

# Day 21.

## 몬스터 애니메이션/공격 구현

### 21-1. Skeleton 애니메이션

Skeleton의 애니메이션에 대한 구현은 캐릭터 카타나의 전례가 있어서 어렵지는 않았습니다. 캐릭터들의 ABP 구조와 유사하게 공격이나 Hit, Death 등은 Montage로 묶어서 처리하면 되었고 공격의 설계나 충돌처리도 카타나의 근접공격들과 비슷하게 처리하면 됐습니다. Montage와 애니메이션 시퀀스들에 대한 간단한 리깅과 Root처리만 하고 Montage를 등록하여 애니메이션과 공격을 구현하였습니다.

### 21-2. Dragon 애니메이션



Dragon도 전체적인 구조는 Skeleton과 비슷하게 짜면 됐는데 아무래도 강력한 몬스터의 포지션에 위치하다보니 여러 가지 공격 패턴이나 비주얼이 필요했습니다. 랜덤 패턴을 만들거나 BehaviorTree Task를 만들기에 앞서서 다양한 공격 패턴들을 만들 필요가 있었는데 근접 공격의 경우는 애니메이션 시퀀스들을 섞어 공격 Montage를 만들었고 이 공격들에 피격당할 경우 캐릭터의 charactermovemnt에 launch와 관련된 벡터 값을 주어 넉백 당하도록 구현했습니다. 원거리 공격이 문제였는데 머리를 전방으로 돌리며 브레스를 내뿜는 패턴의 경우 애니메이션은 있지만 45도 각도로 지속적으로 브레스를 내뿜는 것을 표현할 마땅한 이펙트가 없어서 공격 애니메이션의 특정 타이밍마다 Animnotify를 줘서 기존의 브레스 이펙트를 계속 재활용하는 방식으로 구현하였습니다. 그리고 파이어볼을 구현하는 것도 마법사 캐릭터처럼 마법구체 액터를 Spawn하는 방식으로 하려고 했으나 입안을 Socket으로 할 경우 구체의 Owner와의 충돌을 무시하는 속성을 주더라도 특정 상황에서 구체가 바로 충돌해서 사라져버릴 상황이 몇몇 존재했습니다. 그래서 해결법을 찾아 보았고 FActorSpawnParameters 값에 .SpawnCollisionHandlingOverride = ESpawnActorCollisionHandlingMethod::AdjustIfPossibleButAlwaysSpawn 속성을 주어 가능한 충돌을 피하기 위해 위치를 조정하고 무조건 액터가 스폰되는 속성을 주는 방식으로 해결하였습니다.