

3D콘텐츠 이론 및 활용

유길상

2주. 유니티 인터페이스

1차시 : 패널과 화면구성 및 조작

2차시 : 오브젝트 조작

1차시 학습 목표

- 유니티를 설치하고 기본 화면구성에 대하여 이해한다.
- 유니티 인터페이스의 구성과 배치를 이해한다.
- 씬 뷰의 특징을 이해하고 조작할 수 있다.
- 카메라 시점 전환을 이해하고 작동시킬 수 있다.
- 3D에 대한 기본적인 개념을 이해한다.
- 공간 좌표계에 대하여 이해한다.
- 객체를 구성하는 용어에 대하여 이해한다.

1차시 학습 내용

- 유니티 설치 및 기본 구성 살펴보기
- 프로젝트를 생성하고 열기
- 시점 조작하기
- 로컬공간과 월드공간의 차이점
- 프로젝션 모드
- Polygon, Edge, Vertex, Mesh
- 재질, 텍스처, 셰이더

생각해보기

1. 3D 가상 공간에서 사용되는 객체는 어떻게 구성되어 있을까?
2. 가상공간에서 힘과 중력은 어떻게 구현할까
3. 객체간의 충돌을 어떻게 감지할까?

1. 유니티 설치

1) 유니티 코리아 접속

<http://unity3d.com/kr/>



1. 유니티 설치

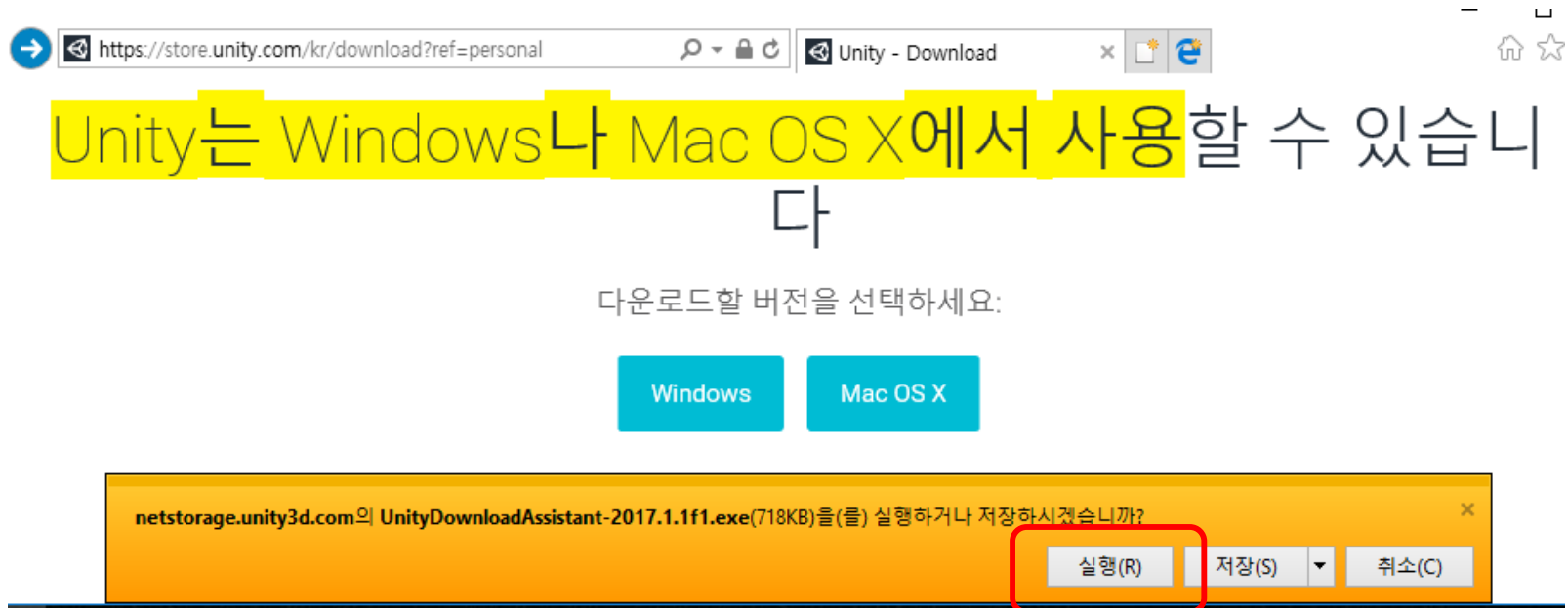
2) 유니티 종류 선택

The screenshot shows the Unity Store website with three pricing tiers: Personal, Plus, and Pro. The Personal tier is highlighted with a red box around the 'Try Personal' button.

Personal	Plus <small>Best Seller</small>	Pro
무료	₩40,250 매월	₩143,750 매월
입문자, 학생 등 Unity를 체험해 보고자 하시는 분들에게 추천합니다.	상상하는 것들을 현실로 구현해보고자 하시는 분들에게 추천합니다.	고급 기능과 커스터마이징 기능을 원하시는 분들에게 추천합니다.
Try Personal	Get Plus 자세히 알아보기	Pro 알아보기
자세히 알아보기	Now included: FREE Essentials Pack (\$150+ value)	Now included: FREE Essentials Pack (\$150+ value)

