

02_ 언리얼 에디터에 익숙해지기

<제목 차례>

02_ 언리얼 에디터에 익숙해지기	1
1. 개요	2
2. 언리얼 에디터에 익숙해지기	4
3. 언리얼 에디터 레이아웃 살펴보기	8
4. 뷰포트의 좌표계와 투사 유형 이해하기	11
5. 뷰포트에서 뷰포트 카메라 조작	16

인천대학교 컴퓨터공학부 박종승
무단전재배포금지

1. 개요

이 장에서는 언리얼 에디터에 익숙해지기 위한 학습을 하자.

언리얼 엔진의 기본적인 주요 용어에 대해서 알아보자. 언리얼 에디터, 레벨 에디터, 레벨, 애셋, 액터 등의 용어에 대해서 알아보자.

게임 엔진은 게임 제작을 위한 저작 도구를 의미하는 용어이다. **언리얼 엔진(Unreal engine)**은 대표적인 게임 엔진의 이름이다. 따라서 **언리얼 엔진**은 게임 제작을 위한 각종 저작 도구와 기타 모든 지원을 포함하는 포괄적인 용어이다. 한편, **언리얼 에디터(Unreal editor)**는 **언리얼 엔진**의 저작 도구의 실제적인 소프트웨어를 의미하는 용어이다.

바탕화면에 있는 언리얼 엔진 실행 아이콘을 더블클릭하는 것을 **언리얼 엔진**을 실행하는 것으로 언급하기도 한다. 실제로는 **언리얼 에디터**에 해당하는 실행파일인 **UnrealEditor.exe**를 실행하는 것에 해당한다. **언리얼 에디터**는 **언리얼 엔진**에서 가장 중심적인 중요한 부분이므로 흔히 **언리얼 엔진**과 **언리얼 에디터**를 같은 의미로 섞어서 사용한다. 정확하게는 언리얼 엔진을 실행하는 것이기보다는 언리얼 에디터를 실행하는 것이 더 명확하다. 그러나 **언리얼 에디터**가 **언리얼 엔진**의 일부라고 생각해도 되고 같은 것으로 생각해도 되므로 **언리얼 엔진**과 **언리얼 에디터**를 혼용해서 사용하더라도 이에 민감하게 생각하지 말자.

이제, **레벨 에디터**에 대해서 알아보자. 언리얼 엔진의 실제적인 저작도구 소프트웨어를 **언리얼 에디터**라고 하였다. 그런데, **언리얼 에디터**는 하나의 에디터 이름이 아니라 다양한 기능의 여러 세부적인 에디터를 포함하고 있는 에디터의 집합체를 의미한다. 이 중에서 디폴트로 뜨는 창인 가장 대표적인 에디터가 **레벨 에디터**이다. 우리는 앞으로 **언리얼 에디터**를 실행하는 것을 **레벨 에디터**를 실행하는 것으로 의미할 것이다. 만약 **블루프린트 에디터**와 같이 **레벨 에디터**가 아닌 다른 언리얼 에디터를 언급할 때에는 세부 에디터 명칭을 구체적으로 언급할 것이다.

레벨 에디터에서는 **레벨(level)**을 편집한다. **레벨**이라는 용어에 대해서 알아보자. **레벨**은 게임 제작에서 사용되는 중요한 용어이다. **레벨**은 게임 플레이어가 게임 플레이 시에 접하게 되는 게임 무대를 의미한다. 즉 레벨은 여러 객체를 배치하여 플레이어가 경험하게 될 월드를 정의하는 게임 씬(scene)을 의미한다. **레벨**을 맵(map)이라고도 하는데, 이 두 용어는 동일한 의미의 용어이다. 레벨은 스테이지(stage)나 월드(world)와도 유사한 용어이다.

레벨을 만드는 것은 게임의 무대를 만드는 것으로 게임의 제작에 있어서 매우 중요한 작업이다. 게임 엔진은 레벨 단위로 게임을 실행한다. 하나의 게임은 게임의 스토리에 따라서 필요한 만큼의 여러 레벨로 구성된다. 각 레벨을 제작하는 **레벨 에디터**는 언리얼 에디터에서 가장 중심에 있는 에디터이고 대부분의 게임 제작 작업이 이 **레벨 에디터**에서 수행된다.

언리얼 에디터에서 사용되는 콘텐츠를 **애셋(asset)**이라고 한다. 애셋은 언리얼 에디터 내부에서 제작할 수도 있고 외부의 디지털 콘텐츠 생성 툴(3ds Max, Maya, Blender, Photoshop 등)에서 제작할 수도 있다. 언리얼 에디터에서 애셋을 사용하기 위해서는 언리얼 에디터 내의 **콘텐츠 브라우저** 탭에서 브라우징할 수 있도록 임포트되어야 한다. 콘텐츠 브라우저는 모든 애셋을 관리하는 창이라고 생각하면 된다.

레벨은 게임 세계를 표현하는 객체들의 묶음이다. 레벨에 배치될 수 있는 객체를 **액터(actor)**라고 한다. **콘텐츠 브라우저** 탭에서 레벨에 배치될 수 있는 애셋의 경우에는 간단한 마우스 드래깅을 통해 레벨에 추가할 수 있다. 레벨에 포함된 액터의 목록은 언리얼 에디터 내의 **아웃라이너** 탭에서 나열된다. 레벨에는 동일한 유형의 액터를 여러 개를 추가할 수도 있다.

<참고> 언리얼 에디터 인터페이스에 대한 자세한 내용은 다음의 링크를 참조하자.

<https://docs.unrealengine.com/unreal-editor-interface/>

지금까지 언리얼 엔진의 기본적인 주요 용어에 대해서 알아보았다. 이제, 언리얼 에디터에 익숙해지기 위한 학습을 하자.

2. 언리얼 에디터에 익숙해지기

이 절에서는 언리얼 에디터의 인터페이스를 이해하고 기본적인 조작 방법에 익숙해지도록 한다. 게임 제작을 위해서는 언리얼 에디터에서 많은 작업을 해야 하므로 에디터의 사용자 인터페이스에 익숙해져야 한다.

