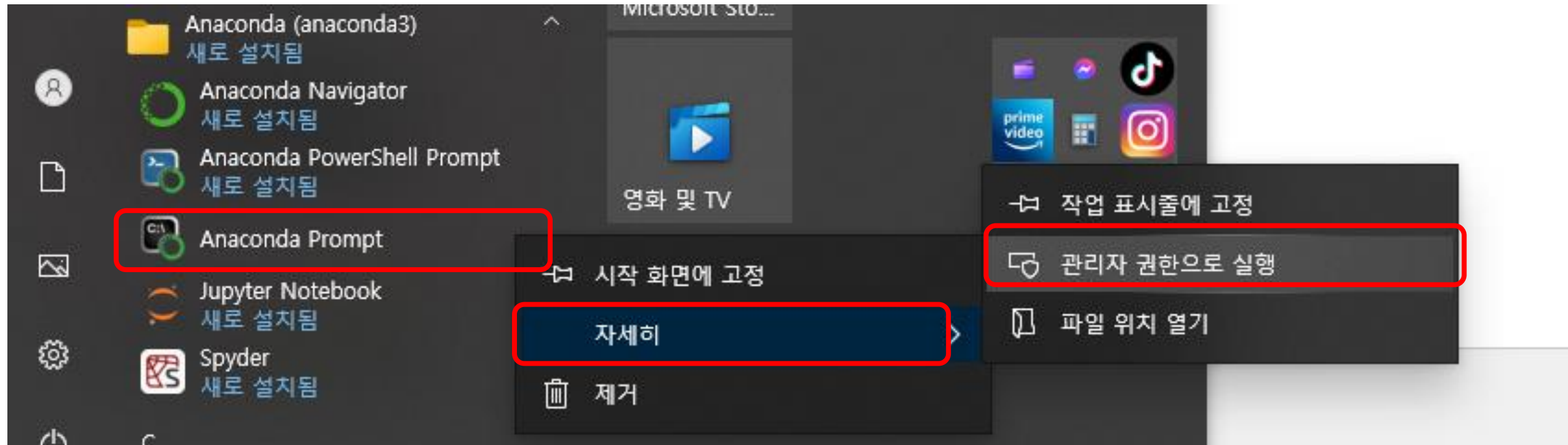
3. 가상 환경 생성

가상 환경은 각 프로젝트가 필요한 파이썬 버전과 라이브러리를 서로 독립적으로 관리하기 위해 사용됩니다

가상 환경

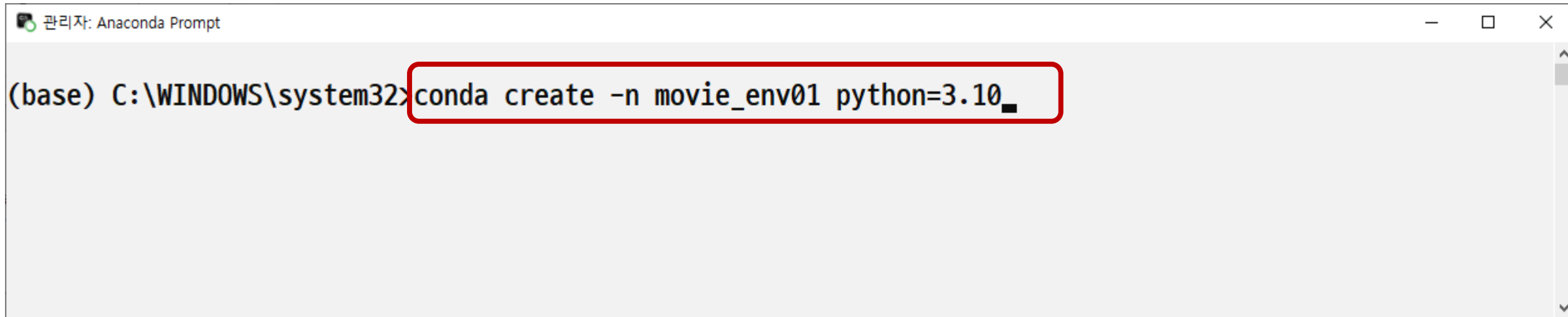
- Anaconda 가상 환경은 데이터 과학, 머신러닝, 또는 소프트웨어 개발을 위한 독립적이고 격리된 작업 환경을 제공하는 기능입니다. 가상 환경은 각 프로젝트가 필요로 하는 특정 버전의 파이썬과 라이브러리를 독립적으로 관리할 수 있도록 도와줍니다. Anaconda는 이를 쉽게 생성하고 관리할 수 있는 도구를 제공합니다.
- 주요 특징
 - 독립성: 각 가상 환경은 고유의 파이썬 버전과 라이브러리를 가질 수 있습니다. 이는 다른 프로젝트와의 충돌을 방지합니다.
 - 라이브러리 관리: Anaconda의 패키지 매니저인 conda를 사용하여 라이브러리를 간편하게 설치, 삭제, 업데이트할 수 있습니다.
 - 재현성: 가상 환경에 설치된 패키지를 목록으로 만들어 다른 사용자나 시스템에서 동일한 환경을 재현할 수 있습니다.
- 사용 예시
 - 데이터 분석 프로젝트 A에서는 Python 3.8과 Pandas 1.3.0을, 머신러닝 프로젝트 B에서는 Python 3.9와 TensorFlow 2.10.0을 필요로 한다고 가정하면, 각각의 가상 환경을 만들어 프로젝트에 맞는 설정을 유지할 수 있습니다. 이를 통해 효율적이고 안정적인 개발 환경을 구축할 수 있습니다.

