**3D graphics Fundamentals**

알고리즘: 화면에 표현한 것을 보여주기 위한 방법

<게임>

2D 게임 – 2D로 표현된 게임

3D 게임 – 3D로 표현된 게임

* x, y축만 사용한다고 2D게임이라고 할 수 없음
* 3D🡪2D 표현 가능, 그 역은 불가능 ( 없는 정보가 있기 때문)

변환 – 화면에 변환한 것을 보여주는 것 (조명, 그림자 등의 변화도 포함)

<렌더링(Rendering)>

– 3차원 표현을 2차원 표현으로 그리는 것

* 화가의 눈: 카메라
* 화가의 그림: 화면에 보이는 화면
* 화가가 충경을 보면서 그리는 것 == 게임 세계를 모니터에 그리는 것
* 렌더러도 화가처럼 그림 그리는 방법을 이용하여(흉내 내어) 그려야 함 좋은 알고리즘을 이용하여
* 원근을 이용하여 이미지(2D)를 3D로 보이도록 해줌
* 수학적인 표현을 이용해서 그림

어떤 방식으로 overhead를 줄여가면 렌더링을 할 수 있나?

* Data를 정적/ 동적으로 따로 모아두기
* Ex) 그림자/ 조명등 구분하기
* 원하는 걸 골라서 그거만 빠르게 동작 할 수 있음

<객체>

* == 모델 == 메쉬
* 가장 간단한 다각형 == 삼각형
* 삼각형은 선분들의 집합
* 선분은 점들의 집합 🡪 점 == 실수
* 따라서 모델은 점들의 집합, 선분들의 집합, 실수들의 집합이므로
* 벡터들의 집합이다 (점 3개 == 벡터)

<벡터>

스칼라와 덧셈으로 정의 돼있고 닫혀 있으며 연산이 가능한 것

*크기와 방향을 가지고 있는 것*

* 순서쌍들의 집합 🡪 점 표현
* 점들의 집합 🡪 폴리곤 표현
* 폴리곤 표현 🡪 메쉬 표현
* 위치와 방향: 언제든 바뀔 수 있는 것 (애니메이션)

<좌표계>

- 모델 좌표계: 절대로 변하지 않음, 모델마다 가지고 있음

- 월드 좌표계: 게임 세상에서 내가 설정한 원점, 하나만 존재