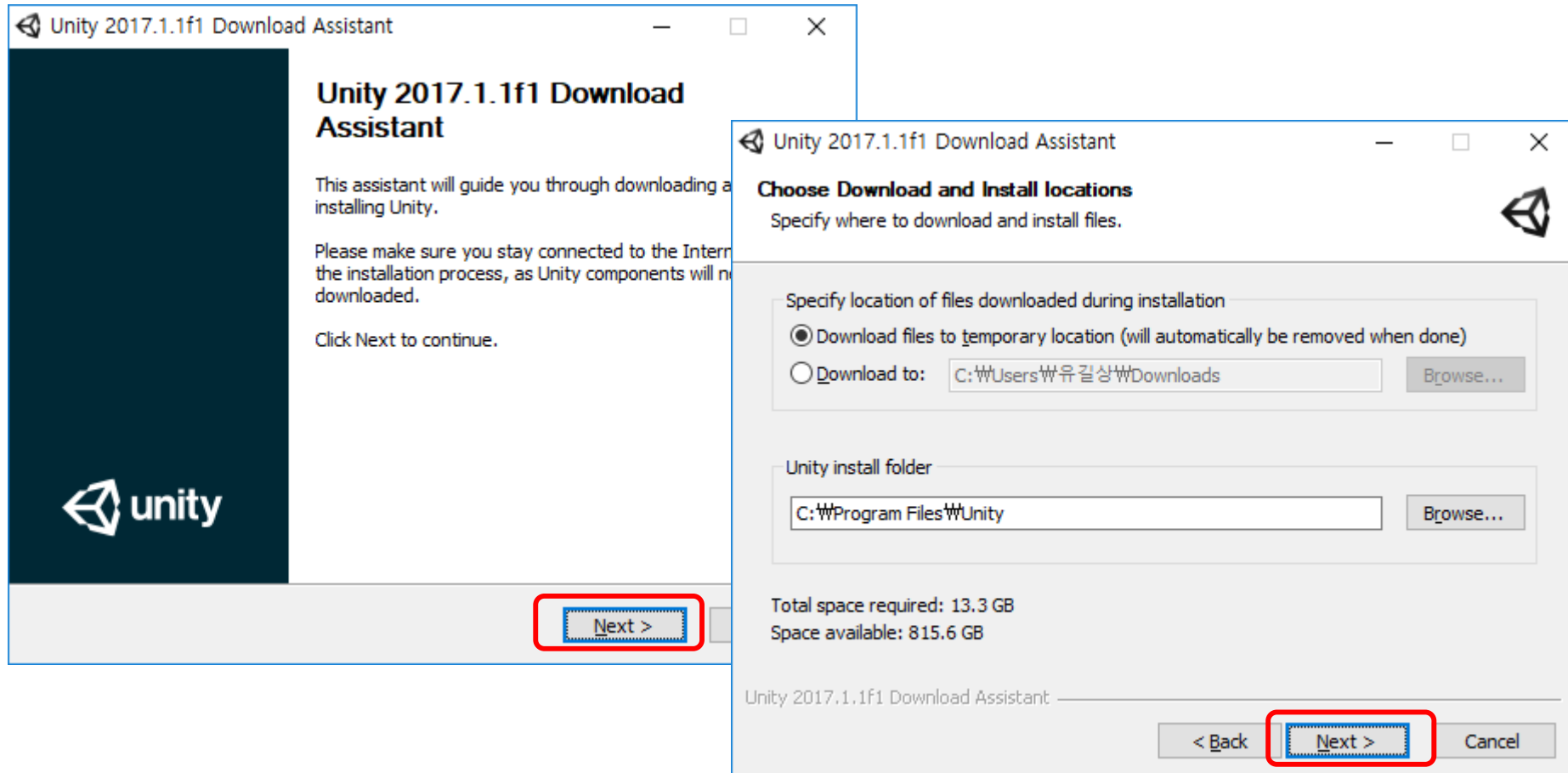
1. 유니티 설치

3) 다운로드



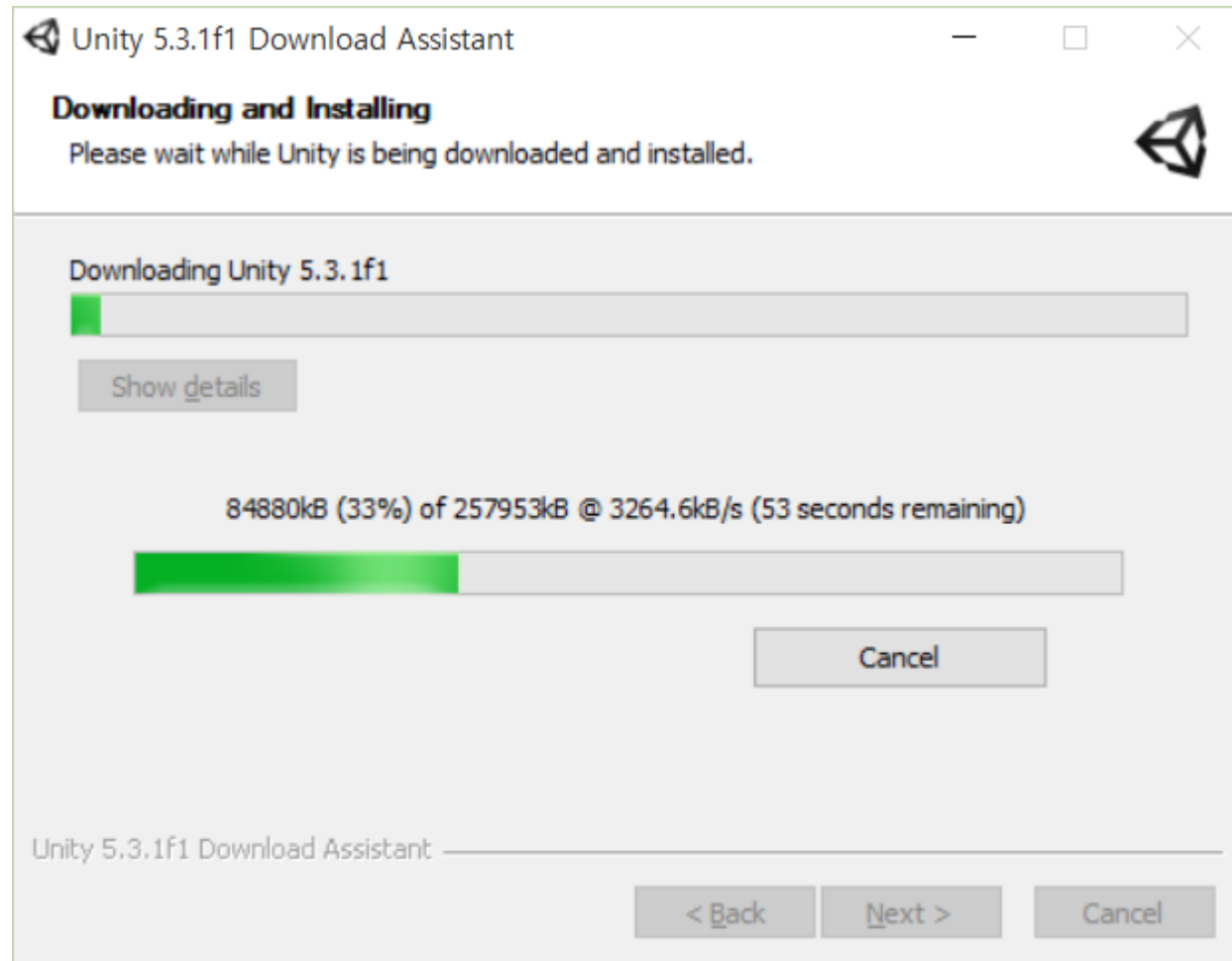
1. 유니티 설치

4) 설치 실행



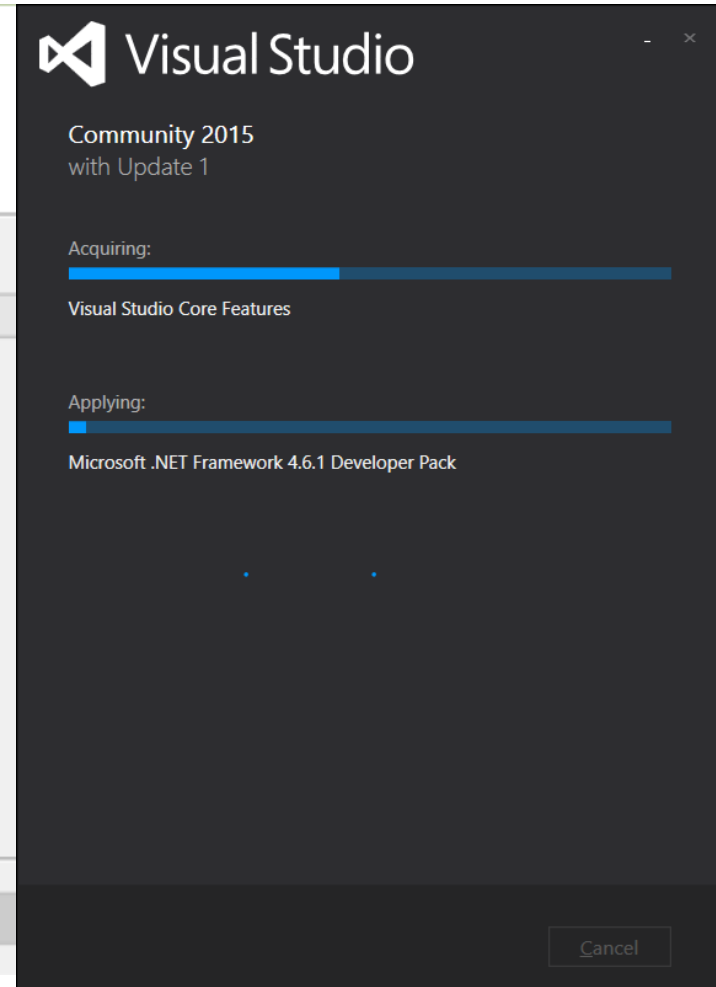
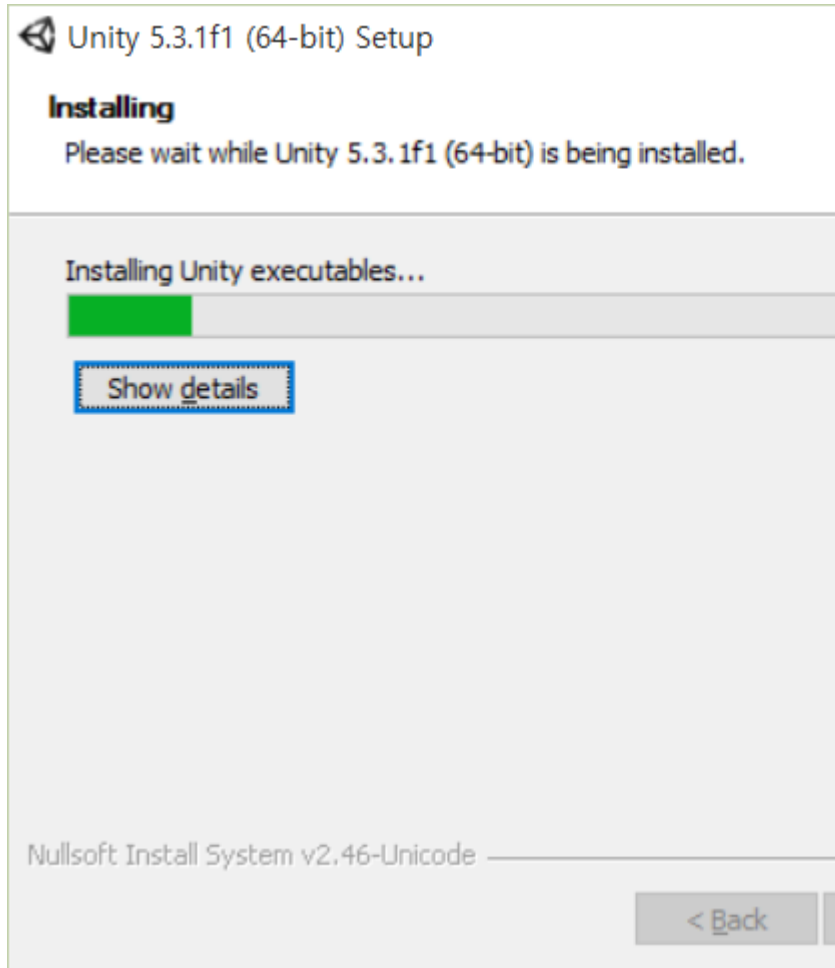
1. 유니티 설치

5) 다운로드 진행



1. 유니티 설치

6) C#개발통합도구 설치(선택)



1. 유니티 설치

7) 계정생성 및 Sign in



Unity 5.3.1f1

Welcome to Unity

Sign in License Survey Thank you

Sign into your Unity Account

If you don't have a Unity Account, please [create one](#) to access Unity services and resources.

Email * Password *

username@domain.com

Forgot your password?
Can't find your confirmation email?

Remember me ☒

Sign In

Create a Unity Account

하기 ☒ 학습하기 ☐ 정리하기

1. 유니티 설치

8) 계정생성

You need to create a Unity Account to shop in the Online and Asset Stores, participate in the Unity Community and manage your license portfolio.

Already have an account? [Sign in.](#)

Name

Username

Numbers, letters and underscores only!

Email

Country

Password

Confirm password

Security question: What is the sum of $9 + 1$?

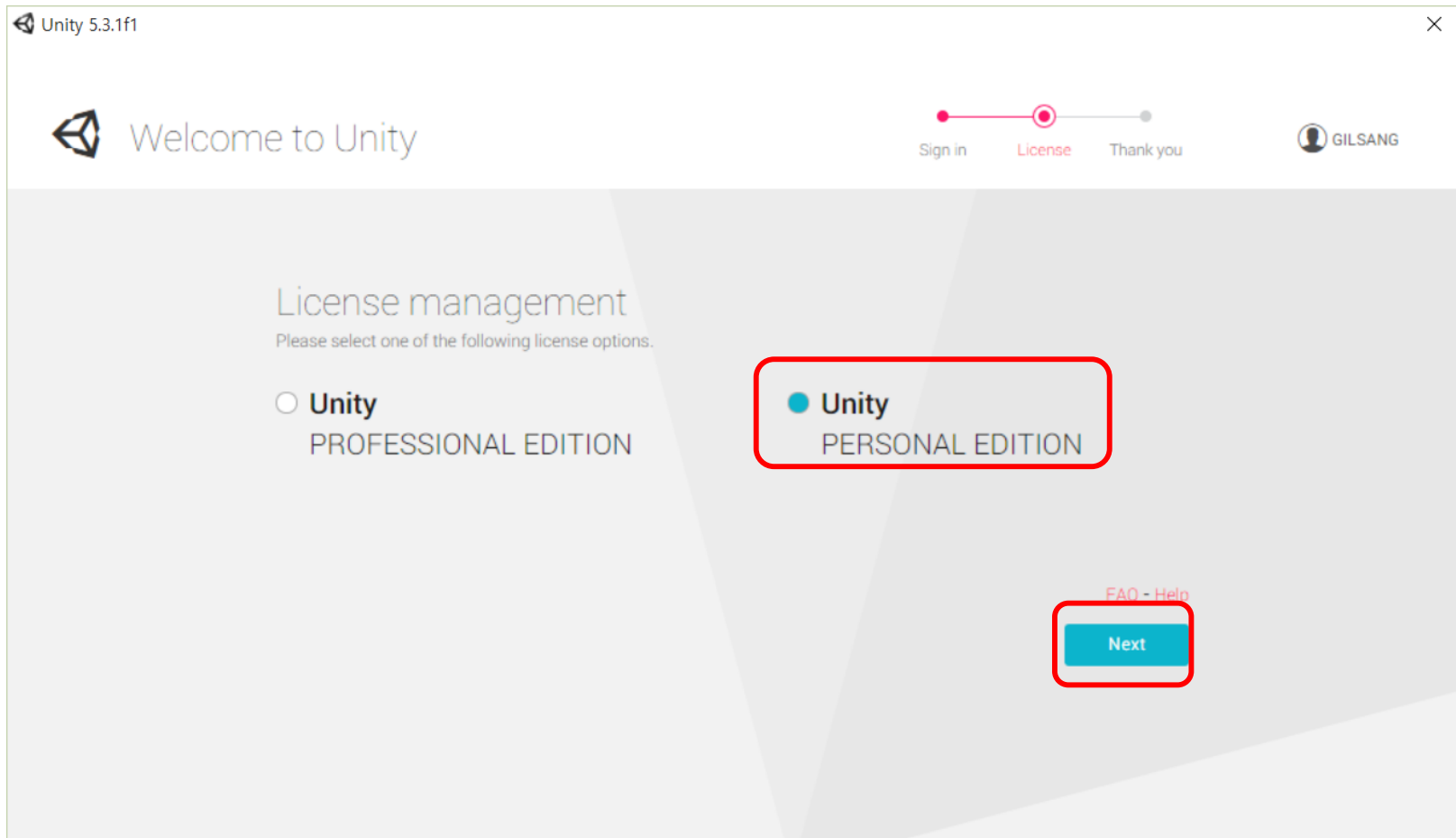
☐ I agree to the Unity [Terms of Use](#) and [Privacy Policy](#)

☐ Get Unity news, discounts and more!

Create account

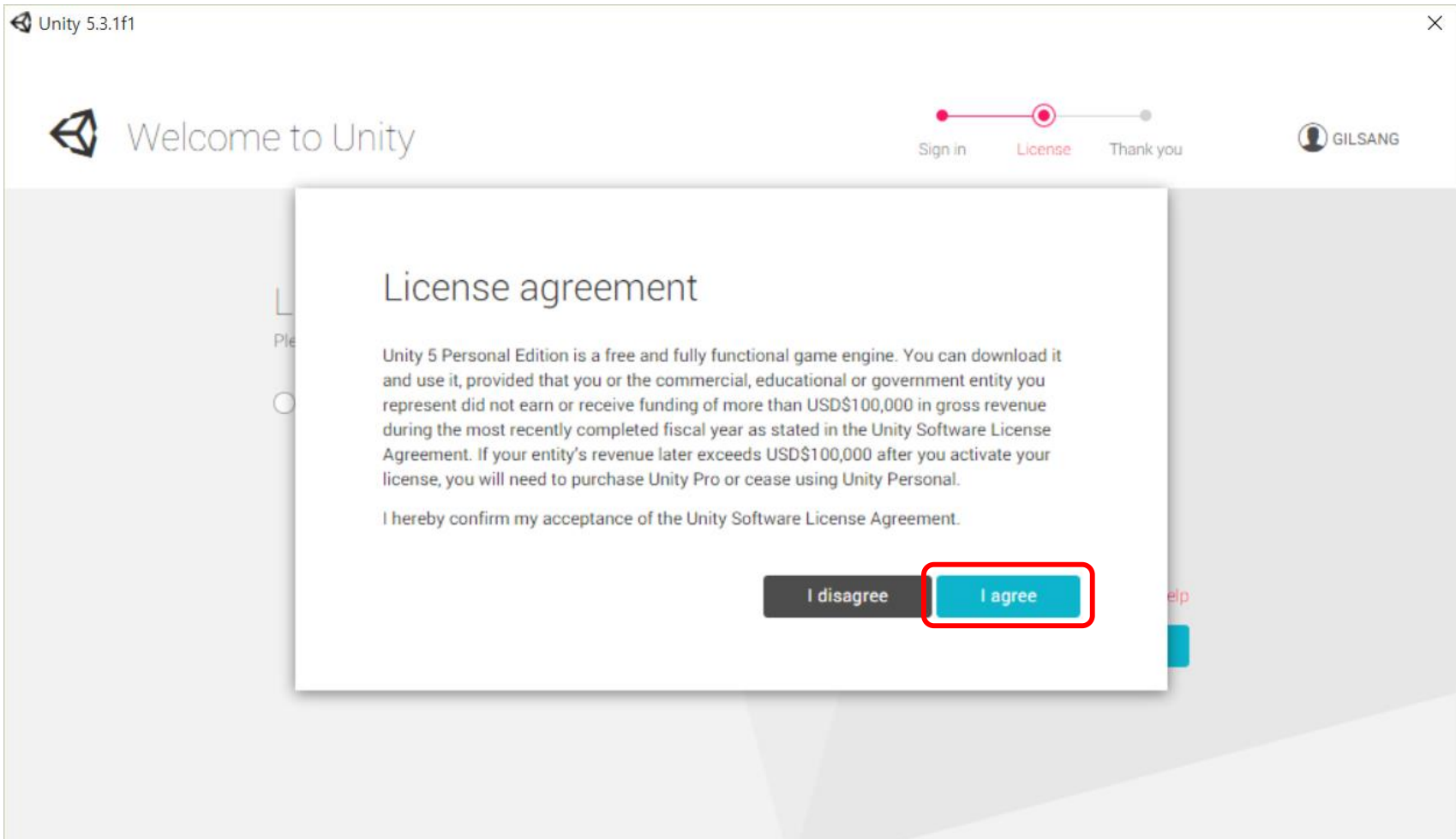
1. 유니티 설치

9) 개인용 버전 선택



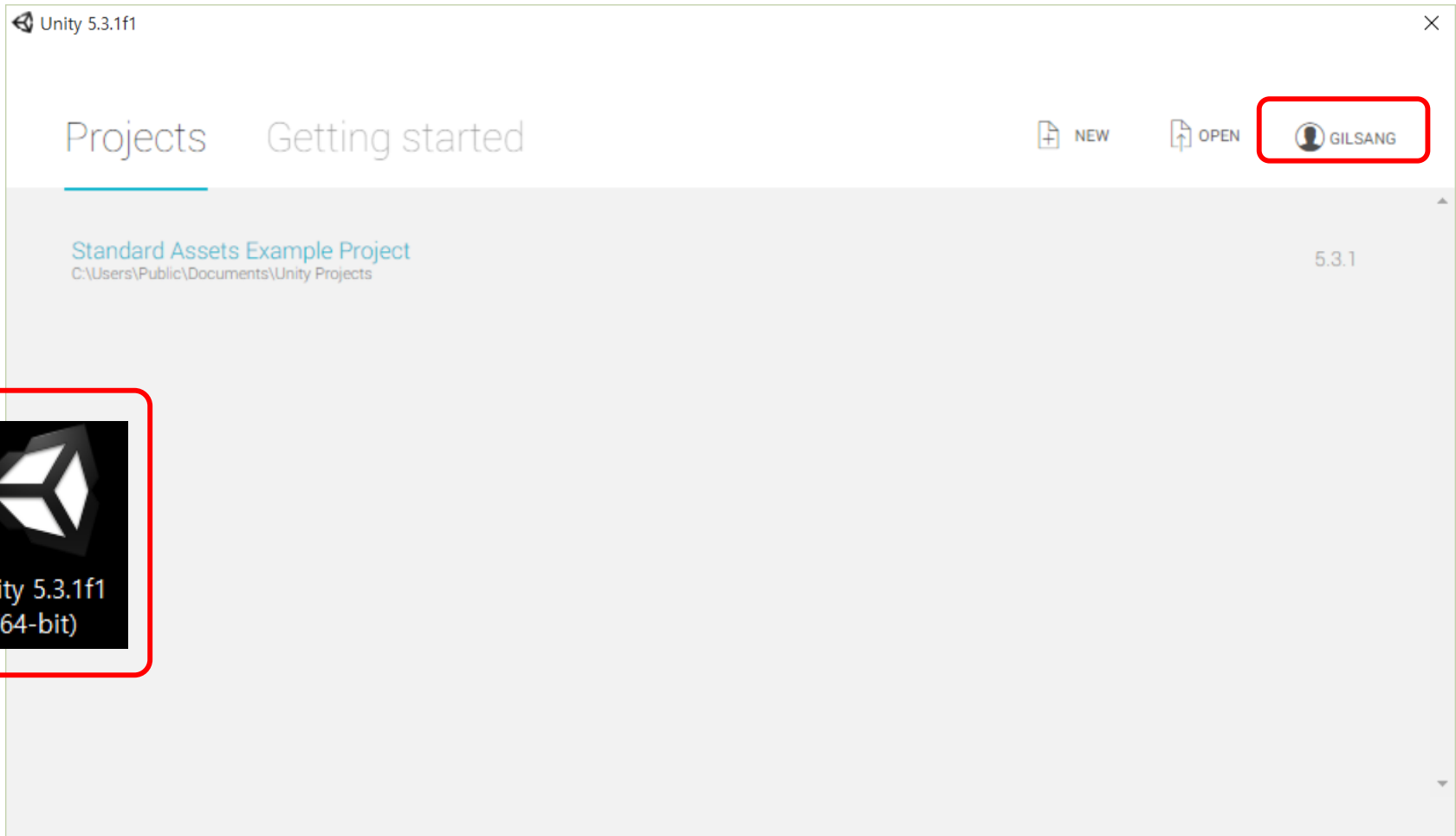
1. 유니티 설치

10) 개인용 버전 선택 (라이선스 동의)