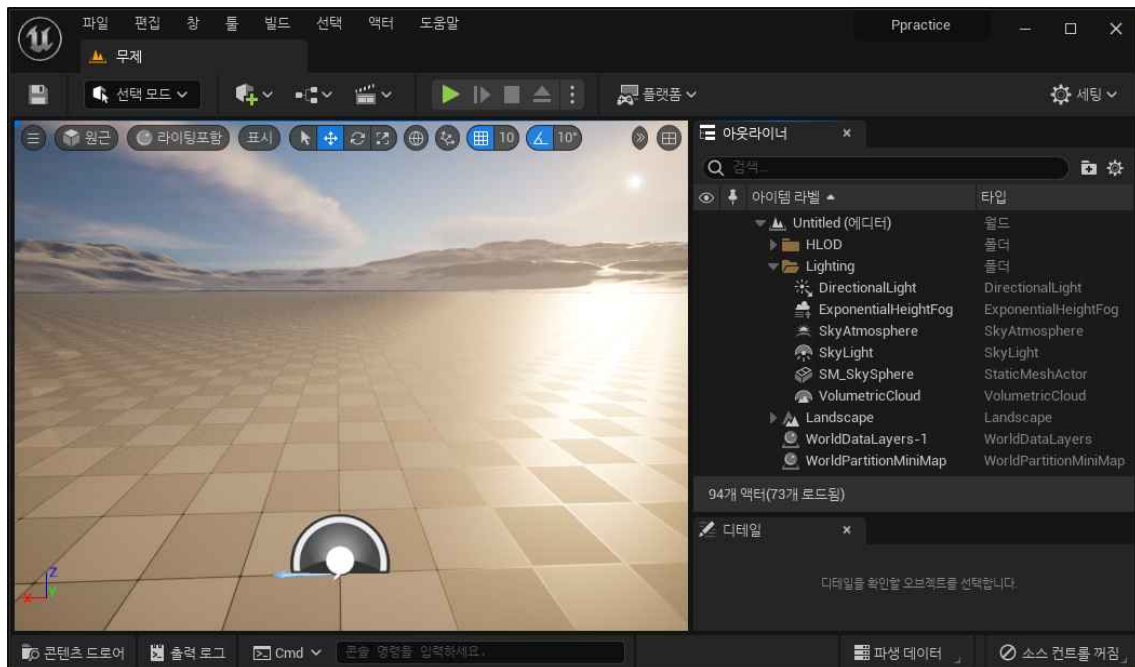
1. 새 언리얼 프로젝트 Ppractice를 생성하자.

이전의 Pfirst와 동일한 방법으로 생성하면 된다.

생성 절차를 다시 설명하면 다음과 같다. 먼저, 바탕화면에 있는 언리얼 엔진 아이콘을 더블클릭하여 언리얼 엔진을 실행하자. 실행하게 되면 먼저 언리얼 프로젝트 브라우저가 보일 것이다. 왼쪽의 게임 탭을 클릭하자. 오른쪽의 템플릿 목록에서 기본 템플릿을 선택하자. 프로젝트 이름은 Ppractice로 입력하고 생성 버튼을 클릭하자.

프로젝트가 생성된 후에는 자동으로 언리얼 에디터 창이 나타날 것이다.

언리얼 에디터는 언리얼 엔진의 메인 창으로 아래와 같은 모습으로 나타날 것이다. 이 화면에 해당하는 창을 언리얼 에디터 중에서 더욱 구체적으로 레벨 에디터라고 한다.



2. 창을 둘러보자. 레벨 에디터 창의 레이아웃의 세부 명칭은 아래의 그림과 같다.



이제부터 창의 각 부분의 명칭과 기능에 대해서 알아보자.

3. 창의 상단에는 **메뉴바**가 있다. 여러 종류의 메뉴를 선택할 수 있다.

메뉴바의 오른쪽에는 현재 로드되어 있는 프로젝트 이름인 **Ppractice**가 표시되어 있다.

4. 메뉴바의 바로 아래에는 메인 **탭바**가 있다.

이 탭바 영역에는 언리얼 에디터의 창을 드래그하여 탭으로 붙일 수 있고 또한 탭을 밖으로 드래그하여 창으로 분리할 수도 있다. 레벨 에디터가 아닌 다른 다양한 종류의 에디터를 탭으로 붙일 수 있다. 탭을 클릭하여 활성 탭을 바꾸면 창의 레이아웃이 활성 탭에 대한 레이아웃으로 바뀐다. 즉 활성 탭이 바뀌면 메뉴바와 툴바를 포함한 그 아래의 모든 탭 배치가 활성 탭에 대한 레이아웃으로 바뀐다.

기본적으로 메인 탭바에는 레벨 에디터 탭이 있다. 탭 제목으로 현재 작업중인 레벨의 이름이 표시된다. 현재 작업중인 레벨에 이름이 아직 부여되어 있지 않은 경우에는 **무제**로 표시된다. 레벨에 이름을 정하고 저장한 후에는 레벨 이름으로 바뀐다.

5. 메인 탭바 아래에는 **툴바**가 있다.

레벨 에디터에 대한 툴바는 많이 사용하는 메뉴나 작업을 빨리 접근할 수 있도록 하는 기능을 제공한다.

툴바에 있는 각 버튼에 대해서 알아보자.

가장 왼쪽에는 **저장** 아이콘이 있다. 클릭하면 현재 작업중인 레벨을 저장한다. 이것은 Ctrl-S를 누르는 것과 같다.

그다음에는 **에디터 모드** 드롭다운 리스트가 있다. 레벨 에디터의 여러 작업 모드 중의 하나를 선택할 수 있도록 해준다. 선택에 따라서 선택된 작업과 관련된 탭이 윈도우의 왼쪽에 추가된다. 에디터 모드에는 **선택**, **랜드스케이프**, **폴리지** 등이 있다. 디폴트로 **선택** 모드로 되어 있다.

선택 모드에서는 액터를 레벨에 배치하는 작업에 대해서 편리한 환경을 제공한다. 액터 배치 탭이나 콘텐츠 브라우저에서 액터를 선택하여 레벨에 배치하여 레벨을 만드는 과정은 **선택** 모드에서 두면 된다. 대부분의 일반적인 경우는 **선택** 모드에 해당한다.

그다음에는 **액터 배치** 아이콘이 있다. 클릭하면 드롭다운 메뉴가 나타나며 액터 배치에 대한 여러 기능을 제공한다. 액터 배치 탭을 에디터에 붙이지 않아도 액터를 배치하는 작업을 바로 수행할 수 있다.

그다음에는 **블루프린트** 아이콘이 있다. 블루프린트 클래스를 생성하거나 레벨 블루프린트를 여는 기능이 있다.

그다음에는 **시네마틱** 아이콘이 있다. 시퀀스를 생성할 수 있다.

그다음에는 **플레이** 아이콘 묶음이 있다. 레벨을 실행하는 기능을 제공한다.

그다음에는 **플랫폼** 아이콘이 있다. 윈도우 외의 안드로이드나 리눅스나 홀로렌즈 등의 다양한 플랫폼에 대한 기능을 제공한다.

가장 오른쪽에는 **세팅** 아이콘이 있다. 월드 세팅, 프로젝트 세팅, 에디터 관련 세팅 등의 기능을 제공한다.

<참고> 툴바에 대해서는 다음의 문서를 참조하자.

<https://docs.unrealengine.com/level-editor-toolbar-in-unreal-engine/>

에디터 모드에 대해서는 다음의 문서를 참조하자.

