유니티와 ML-Agents 의 설치

1. 유니티 설치

1.1 유니티 허브 다운로드 및 설치

먼저 유니티 허브(Unity Hub)와 유니티의 설치를 진행하겠습니다. 유니티 허브는 유니티의 프로젝트 및 다양한 버전의 유니티를 설치, 관리할 수 있는 프로그램입니다. 유니티 홈페이지(https://unity.com/kr)로 이동한 다음 [다운로드]나 [시작하기] 버튼을 클릭합니다.

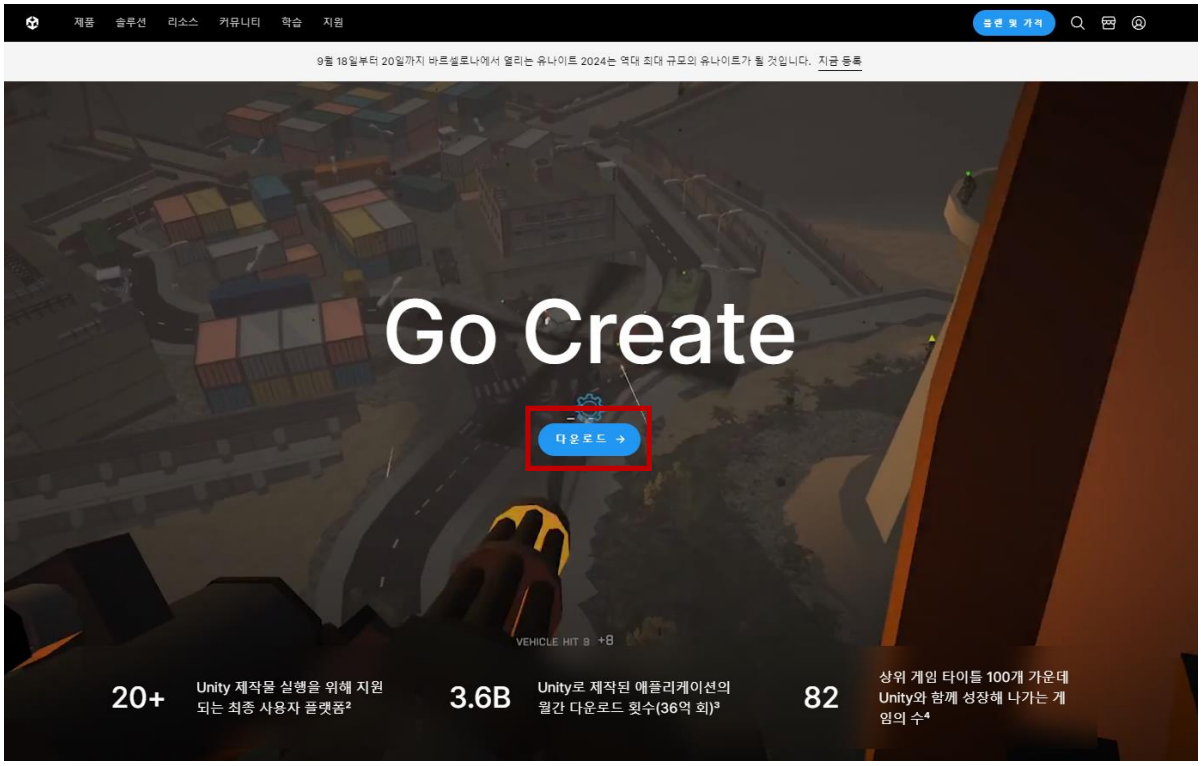


그림 1. 유니티 홈페이지에서 [다운로드] 버튼 클릭

[다운로드] 버튼을 클릭하면 유니티 허브를 다운받을 수 있는 페이지로 이동합니다. 여기서 [다운로드] 버튼을 클릭하면 유니티 허브를 내려받을 수 있습니다. 유니티 허브는 유니티 프로젝트를 관리하고, 유니티 에디터의 버전을 관리할 수 있는 유용한 툴입니다.

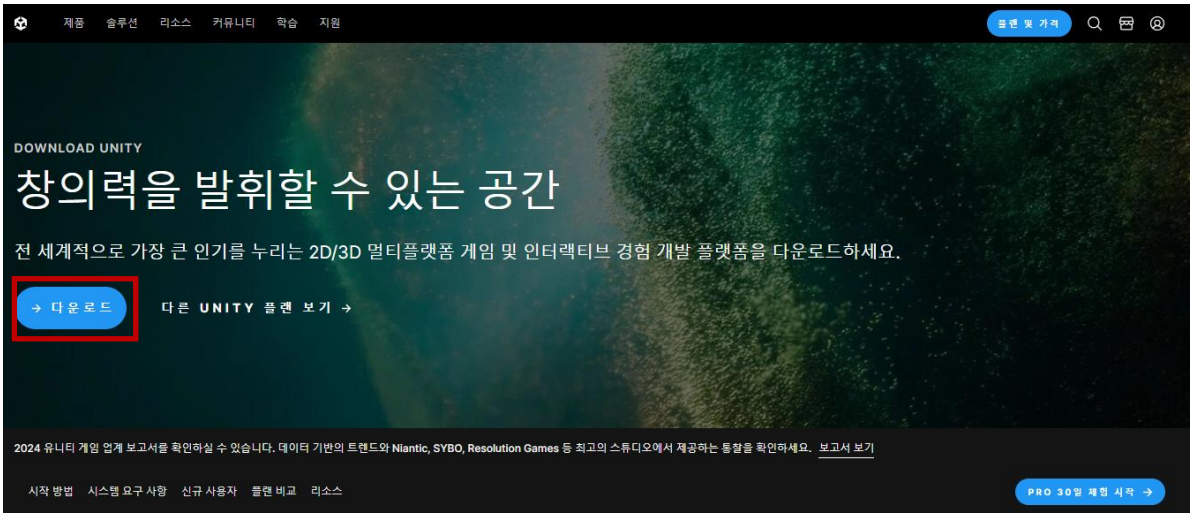


그림 2. 유니티 허브 다운로드

내려받은 파일을 실행해 설치를 진행하고, 설치가 끝나면 유니티 허브를 실행합니다. 다음과 같은 화면이 나오면 **[로그인]** 버튼을 클릭한 다음 유니티 계정을 이용해 로그인합니다. 계정이 없다면 **[계정 생성]** 링크를 클릭해 계정을 생성한 다음 로그인을 진행합니다.

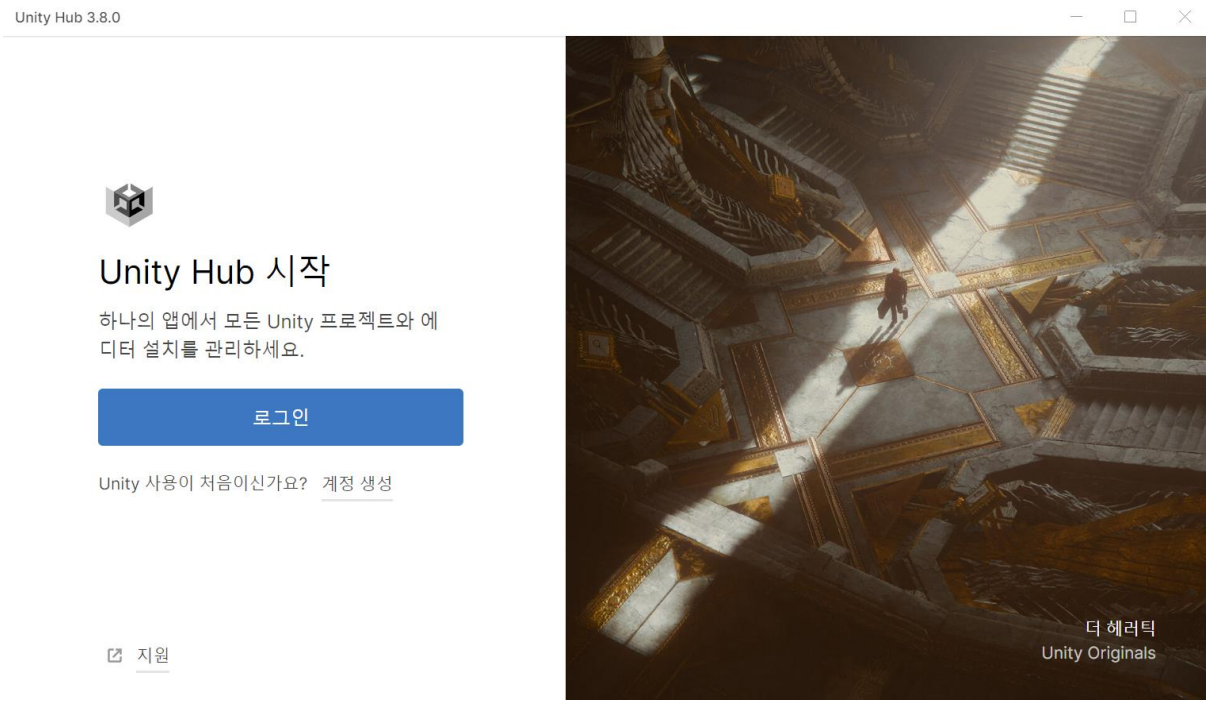


그림 3. 유니티 허브 실행 화면

유니티 허브를 설치하고 실행하는 과정은 유니티 홈페이지나 유니티 허브 버전이 변경됨에 따라 조금씩 바뀔 수 있지만, 전체적인 과정은 크게 다르지 않습니다.

1.2 유니티 라이선스 활성화

유니티 허브를 처음 실행하면 활성 라이선스가 없이 때문에 활성 라이선스가 없다는 알림을 확인할 수 있습니다. **[라이선스 관리]** 버튼을 클릭해 라이선스 관리 창으로 이동합니다.

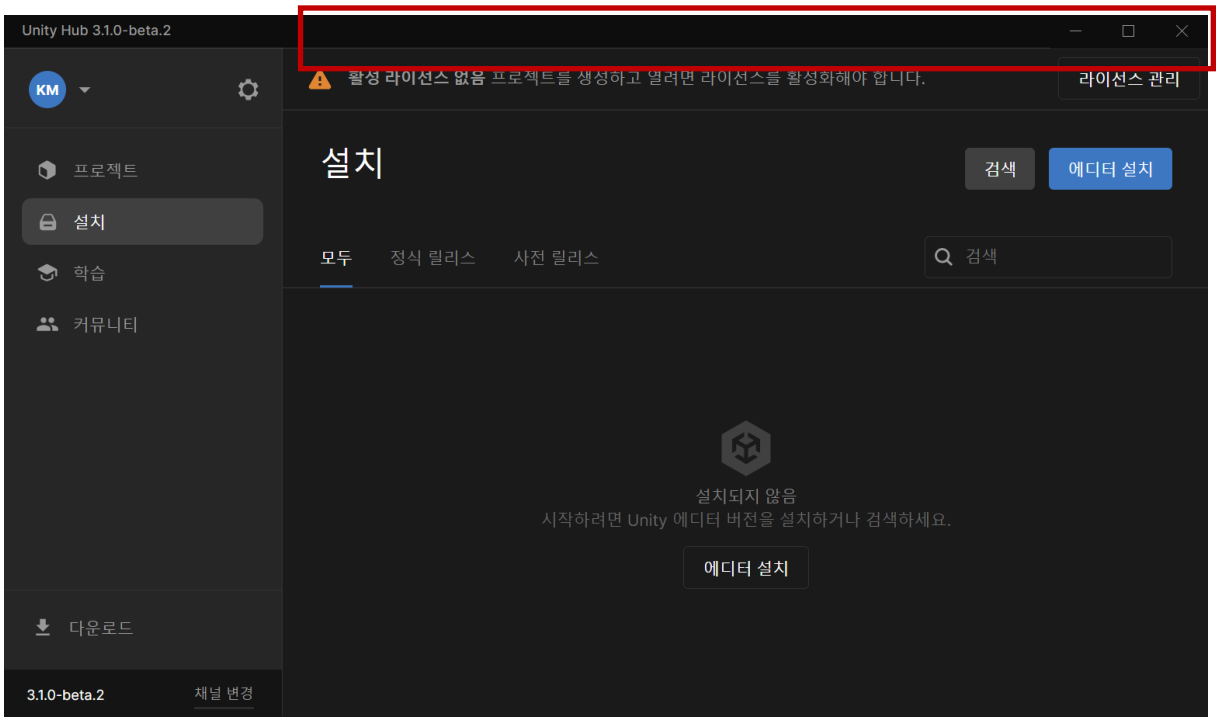


그림 4. 활성 라이선스가 없는 경우 나오는 알림창

환경 설정 창이 나오면 **[라이선스 추가]** 버튼을 클릭합니다.

