|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **주차** | 1 주차 | **기간** | 6.24 ~ 7.2 | **지도교수** | (서명) |
| 이번주 한일 요약 | 언리얼 c++ 리플렉션/ 델리게이트/ 인터페이스 | | | | |

<상세 수행내용>

언리얼 엔진에서의 c++에는 생각보다 다른 부분들이 많아서 언리얼 도큐먼트에서 참고하면서 언리얼c++에서 있는 기능들을 공부했습니다. 해당 공부를 하면서 문서에 있는 글만 읽어서는 부족한 부분들이 있어서 언리얼 프로젝트를 하나만들어서 해당 기능들을 사용해서 log를 찍어보면서 어떻게 활용하는지 공부했습니다. 실제로 델리게이트를 활용하면 보스몬스터와 그 보스몬스터가 생성한 부하몬스터가 있을때 보스몬스터가 죽으면 부하몬스터에 메시지를 보내서 같이 소멸하도록 만들수 있습니다. 이를 log로 찍어서 만들어 보았습니다. 리플렉션은 "모듈명".generated.h” 이라는 헤더파일이 class를 만들때 추가 되어있는데 열거형UENUM(), 클래스UCLASS(), 구조체USTRUCT(), 함수UFUNCTION(), 멤버 변수UPROPERTY() 라는 매크로 함수를 위에 선언해 주면

모듈명".generated.h에 기록되고 이를 활용할수 있습니다.

<https://docs.unrealengine.com/5.2/ko/reflection-system-in-unreal-engine/>

https://www.stevestreeting.com/2020/11/02/ue4-c---interfaces---hints-n-tips/

|  |  |  |  |
| --- | --- | --- | --- |
| **문제점 정리** | X | | |
| **해결방안** | X | | |
| **다음주차** | 2 | **다음기간** | 7.03 ~ 7.10 |
| **다음주 할일** | 언리얼 컨테이너, 직렬화, 패키지/에셋 | | |
| **지도 교수**  **Comment** |  | | |