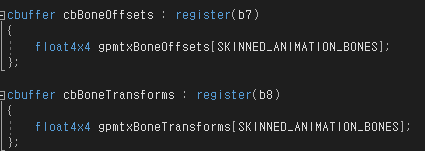
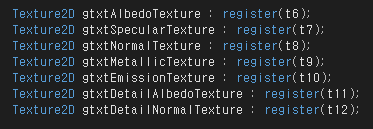
**2018180020 박재우(클라이언트) 10주차 기록**

1. 모델이 포함한 정보를 저장할 객체를 만들어야 한다. (텍스쳐, 머터리얼, 애니메이션 등등..)
2. 모델이 포함한 정보를 읽을 함수를 만들어야 한다.
3. 모델을 렌더링할 때 사용할 쉐이더를 만들어야 한다.
4. 모델이 가지고 있는 리소스를 쉐이더로 넘겨줘야 한다.

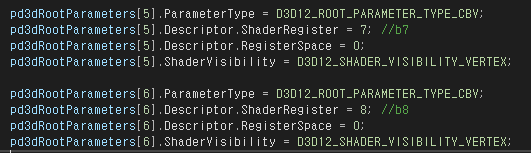
**모델 띄우기 4번**

모델과 쉐이더가 준비되었기 때문에 마지막으로 루트 파라미터를 통해 쉐이더 변수로 값을 넘겨주는 작업이 남았다.

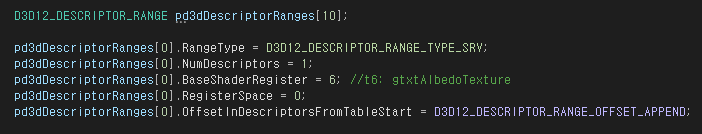
쉐이더가 필요로 하는 변수를 먼저 정리한다.



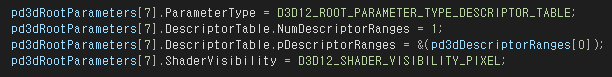
t6~t12 까지의 텍스쳐 변수들과 b7과 b8의 본들의 정보들을 루트 파라미터에 추가해야한다. t6~t12는 DescriptorRange를 이용한다.



우선 b6과 b8은 cbuffer이기 때문에 타입은 CBV다. 해당 데이터는 버텍스 쉐이더에서만 접근하는 데이터 이므로 ShaderVisibility는 VERTEX로 설정한다.



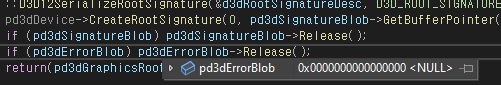
남은 텍스쳐들은 DescriptorRange로 선언하고 위와 같이 설정해준다. BaseShaderRegister만 t값에 맞게 바꿔 값을 넣어준다.



만들어준 DescriptorRange를 이용해 루트 파라미터에 값을 넣어준다.

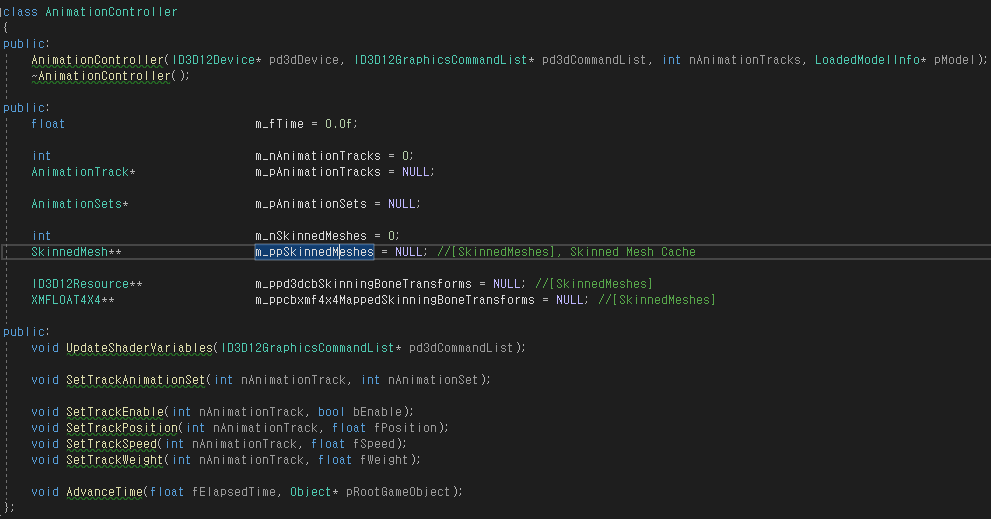
타입은 DESCRIPTOR TABLE이고 픽셀 쉐이더에서만 접근하는 데이터 이기 때문에 ShaderVisibility는 PIXEL로 한다. DescriptorTable의 NumDescriptorRanges는 모든 DescriptorRange마다 하나씩 루트파라미터에 넣어주기 때문에 1로 설정하고 pDescriptor Ranges는 위에서 만든 DescriporRange의 주소 값이므로 각 데이터에 맞는 값을 넣어준다.