그림 5. [라이선스 추가] 버튼 클릭

새 라이선스 추가 창이 나오고 라이선스를 활성화 할 수 있는 다양한 방법을 확인할 수 있습니다. 이 책에서는 학습을 위한 무료 라이선스를 사용하고 있으므로 **[무료 Personal 라이선스 받기]**를 클릭합니다.

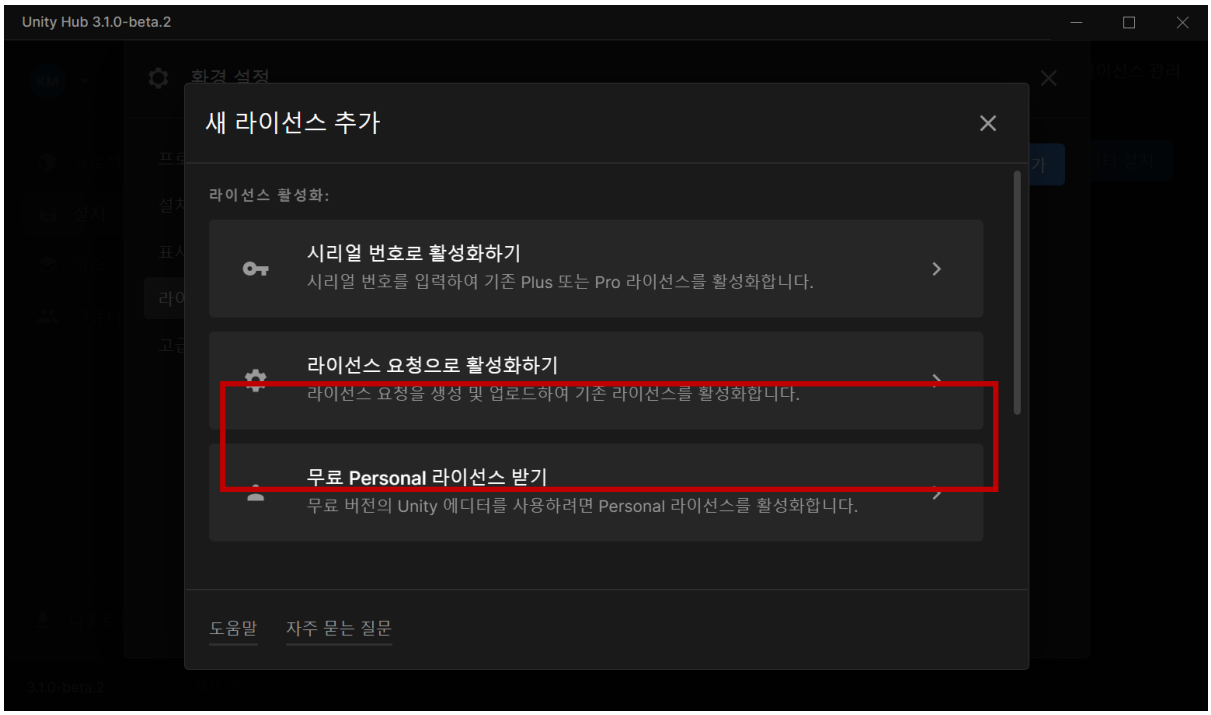


그림 6. [무료 Personal 라이선스 받기] 버튼 클릭

Personal 라이선스 받기 창이 나오면 **[동의하고 Personal 라이선스 받기]** 버튼을 클릭합니다.

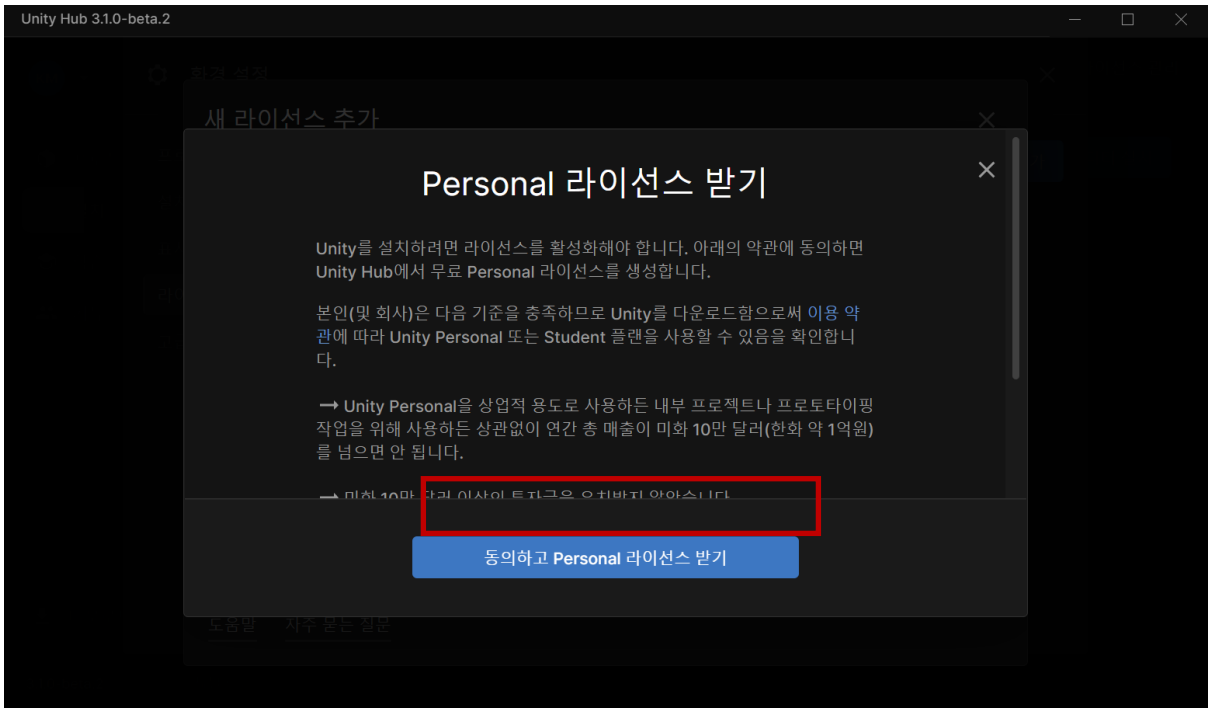


그림 7. [동의하고 Personal 라이선스 받기] 버튼 클릭

여기까지의 내용을 진행하면 다음과 같이 Personal 라이선스가 활성화 된 것을 확인할 수 있습니다.

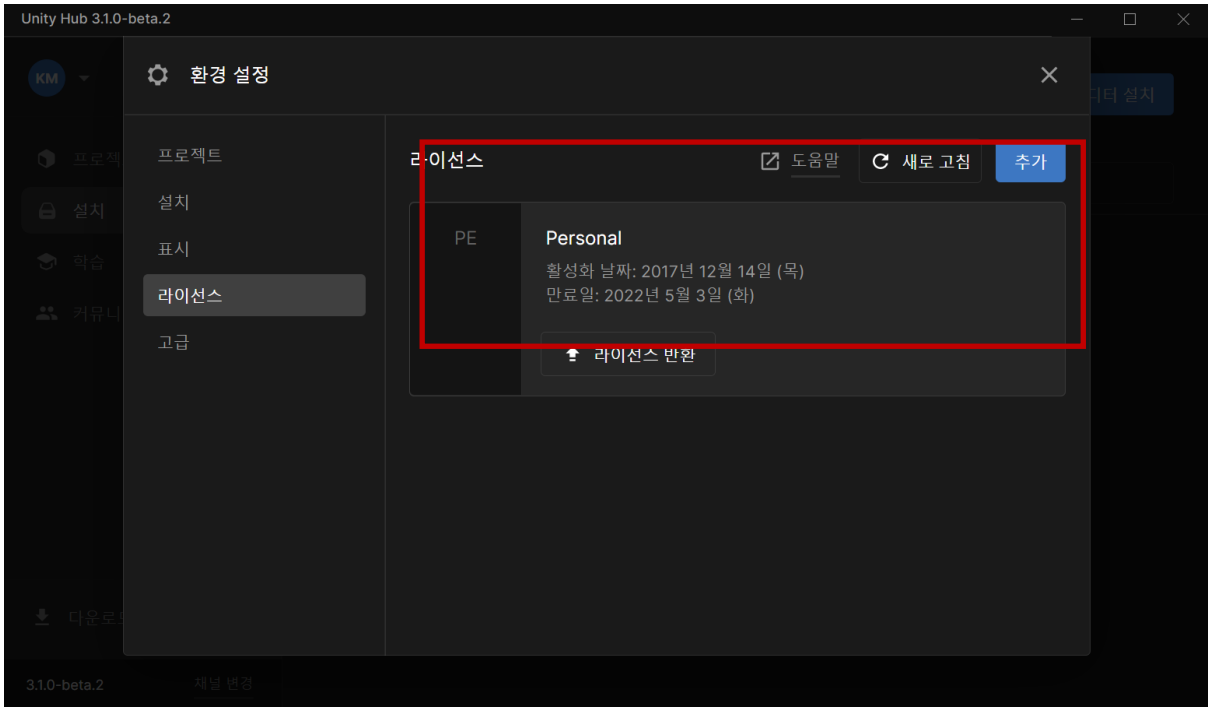


그림 8. 활성화 된 Personal 라이선스

1.3 유니티 에디터 설치

이제 유니티를 설치하겠습니다. 환경 설정 창을 닫고 유니티 허브 화면으로 돌아온 뒤 왼쪽 메뉴에서 [설치]를 클릭합니다.

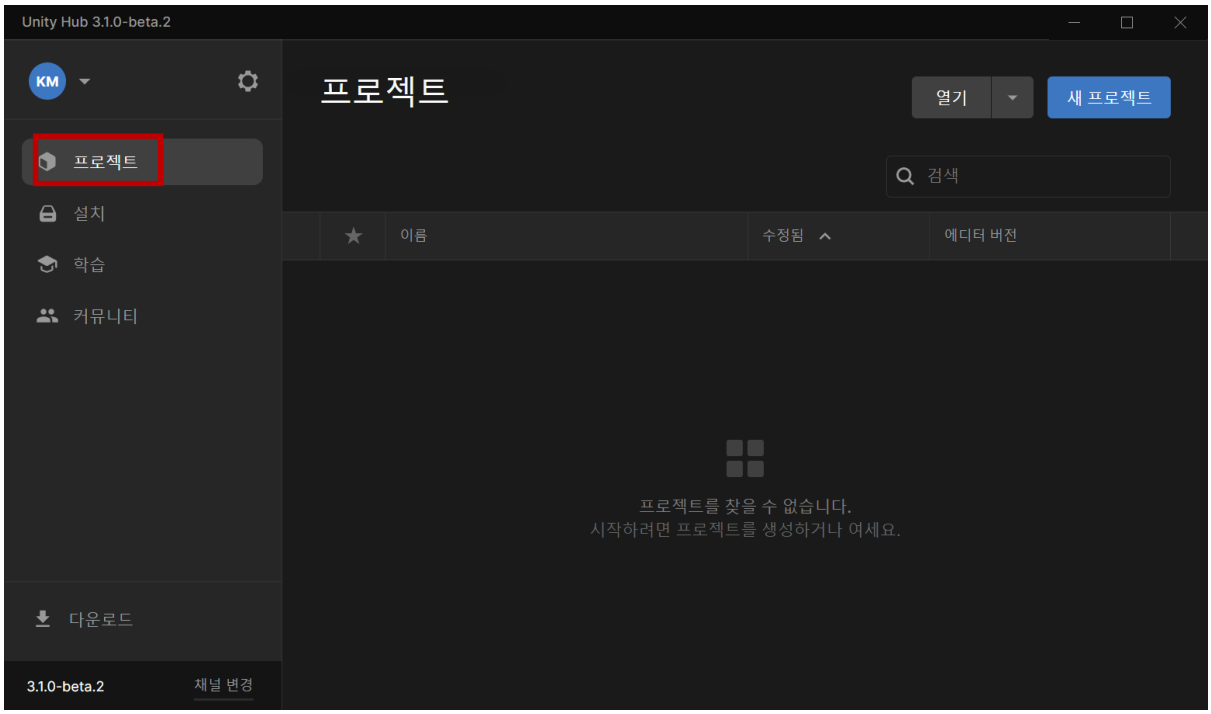


그림 9. [설치] 메뉴 클릭

화면 가운데 또는 오른쪽 상단에 있는 [에디터 설치] 버튼을 클릭합니다.