1. 유니티 설치

11) 유니티 실행 및 sign in



2. 좌표계

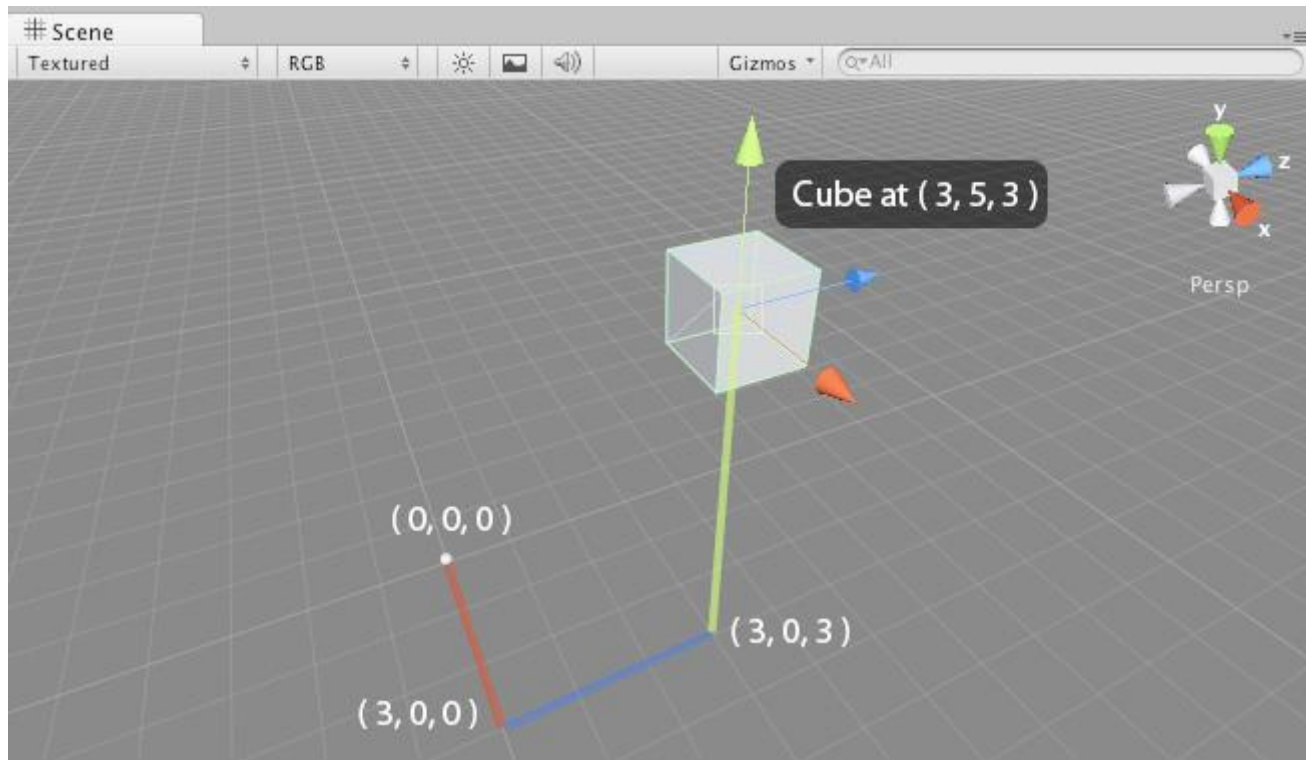
1) 유니티 좌표계 표현

- X축 수평, Y축 높이, Z축 깊이

2) 3D상에서의 차원, 회전, 위치 등도 이 방식으로 표현가능

3) 다음과 같은 방식으로 좌표 정보를 전달함 : (3,5,3)

- 그림에서 큐브는 3D공간상에서 (3,5,3)에 위치함



3. 로컬 공간과 월드 공간

- 로컬(local) 공간과 월드(world) 공간 간의 차이점
- 모든 3D 공간에는 기준점(origin) 또는 월드 영점 (world zero)라고 불리는 일종의 기준이 되는 지점이 있음 : 이 지점은 (0,0,0)에 위치함
- 3D 공간에서 오브젝트의 월드상 위치는 바로 이러한 기준점으로부터의 좌표를 의미함
- 그러나 경우에 따라서는 오브젝트 간의 상대적 위치를 정의하는 로컬공간(또는 오브젝트 공간)이 더 편할 수 있음

예)

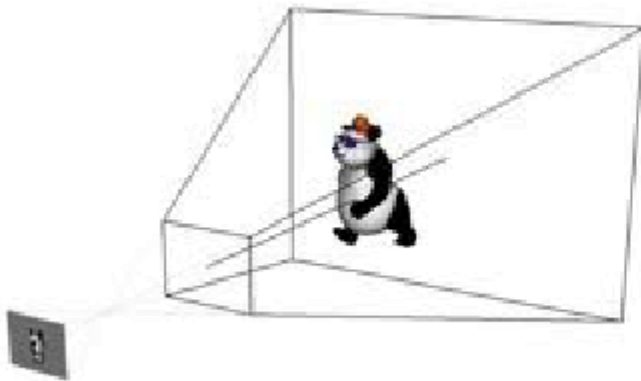
- 아파트 공사에서 남향의 기준이 되는 것이 월드 좌표계
- 방안에 가구를 배치할 때 가구를 기준으로 방 내부에 배치하는 것이 편리함 (로컬좌표계)

4. 카메라

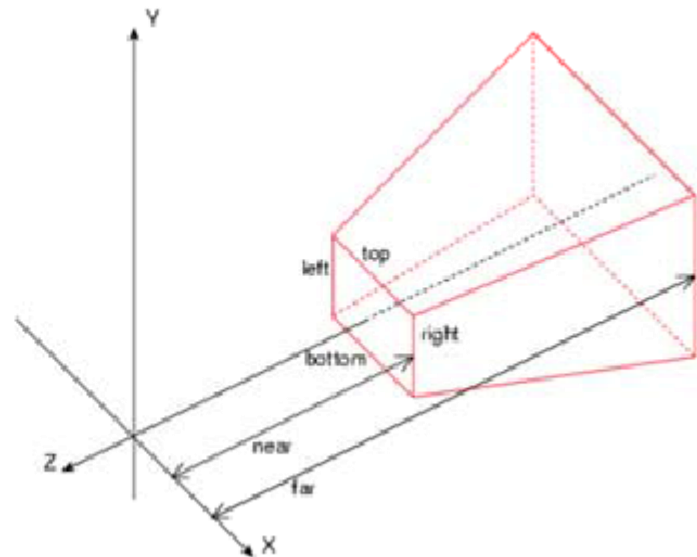
- 화면을 위한 뷰포트(viewport)로 사용된다는 점에서 카메라는 3D에서 매우 중요한 요소임
- 카메라는 월드공간의 어느 지점에도 위치 가능. 또한 애니메이션되거나 게임에 따라서는 캐릭터나 다른 오브젝트에도 부착가능
- 하나의 씬에 여러 카메라를 설치 가능하나, 플레이어가 보는 화면은 하나의 주된 카메라가 렌더링하는 것이 일반적임
- 유니티가 새로운 씬을 만들 때 메인 카메라를 기본적으로 제공하는 것도 이러한 이유 때문임

5. 프로젝션 모드

- Perspective Projection(원근 투영), Orthographic Projection(직교 투영)
 - 카메라의 프로젝션 모드(Projection Mode)는 카메라가 대상을 3D (Perspective)로 담느냐 아니면 2D (Orthographic)로 담느냐를 결정함
- 일반적으로 카메라는 퍼스펙티브 프로젝션 모드를 사용하고, 그 결과 피라미드 모양의 FOV(Field of View)를 가짐

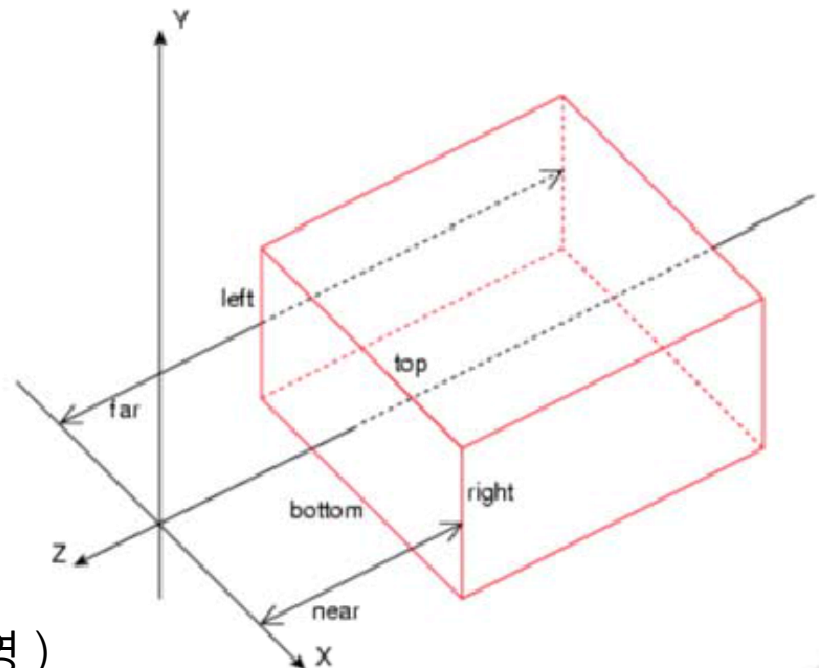
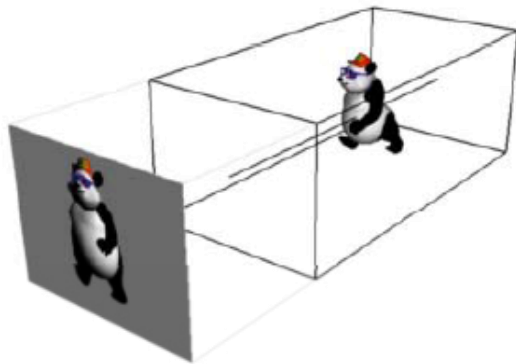


Perspective Projection(원근 투영)



5. 프로젝션 모드

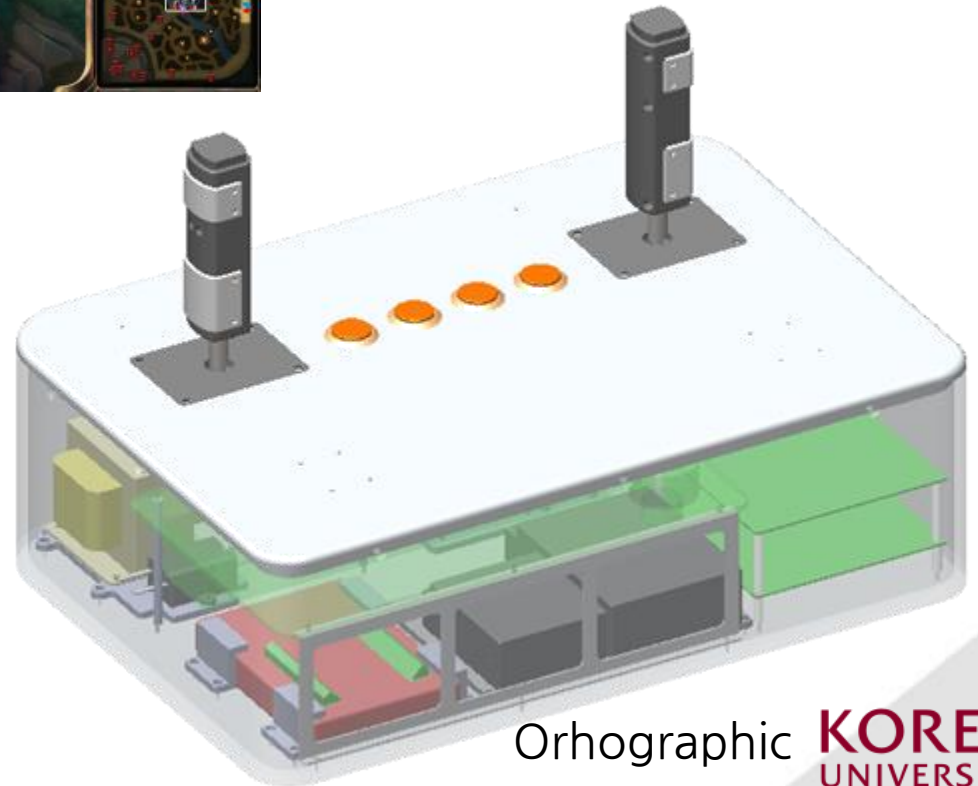
- 대상을 2D로 렌더링하는 Orthographic 프로젝션 모드도 사용가능함. 이는 직육면체 모양의 FOV(Field of View) 를 가짐
- 이 모드는 2D 게임의 메인 카메라에 사용되거나 3D/2D게임의 미니맵 등을 렌더링하기 위한 보조 카메라로 사용됨



Orthographic Projection(직교 투영)