값을 다 넣었으므로 우선 빌드가 정상적으로 잘 되는지 본다.

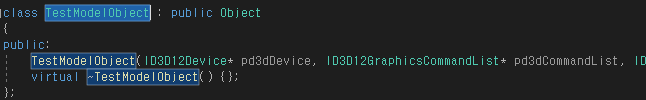


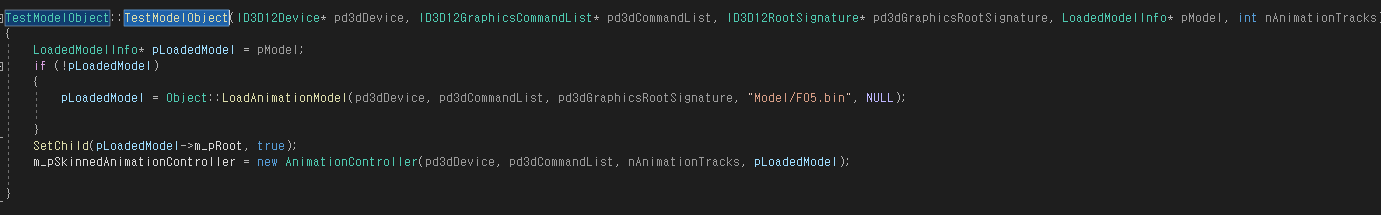
RootSignature를 만드는 부분의 마지막에 ErrorBlob이 NULL인 것을 보니 RootSignature는 정상적으로 잘 만들어 졌다고 생각 할 수 있다.

잊어버린 부분이 있다. 애니메이션을 관리하는 AnimationController가 필요했다. AnimationSet은 애니메이션이 진행되었을 때의 Mesh의 위치 데이터만 갖고 있을 뿐이지 실제 Mesh를 가지지 않는다. AnimationController가 Mesh의 포인터와 AnimationSet을 이용해 실제 애니메이션을 재생하게 된다.



Track으로 위치 속도 가중치를 조절할 수 있다.

이제 테스트 용도의 오브젝트 클래스를 만들어 주고 생성자에서 읽은 모델의 Root오브젝트를 Child로 설정한다.

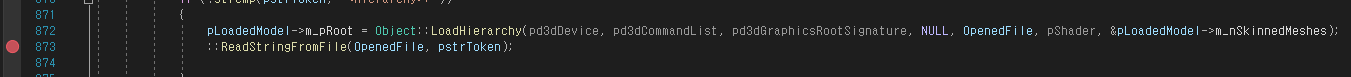


모델의 데이터를 넘겨주긴 하지만 만약을 위해 if문으로 모델이 넘어오지 않았을 때를 대비해준다. 마지막으로 AnimationController를 생성해준다.



GameScene의 BuildObject에서 파일을 읽고 객체를 생성한다. 여기서 파일을 잘 읽을 수 있는지 확인하기 위해 중단점을 찍고 실행해 본다.

실행해본 결과 LoadAnimationModel 함수 어딘가에서 무한 루프를 돌고 있다.



안쪽으로 들어가보니 LoadHierarcy 함수에 문제가 생긴 모양이다.

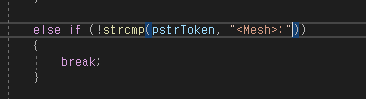
더 안쪽으로 들어가며 찾아보니 LoadMeshFromFile 함수에서 무한 루프를 돌고 있는 것을 발견했다.

왜 문제가 되는 것인가 디버깅하며 한 줄 한 줄 실행 해 봤다.