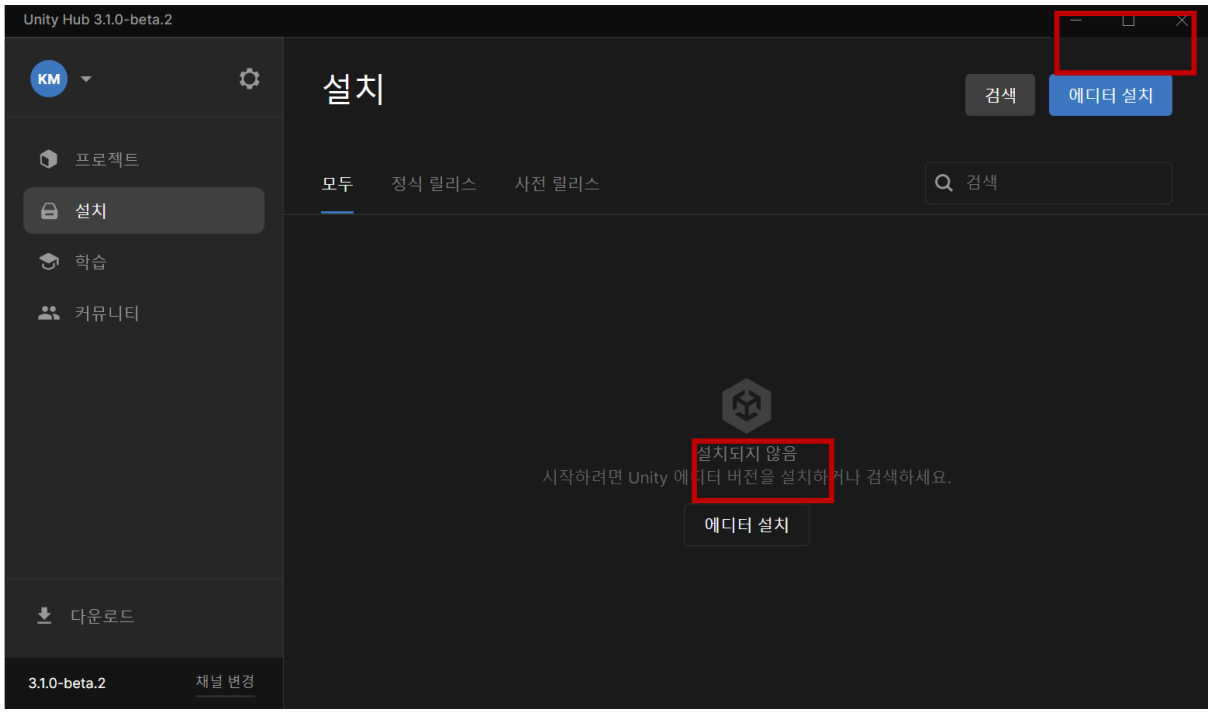


그림 10. [에디터 설치] 버튼 클릭

이 책에서는 유니티 에디터 2021.1.18 버전이나 2022.3.4 버전을 사용합니다. 만약 정식 릴리즈 버전의 유니티 에디터를 사용하는 경우 **[정식 릴리스]** 탭에서 원하는 버전 오른쪽에 있는 **[설치]** 버튼을 클릭합니다.

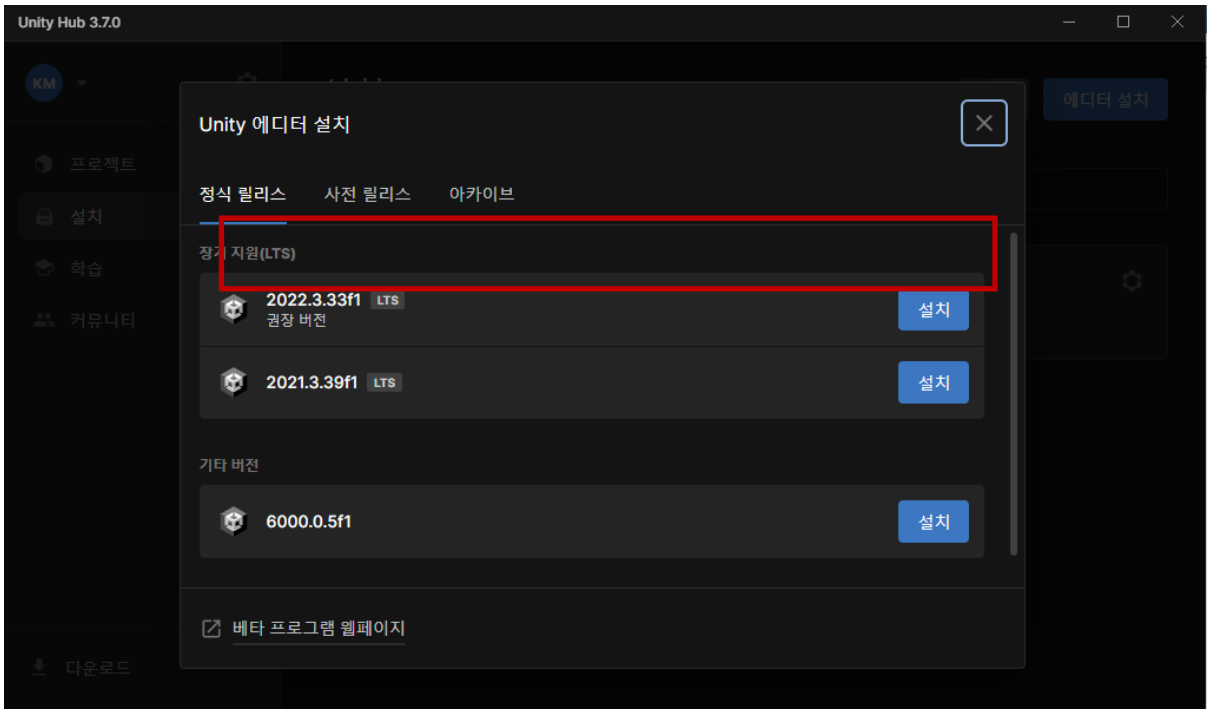


그림 11. [정식 릴리스] 버전 설치

특정 버전의 유니티 에디터를 내려받고 싶다면? 유니티 허브 메뉴에서 [설치]를 클릭한 다음 [에디터 설치] 버튼을 클릭해 유니티 에디터 설치 창을 엽니다. [아카이브] 탭에서 [아카이브 다운로드] 링크를 클릭해 유니티 다운로드 아카이브 페이지로 이동합니다.

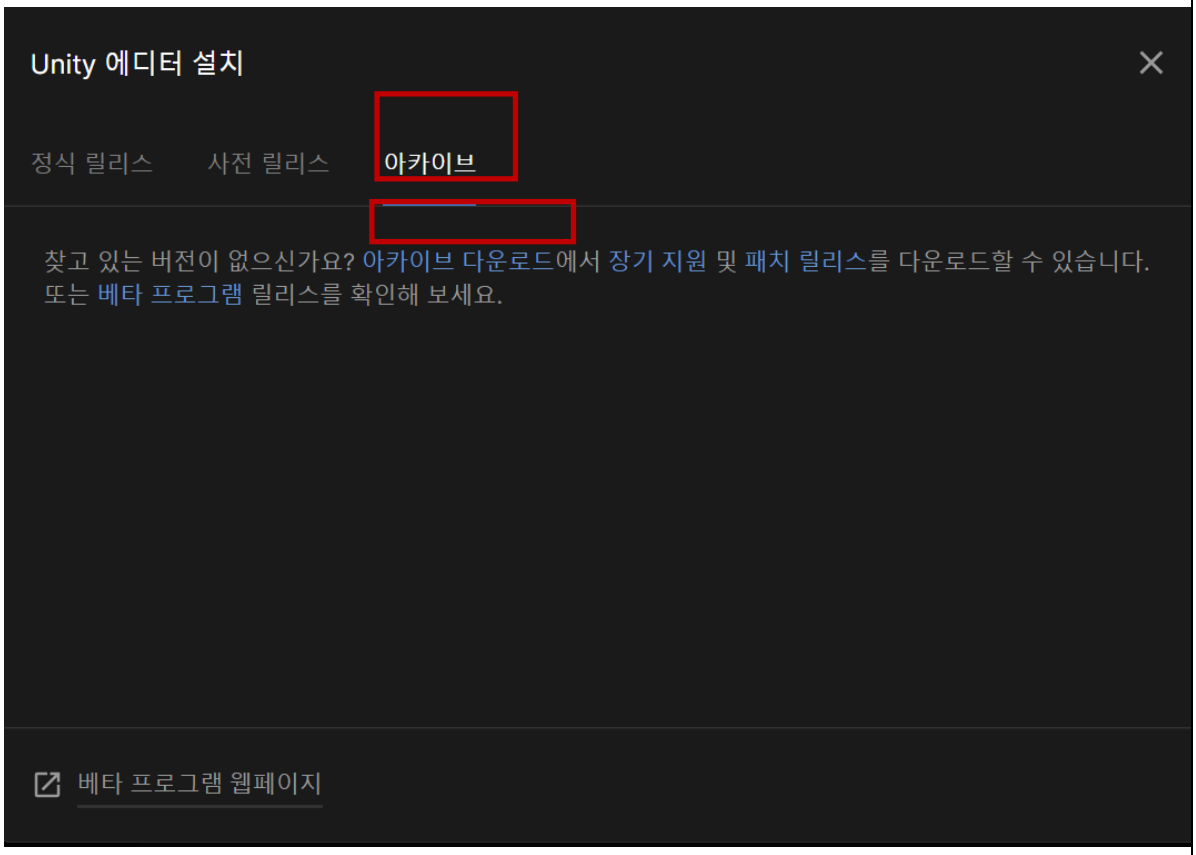


그림 12. [아카이브 다운로드] 링크 클릭

유니티 다운로드 아카이브 페이지에서는 그동안 유니티에서 제공한 모든 버전의 유니티 에디터를 내려받을 수 있습니다. 원하는 버전 오른쪽에 있는 [Unity Hub] 버튼을 클릭합니다.

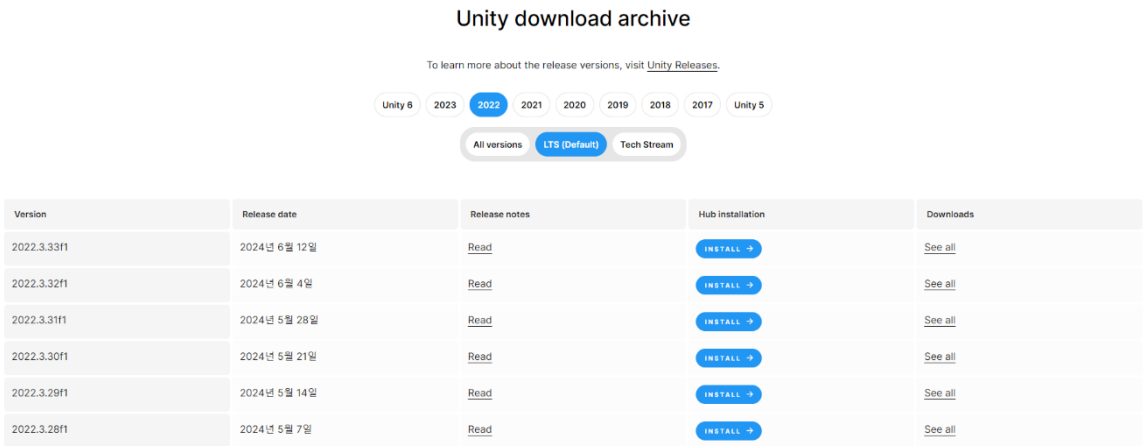


그림 13. 유니티 다운로드 아카이브 페이지

다음 단계에서는 유니티에 다양한 모듈을 추가할 수 있습니다. 특히 플랫폼 항목에서는 제작한 환경을 빌드할 플랫폼을 선택할 수 있습니다. 윈도우, macOS, 리눅스에서 환경을 빌드할 수 있도록 다음과 같이 세 가지 플랫폼을 선택합니다. 플랫폼을 모두 선택했으면 오른쪽 아래에 있는 **[설치]** 버튼을 클릭합니다.