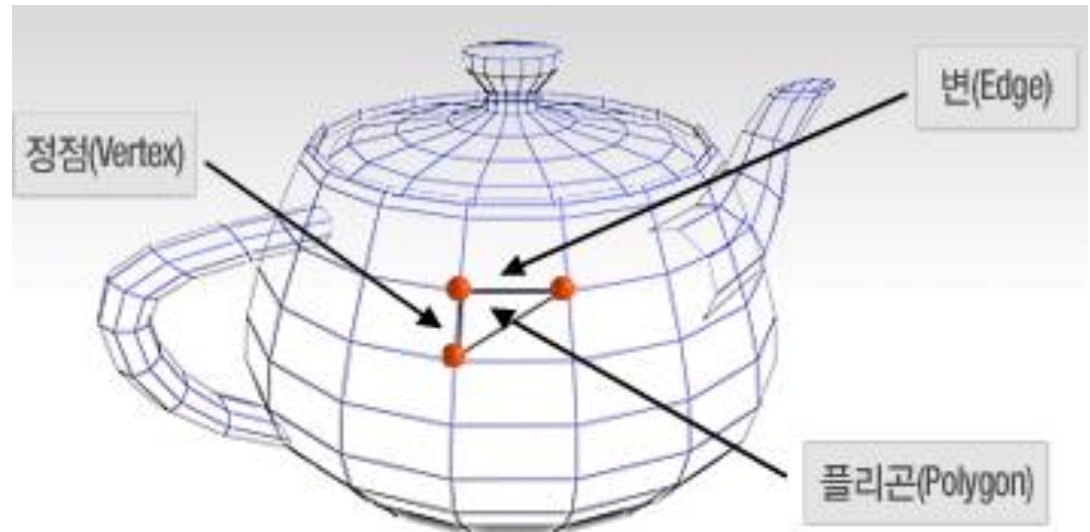
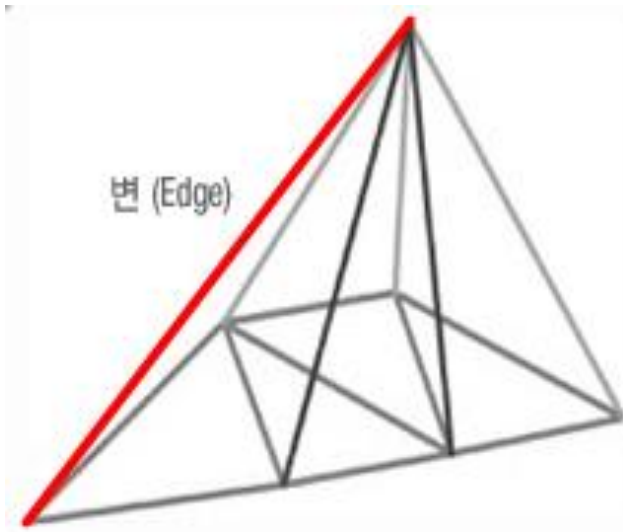


〈league of legends〉
Perspective mode



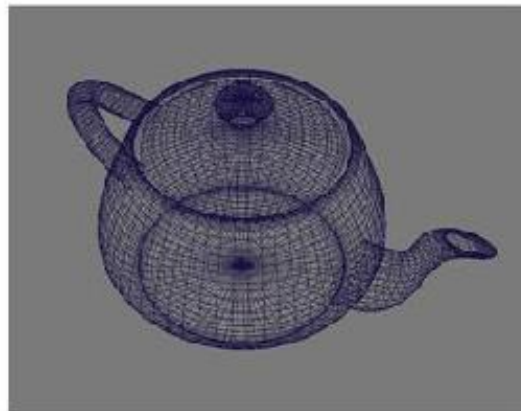
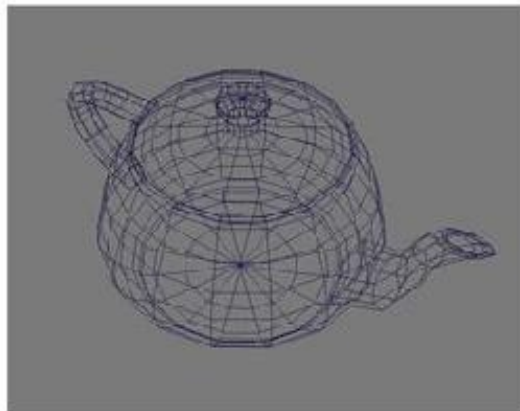
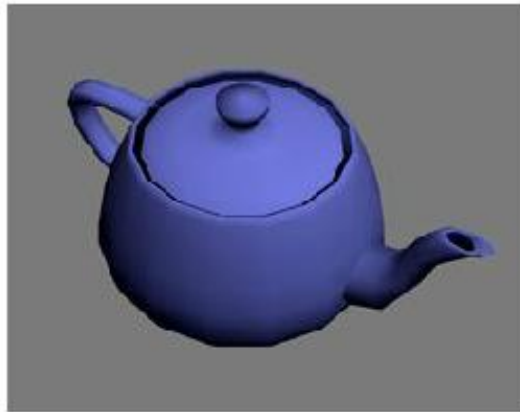
6. Polygon, Edge, Vertex, Mesh

- 정점(Vertex)
- 선(Edge): 버텍스와 버텍스를 연결하는 선들
- 면(Polygon): 3개 이상의 정점이 모이면 한 개의 면이 생성
- 메시(Mesh): 폴리곤들이 모여 하나의 3차원 물체를 생성하는 것



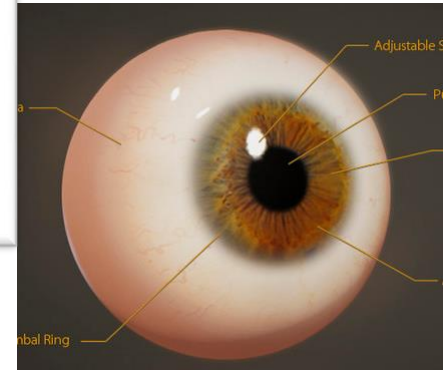
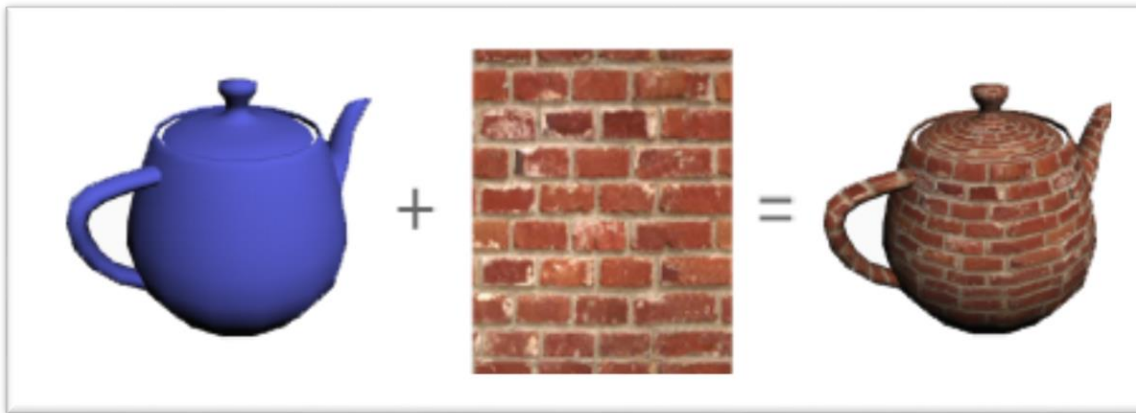
폴리곤 수의 영향력

- 폴리곤의 수는 물체의 섬세함에 직접적인 영향을 줌
- 같은 물체를 구성하는 폴리곤의 양이 많으면 매우 부드럽고 정밀한 물체의 형태를 만들어 줌
- 그러나, 폴리곤의 양이 많으면 그 만큼 처리해야 하는 데이터의 양이 늘어나게 되어 시스템의 고사양을 요구하게 됨



7. Texture, Shade

- 텍스처(Texture)
 - Vertex만으로 물체를 보여주기에는 제약이 많음
 - 섬세한 물체를 표현하기 위해서는 많은 수의 정점이 요구됨
- 텍스처 맵핑
 - 3차원 정점에 2차원 이미지를 입히는 것을 Texture mapping 이라고 함



7. Texture, Shade

■ 텍스처 맵 제작과정

1단계

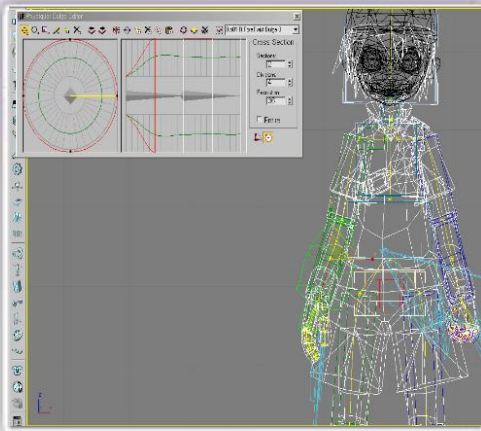


그림 모델링의 UVW 좌표

UVW좌표란 각 오브젝트에 적용되는 이미지의 좌표

2단계



그림 텍스처 이미지

텍스처는 물체 표면에 대한 시각적 느낌을 나타내는 것을 말함

3단계

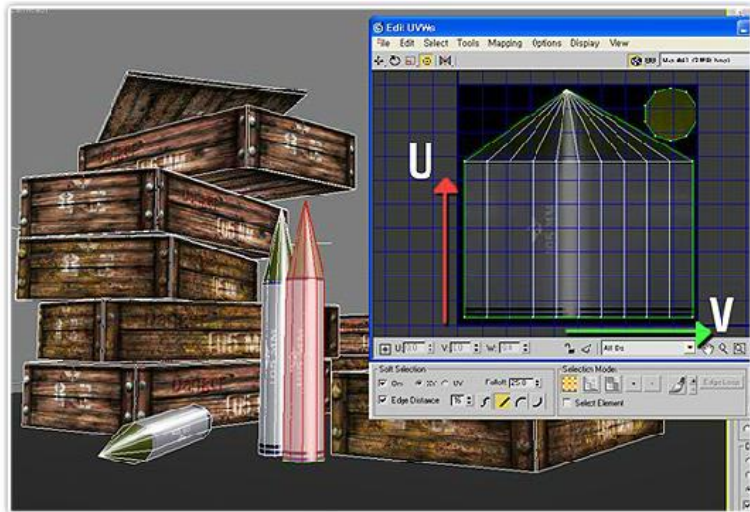


그림 모델링 Mapping하기

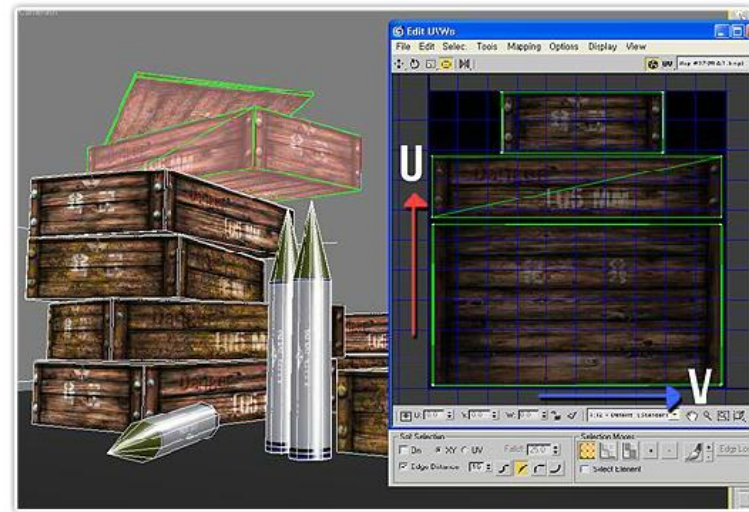
Mapping은 오브젝트 표면에 UVW 좌표를 입혀 주는 과정

7. Texture, Shade

■ 텍스처 맵 적용 예



〈총알 오브젝트에 적용되는 UVW좌표〉



〈상자 오브젝트에 적용되는 UVW좌표〉

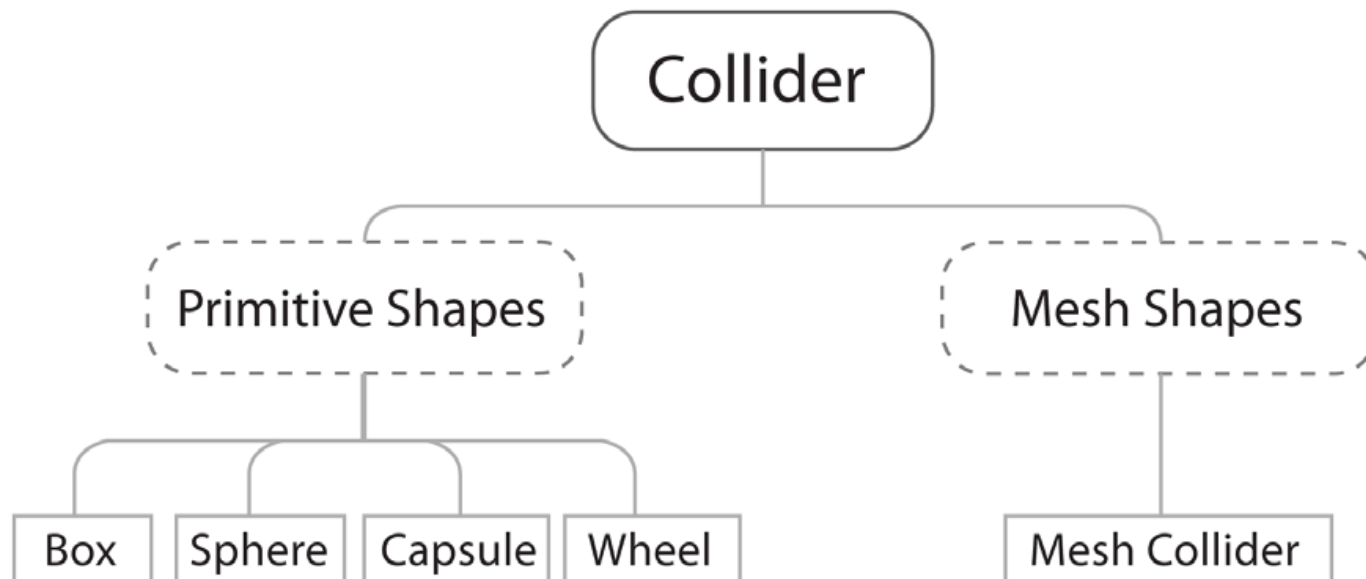
8. 물리효과

* 리지드 바디 다이내믹스

- 게임 엔진을 사용하는 개발자에게, 물리엔진은 게임 속 오브젝트가 실제 세계에서 반응하는 방식을 재현하는 수단을 제공함
- 유니티는 널리 사용되고, 매우 정교한 물리 효과를 재현하는 상용 엔진인 엔비디아(Nvidia)의 PhysX엔진을 사용함