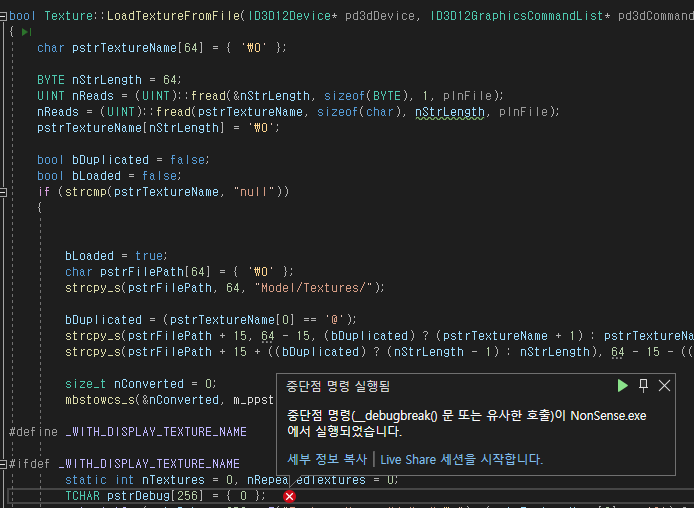


원인은 if문의 중괄호가 꼬여 있었다. TextureCoords0 이후의 문구가 전부 이 if문 안에 들어있었다. 때문에 다음 문구인 TextureCoords1이 나와도 찾지 못하고 무한으로 돌고 있던 것이다.

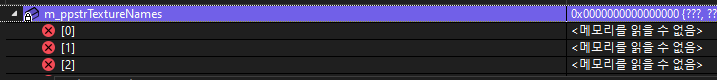
그리고 또 다른 문제점



마지막에 끝나는 문구는 </Mesh> 이지만 <Mesh>: 로 해버리는 어이없는 실수를 했다. 복사 붙여넣기를 하다 보니 놓친 모양이다.



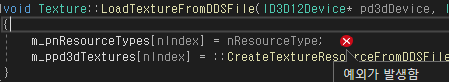
문구의 오류를 고치고 나니 LoadTextureFromFile에서 오류가 발생하여 프로그램이 꺼졌다.



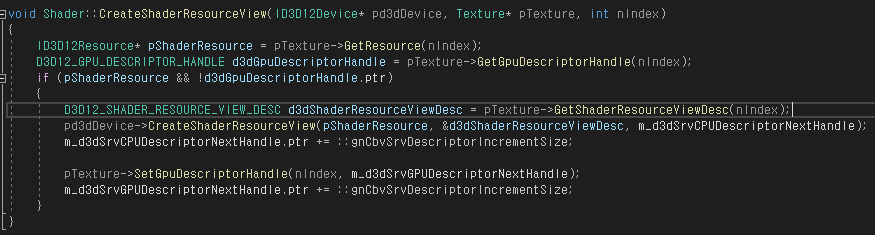
m\_ppstrTextureNames가 전부 NULL이었던 모양이다.



Texture의 생성자에서 m\_ppstrTextureNames를 초기화 해줬다.

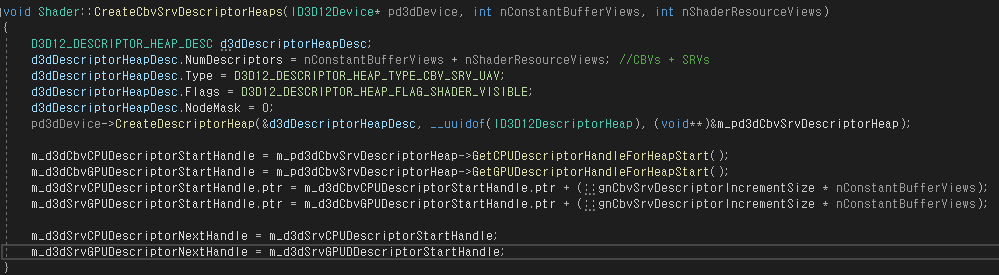


이번엔 m\_pnResourceTypes가 NULL인 것 같다. 여러 군데 구멍이 많은 모양이다. 디버깅하면서 전부 찾아내야한다.

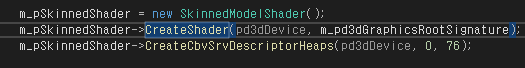


쉐이더에서 Descriptor Heap을 만들어 주지 않고 있었다. 이 때문에

CreateShaderResourceView함수에서 m\_d3dSrvCPUDescriptorNextHandle이 NULL이어서 프로그램이 중단됐다.