- Windows Build Support (IL2CPP)
  - Mac Build Support (Mono)
- Linux Build Support (Mono)

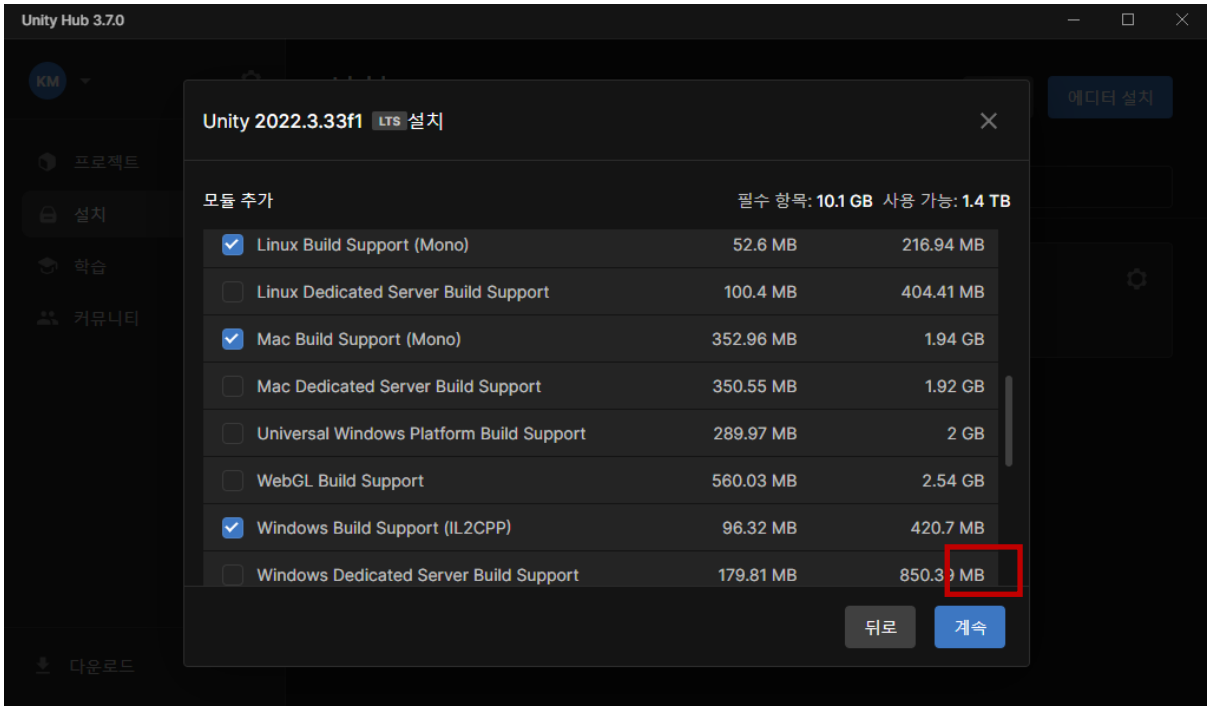


그림 14. 유니티 에디터 설치를 위한 모듈 추가

다음과 같이 유니티 에디터의 설치가 진행되는 모습을 확인할 수 있습니다. 설치가 진행되는 동안 잠시 기다립니다.

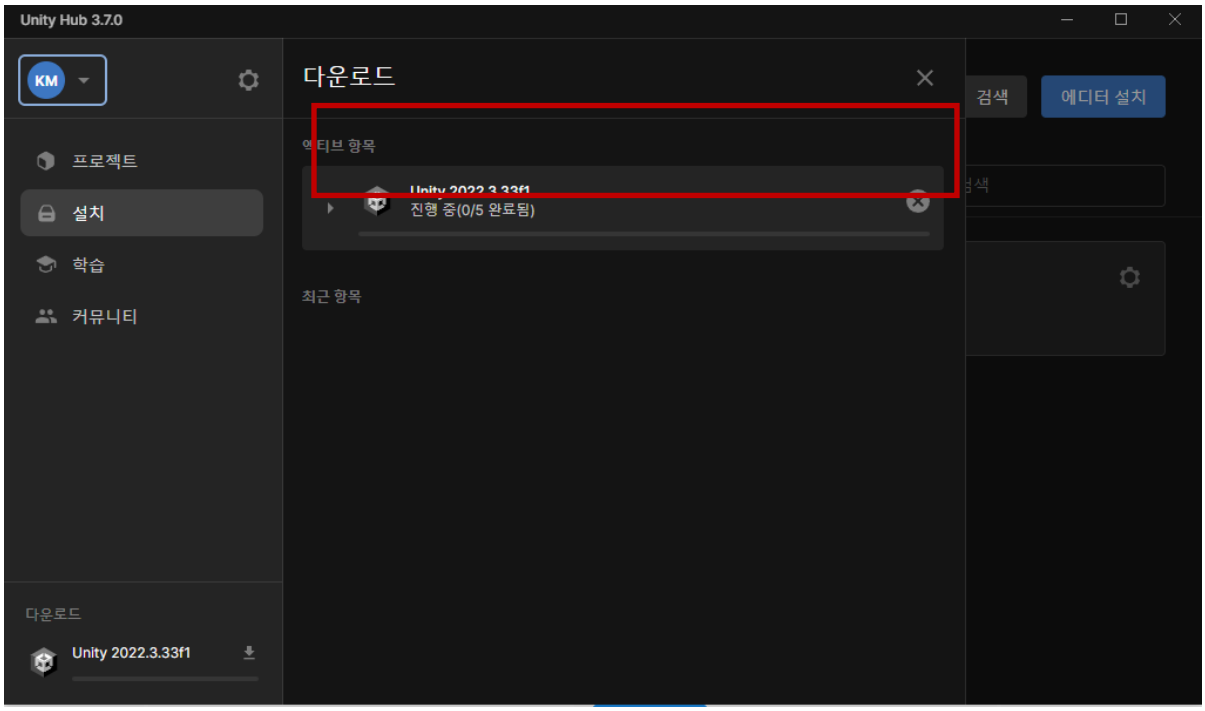


그림 15. 유니티 에디터 설치 진행

2. ML-Agents 설치

이번 절에서는 ML-Agents(유니티 머신러닝 에이전트, Unity Machine Learning Agents)의 설치 방법을 알아보겠습니다. ML-Agents 의 경우 유니티와 파이썬에 따로 설치를 진행해야 합니다. 유니티에서는 유니티 패키지를 이용해 설치를 진행하여 유니티로 만든 환경에 다양한 강화학습과 관련된 설정을 수행할 수 있습니다. 파이썬에서는 라이브러리로 설치를 진행하여 ML-Agents 에서 제공하는 강화학습 알고리즘들을 사용할 수 있으며, 파이썬 코드를 통해 유니티 환경과 통신할 수 있도록 설정합니다.

2.1 ML-Agents 파일 내려받기

먼저 유니티에 ML-Agents 를 설치하겠습니다. 먼저 ML-Agents 의 깃허브에서 필요한 파일들을 내려받습니다. ML-Agents 의 깃허브 주소는 다음과 같습니다.

깃허브 링크: https://github.com/Unity-Technologies/ml-agents

해당 페이지에서 아래쪽으로 스크롤을 내려 보면 지금까지 배포된 다양한 버전의 ML-Agents 를 살펴볼 수 있습니다. 이 책에서는 유니티 패키지 2.3.0 버전에 해당하는 Release 20 을 사용할 것입니다. 만약 최신 버전인 Release 21 을 사용하는 경우 깃허브 페이지에서 바로 Download 를 클릭하여 다운받습니다.

Version	Release Date	Source	Documentation	Download	Python Package	Unity Package
develop (unstable)	--	source	docs	download	--	--
Release 21	October 9, 2023	source	docs	download	1.0.0	3.0.0

그림 16. 깃허브 페이지 내부의 Releases 페이지

현재 페이지에서 Release 20 은 찾아볼 수 없기 때문에 다른 방법으로 내려받도록 하겠습니다. 만약 Release 20 처럼 깃허브 페이지 하단에 원하는 버전이 없는 경우 깃허브 페이지 오른쪽에 있는 **Releases** 페이지로 이동해서 원하는 버전의 ML-Agents 를 찾아줍니다.

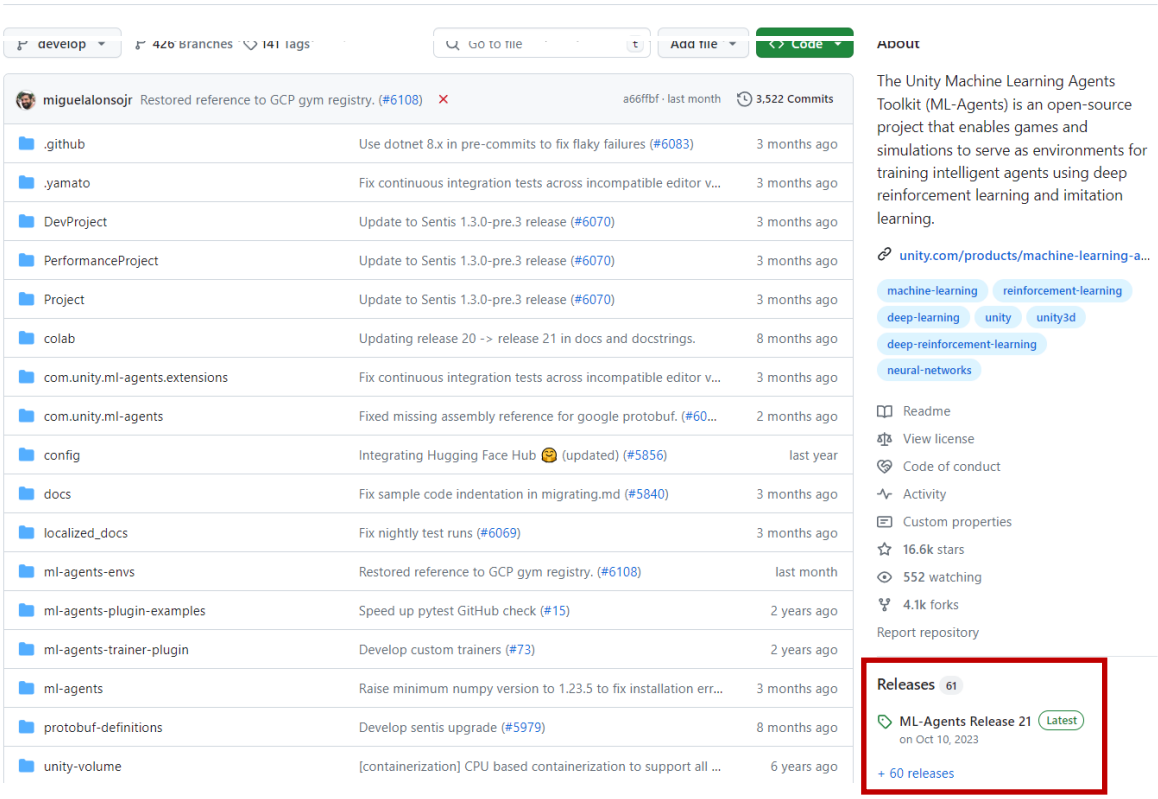
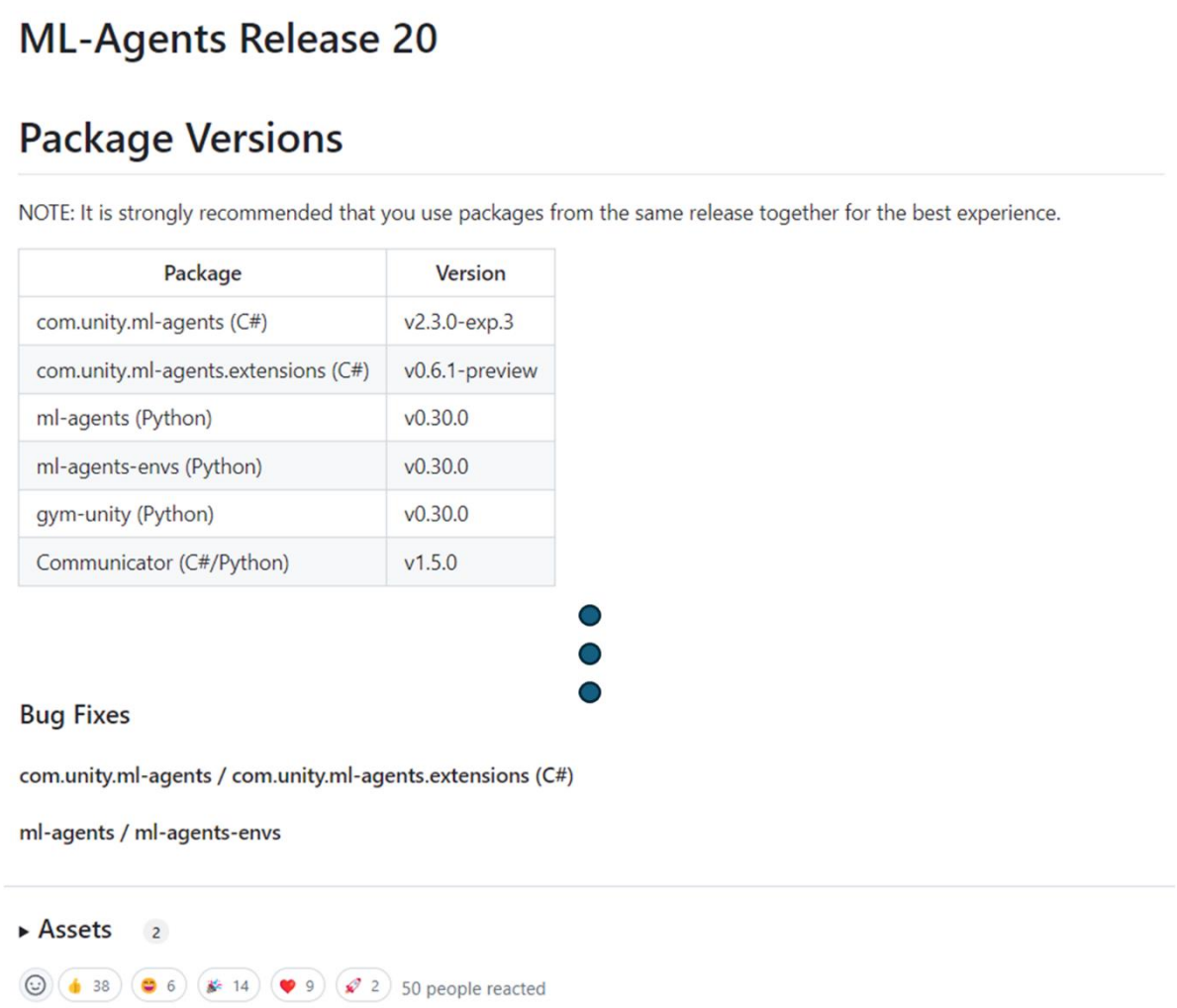