<https://docs.unrealengine.com/level-editor-modes-in-unreal-engine/>

6. 툴바 아래에는 **뷰포트**가 있다.

뷰포트는 언리얼 에디터에서 만든 월드를 보는 창이다. 뷰포트는 레벨을 만드는데 있어서 매우 중요한 역할을 하는 부분이다. 뷰포트에 대해서는 나중에 상세하게 다룬다.

뷰포트는 창의 레이아웃에 있어서 가장 중심이 되는 영역이다. 뷰포트를 중심으로 상하좌우에 다양한 종류의 창을 탭으로 붙일 수 있다.

7. 뷰포트의 오른쪽에 **아웃라이너** 탭이 있다.

아웃라이너(outliner) 탭은 레벨에 배치된 모든 액터를 계층 트리 뷰로 보여준다.

레벨에서 액터를 선택할 때에 뷰포트에서 마우스로 클릭하여 선택하는 것이 일반적이지만 **아웃라이너**에서 클릭하여 선택해도 된다.

8. **아웃라이너** 탭의 아래에는 **디테일** 탭이 있다.

디테일 탭은 뷰포트에서 현재 선택된 액터에 대한 자세한 정보를 표시한다.

9. 레이아웃의 가장 아래에는 **하단바**가 있다.

하단바에는 가장 왼쪽에 **콘텐츠 드로어**가 있다. **콘텐츠 드로어(Content Drawer)**는 임시로 표시되는 콘텐츠 브라우저이다. 열렸다가 자동으로 닫히는 서랍인 드로어(drawer)처럼 작동한다. 클릭하면 콘텐츠 브라우저가 나타난다. 콘텐츠 브라우저를 상시적으로 표시하지 않고 필요할 때에만 나타나게 하고 평소에는 뷰포트를 넓게 사용하도록 하기 위함이다.

그다음에는 **출력 로그** 아이콘이 있다. 클릭하면 **출력 로그** 탭이 나타난다. 개발 과정에서 디버깅을 위해 유용한 정보를 출력해서 확인할 수 있다.

그다음에는 **콘솔 명령** 입력 상자가 있다. 이것은 에디터가 인식하는 특수 콘솔 명령을 입력할 수 있는 입력 상자이다. 자동 완성 기능이 있어 현재 입력된 내용에 일치하는 모든 명령이 자동으로 나열된다.

그다음에는 **파생 데이터** 아이콘이 있다. **파생 데이터**는 엔진이 애셋 파일(.uasset)을 읽어서 자신의 버전이나 플랫폼에 맞도록 수정하여 별도의 위치에 저장해두고 관리하는 기능을 의미한다. 클릭하면 이와 관련된 통계 데이터를 보여준다.

가장 오른쪽에는 **소스 컨트롤** 아이콘이 있다. 외부의 소스 컨트롤 소프트웨어를 UE 내부에서 사용할 수 있도록 해준다. 클릭하면 소스 컨트롤 접속 설정이나 제출 기능을 수행할 수 있다.

지금까지 레벨 에디터의 디폴트 레이아웃에 대해서 알아보았다.

<참고> 본 교재에서의 용어 사용에 대한 혼란을 방지하기 위해서 다음을 참고해두자. 레벨 에디터 내부에는 뷰포트와 더불어 **아웃라이너**, **디테일** 등의 여러 세부 영역이 있다. 이러한 영역을 ‘패널’이라고도 하고 ‘탭’이라고도 한다. 용어의 일관된 사용을 위해서 우리는 앞으로 ‘탭’으로만 언급할 것이다. 즉 **아웃라이너** 패널, **디테일** 패널 대신에 **아웃라이너** 탭, **디테일** 탭으로 언급할 것이다.

3. 언리얼 에디터 레이아웃 살펴보기

언리얼 에디터 레이아웃에 대해서 더 알아보자.

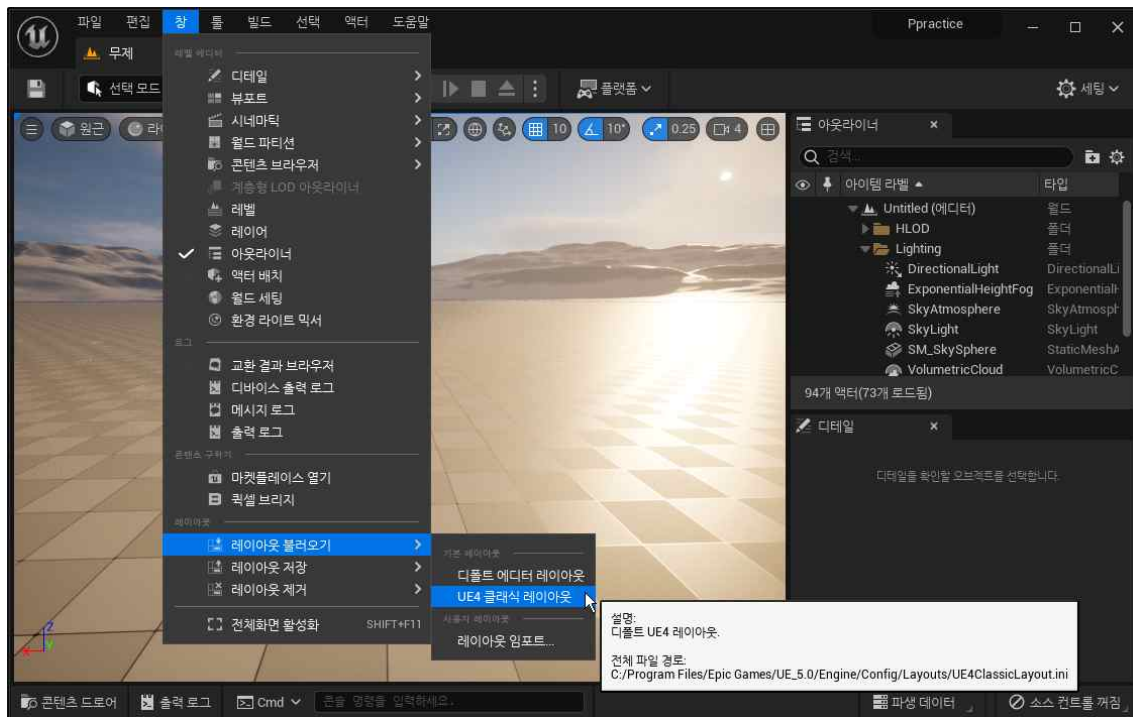
1. 이전 예제의 프로젝트가 있다면 그대로 이어서 계속하자.