가상 환경 생성



가상 환경 생성

■ 다음을 입력 합니다



```
(base) C:\WINDOWS\system32>conda create -n movie_env01 python=3.10_
```

- 이 명령어는 Anaconda에서 movie_env01이라는 이름의 가상 환경을 생성합니다. 그리고 파이썬 3.10 버전을 해당 환경에 설치합니다.
- 구성 요소별 설명:
 - conda: Anaconda에서 제공하는 패키지과 환경 관리 명령어입니다.
 - create: 새로운 가상 환경을 생성하라는 명령어입니다.
 - -n movie_env01: movie_env01이라는 이름으로 가상 환경을 만듭니다.-n은 이름(name)을 지정하기 위한 옵션입니다.
 - python=3.10: 가상 환경에 설치할 파이썬의 버전을 지정합니다.

가상 환경 생성

```
ca-certificates-2023.12.12 | haa95532_0 | 127 KB
openssl-3.0.12 | h2bfff1b_0 | 7.4 MB
pip-23.3.1 | py310haa95532_0 | 2.9 MB
python-3.10.13 | he1021f5_0 | 15.8 MB
setuptools-68.2.2 | py310haa95532_0 | 942 KB
wheel-0.41.2 | py310haa95532_0 | 127 KB
xz-5.4.5 | h8cc25b3_0 | 593 KB
```

```
-----
Total: 27.8 MB
```

The following NEW packages will be INSTALLED:

```
bzip2 | pkgs/main/win-64::bzip2-1.0.8-he774522_0
ca-certificates | pkgs/main/win-64::ca-certificates-2023.12.12-haa95532_0
libffi | pkgs/main/win-64::libffi-3.4.4-hd77b12b_0
openssl | pkgs/main/win-64::openssl-3.0.12-h2bfff1b_0
pip | pkgs/main/win-64::pip-23.3.1-py310haa95532_0
python | pkgs/main/win-64::python-3.10.13-he1021f5_0
setuptools | pkgs/main/win-64::setuptools-68.2.2-py310haa95532_0
sqlite | pkgs/main/win-64::sqlite-3.41.2-h2bfff1b_0
tk | pkgs/main/win-64::tk-8.6.12-h2bfff1b_0
tzdata | pkgs/main/noarch::tzdata-2023c-h04d1e81_0
vc | pkgs/main/win-64::vc-14.2-h21ff451_1
vs2015_runtime | pkgs/main/win-64::vs2015_runtime-14.27.29016-h5e58377_2
wheel | pkgs/main/win-64::wheel-0.41.2-py310haa95532_0
xz | pkgs/main/win-64::xz-5.4.5-h8cc25b3_0
zlib | pkgs/main/win-64::zlib-1.2.13-h8cc25b3_0
```