그림 17. 깃허브 페이지 내부의 Releases 페이지

원하는 버전의 ML-Agents 를 찾은 후 하단의 **Source code** 를 클릭해 압축 파일을 내려받습니다.



ML-Agents Release 20

Package Versions

NOTE: It is strongly recommended that you use packages from the same release together for the best experience.

Package	Version
com.unity.ml-agents (C#)	v2.3.0-exp.3
com.unity.ml-agents.extensions (C#)	v0.6.1-preview
ml-agents (Python)	v0.30.0
ml-agents-envs (Python)	v0.30.0
gym-unity (Python)	v0.30.0
Communicator (C#/Python)	v1.5.0

Bug Fixes

com.unity.ml-agents / com.unity.ml-agents.extensions (C#)

ml-agents / ml-agents-envs

▸ Assets 2

38 6 14 9 2 50 people reacted

그림 18. Release 페이지에서 원하는 버전의 압축 파일 내려받기

압축을 해제한 파일의 구성 요소들과 이 책에서 주로 다룰 요소는 다음과 같습니다. 해당 요소에 대해서는 앞으로 하나씩 사용해보면서 자세히 살펴보겠습니다.

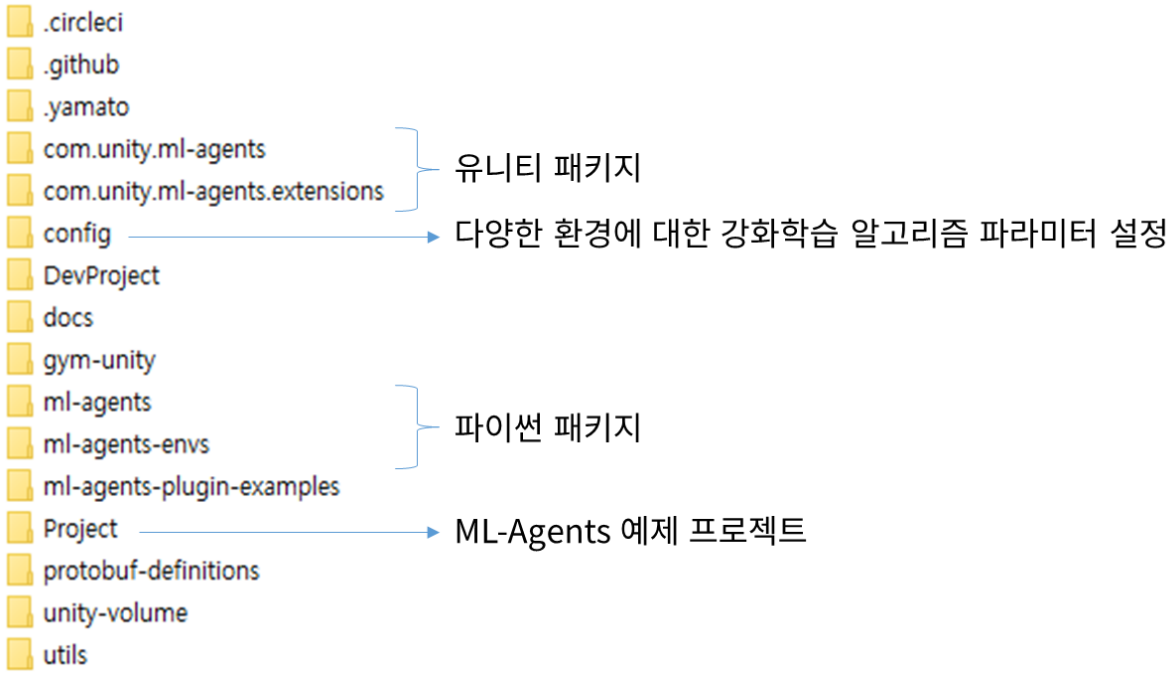


그림 19. ML-Agents 압축 파일의 구성 요소

이제 본격적으로 ML-Agents 의 설치를 진행해보겠습니다. ML-Agents 는 유니티와 파이썬 각각 설치를 진행해야 합니다. 유니티와 파이썬에 어떻게 ML-Agents 설치하는지 하나씩 살펴보겠습니다.

2.2 유니티에 ML-Agents 설치하기

ML-Agents 를 유니티에 설치하는 방법은 총 3 가지가 있습니다.

1. 유니티 패키지에서 기본적으로 제공하는 ML-Agents 패키지 설치
2. 직접 ML-Agents 패키지 설치
3. ML-Agents 의 예제 프로젝트 이용

이후 각 방법에 대해 자세히 설명하겠지만 일반적으로는 2, 3 번 방법이 가장 안정적으로 설치되는 편이므로 해당 방법으로 설치하는 것을 권장합니다.

설치 방법 1. 유니티 패키지에서 기본적으로 제공하는 ML-Agents 패키지 설치

먼저 유니티 프로젝트의 상단 메뉴에서 **[Window]** → **[Package Manager]**를 클릭합니다.

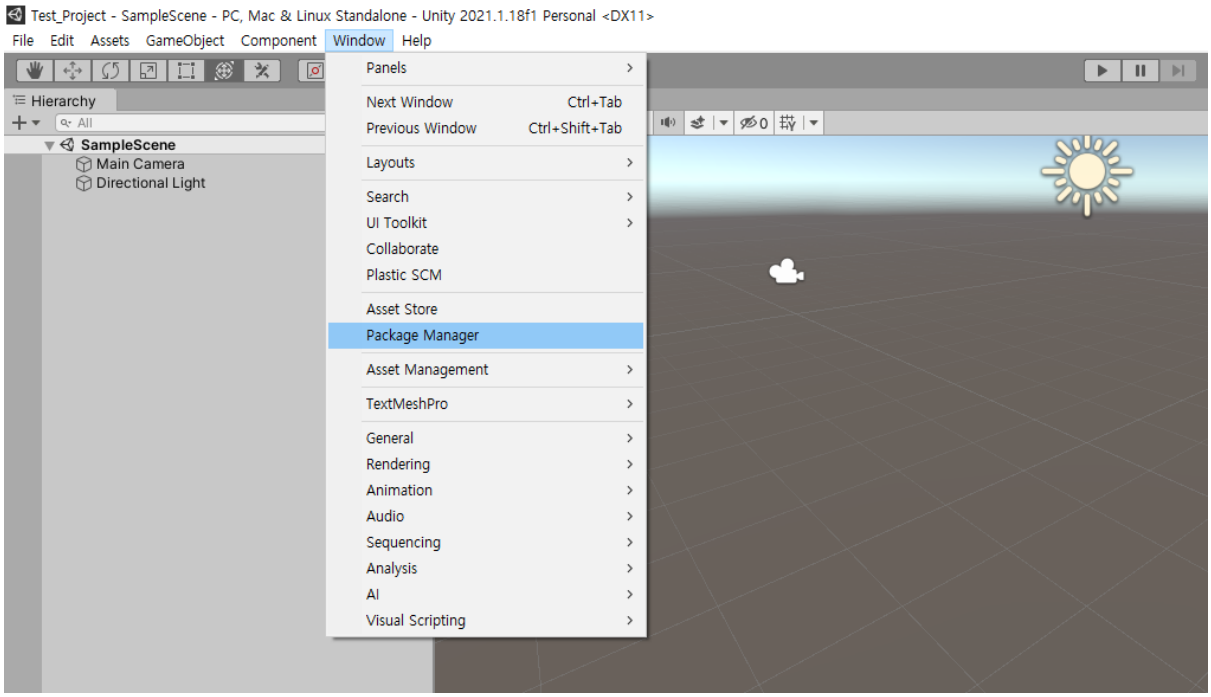


그림 20. [Window] → [Package Manager] 클릭

Package Manager 창이 나오면 상단에서 **[Packages]** → **[Unity Registry]**를 선택합니다.