이전의 프로젝트가 없다면 동일한 방식으로 새 프로젝트 **Ppractice**를 다시 생성하자. 먼저, 언리얼 엔진을 실행하고 **언리얼 프로젝트 브라우저**에서 왼쪽의 **게임** 탭을 클릭하자. 오른쪽의 템플릿 목록에서 **기본** 템플릿을 선택하자. **프로젝트 이름**은 **Ppractice**로 입력하고 **생성** 버튼을 클릭하자. 프로젝트가 생성되고 언리얼 에디터 창이 뜰 것이다.

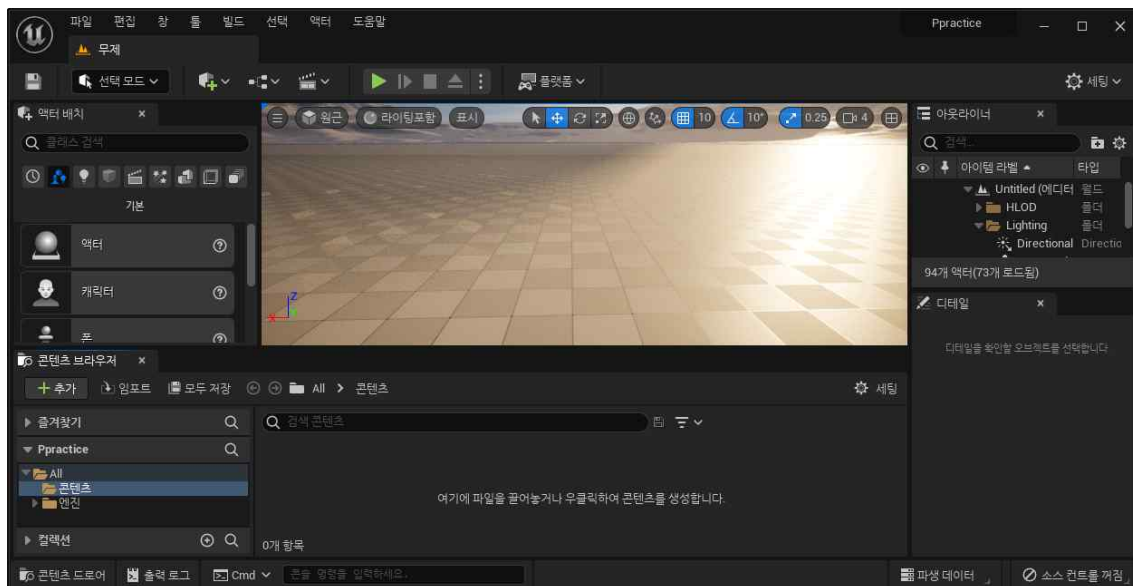
현재의 언리얼 에디터 창의 레이아웃은 언리얼 에디터가 제공하는 **기본 에디터 레이아웃**이다.



2. 언리얼 에디터의 **기본 에디터 레이아웃**에서는 하단에 **콘텐츠 브라우저**가 표시되지 않는다. 언리얼의 이전 버전에 익숙한 사용자라면 UE4의 레이아웃을 더 선호할 수 있다. 현재의 레이아웃을 UE4의 레이아웃으로 바꾸려면 메뉴바에서 **창** » **레이아웃 불러오기** » **UE4 클래식 레이아웃**을 선택하면 된다.



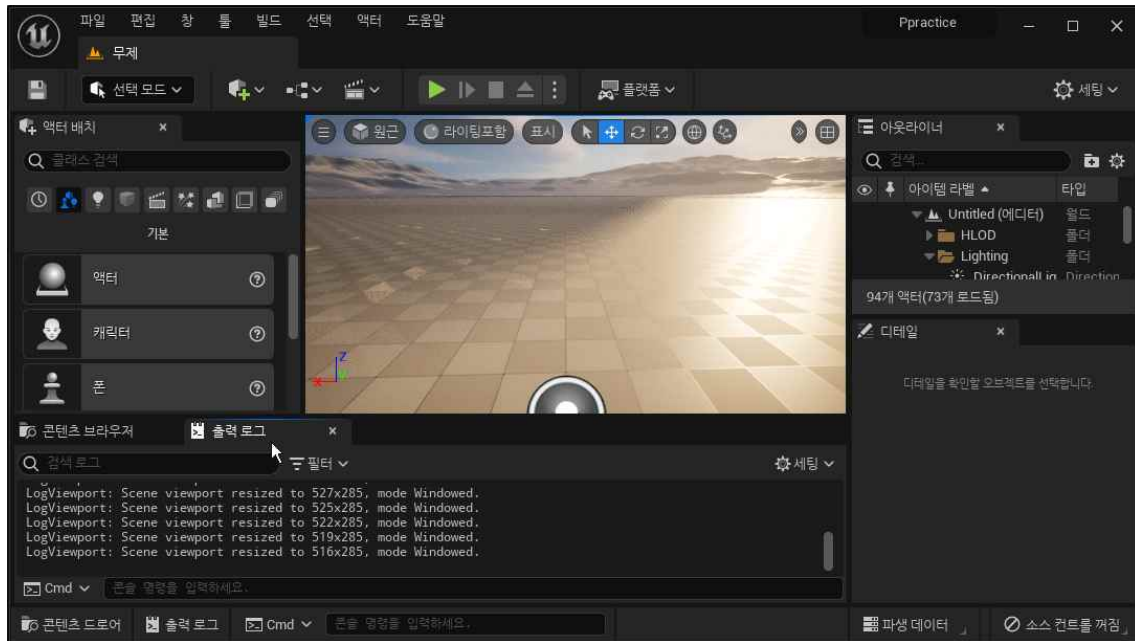
3. 언리얼 에디터가 다음의 모습으로 보일 것이다.



4. 언리얼 에디터는 고도의 커스터마이징이 가능하다.

마우스로 창 내에서 크기와 위치를 마음대로 수정할 수 있다. 또한 추가적인 창을 생성해서 탭으로 붙일 수 있다.

메뉴바에서 **창** 메뉴를 클릭하고 그 아래의 메뉴에서 새로운 창을 생성해보자. 예시로 **창 » 출력 로그** 메뉴를 선택해보자. 새로운 **출력 로그** 창이 나타날 것이다.



창의 **출력 로그** 탭 부분을 마우스로 드래그하여 **레벨 에디터** 위로 가져가자. 마우스 드롭 시에 놓여질 배치가 미리 보여질 것이다. 다양한 위치에 배치될 수 있음을 알 수 있다. 기존의 탭 옆에도 배치해보고 새로운 레이아웃 형태로도 배치해보자.

언리얼 에디터에는 **출력 로그** 창 이외에도 많은 창들이 있다. 메뉴바에서 **창** 메뉴 아래의 **출력 로그** 이외의 다른 메뉴도 선택해서 창을 생성해보자. 그 내용을 알 수는 없더라도 창의 모습을 대략적으로 살펴보자. 그리고 여러 위치에 탭으로 붙여보자.

