|  |
| --- |
| 3DGP |
| 과제 3 |
| Rollercoaster Application |

|  |
| --- |
| 후안  2016180046 |

# 구현한 기능

|  |  |  |
| --- | --- | --- |
| **Category** | **Feature** | **Details** |
| HLSL | Instancing Shades하고, Diffused Shaders를 파일 따로 분리 | VSInstancing, PSInstancing “InstancingShader.hlsl”에 넣고 VSDiffused, PSDiffused는 “DiffusedShader.hlsl”에 넣었습니다. |
| Include  클래스  (ID3DInclude의 Child 클래스 생성) | Shader들이 파일 따로 했지만 VSDiffused하고 VSInstancing Vertex Shader들이 둘다 Root Signature가 정한 Constant Buffer를 필요합니다. 그래서 중복성을 없애기 위해서 Constant Buffer정의를 “CBStruct.hlsl” 파일에 넣고 InstancingShader와 DiffusedShader HLSL 파일들이 둘다 CBStruct HLSL 파일 Include합니다. Shader Compile할때 ShaderInclude 클래서를 통해서CBStruct.hlsl Compile데이터에 include할 수 있습니다. |
| Instancing | Column Instancing | 정해진 시간마다 레일 생성하면서 Instancing 기능을 쓰는 Column Object를 생성해서 Terrain이나 y = 0 플레인에 붙이는 겁니다. 이런식으로 해서 실제 물리적인 Rollercoaster 느낌이 납니다. |

# Controls

(Rollercoaster Scene의 Process Input 함수에서 처리)

|  |  |
| --- | --- |
| **Key** | **Action** |
| ‘W’ | 생성할Rail들이 Pitch 양수 회전을 해서 생성합니다. |
| ‘S’ | 생성할Rail들이 Pitch 음수 회전을 해서 생성합니다 |
| ‘D’ | 생성할 Rail들이 Yaw 양수 회전을 해서 생성합니다 |
| ‘A’ | 생성할 Rail들이 Yaw 음수 회전을 해서 생성합니다 |
| 'Q' | 생성할 Rail들이 Roll 양수 회전을 해서 생성합니다 |
| 'E' | 생성할 Rail들이 Roll 음수 회전을 해서 생성합니다 |
| Spacebar | Spacebar 계속 누르고 있으면 게임 시간이 10배로 더 느리게 됩니다. Spacebar release하면 게임 원래 속도로 됩니다. |
| F1 | 1인칭 카메라. 이 카메라는 Player 회전에 따라서 회전을 합니다. |
| F2 | 3인칭 카메라. 이 카메라는 Player의 위치를 따라하지만 Player의 회전 정보 상관없고 Player위치만 바라보는 겁니다. |
| F3 | 3인칭 카메라. 이 카메라는 Player 회전에 따라서 회전을 합니다. |
| F4 | 위치를 고정되어 있는 카메라. 이 카메라로 계속 Wagon을 보이지만 위치를 안 바꿔서 Rollercoaster를 멀리 모습을 보일 수 있습니다. |

# Framework 구조 설명

Scene에는 Shader만 있는 이유는 Shader 순서 없이 각 Object에 설정하면 SetGraphicsPipeline 많이 호출할 거라 성능 떨어질 수가 있어서 Shader으로 하면 SetGraphicsPipeline 몇번만 호출하기 때문입니다. 따라서, Instancing할 거와 Instancing 안 할 옵젝트들을 따로 Shader들에 분리했습니다. 예를 들면 Tree Instance는 Tree Shader, Rail Instance들이 Rail Shader, Cube Instance들 Cube Instance 있습니다. 이런 Shader들이 하나로 합칠 수가 있었지만 Rail이 특별한 변수와 함수들이 있어서 따로 분리했습니다.

# Mesh

# 소감

# Gameplay Screenshots

Camera Mode F1 (1인칭 카메라, 플레이어 회전 의존성이 있습니다)

Camera Mode F2 (Orbital Camera)

Camera Mode 3 (3인칭 카메라)

Camera Mode 4 (위치 고정된 카메라)