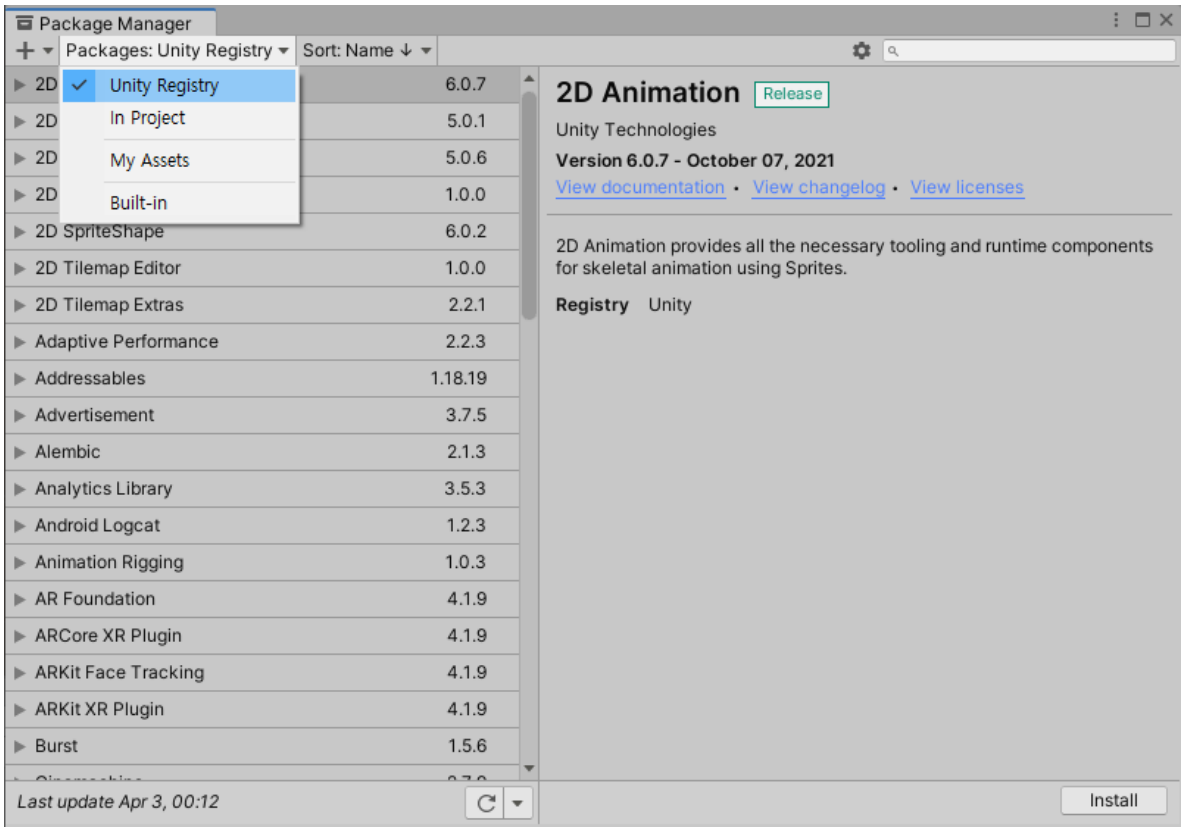


그림 21. [Packages] → [Unity Registry] 선택

이렇게 진행하면 유니티에 등록된 패키지들의 리스트를 확인할 수 있습니다. 이 리스트 중에서 ML Agents 를 찾고 오른쪽 아래에 있는 **[Install]** 버튼을 클릭하면 ML-Agents 패키지를 설치할 수 있습니다.

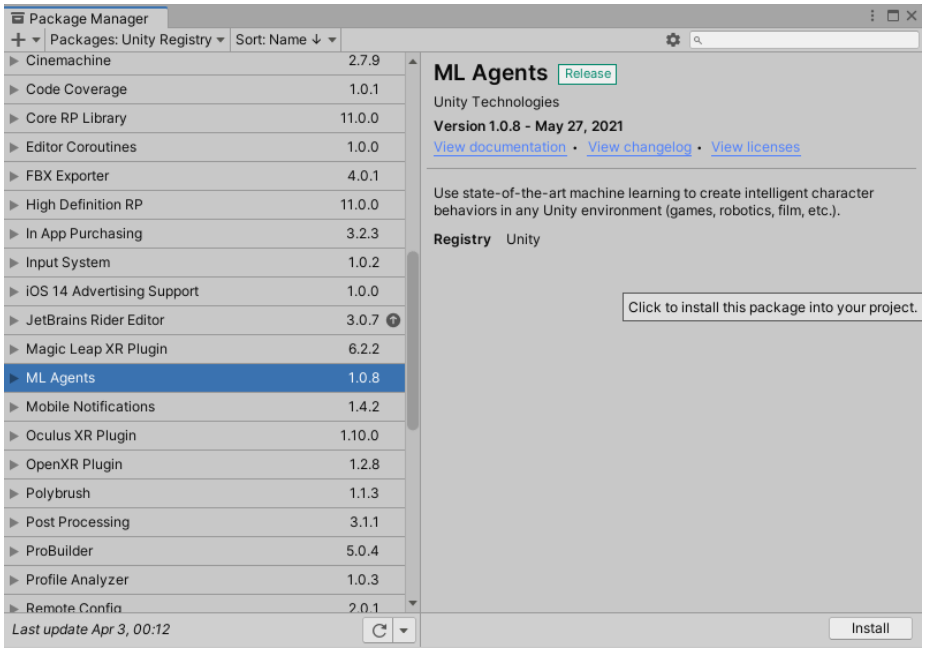


그림 22. ML Agents 패키지 설치

패키지 설치가 완료되면 프로젝트 창에서 Packages 에 ML Agents 가 포함된 것을 확인할 수 있습니다.

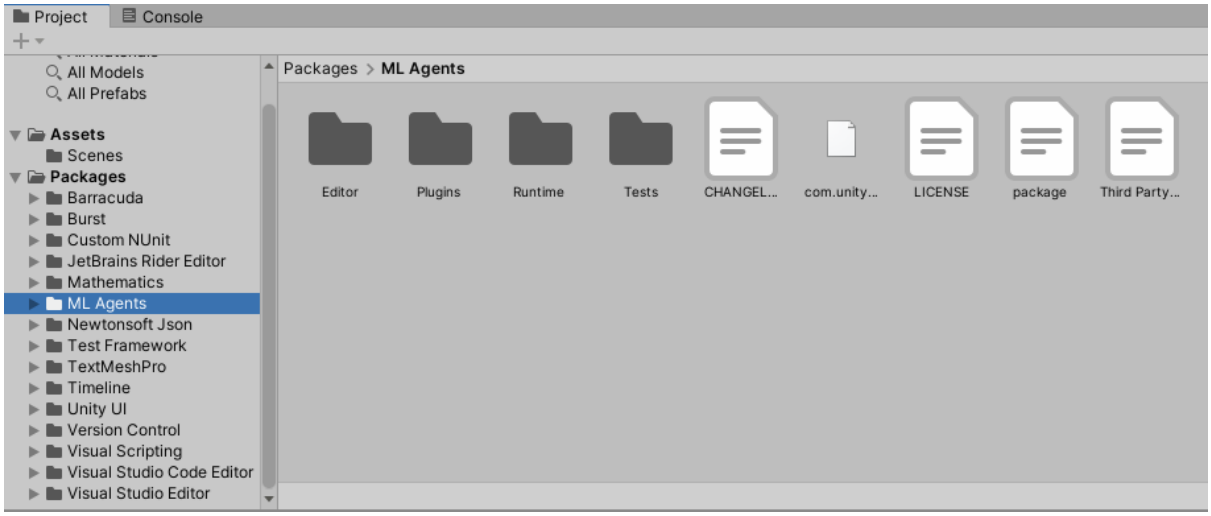


그림 23. 프로젝트 창에 ML Agents 패키지 포함

ML-Agents 깃허브에서 내려받은 요소들 중 Project 폴더는 ML-Agents 예제 프로젝트를 포함하는 폴더입니다. 해당 예제 프로젝트의 일부 내용을 이용할 예정이므로 해당 프로젝트의 내용을 옮기는 방법을 살펴보겠습니다.

깃허브에서 받은 파일에서 Project/Assets/ML-Agents 폴더를 현재 유니티 프로젝트의 프로젝트 뷰로 드래그 앤드 드롭합니다.

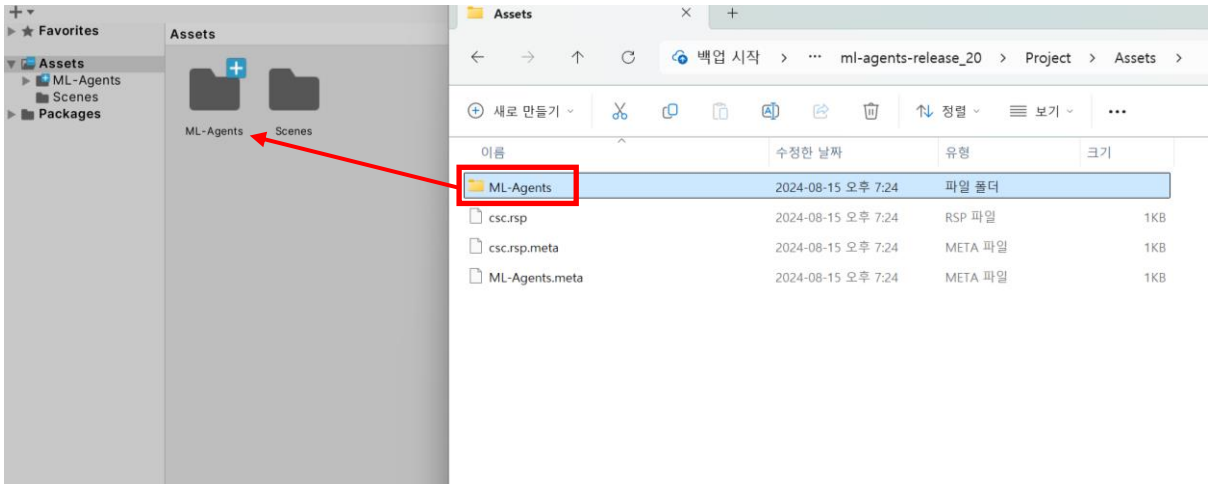


그림 24. 유니티 프로젝트로 ML-Agents 예제 프로젝트 옮기기

이 경우 콘솔을 확인하면 다음과 같이 많은 에러들이 발생하는 것을 확인할 수 있습니다. 첫 번째 방법으로 ML-Agents 를 설치하는 경우 고정된 버전의 ML-Agents 가 설치되므로 예제 프로젝트의 버전에 따라 서로 호환이 되지 않아 에러가 발생하는 경우가 많습니다.

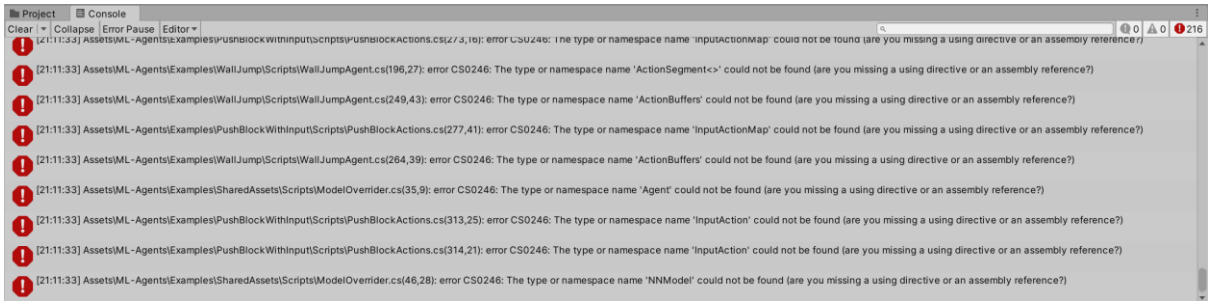


그림 25. 콘솔에서 발생하는 오류 확인

지금까지 유니티에 ML-Agents 를 설치하는 첫 번째 방법을 알아봤습니다. 이 방법은 설치 방법은 매우 간단하지만 원하는 버전의 ML-Agents 를 설치할 수

없다는 점과, 기본 유니티 프로젝트에 ML-Agents 만 설치하는 방식이기 때문에 버전에 따라서는 에러들이 발생할 수 있다는 단점이 있습니다.

설치 방법 2. 직접 ML-Agents 패키지 설치

두 번째 방법은 깃허브에서 내려받은 유니티 패키지를 직접 설치하는 방법입니다. 우선 설치 방법 1 과 동일하게 상단 메뉴에서 **[Window]** → **[Package Manager]**를 클릭합니다. Package Manager 창이 나오면 왼쪽 상단의 **[+]** 버튼을 클릭하고 **[Add package from disk]**를 클릭합니다. 이 기능을 이용해 로컬 컴퓨터에 있는 유니티 패키지를 현재 프로젝트에 추가할 수 있습니다.