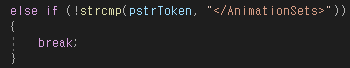


CreateCbvSrvDescriptorHeaps 함수를 만들었고



GameScene의 BuildObject 함수에서 쉐이더를 만든 후 호출해 주었다. ShaderResourceView의 수를 3번째 인자를 넣어줬는데 예제에서는 76을 넣었다. 왜 76인지는 아직 공부가 부족하다. 76보다 더 큰 수를 넣어도 문제는 없지만 작은 수를 넣었을 때 문제가 생겼던 기억이 있다. 이에 관해 더 공부를 해야 한다.

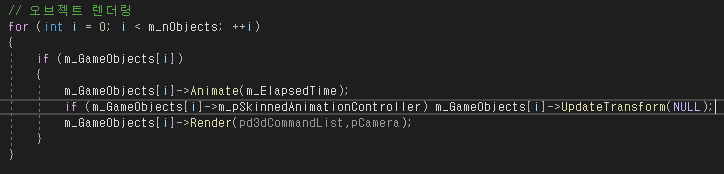
Hierarchy함수가 끝나고 Animation을 읽는 함수에서도 어김없이 무한 루프를 돌고 있다. 원인은 </AnimationSets>로 구문의 끝을 구분하여 break를 해줘야 하는데 이 부분이 빠져 있었다.



위의 코드를 추가하여 실행을 해보니



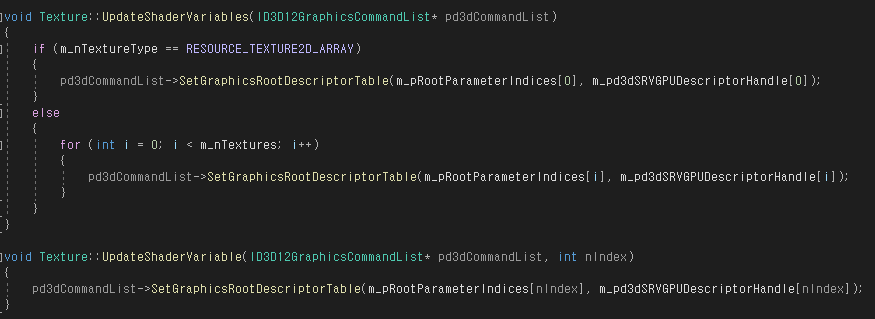
정상적으로 프로그램이 실행되었다. 아직 렌더링 관련한 처리를 해주지 않아 아무것도 그려지지 않는 게 당연하고 문제없이 윈도우창이 뜬 걸 보니 파일을 읽는 기능은 잘 작동한다고 생각된다. 이제 렌더링을 해보고 문제없이 모델이 잘 그려지는지 확인하는 것만 남았다.



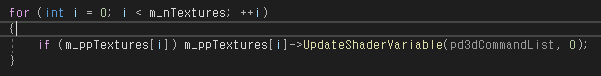
렌더링의 과정은 위와 같다. 애니메이션을 진행한 후 만약 애니메이션 컨트롤러가 있다면 Transform을 갱신해준다. 그리고 마지막에 렌더링을 해주는데 아직 저 함수들은 최상위 부모의 함수들로 비어 있는 상태이다. Render함수는 작동은 하지만 우리가 렌더링할 오브젝트는 지금까지 만든 SkinnedMesh를 이용해 렌더링 하기 때문에 관련된 함수들도 전부 수정해 주어야한다.

Material과 관련된 쉐이더 변수들을 넘겨주는 UpdateShaderVariable 함수를 만들려고 했으나 이미 Material의 데이터를 Scene에서 넘겨주고 있었다. 일단 패스

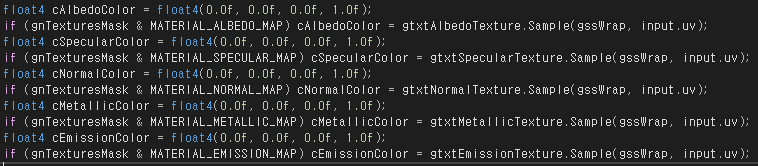
Material에는 Texture의 관한 데이터도 포함되어 있다. Texture를 쉐이더로 넘겨주기 위한 함수도 만든다.