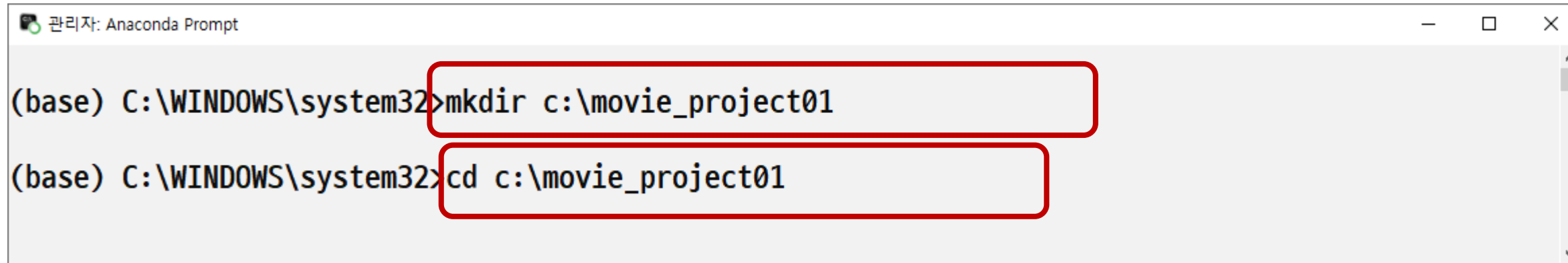
Proceed ([y]/n)? y

4. 작업 폴더 생성

실습을 위한 작업 폴더를 생성 합니다

작업 폴더 생성

- 아래 명령을 입력 합니다



```
관리자: Anaconda Prompt
(base) C:\WINDOWS\system32>mkdir c:\movie_project01
(base) C:\WINDOWS\system32>cd c:\movie_project01
```

- 명령어 1:

- mkdir c:\movie_project01이 명령어는 새로운 폴더를 만드는 것입니다.
- 구성 요소:
 - mkdir: Make Directory의 줄임말로, 새 폴더를 만들라는 명령어입니다.
 - c:\movie_project01: 폴더 이름과 위치를 지정합니다. 여기서는 C 드라이브에 movie_project01이라는 폴더를 만듭니다.

- 명령어 2:

- cd c:\movie_project01이 명령어는 우리가 방금 만든 폴더로 이동하는 것입니다.
- 구성 요소:
 - cd: Change Directory의 줄임말로, 다른 폴더로 이동하라는 명령어입니다.c:\movie_project01: 이동할 폴더의 위치를 지정합니다.

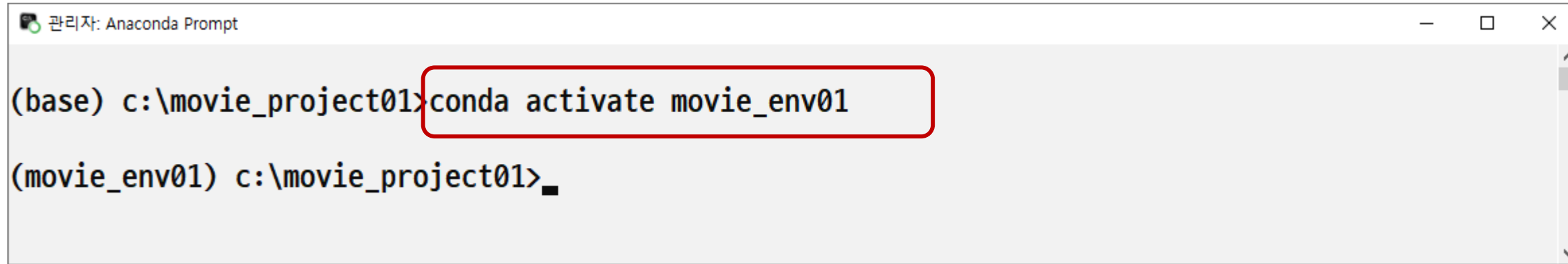
5. 개발 환경 설치

가상 환경에 인공지능 코딩을 하는 Jupyter notebook 설치 합니다

.

개발 환경 설치

- 아래 명령을 입력 합니다



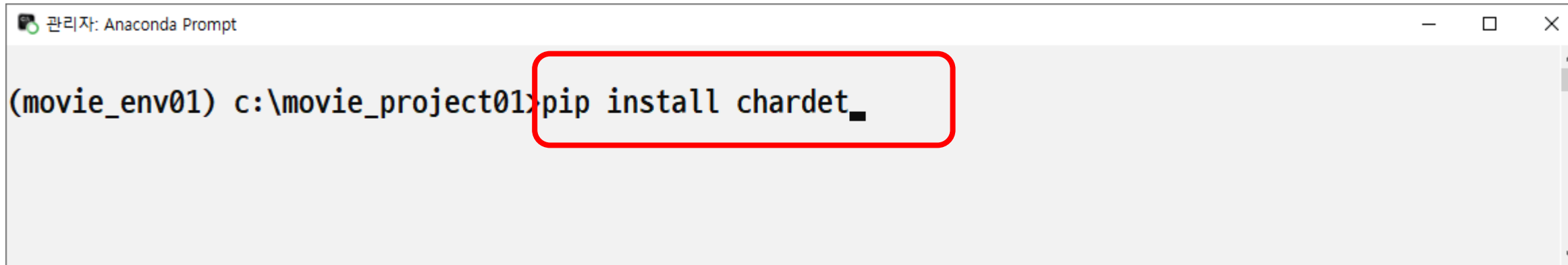
```
관리자: Anaconda Prompt
(base) c:\movie_project01>conda activate movie_env01
(movie_env01) c:\movie_project01>_
```