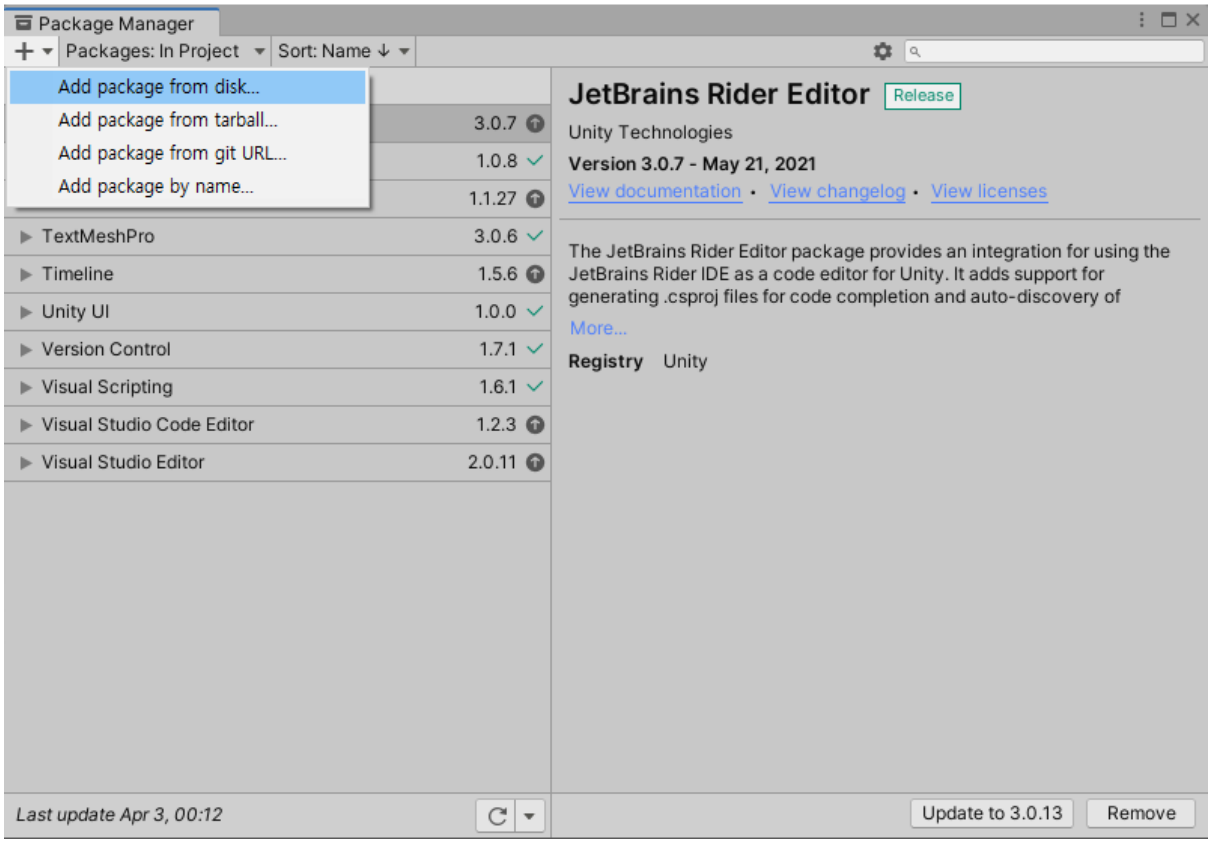


그림 26. Package Manager 창 왼쪽 상단의 [+] 버튼을 클릭하고 [Add package from disk] 클릭

파일 선택 창이 나오면 깃허브에서 내려받은 항목들 중 com.unity.ml-agents 폴더 내부로 이동하고 package.json 파일을 선택해 ML-Agents 패키지를 직접 추가합니다. 같은 방법으로 com.unity.ml-agents.extensions 내부의 package.json 을 선택하면 ML-Agents 패키지 추가가 마무리됩니다.

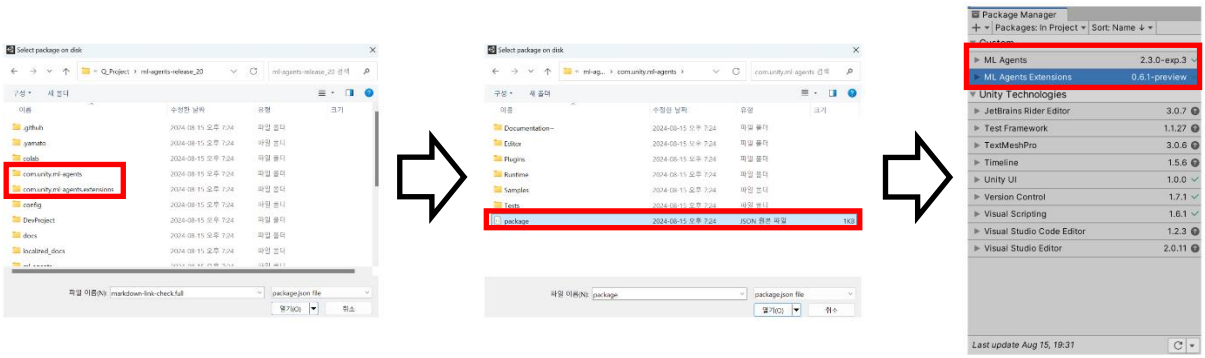


그림 27. ML-Agents 패키지 추가 과정

ML-Agents 의 예제 프로젝트를 옮기는 과정은 설치 방법 1 과 동일합니다. 깃허브에서 내려받은 파일 중에서 Project/Assets/ML-Agents 폴더를 현재 유니티 프로젝트의 프로젝트 뷰로 드래그 앤드 드롭합니다. 이 경우에도 PushBlockWithInput 환경에서 에러가 발생하는 경우가 있습니다. 에러가 발생한다면 프로젝트 뷰에서 Assets/ML-Agents/Examples/PushBlockWithInput 폴더를 삭제합니다.

지금까지 유니티에 ML-Agents 를 설치하는 두 번째 방법을 알아봤습니다. 두 번째 방법을 이용해 설치하면 원하는 버전의 ML-Agents 를 설치할 수 있지만 초기 유니티 프로젝트에 ML-Agents 패키지만 추가하는 방식이기 때문에 버전에 따라서는 사소한 에러가 발생한다는 단점이 있습니다.

설치 방법 3. ML-Agents 의 예제 프로젝트 이용

세 번째 설치 방법은 깃허브에서 받은 항목 중에서 ML-Agents 예시 프로젝트인 Project 폴더를 직접 이용하는 방법입니다. 깃허브에서 내려받은 항목 중 Project 폴더를 복사하고 원하는 경로에 해당 폴더를 붙여 넣습니다.

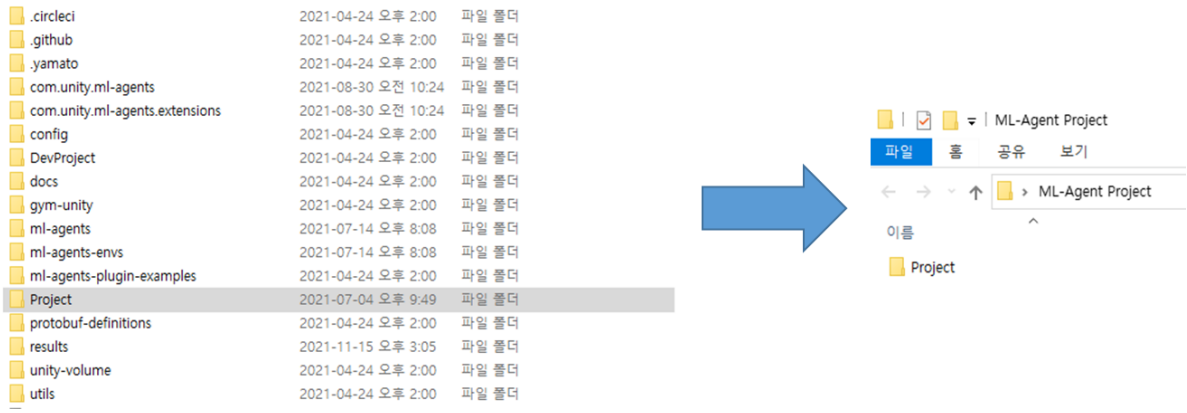


그림 28. 깃허브 항목 중 Project 폴더를 복사한 다음 원하는 경로에 붙여넣기

유니티 허브를 열고 **[프로젝트]** 메뉴를 선택한 다음 오른쪽 상단에 있는 **[열기]** 버튼을 클릭합니다. 폴더 선택 창이 나오면 Project 폴더를 붙여 넣은 경로로 이동한 다음 Project 폴더를 선택하고 **[열기]**를 클릭합니다.

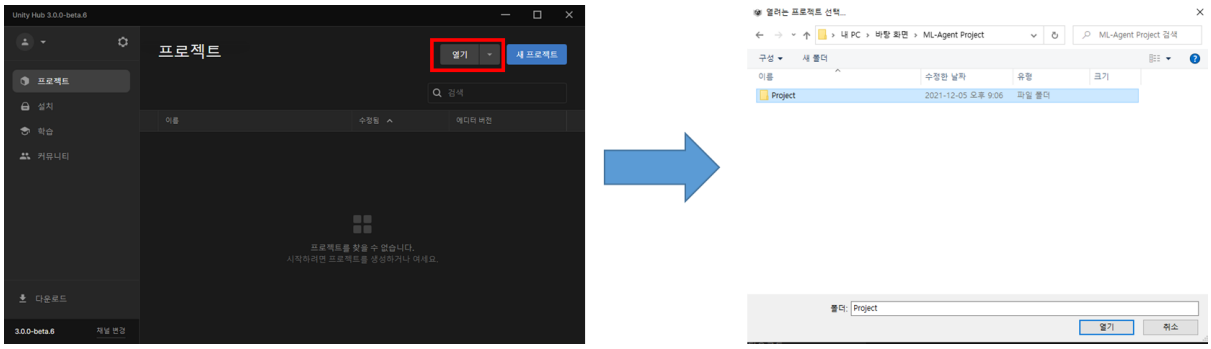


그림 29. 유니티 허브에서 Project 폴더 열기

이 경우 다음과 같은 메시지가 나올 수 있습니다. 이는 해당 유니티 프로젝트 (Project 폴더)가 현재 설치된 유니티 에디터 버전과 다르기 때문에 나오는 것입니다.

그림 30. 유니티 허브에서 Project 폴더를 열 때 발생할 수 있는 메시지

이 경우 하위의 **[다른 에디터 버전 선택]**을 클릭하고 현재 설치된 버전의 유니티 에디터를 선택해 열어줍니다.

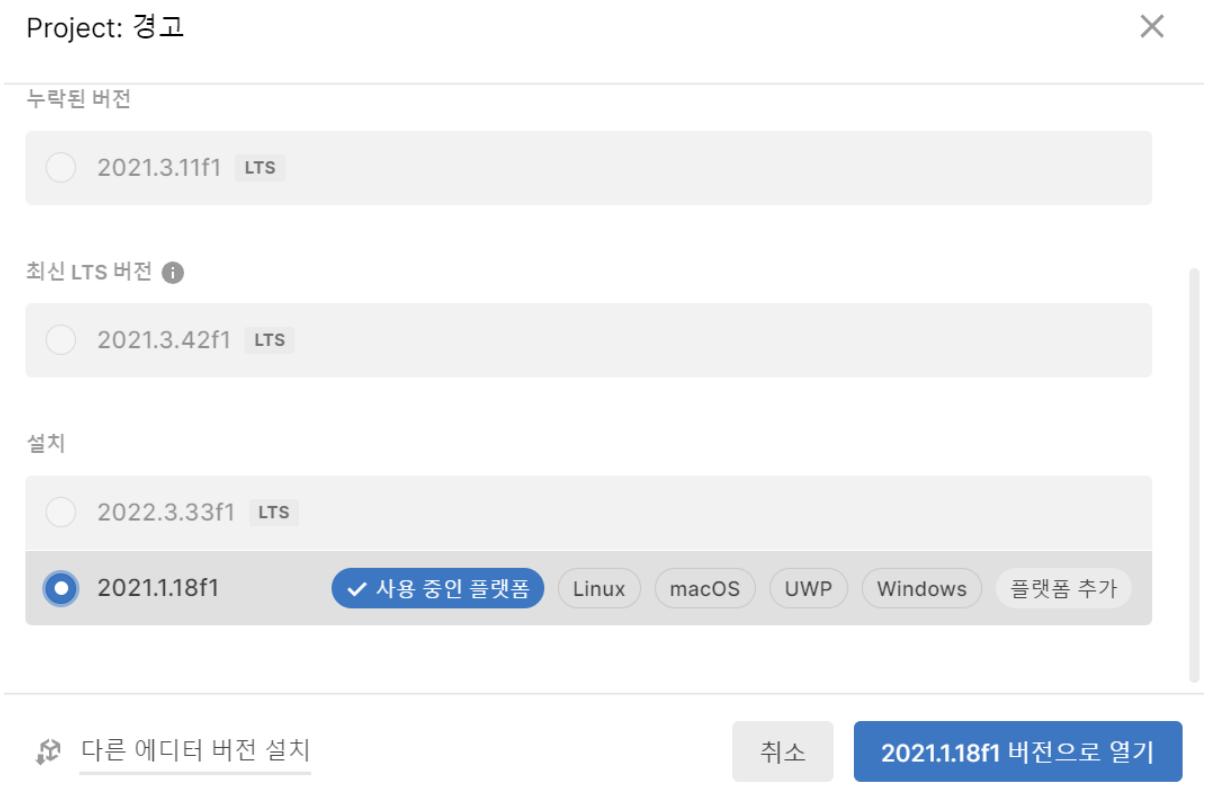


그림 31. 현재 설치된 버전의 유니티 에디터로 프로젝트 열기

프로젝트가 열리는 중에 오류가 발생한다는 경고가 나오는데 이는 현재 프로젝트에 설정된 ML-Agents 관련 패키지의 경로가 바뀐 관계로 발생하는 오류입니다. ML-Agents 관련 패키지의 경로는 재설정할 예정이므로 해당 오류는 무시할 것입니다. **[continue]**와 **[Ignore]**를 클릭하며 진행합니다. 프로젝트가 실행되고 나면 콘솔창에 다음과 같이 많은 오류가 발생하는 것을 확인할 수 있습니다.