5. 충분히 연습한 후에는 원래대로 되돌려두자. 메뉴바에서 **창** » **레이아웃 불러오기** » **기본 에디터 레이아웃**을 선택하면 된다.

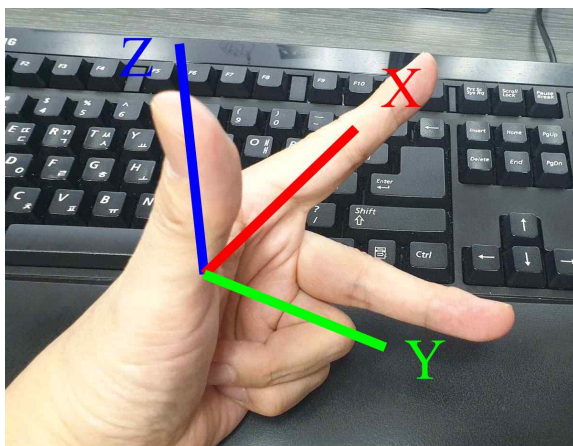
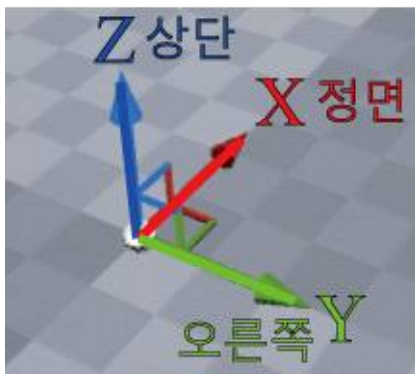
지금까지 언리얼 에디터의 레이아웃 변경에 대해서 알아보았다.

4. 뷰포트의 좌표계와 투사 유형 이해하기

뷰포트는 레벨 제작에 있어서 매우 중요한 부분이다. 뷰포트에 익숙해지도록 연습해보자. 뷰포트의 이해를 위해서는 먼저 뷰포트의 좌표계와 투사 유형에 대해서 알아야 한다.

먼저 언리얼 엔진에서의 좌표계에 대해서 알아보자.

언리얼 엔진은 **왼손 좌표계**를 사용한다. 그리고 위가 Z축(**Z-up**)이고 정면이 X축(**X-front**)인 방식 사용한다. 따라서 엄지, 검지, 중지의 세 손가락을 서로 직각이 되도록 펼치고 엄지가 Z축에 해당하는 하늘 위를 향하도록 하면 검지가 X축인 정면을 가리키고 중지가 Y축인 오른쪽을 가리킨다. 아래의 그림을 참고하자. 언리얼에서는 X,Y,Z축을 각각 빨간색(red), 초록색(green), 파란색(blue)의 세 컬러로 표시한다.



이러한 언리얼 좌표계에 익숙해져야 한다.

<참고> 왼손 좌표계를 사용하는 소프트웨어에는 언리얼 엔진, 유니티, DirectX, Zbrush 등이 있다. 오른손 좌표계를 사용하는 소프트웨어에는 OpenGL, 3ds Max 등이 있다.

한편 유니티는 언리얼과 같은 왼손 좌표계를 사용하지만 위가 Y축(**Y-up**)이고 정면이 Z축(**Z-front**)인 좌표계이다.

지금부터 뷰포트의 좌표계와 투사 유형에 대해서 학습해보자.

1. 이전 예제의 프로젝트가 있다면 그대로 이어서 계속하자.

이전의 프로젝트가 없다면 동일한 방식으로 새 프로젝트 **Ppractice**를 다시 생성하자. 먼저, 언리얼

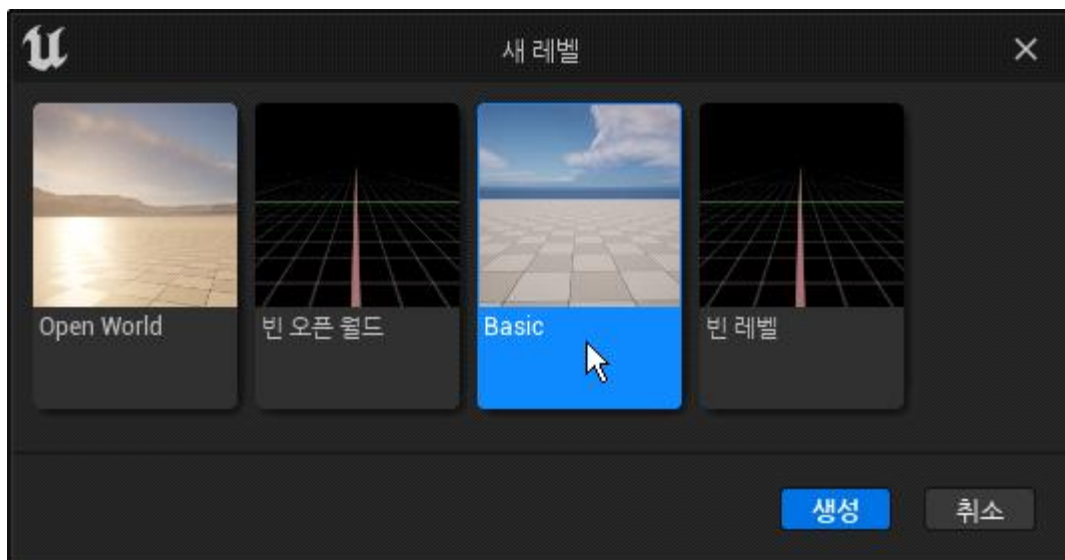
엔진을 실행하고 **언리얼 프로젝트 브라우저**에서 왼쪽의 **게임** 탭을 클릭하자. 오른쪽의 템플릿 목록에서 **기본** 템플릿을 선택하자. **프로젝트 이름**은 **Ppractice**로 입력하고 **생성** 버튼을 클릭하자. 프로젝트가 생성되고 언리얼 에디터 창이 뜰 것이다.



2. 프로젝트가 생성되면 바로 레벨 제작을 시작할 수 있도록 디폴트 레벨을 생성하여 뷰포트에 보여준다. 디폴트로 생성되는 레벨은 **오픈 월드(Open World)** 템플릿의 레벨이다. **오픈 월드**란 자동으로 월드 공간을 격자로 나누고 필요한 셀을 스트리밍하는 UE의 최신 기술이다. 디폴트 템플릿은 이 기술을 적용한 템플릿 레벨이다.