9. 충돌감지

- 오브젝트에 충돌체(collider) 컴포넌트를 추가하는 것은 오브젝트 주위로 보이지 않는 그물을 치는 것에 비유할 수 있음
- 이 그물은 대개 그 오브젝트와 유사한 형태를 띠며, 다른 충돌체와의 충돌 여부를 보고해서 엔진이 그에 따라 반응하도록 하는 역할을 담당함
- 유니티는 2가지 유형의 충돌체를 제공함



9. Mesh Collider

- 메쉬 충돌체를 이용해서 복잡한 충돌 상황을 감지할 때, 게임 엔진은 버텍스를 이용하여 충돌지점을 계산함
- 충돌감지에 쓰이는 메쉬는 실제 오브젝트의 메쉬보다 단순하게 표현함
- 물리엔진이 충돌을 위해 복잡한 메쉬를 처리하지 않기 때문에 성능향상을 꾀할 수 있음

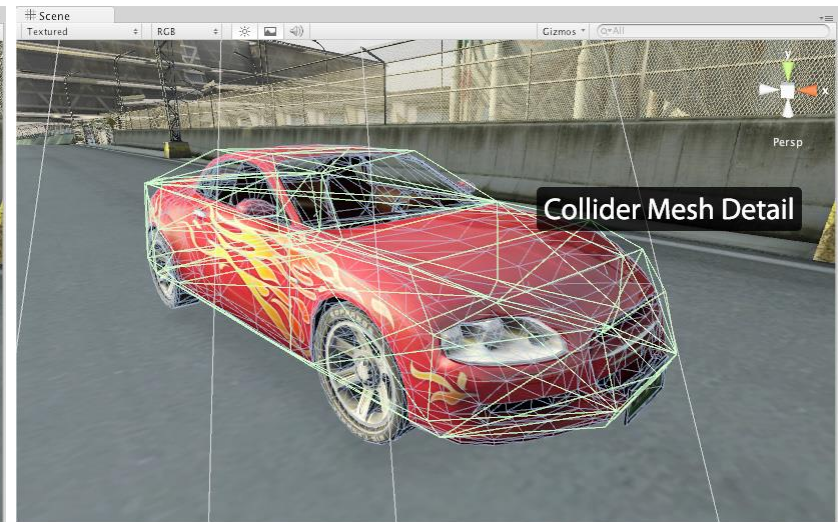


그림 (좌)자동차 오브젝트 메쉬, 그림 (우) 자동차 오브젝트의 충돌 메쉬

1

월드(또는 글로벌) 좌표는 게임 화면을 기준으로 한 절대 좌표, 로컬 좌표는 오브젝트를 기준으로 한 좌표이다.

2

카메라의 프로젝션 모드(Projection Mode)는 Perspective, 또는 Orthographic 모드가 있다.

3

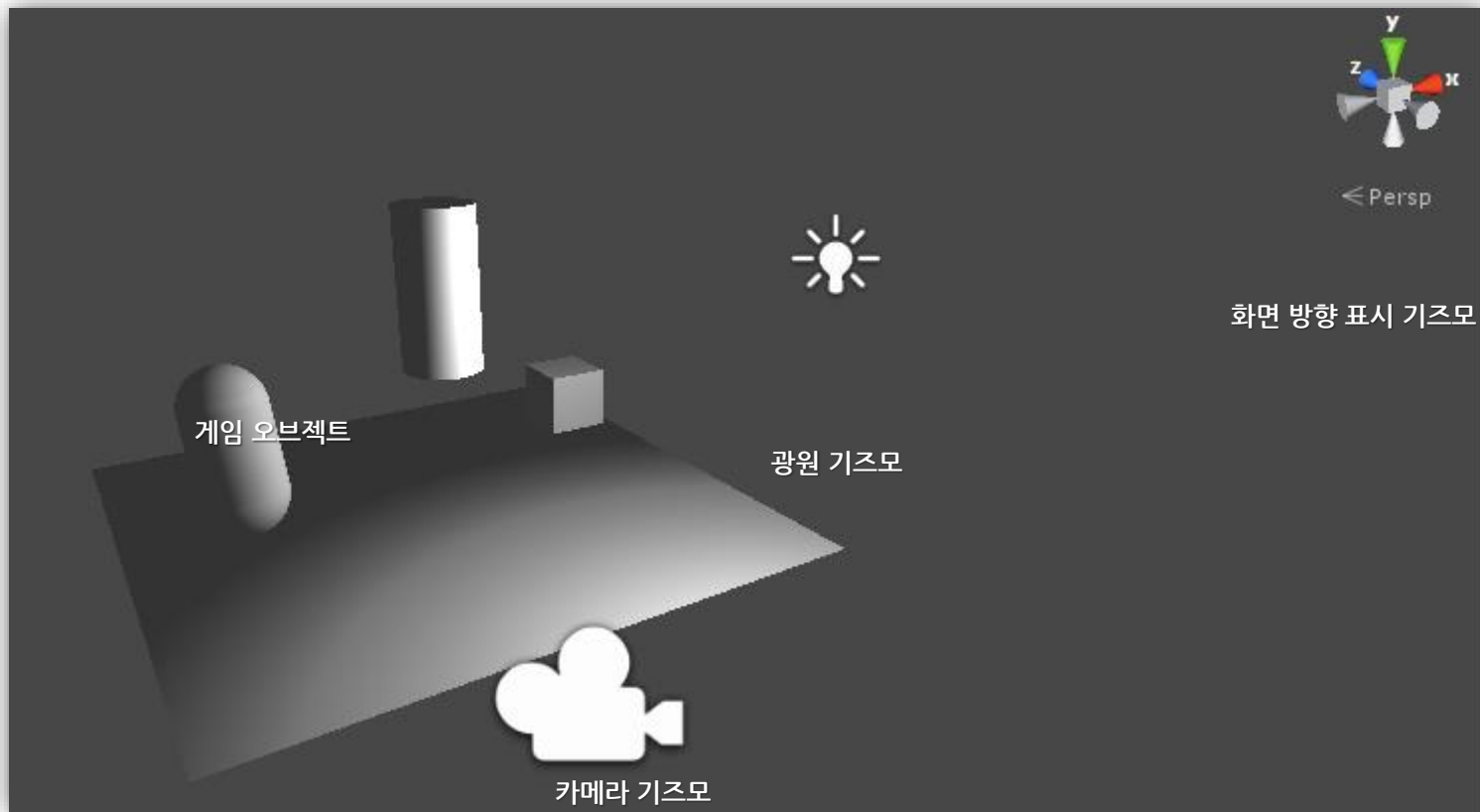
3D 형태를 가진 모든 오브젝트는 기본적으로 폴리곤을 조합함으로써 메쉬라고 하는 복잡한 객체 형태를 만든다

4

오브젝트에 충돌체(collider) 컴포넌트를 사용하면 다른 충돌체와의 충돌 여부를 알 수 있다

생각해보기

- 유니티 가상공간의 객체들(Object, Gizmo)
기즈모 = 사전적의미로 간단한 장치, 유니티에서는 보이지 않는 객체들을 표현하기 위한 것들을 말함. 예를 들어 카메라를 보면 카메라 객체가 보이지만 실제 플레이해 보면 볼 수 없음, 단지 카메라 뷰 포인트를 역할을 수행

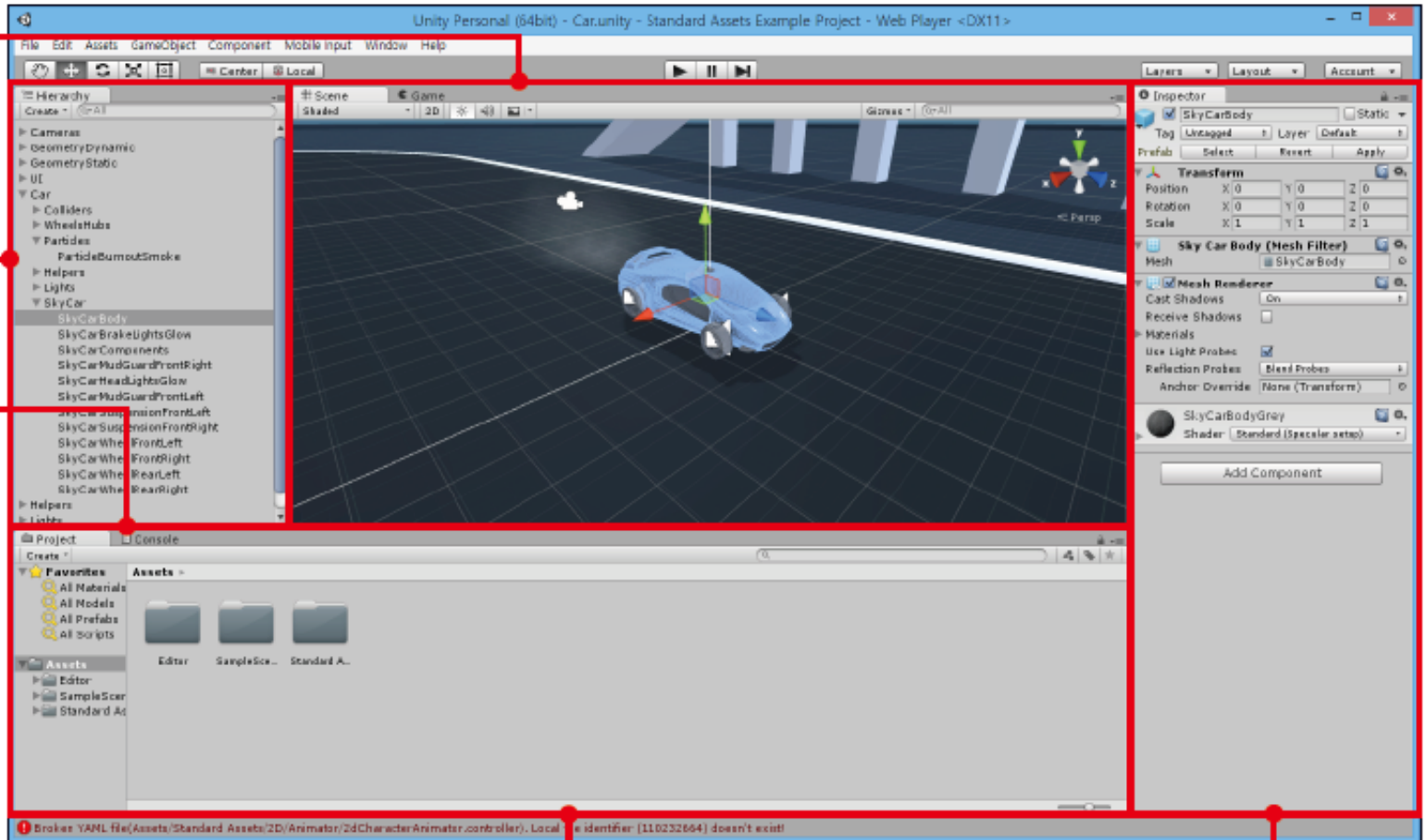


1. 유니티 개발 화면

Scene 뷰
(Game 뷰)