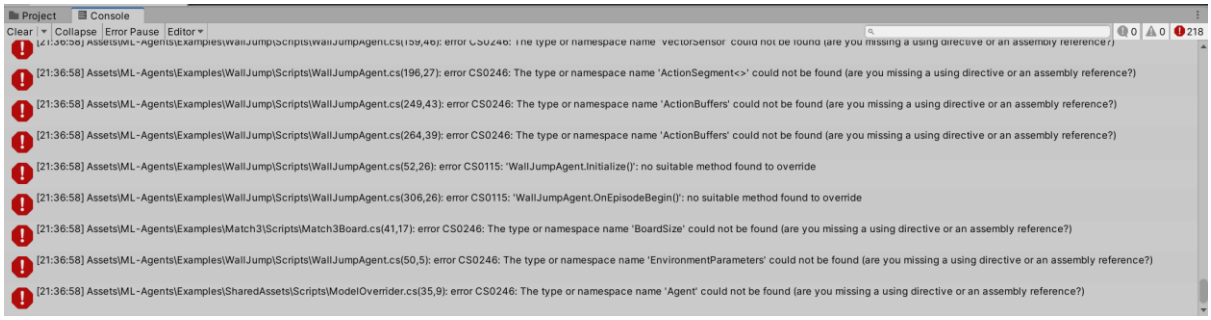


그림 32. 프로젝트 실행 시 콘솔창에 발생하는 에러 로그

설치 방법 1, 2 와 동일하게 상단 메뉴에서 **[Window]** → **[Package Manager]**를 클릭합니다. Package Manager 창이 나오면 현재 프로젝트의 패키지 (In

Project)에서 ML-Agents, ML-Agents Extensions 의 경로를 찾지 못하는 것을 확인할 수 있습니다.

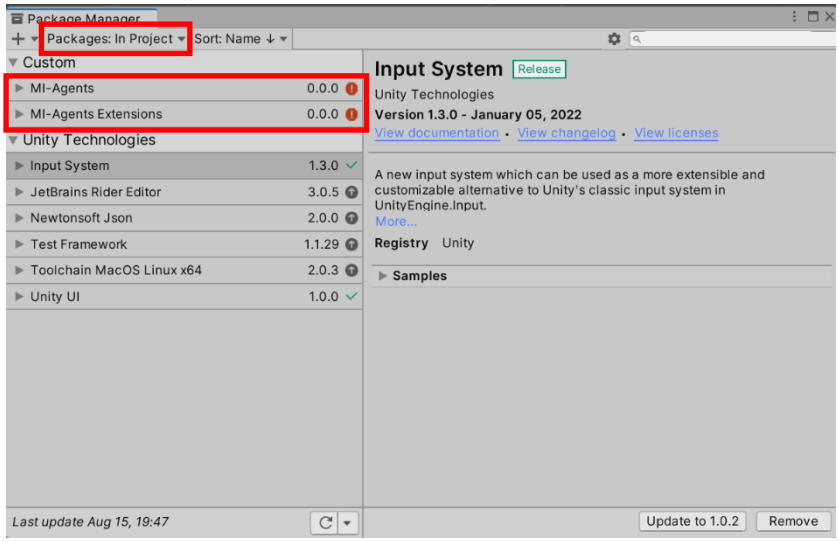


그림 33. Package Manager 에서 ML-Agents 관련 패키지 확인

이제 설치 방법 2 와 동일하게 좌측 상단의 **[+]** 버튼을 클릭하고 **[Add package from disk]**를 클릭합니다. 그리고 깃허브에서 내려받은 항목들 중 com.unity.ml-agents 폴더에 있는 package.json 파일을 선택합니다. 같은 방법으로 com.unity.ml-agents.extensions 폴더에 있는 package.json 을 선택하면 ML-Agents 와 관련된 패키지의 경로가 제대로 지정된 것을 확인할 수 있습니다.

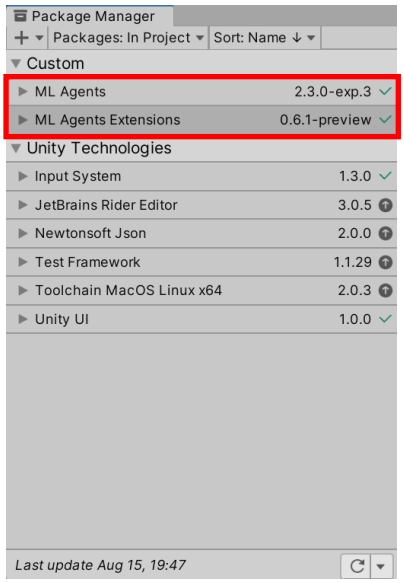


그림 34. ML-Agents 와 관련된 패키지 추가 완료

여기까지의 내용을 진행하면 콘솔의 모든 에러가 사라진 것을 확인할 수 있습니다. 설치 방법 3 을 통해 설치하는 경우 ML-Agents 에서 기본적으로 설정을 마친 예제이므로 ML-Agents 에 대한 패키지만 재설정하면 일반적으로 원하는 버전의 ML-Agents 를 에러 없이 사용할 수 있습니다. 지금까지 유니티에 ML-Agents 를 설치하는 세 가지 방법을 모두 살펴봤습니다.

2.3.3 ML-Agents 파이썬 패키지 설치하기

ML-Agents 파이썬 패키지의 설치는 `pip` 를 이용해 간단하게 진행할 수 있습니다. 먼저 유니티 패키지의 버전에 대응되는 파이썬 패키지의 버전을 확인해야 합니다. 이 책에서 설치한 유니티 패키지의 버전 2.3.0 에 대응되는 파이썬 패키지의 버전은 0.30.0 입니다. 따라서 다음의 명령어를 이용해 `pip` 로 mlagents 파이썬 패키지를 설치할 수 있습니다.

```
pip install mlagents==0.30.0
```

다음과 같이 "Successfully installed"라는 메시지가 나오면 잘 설치된 것입니다.

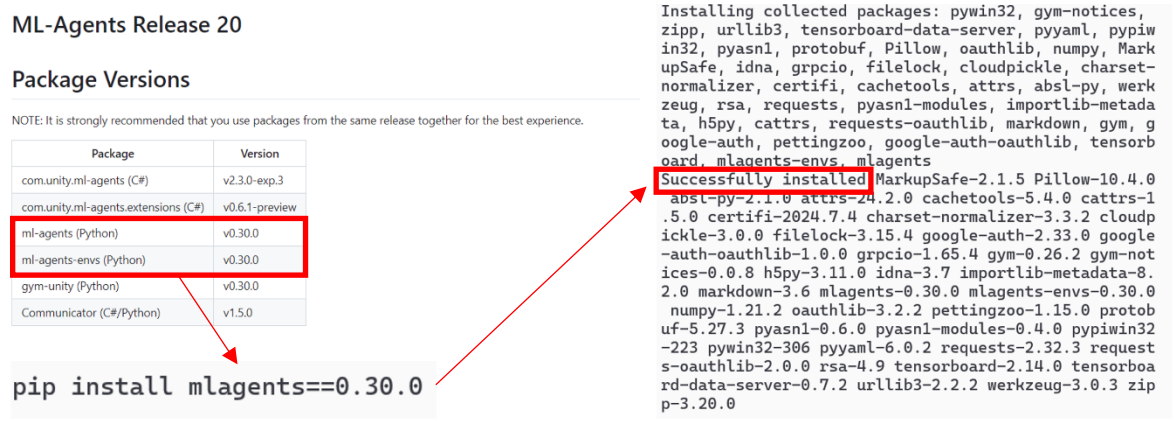


그림 35. pip 을 이용한 mlagents 파이썬 패키지 설치

ML-Agents 관련 파이썬 패키지가 잘 설치됐는지는 다음의 명령어로 확인할 수 있습니다.

```
pip show mlagents
pip show mlagents-envs
```

다음과 같이 명령어를 입력했을 때 이름과 버전이 나오면 잘 설치된 것입니다.

```
(mlagent) C:\Users\kyush\Desktop\Q_Project\ml-agents-release_20>pip show mlagents
Name: mlagents
Version: 0.30.0
Summary: Unity Machine Learning Agents
Home-page: https://github.com/Unity-Technologies/ml-agents
Author: Unity Technologies
Author-email: ML-Agents@unity3d.com
License:
Location: c:\users\kyush\anaconda3\envs\mlagent\lib\site-packages
Requires: attrs, cattrs, grpcio, h5py, mlagents-envs, numpy, Pillow, protobuf, pypiwin32, pyyaml, tensorboard
Required-by:

(mlagent) C:\Users\kyush\Desktop\Q_Project\ml-agents-release_20>pip show mlagents-envs
Name: mlagents_envs
Version: 0.30.0
Summary: Unity Machine Learning Agents Interface
Home-page: https://github.com/Unity-Technologies/ml-agents
Author: Unity Technologies
Author-email: ML-Agents@unity3d.com
License:
Location: c:\users\kyush\anaconda3\envs\mlagent\lib\site-packages
Requires: cloudpickle, filelock, grpcio, gym, numpy, pettingzoo, Pillow, protobuf, pyyaml
Required-by: mlagents
```

그림 36. mlagents 파이썬 패키지가 잘 설치됐는지 확인

이렇게 설치하는 경우 파이토치에 대한 이전 버전 설치 페이지로 이동 후 다음과 같이 다양한 파이토치 설치 명령어 중 적절한 버전에 대한 명령어를 확인하고 설치를 진행합니다.

- 파이토치 이전 버전 설치 페이지: https://pytorch.org/get-started/previous-versions/

v2.0.0

Conda

OSX

```
# conda
conda install pytorch==2.0.0 torchvision==0.15.0 torchaudio==2.0.0 -c pytorch
```

Linux and Windows

```
# CUDA 11.7
conda install pytorch==2.0.0 torchvision==0.15.0 torchaudio==2.0.0 pytorch-cuda=11.7 -c pytorch -c nvidia
# CUDA 11.8
conda install pytorch==2.0.0 torchvision==0.15.0 torchaudio==2.0.0 pytorch-cuda=11.8 -c pytorch -c nvidia
# CPU Only
conda install pytorch==2.0.0 torchvision==0.15.0 torchaudio==2.0.0 cpuonly -c pytorch
```

그림 37. 파이토치 이전 버전 설치 페이지와 명령어 예시

지금까지 mlagents 파이썬 패키지를 설치하고, 잘 설치됐는지 확인하는 방법을 살펴봤습니다. 이제 설치는 모두 완료됐으니 본격적으로 ML-Agents 와 관련된 요소들을 살펴보고 직접 사용해보겠습니다.