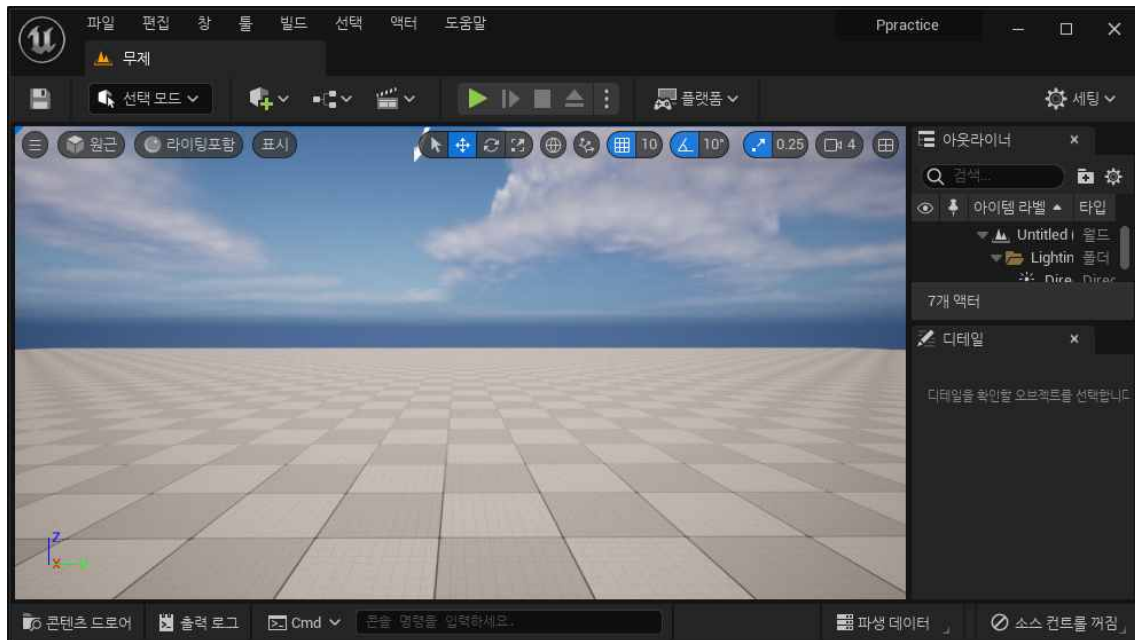
우리는 학습을 위해서 **베이직** 템플릿의 레벨로 바꾸도록 하자.

메뉴바에서 **파일** » **새 레벨**을 선택하자. 아래와 같은 템플릿 선택 창이 뜬다.



레벨의 템플릿은 **오픈 월드**, **빈 오픈 월드**, **베이직**, **빈 레벨**의 네 가지가 있다. **베이직**과 **빈 레벨**은 고전적인 형태이고 **오픈 월드**와 **빈 오픈 월드**는 **베이직**과 **빈 레벨**의 오픈 월드 버전이다. 우리는 **베이직** (Basic)을 선택하고 **생성**을 클릭하자.

3. 이제 베이직 템플릿 레벨이 화면에 보일 것이다.



4. 뷰포트의 왼쪽 아래를 보자. 좌표계가 표시되어 있다.



언리얼에서의 좌표계는 **왼손 좌표계**이고 **Z-up**이고 **X-front**임을 기억하자. 또한 X,Y,Z축이 각각 빨간색, 초록색, 파란색임을 기억하자.

좌표계의 아래에 있는 작은 물음표 아이콘을 클릭하면 뷰포트 조작법 도움말 문서 사이트로 이동한다. 뷰포트 조작법에 대해서는 별도로 자세히 설명한다.

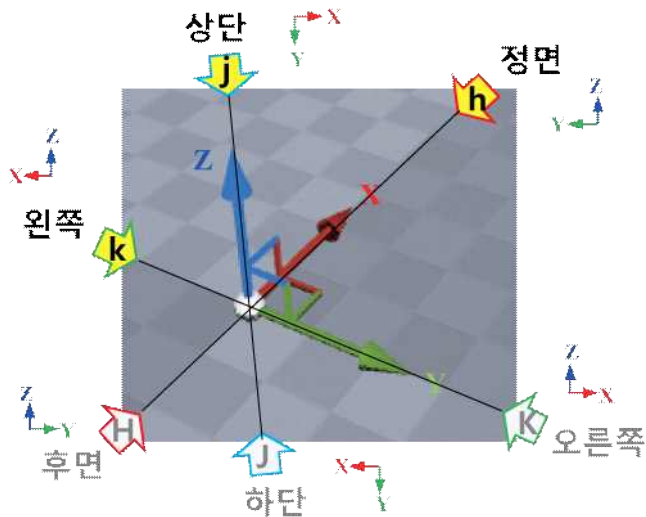
좌표계에 대해서 충분히 익숙해지도록 살펴보자.

5. 뷰포트의 왼쪽 위에 **원근** 아이콘이 있다. 이것을 클릭해보자. 드롭다운 메뉴에서 뷰포트의 투사 유형을 바꿀 수 있다.



뷰포트의 투사(projection) 유형에는 두 종류가 있다. 원근(perspective) 투사와 직교(orthographic) 투사 유형이다. 원근 투사 유형은 게임에서 실제로 3D로 보는 모습과 같은 원근법이 나타나는 방식이다. 직교 투사 유형은 건축 설계도와 같은 도면 작업이 가능한 평면적으로 표현되는 방식이다. 원근 유형이 디폴트로 표시된다.

6. 직교 유형으로 바꾸기 위해서는 6개의 옵션 중에서 하나를 선택하면 된다. 직교 유형의 각각에 대해서 클릭하여 살펴보자. 상단(top)과 하단(bottom)은 Z축 방향의 직교 투사이고, 왼쪽(leftside)과 오른쪽(rightside)은 Y축 방향의 직교 투사이고, 정면(front)과 후면(back)은 X축 방향의 직교 투사이다. 또한, 상단, 오른쪽, 정면은 축의 양의 방향에서 음의 방향으로의 직교 투사이고 그 외의 것들은 그 반대 방향의 직교 투사이다. 아래의 그림을 참조하자.