Hierarchy
뷰

Console
뷰

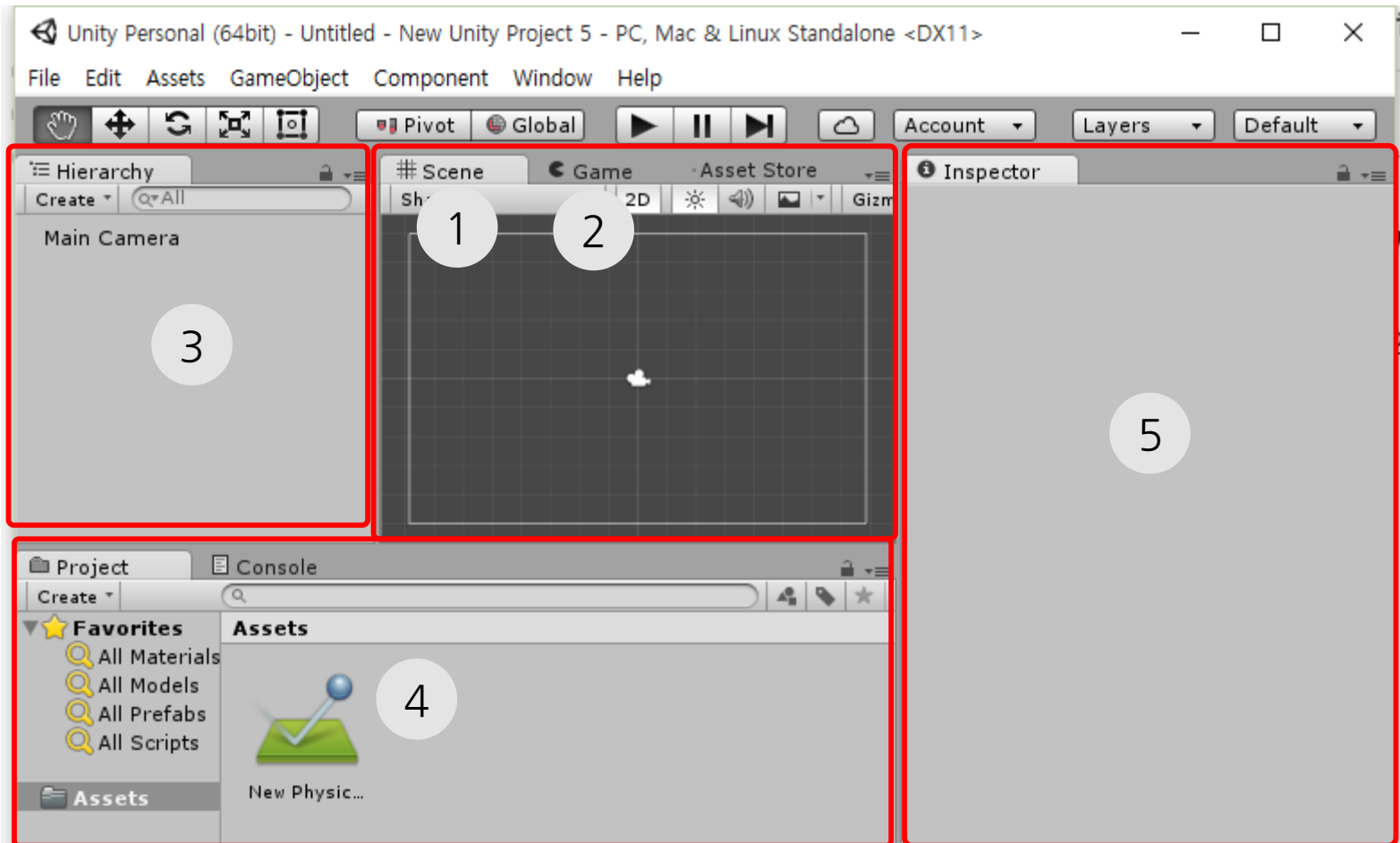


Project 뷰

Inspector 뷰

유니티 창(패널, 뷰)살펴보기

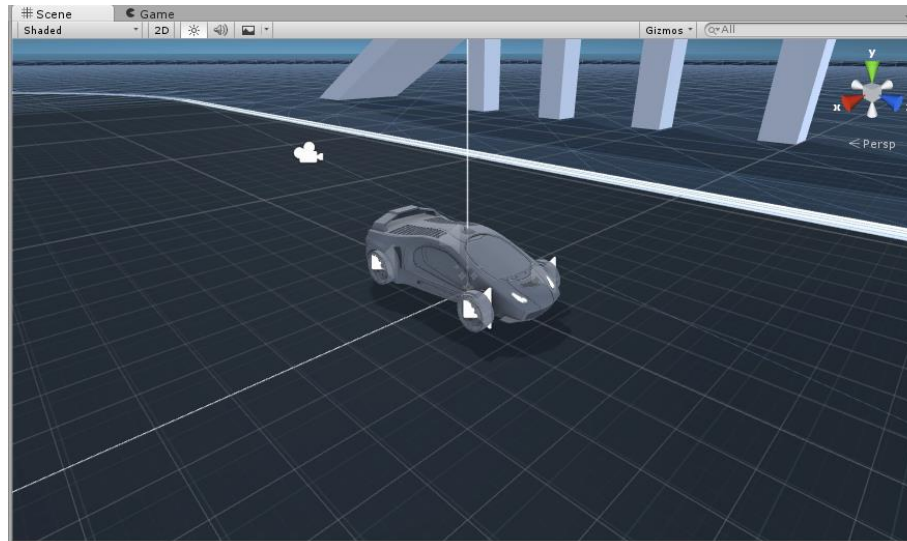
- [1] - Scene (무대 창)
- [2] - Game (카메라 촬영결과 출력 창)
- [3] - Hierarchy (무대에서 사용된 소품을 관리하는 창)
- [4] - Project (소품 참고)
- [5] - Inspector (소품에 대한 설정을 변경해 줄 수 있는 창)



1. 유니티 개발 화면

1) Scene 뷰

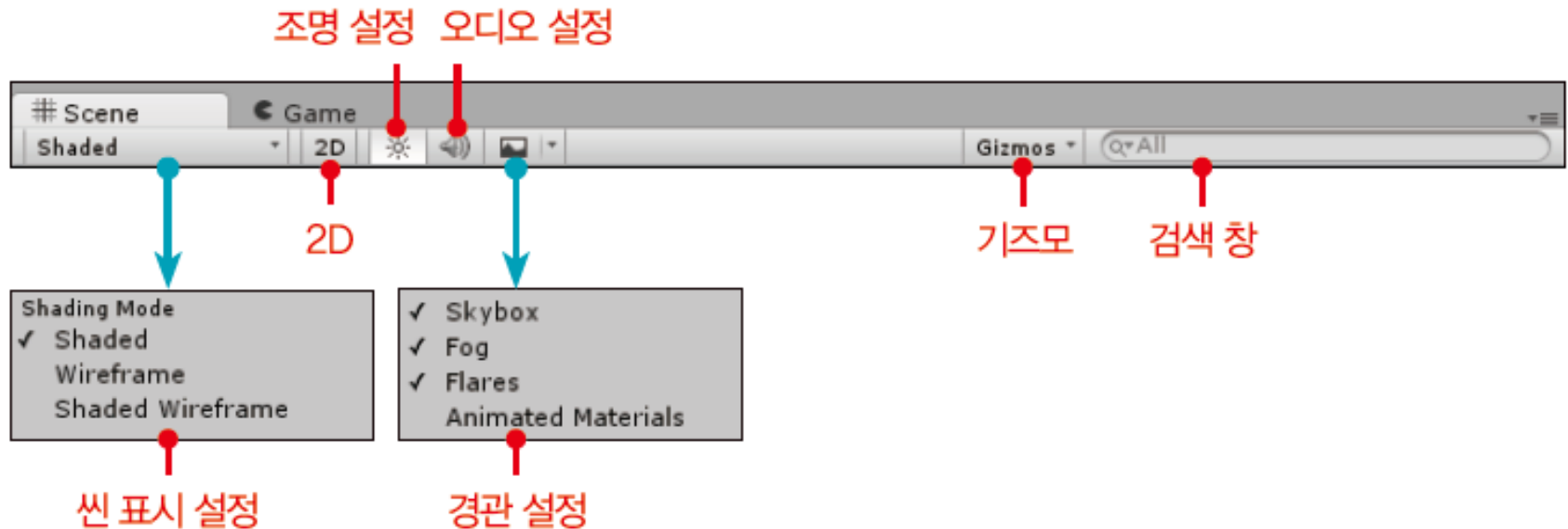
- 작성 중인 게임의 게임 오브젝트 표시
 - 실제로 확인하면서 위치나 크기 등을 조정 가능
- 플레이 중에 변경한 내용
 - 플레이 모드 종료와 동시에 원래대로 돌아감
- 게임 만드는 동안 유니티에서 가장 많이 사용하는 뷰



1. 유니티 개발 화면

1) Scene 뷰 - 컨트롤 바

- 씬의 표시 등을 설정

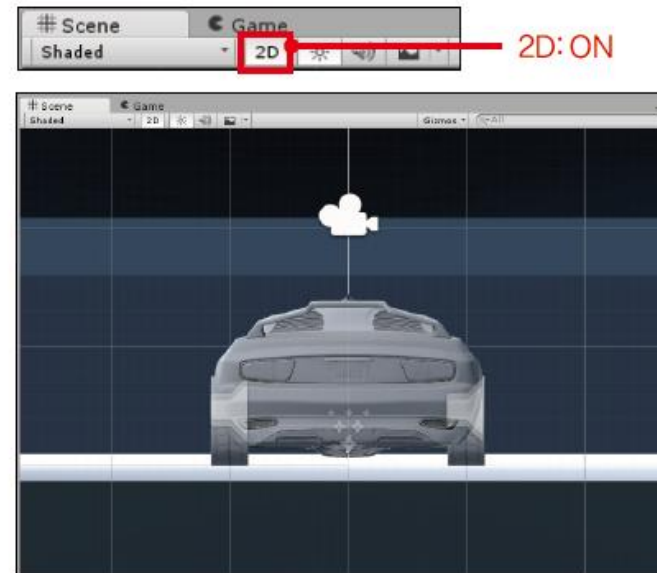
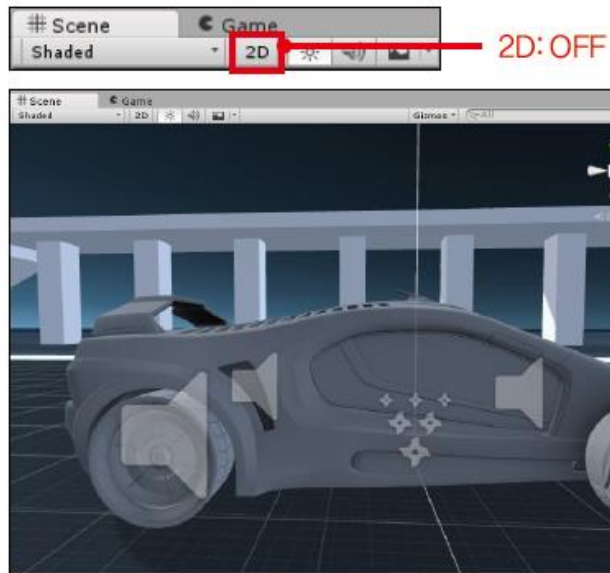


1. 유니티 개발 화면

1) Scene 뷰 - 컨트롤 바

■ 2D

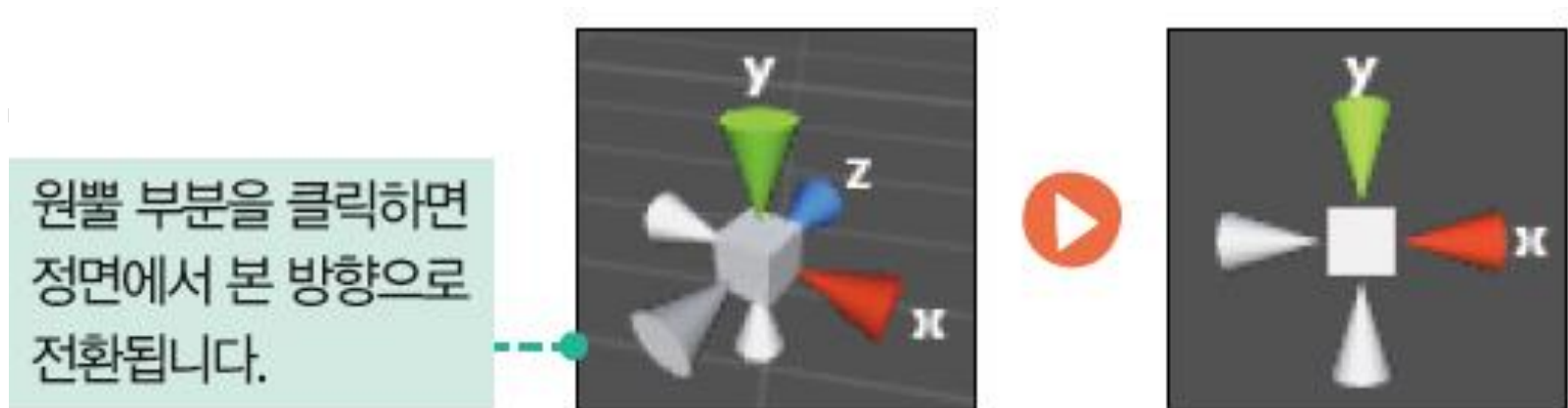
- Scene 뷰 표시 2D로 전환
- 2D 게임 만들 때 사용
- 화면 오른쪽 상단의 썸 기즈모가 사라지고, X와 Y 2개의 좌표로 오브젝트 표시



1. 유니티 개발 화면

1) Scene 뷰 - 컨트롤 바

- 씰 기즈모

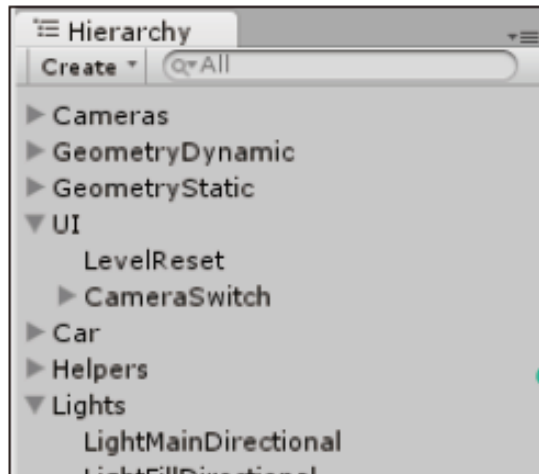


- 검색 창 - Scene 뷰의 오브젝트 검색 가능

1. 유니티 개발 화면

2) Hierarchy 뷰 (계층 뷰)

- 현재 선택된 씬에 배치된 모든 오브젝트 포함
- 오브젝트의 계층 구조 확인하고 편집
 - 캐릭터나 도형 등의 리소스를 Hierarchy 뷰에 추가
 - 씬에 오브젝트 배치
 - 오브젝트 추가는 스크립트 통해서도 가능

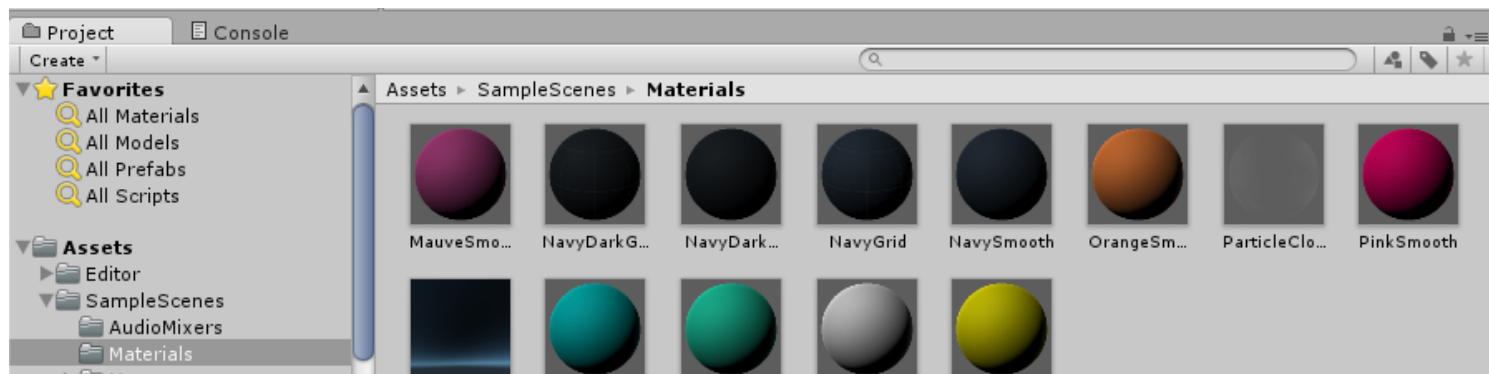


이 그림은 프로젝트를 만든 후 오브젝트가 추가된 상태입니다. 새 프로젝트를 만들면 일반적으로 Main Camera와 Directional Light만 배치되어 있습니다.

