OpenGL

간단하게 말해보자 **Open Graphics Library**의 약자로 그래픽하드웨어의 기능을 사용하게 해주는 API라고 생각하면 된다. 하드웨어/운영체제와 독립적으로(Windows, OS X, Linux, IOS, Android 등 플랫폼에 구애를 받지 않고 사용할 수 있다.) 사용할 수 있는 인터페이스를 제공한다.

그래픽스는 3D -> 2D 영상으로 만드는 작업인데 제작 단계로는

Modeling-> rigging -> animation-> rendering -> post-processing 으로 진행된다.

이중에서도 animation 에서부터 post-processing 은 runtime에 진행된다.

Offline 에서 artist 들이 작업하는 modeling 혹은 animation과 달리 runtime에 진행되는 과정들은 game application 에서 애니메이션이 replay되는 형태로 진행된다.

어떤 구성으로 되어있을지 알아보자

|  |
| --- |
| **Application** |
| **Game engine** |
| **Graphics API** |
| **GPU** |

위 표와 같이 모든 단계는 자신 아래 있는 단계를 기반으로 실행된다. 따라서 **OpenGL**은 **GPU** 구동을 위한 **API** 이다. 보통 CPU에 저장되어있는 model의 vertices 정보나 normal, texture, lighting 정보들을 GPU에 넘겨주는 역할을 한다고 보면 된다. 그렇게 되면 GPU가 그려내고 결국 output으로 원하는 정보들을 표현해준다!

모델링하고 처리하는 과정에서 그것들이 처리가 되려면 어떠한 과정이 필요한가에 대한 수학적이고 컴퓨터 공학적인 내용을 배운다고 생각한다.

이번에 시작하는 Metanimal팀 논문을 작성하기 위해서는 기본적인 2D가 3D로 가는 작업들을 실제로 수행하기 때문에 vertex shader나 fragment shader는 작성을 해줘야 한다. 실제 데이터를 원하는 결과로 구현하기위한 능력을 함양하려고 한다.

언뜻 보면 이런 생각이 들었다. 왜 내가 이걸 배워야하지 그냥 Unity 쓰면 되는 것 아닌가 싶었다. 하지만 찾아본 결과 어떤 방식으로 그런 것들이 진행되는지 알아야 문제점을 찾고 해결할 수 있을 것이다. 결국 동작되는 방법과 사용하는 방법을 알아야 실력 있는 프로그래머가 될 것이라고 생각된다. 사실 이런 기본적인것을 안 배운다는 건 계산기가 다 해줄 건데 왜 수학을 배워와 같은 느낌인 것 같다.