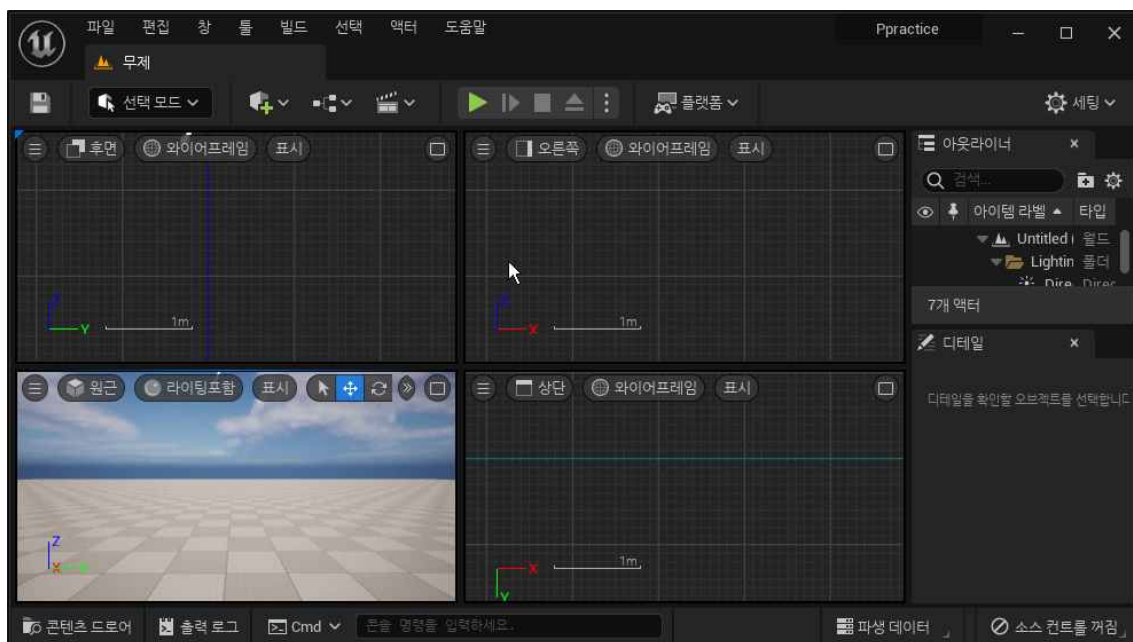
위의 그림에 충분히 익숙해지도록 하자. 각 직교 유형 명칭의 옆에 표시된 2차원적 좌표계 표시는 뷰포트의 유형을 실제로 바꾸었을 때에 뷰포트의 왼쪽 아래에 좌표계 표시가 어떻게 나타나는지를 보여준다.

7. 투사 유형에 선택에 대해서는 단축키가 부여되어 있다. Alt 키를 누른 상태에서 g, h, j, k 키를

누르면 각각 **원근**, **정면**, **상단**, **왼쪽** 투사 유형으로 바뀐다. 보다 쉽게 외우기 위해서는 아래의 그림을 활용하자. 직교 투사 유형에 대한 **h, j, k** 키에 대해서, **Alt** 키와 **Shift** 키를 함께 눌러 대문자로 입력하면 **정면**, **상단**, **왼쪽** 유형 대신 **후면**, **하단**, **오른쪽** 유형이 선택된다.



8. 이제, 뷰포트 레이아웃에 대해서 살펴보자. 디폴트로 뷰포트에서는 하나의 원근 유형만 표시된다. 뷰포트의 오른쪽 가장 위의 모서리에 있는 사각격자 모양의 아이콘을 클릭하자. 뷰포트가 2x2 격자 형태로 바뀌며 4개의 투사 유형이 모두 표시될 것이다. 디폴트로 후면, 오른쪽, 원근, 상단의 유형이 표시된다.



각각의 뷰포트에서 오른쪽 가장 위의 사각형 모양의 아이콘을 클릭하면 그 뷰포트가 최대화된다. 직접 클릭하여 보자.

또한, 각각의 뷰포트에서 왼쪽의 투사 유형을 클릭하고 다른 투사 유형을 선택할 수 있다. 직접 클릭하여 선택해보자.

9. 뷰포트 창을 추가적으로 열 수도 있다. 메뉴바에서 **창 » 뷰포트**를 클릭하면 총 4개의 뷰포트 창을 열 수 있음을 알 수 있다. **뷰포트 1**은 현재 레벨 에디터에서 보이는 뷰포트를 의미한다. **뷰포트 2**를 선택해보자. 새로운 뷰포트 창이 뜰 것이다. 새로운 뷰포트 창에서 투사 유형도 바꾸어보고 레이아웃도 바꾸어보자.

뷰포트 창을 여러개 열더라도 내부적인 모든 데이터는 서로 연동된다. 작업의 편리를 위해서만 중복해서 제공하는 것이다. 한 컴퓨터에서 여러 모니터를 사용하는 경우에 특히 유용한 기능이다. 이러한 기능은 **뷰포트** 창 외에도 **디테일** 창과 **콘텐츠 브라우저** 창에서도 지원된다. 모두 각각 최대 4개의 창을 허용하고 있다.

지금까지 뷰포트의 좌표계와 투사 유형에 대해서 학습하였다.

우리는 앞으로 원근 투사 유형에 대해서는 간단하게 **원근 뷰**로 언급하고 그 외의 6가지의 직교 투사 유형에 대해서는 **직교 뷰**로 언급할 것이다.

5. 뷰포트에서 뷰포트 카메라 조작

뷰포트 카메라는 뷰포트에 보이는 레벨의 모습을 결정한다. 뷰포트 카메라를 조작하는 방법을 학습하자.

앞으로 마우스의 왼쪽 버튼, 오른쪽 버튼, 가운데 휠 버튼 클릭을 **좌클릭**, **우클릭**, **휠클릭**이라고 줄여서 언급할 것이다.

<참고> 뷰포트 컨트롤에 대한 자세한 내용은 다음의 링크를 참조하자.

<https://docs.unrealengine.com/viewport-controls-in-unreal-engine/>

뷰포트에서 뷰포트 카메라를 조작하는 방법을 학습하자.

1. 이전 예제의 프로젝트가 있다면 그대로 이어서 계속하자.

이전의 프로젝트가 없다면 동일한 방식으로 새 프로젝트 **Ppractice**를 다시 생성하자. 먼저, 언리얼 엔진을 실행하고 **언리얼 프로젝트 브라우저**에서 왼쪽의 **게임** 탭을 클릭하자. 오른쪽의 템플릿 목록에서 **기본** 템플릿을 선택하자. **프로젝트 이름**은 **Ppractice**로 입력하고 **생성** 버튼을 클릭하자. 프로젝트가 생성되고 언리얼 에디터 창이 뜰 것이다.

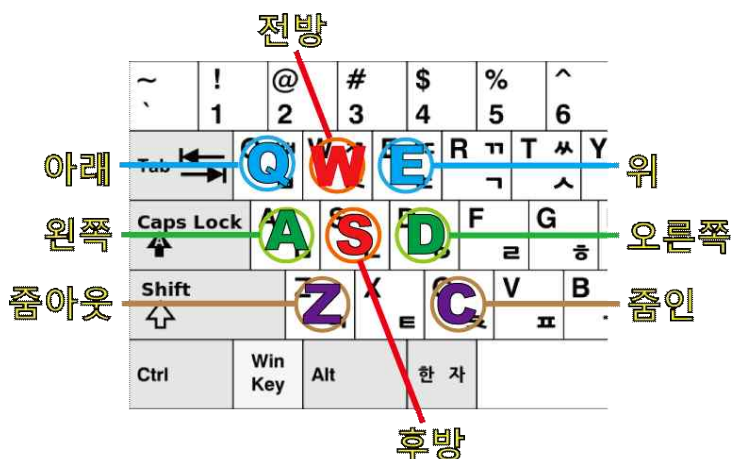
창이 뜨면, 메뉴바에서 **파일 » 새 레벨**을 선택하고 **Basic**을 선택하여 기본 템플릿 레벨을 생성하자.