Texture를 읽는 과정에서 얻은 RootParameterIndices와 SRVGPUDescriptorHandle을 이용해 값을 넘긴다.



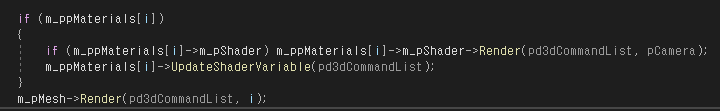
만든 함수를 호출해준다. Texture는 여럿 있으므로 각 Texture마다 함수를 호출한다.



쉐이더를 보면 TexturesMask의 값에 따라 처리해주는 계산도 다르다. 우리는 이 값을 넘겨줘야 하는데 이 값이 Material의 타입이다.



타입도 마찬가지로 3번 인덱스의 루트 파라미터에 넘겨준다.



Material의 쉐이더에서 Render 함수를 불러준다. 쉐이더의 Render에서는 GraphicsPipeLine을 Set해준다. 그 후 Material의 데이터를 쉐이더에 넘겨주기 위해 UpdateShaderVariable 함수를 호출한다. 마지막으로 Mesh의 Render를 호출해서 Mesh를 그려준다.

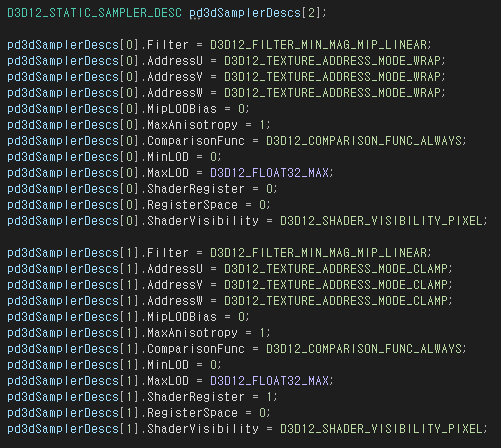


마지막으로 Object의 Render 함수에서 자식과 형제의 Render 함수를 호출해준다.

이제 실행해보고 잘 그려지는 지 확인한다.

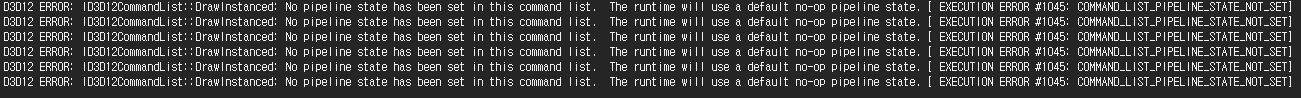


오류가 발생했다. Sampler Descriptor를 생성해주지 않았다.

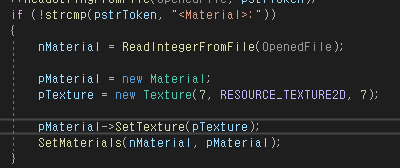




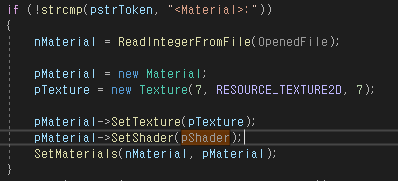
위와 같이 Smapler를 생성해준다.



두번째 오류이다. 쉐이더가 제대로 설정이 되지 않은 것 같다.

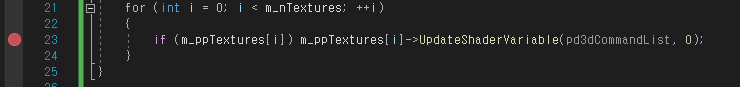


이 부분에서 Texture를 생성하고 Set 했지만 정작 Shader를 설정해주지 않았다.



위와 같이 SetShader를 호출해 줬다.

결과 : 아무것도 그려지지 않았다. 눈에 보이는 오류는 없지만 쉐이더에 값이 제대로 들어가지 않았거나 모델이 잘 불러지지 않았거나 하는 문제가 있을 수 있다.



해당 위치에 중단점을 찍어 봤지만 전혀 들어오지 않는다. m\_nTextures 변수에 문제가 있다.

코드를 다시 읽어보니 m\_nTextres는 Material의 멤버 변수이고 내가 읽은 Texture의 개수는 Texture 클래스 안에 있으므로 m\_pTexture를 이용해 쉐이더 변수를 업데이트 해줘야 한다.