- 이 명령어의 의미:

- conda: Anaconda라는 프로그램에서 사용되는 명령어입니다.
- activate: 특정 가상 환경을 사용하도록 전환하는 명령입니다.
- movie_env01: 이전에 만들어둔 가상 환경의 이름입니다.
- 가상 환경을 활성화하는 것은 프로젝트 전용 작업 공간에 들어가는 것과 같습니다.movie_env01은 특정 작업(예: 영화 프로젝트)을 위한 개인 작업 공간입니다.이 공간 안에서는 다른 프로젝트와 관련 없는 도구나 설정을 사용할 걱정 없이 해당 프로젝트에 최적화된 환경에서 작업할 수 있습니다.즉, 이 작업 공간은 영화 프로젝트를 위한 "전용 책상" 같은 역할을 합니다.

개발 환경 설치

- 아래를 입력 하세요



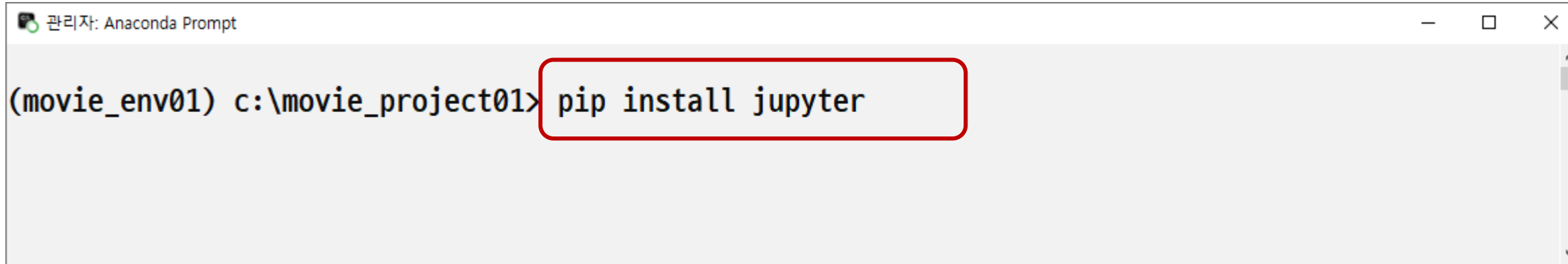
The screenshot shows an Anaconda Prompt window with the title bar '관리자: Anaconda Prompt'. The command prompt shows the current directory as 'c:\movie_project01' and the command 'pip install chardet' is being entered. The command is highlighted with a red rounded rectangle.

```
(movie_env01) c:\movie_project01>pip install chardet_
```

- chardet을 설치 합니다
- chardet는 파일의 문자 인코딩(예: UTF-8, EUC-KR 등)을 자동으로 감지해주는 파이썬 패키지입니다.문자 인코딩이 깨질 때 이 패키지를 사용하면 원인을 파악하고 문제를 해결할 수 있습니다.

개발 환경 설치

- 아래를 입력 하세요

A screenshot of the Anaconda Prompt window. The title bar reads '관리자: Anaconda Prompt'. The command prompt shows '(movie_env01) c:\movie_project01>'. The command 'pip install jupyter' is being typed and is highlighted with a red rounded rectangle. The window has standard Windows window controls (minimize, maximize, close) in the top right corner.

```
(movie_env01) c:\movie_project01> pip install jupyter
```

- Jupyter Notebook 을 설치 합니다
- Jupyter Notebook은 파이썬 코드를 작성, 실행, 결과를 시각적으로 확인할 수 있는 도구로서, 프로젝트 작업을 보다 효율적으로 수행할 수 있도록 지원합니다. 명령어 pip install jupyter를 통해 설치가 가능하며, 설치 후에는 jupyter notebook 명령어를 통해 실행할 수 있습니다. 이를 활용하여 영화 프로젝트와 같은 데이터 분석 작업을 체계적으로 진행할 수 있습니다.