1. 유니티 개발 화면

3) Project 뷰

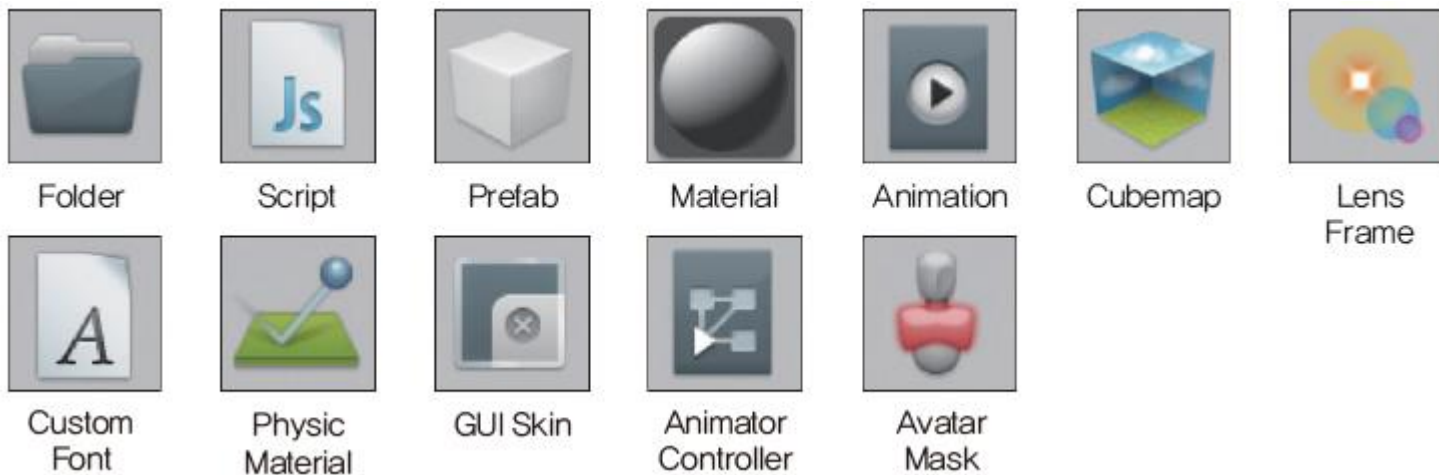
- 만들고 있는 게임 프로젝트에 사용될 모든 오브젝트와 텍스처, 음악 데이터, 스크립트 등 게임 형성 요소 포함
- Project 뷰에서 유니티로 импорт
 - 다른 프로그램에서 만든 캐릭터나 음악 리소스를 다른 뷰에 전달
- 왼쪽 패널은 프로젝트의 폴더 구조를 계층 목록으로 표시
 - 폴더 클릭하여 선택하면 세부 내용이 오른쪽 창에 표시
- 오른쪽 상단에는 검색 창
 - 게임을 구성하는 요소가 많을 때도 이름으로 쉽게 찾을 수 있음



1. 유니티 개발 화면

3) Project 뷰

- Project 뷰의 아이콘
 - 게임 구성 요소를 시각적으로 관리 가능
 - Project 뷰에 표시되는 아이콘도 오브젝트의 종류에 따라 각각 다른 모양으로 식별할 수 있도록 고안



1. 유니티 개발 화면

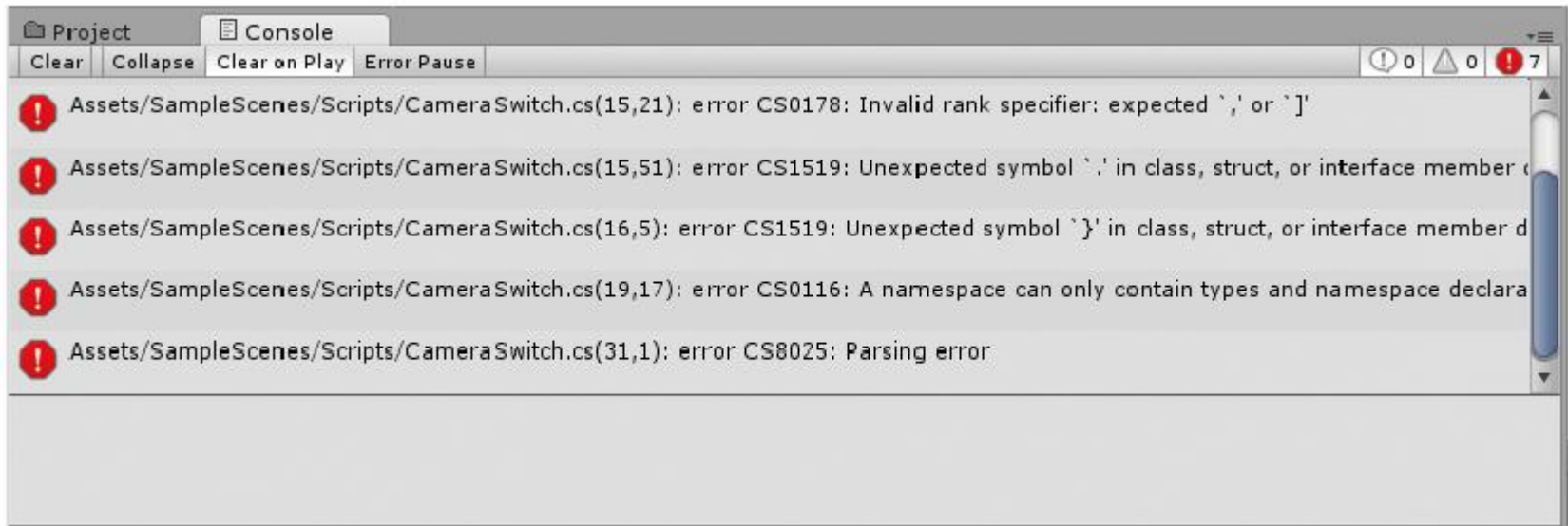
4) Inspector View(인스펙터 뷰)

- 선택한 오브젝트의 구성과 요소확인 및 편집
 - 스크립트, 메시, 오디오, 물리적 동작 등
 - 유니티에서는 이러한 요소를 컴포넌트라고 부름
- 컴포넌트 추가 - Inspector 뷰에 그 정보가 표시
- Inspector 뷰에서 구성 요소 추가/제거가능
- 오브젝트 각각의 정보 확인하거나 설정 변경 가능
 - 오브젝트가 어디에 위치하고 있으며(Position), 어느 각도를 향하고(Rotation), 어느 정도의 크기(Scale)인지 설정
- 각 항목 이름 왼쪽에 있는 ► 클릭해 표시 확장/ 축소

1. 유니티 개발 화면

5) Console 뷰

- 메시지, 경고, 오류 등의 로그 표시
- 뷰 상단 컨트롤 바에서 표시 내용 변경
- 탭을 눌러 Console 뷰와 Project 뷰를 전환하여 표시



1

유니티 인터페이스는 씬뷰, 계층뷰, 인스펙터뷰, 프로젝트뷰, 콘솔 뷰 등으로 구성되어 있다.

2

씬뷰는 작성중인 게임오브젝트를 표시하는 창으로 콘텐츠를 만드는 동안 가장 많이 사용하는 뷰이다.

3

계층뷰에는 게임제작에 사용된 모든 오브젝트들을 계층적으로 관리할 수 있도록 제공 한다.

3D콘텐츠 이론 및 활용

유길상

2주. 유니티 인터페이스

1차시 : 패널과 화면구성 및 조작

2차시 : 오브젝트 조작

2차시 학습목표

- 게임오브젝트로 객체를 생성하고 조작할 수 있다.

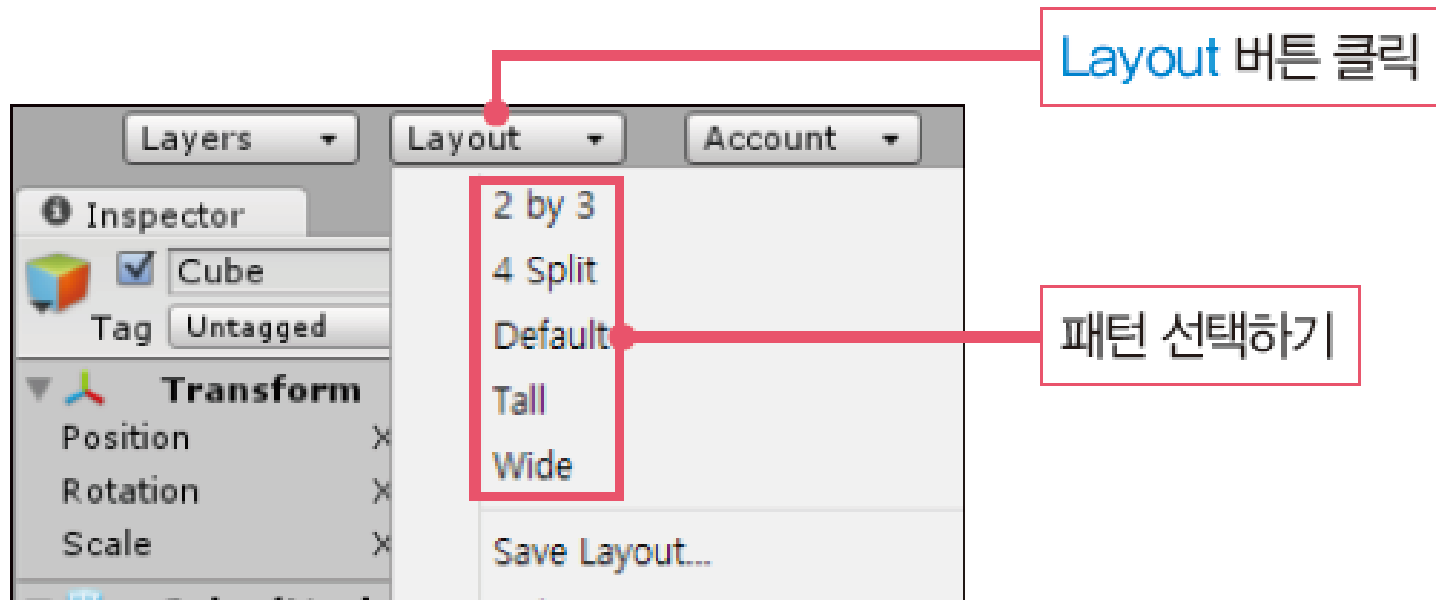
2차시 학습내용

- 게임 오브젝트 생성하기

1. 조작도구

1) 뷰의 배치 변경하기

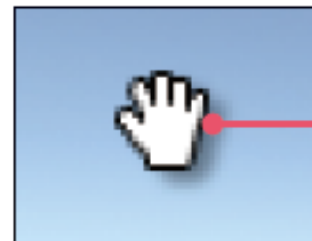
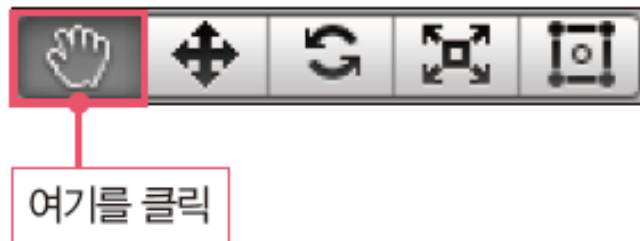
- Inspector 뷰에 있는 Layout 버튼으로 변경
- 각 뷰의 탭 부분 드래그하여 자유롭게 이동
- 원래 위치로 복원하려면 Layout에서 “패턴” 선택



1. 조작도구

2) 손/눈 도구

- Scene 뷰에서 오브젝트 드래그하여 표시 위치 변경
 - 표시 위치만 변경
 - 오브젝트의 속성값(위치, 기울기, 크기)은 변경 X
- 마우스 휠로 표시 영역을 확대/축소
- Scene 뷰에서 Alt 키 누르면 눈 도구로 표시됨
 - 다양한 각도에서 오브젝트 확인



마우스 커서가 손 모양으로 바뀝니다.

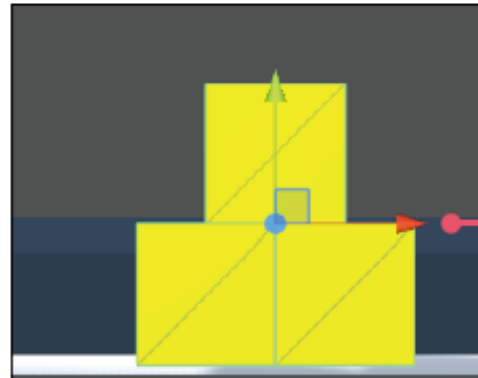
1. 조작도구

3) 오브젝트 이동 도구

- 오브젝트를 Scene 뷰에서 X, Y, Z축 따라 직선으로 움직여 시각적으로 위치 조정
- 축 방향 변경하며 이동시키는 것



여기를 클릭



이동시키고 싶은 방향으로 화살표 (또는 중앙의 사각형)를 드래그합니다.