2. 우클릭 상태에서 키보드 키를 함께 사용하는 뷰포트 카메라 조작 방법에 대해서 알아보자.

- **우클릭+W/S** : 카메라를 전방/후방으로 이동한다.
- **우클릭+A/D** : 카메라를 왼쪽/오른쪽으로 이동한다.
- **우클릭+Q/E** : 카메라를 아래/위로 이동한다.
- **우클릭+Z/C** : 카메라를 줌아웃/줌인 한다.

카메라를 줌아웃/줌인 하는 조작은 카메라를 이동하는 것이 아니라 FOV를 일시적으로 올/내리는 것이다. 우클릭 버튼을 떼면 원래 FOV로 되돌아간다.

매우 중요한 조작이므로 많이 연습해보자. 아래의 그림을 활용하자.



<참고> 실제로는 **우클릭** 대신에 **좌클릭**을 사용해도 동일하게 동작한다. 그러나 공식 문서에서 **우클릭**으로 규정하고 있으므로 **우클릭**을 사용하자.

<참고> 키보드에서 중복해서 정의하고 있다.

키보드의 숫자패드(Numpad) 키로 대체할 수 있도록 정의하고 있다. **W/S**, **A/D**, **Q/E**, **C/Z** 키는 각각 **8/2**, **4/6**, **7/9**, **1/3** 키에 대응한다.

화살표 키로도 대체할 수 있도록 정의하고 있다. **W/S**, **A/D**, **Q/E** 키는 각각 **↑/↓**, **←/→**, **PageDown/PageUp** 키에 대응한다.

3. 마우스만 사용하는 뷰포트 카메라 조작 방법에 대해서 알아보자.

뷰포트에서의 이동 조작은 키는 누르지 않고 마우스 드래그만으로 이루어진다. 이동 조작은 다음과 같다.

- **좌클릭+드래그** : 카메라를 좌우로 회전하고, 전후방으로 수평이동한다.
- **우클릭+드래그** : 카메라를 좌우로 회전하고, 위아래로 회전한다.
- **좌클릭+우클릭+드래그** : 카메라를 좌우로 수평이동하고, 위아래로 수직이동한다. **휠클릭+드래그**도 동일하게 동작한다.

매우 중요한 조작이므로 많이 연습해보자.

4. 키보드 **Alt** 키를 함께 사용하는 뷰포트 카메라 조작 방법에 대해서 알아보자. 이를 3D 저작도구인 마야(Maya)에서 사용하는 조작 방법이라서 마야 스타일 컨트롤이라고도 한다

- **Alt+좌클릭+드래그** : 화면 중앙의 피벗 위치를 중심으로 시선은 피벗을 바라보면서 주변으로 이동한다. 이를 **텀블** 조작이라고 한다.
- **Alt+우클릭+드래그** : 화면 중앙의 피벗 위치 방향으로 전진이나 후진 이동한다. 이를 **돌리** 조작이라고 한다. 마우스를 우측이나 아래로 이동하면 전진 이동하고 좌측이나 위로 이동하면 후진 이동이다.
- **Alt+휠클릭+드래그** : 카메라를 좌우 방향과 상하 방향으로 평행 이동한다. 이를 **트랙**이라고 한다. 먼저 카메라 조작 방법에 관련된 용어인 텀블, 돌리, 트랙 조작 방법에 대해서 이해하자.



- **텀블(tumble)**은 한 피벗(pivot) 또는 관심 지점(POI)을 중심으로 뷰포트를 텀블링시키는 조작으로, 공전(orbit)이라고도 한다.

- **돌리(dolly)**는 한 피벗 또는 관심 지점을 기준으로 카메라를 전후방으로 이동시키는 조작으로, 줌(zoom)이라고도 한다.

- **트랙(track)**은 카메라를 상하좌우 마우스 이동 방향으로 평행 이동하는 조작으로, 팬(pan)이라고도 한다.

매우 중요한 조작이므로 익숙해지도록 충분히 연습해보자.

5. 포커싱에 대해서 알아보자.

뷰포트에서 특정 객체를 선택하고 **F** 키를 눌러보자. 카메라가 선택된 객체의 중심을 적당한 거리에서 바라보도록 카메라를 이동시킨다. 한편, **아웃라이너**에서 해당 객체를 더블클릭해도 동일한 포커싱 기능이 수행된다.

이 포커싱 기능은 여러 경우에 매우 유용하게 사용되는 기능이다. 특히 카메라 **텀블** 회전을 활용하기 위해서는 이 포커싱 키를 꼭 함께 사용해야 한다. 특정 객체의 중심을 피벗으로 정한 후에 텀블 회전하면 그 객체를 중심으로 회전하게 된다. 따라서 객체에 포커싱한 후에 텀블 회전하는 것이

흔히 사용되는 방법이다.

포커싱 후에 텀블 회전을 테스트해 보자. 먼저, 임의의 방향으로 움직여보자. 그다음, 바닥 메시지를 한 번만 클릭하여 선택하자. 그다음, **F** 키를 눌러서 바닥 메시지를 포커싱하자. 그다음, **Alt+좌클릭+드래그**로 텀블 회전해보자. 바닥 메시지 전체를 잘 관찰할 수 있을 것이다.

한편, 포커싱 기능은 단일 물체뿐만 아니라 여러 물체의 그룹을 선택한 후에 포커싱해도 동작한다. 또한, 포커싱 기능은 원근 뷰뿐만 아니라 직교 뷰에서도 동일하게 동작한다.

매우 중요한 조작이므로 익숙해지도록 충분히 연습해보자.

이 절에서는 언리얼 에디터에 익숙해지기 위해서 중요한 뷰포트 조작법에 대해서 학습하였다. 학습한 내용을 다음의 표에 정리하였다. 익숙해질 때까지 연습해보자.

마우스 클릭	추가 조합	조작
우클릭	W/S	전방/후방으로 이동
	A/D	왼쪽/오른쪽으로 이동
	Q/E	아래/위로 이동
	Z/C	줌아웃/줌인
좌클릭	드래그	좌우로 회전, 전후방으로 수평이동
우클릭		회전 (좌우로 회전, 위아래로 회전)
휠클릭		이동 (좌우로 수평이동, 위아래로 수직이동)
좌클릭	Alt+드래그	피벗 위치를 중심으로 피벗을 바라보면서 주변으로 이동
우클릭		피벗 위치 방향으로 전후진 이동
휠클릭		이동 (좌우로 수평이동, 위아래로 수직이동)
객체 선택 후	F	포커싱

지금까지 뷰포트에서 뷰포트 카메라를 조작하는 방법에 대해서 학습하였다.

□