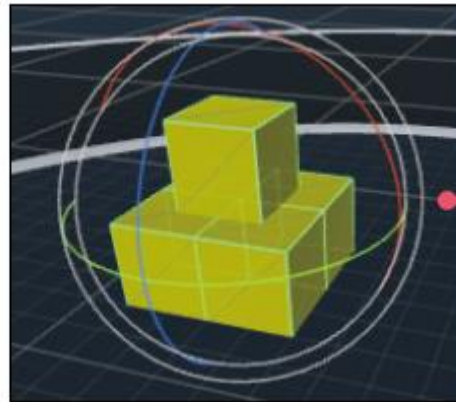
1. 조작도구

4) 오브젝트 회전도구

- 오브젝트 회전시켜 시각적으로 기울기 조정
- 썬 기즈모로 좌표축 전환하여 다양한 방향으로 회전



여기를 클릭

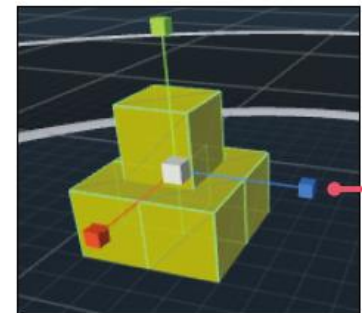


회전하고 싶은 방향으로
마우스를 드래그

1. 조작도구

5) 오브젝트 스케일 도구

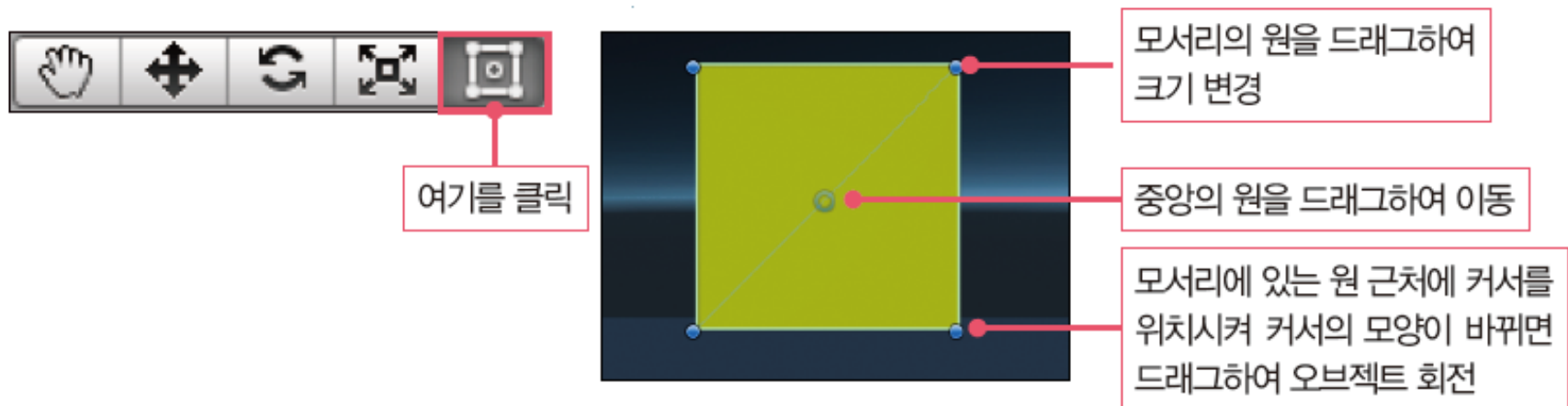
- 오브젝트의 크기 시각적으로 변경
 - 빨간색(X축), 녹색(Y축), 파란색(Z축) 화살표를 바깥쪽으로 당겨서 선택한 오브젝트를 확대(X, Y, Z 방향의 값 크게)
 - 안쪽으로 당겨서 축소(X, Y, Z 방향의 값 작게)
- Scene 뷰에서 오브젝트의 위치, 기울기, 크기 변경
 - Inspector 뷰에 바로 반영
- Inspector 뷰에서 각 항목의 값을 변경
 - Scene 뷰의 오브젝트에 반영
- 스케일 도구 활용
 - 대략적 조정 - 씬 뷰, 마우스 사용
 - 세부적 조정 - Inspector 뷰, 값 조정



1. 조작도구

6) 오브젝트 직사각형 도구

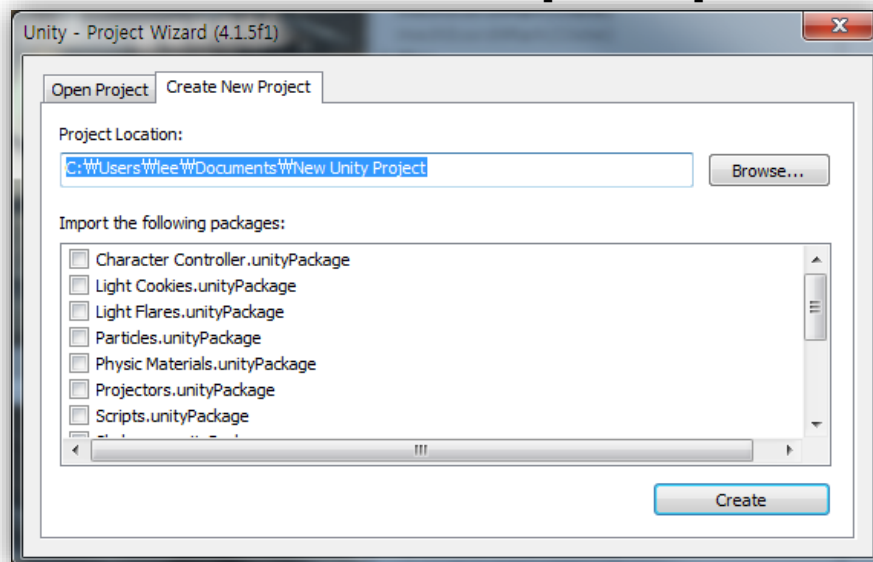
- 유니티 4.6 이후 이동, 회전, 크기 변경을 하나로 수행
- Shift 키 누르고 드래그
 - 종횡비(가로세로비율)를 유지한 채로 확대/축소



2. 게임 오브젝트 추가

1) 새로운 프로젝트 생성

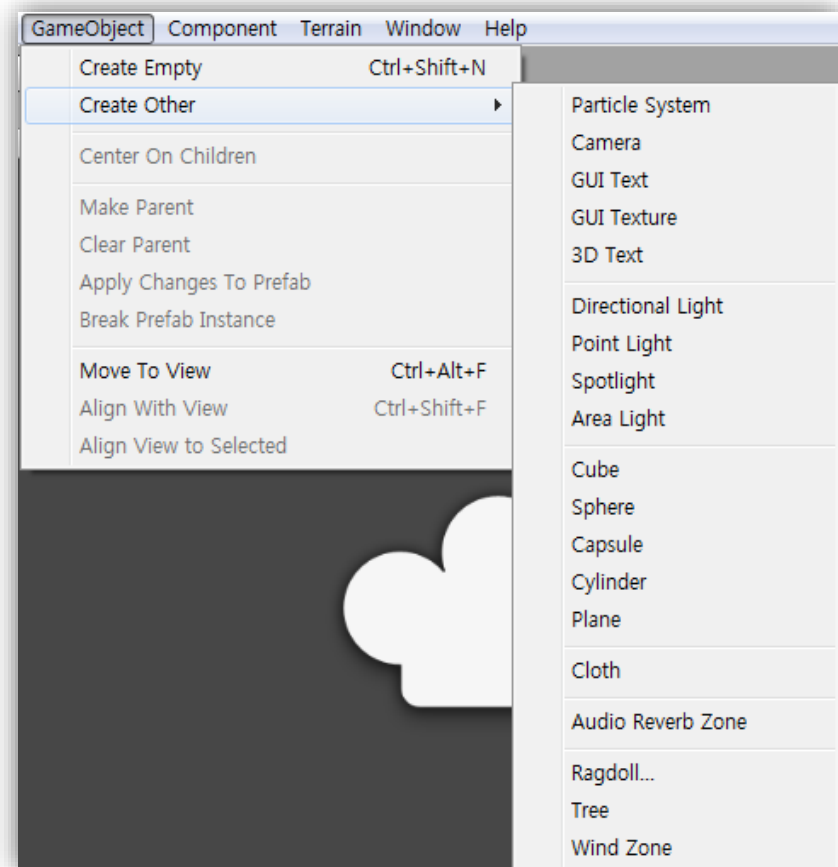
- 유니티는 게임 각각을 프로젝트 단위로 관리
 - [File → New Project] 메뉴 선택
- 한글경로 인식 문제
- 게임에 필요한 기능을 패키지 단위로 제공
 - 빈 프로젝트를 만들고 추후 필요한 패키지를 따로 불러올 수 있음
- 패키지는 아무것도 선택하지 말고 [Create]



2. 게임 오브젝트 추가

2) 게임 오브젝트를 선택하고 추가하기

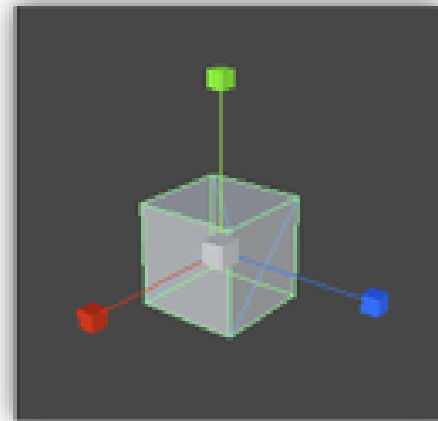
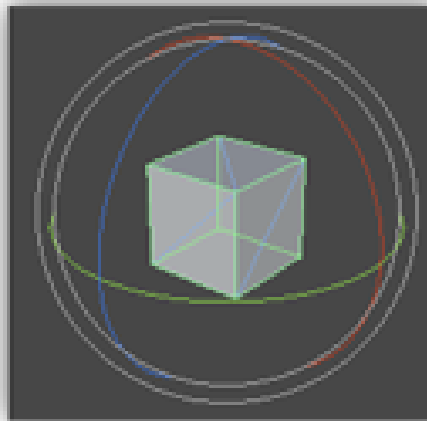
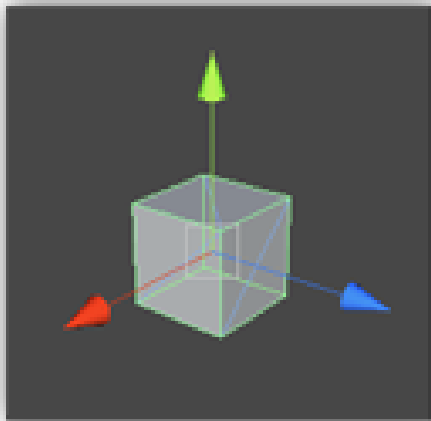
- 게임 오브젝트 [GameObject] → [Create Other]
 - 캐릭터, 총, 카메라, 조명 등
- 삭제
 - 오브젝트 선택 후 [Delete]



3. 실습

1) 오브젝트 선택과 조작

- 이동, 회전, 확대/축소의 기즈모가 표시됨
 - 기즈모(gizmo): 눈에 보이지 않는 요소를 아이콘으로 표시한 것
 - 오브젝트 중앙의 사각형을 드래그 : 임의의 방향 이동/회전 또는 3축 같은 비율 확대/축소



- 오브젝트의 위치와 방향 표시
 - x축: 빨강(red), y축: 초록(green), z축: 파랑(blue)

3. 실습

2) 마우스 버튼 조작

- **왼쪽 버튼**: 게임 오브젝트 선택
- **Ctrl + Alt + 왼쪽버튼 드래그** 또는 **휠 클릭 및 드래그** : 화면 이동.
- **Alt + 우클릭** : 선택한 객체를 기준으로 화면회전 할 경우.
- **오른쪽 버튼**: 현 시점을 기준으로 화면 회전.
- **휠 스크롤**: 화면 확대/축소
- **Alt+마우스 오른쪽 버튼 드래그**: 줌인/줌아웃

3. 실습

3) 시점 변경

- 좌표축 기준

- 물체의 이동, 회전, 확대/축소 시 기준점

[Local]: 오브젝트의 좌표 기준

[Global]: 전체 화면 기준

- 2개 이상의 오브젝트 선택 시

[Pivot]: 맨 나중에 선택한 오브젝트의 좌표 기준

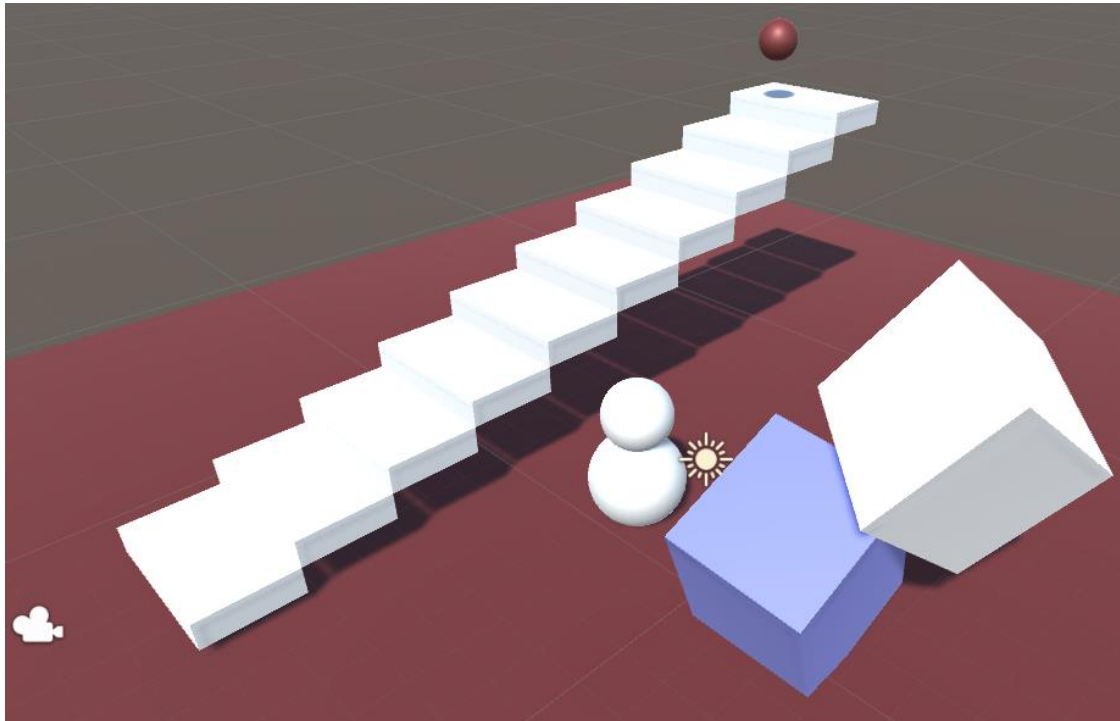
[Center]: 선택된 오브젝트들의 중간 지점 기준

- 사용자 시점 변경

- 기즈모의 x, y, z 원뿔 모양 아이콘 클릭 시 그 방향으로 시점이 전환됨

4. 도형 만들기 실습

- 기본도형 만들기
 - 이동, 회전, 스케일, 위치조정, 오브젝트 직사각형 도구(사형변환 도구)
 - Pivot, Center / Global, Local
 - 기본 매트리얼 적용
- 카메라 조정 하기
- 중력 컴포넌트 주기



5. 석가탑 제작 실습

- 사용된 Materials
 - High Quality Bricks & Walls
 - Yughues Free Ground Materials
- 실습목표
 - 시점 변환 이해하기
 - 에셋이용하기
 - 객체 복제하기
 - 툴 다루기
 - 계층구조 활용하기
 - 텍스처 적용하기
 - 프로젝트 저장
 - 씬 저장
 - 패키지 만들기



1

씬뷰의 조작은 단축키와 마우스로 이동, 회전, 선택을 할 수 있다.

2

유니티 화면 배치(layout)는 개발자의 취향에 따라 변경 할 수 있다.

3

유니티 이전버전을 설치한 경우, 한글 경로는 가능한 사용하지 않는다.

