**문서화**

* **목차**
* **HUD**
* **로직 구현 초기작업**
* **게임 로직**
* **적 생성**
* **Wave 시작**
* **사망 처리**
* **레벨 블루프린트**
* **마을 파괴 처리**

**이름 : 박동원**

**학번 : 201101738**

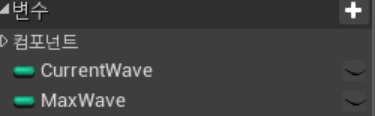
**담당 교수 : 박종승 교수님**

**과목 : 프로젝트 실무(2)**

1. **웨이브를 세기 위한 변수 만들기**

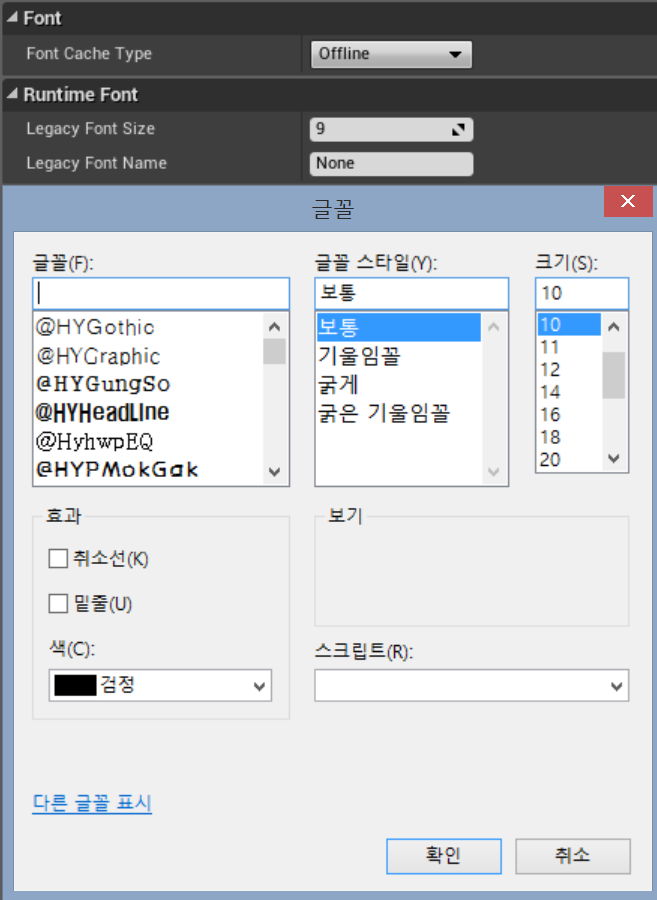
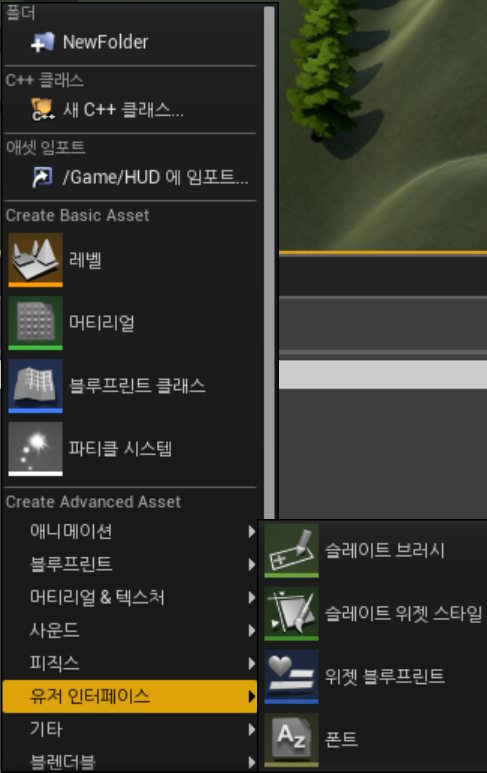
* **게임 모드 클래스의 변수**

**MaxWave, CurrentWave 생성**



1. **HUD용 애셋 만들기**

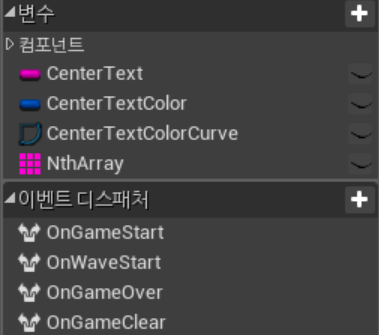
* **폰트**



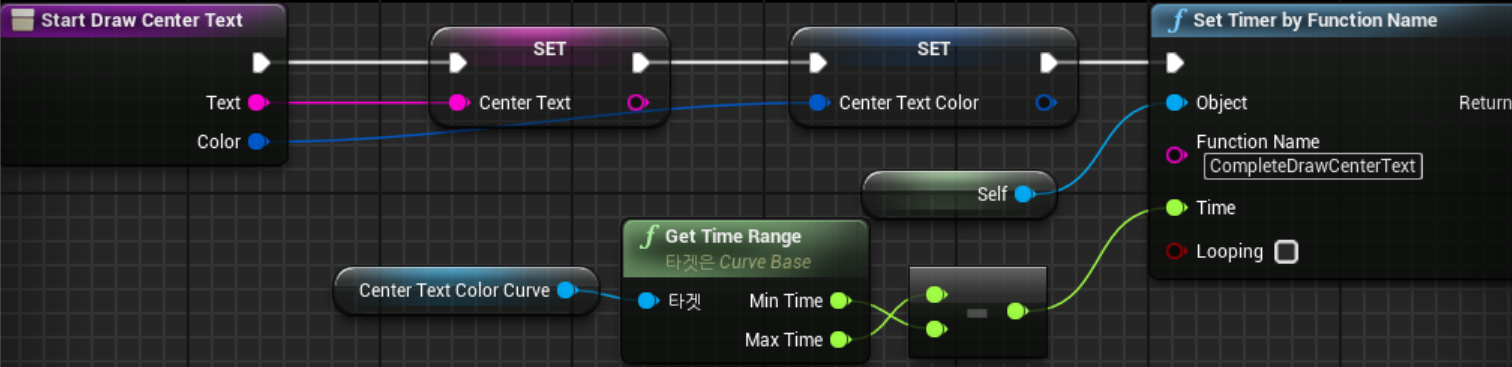
* **본인이 원하는 글꼴을 설정하여 Wave에 해당하는 폰트를 설정합니다.**

1. **HUD 클래스 만들기**

* **사용 하게 될 각종 변수들 추가**



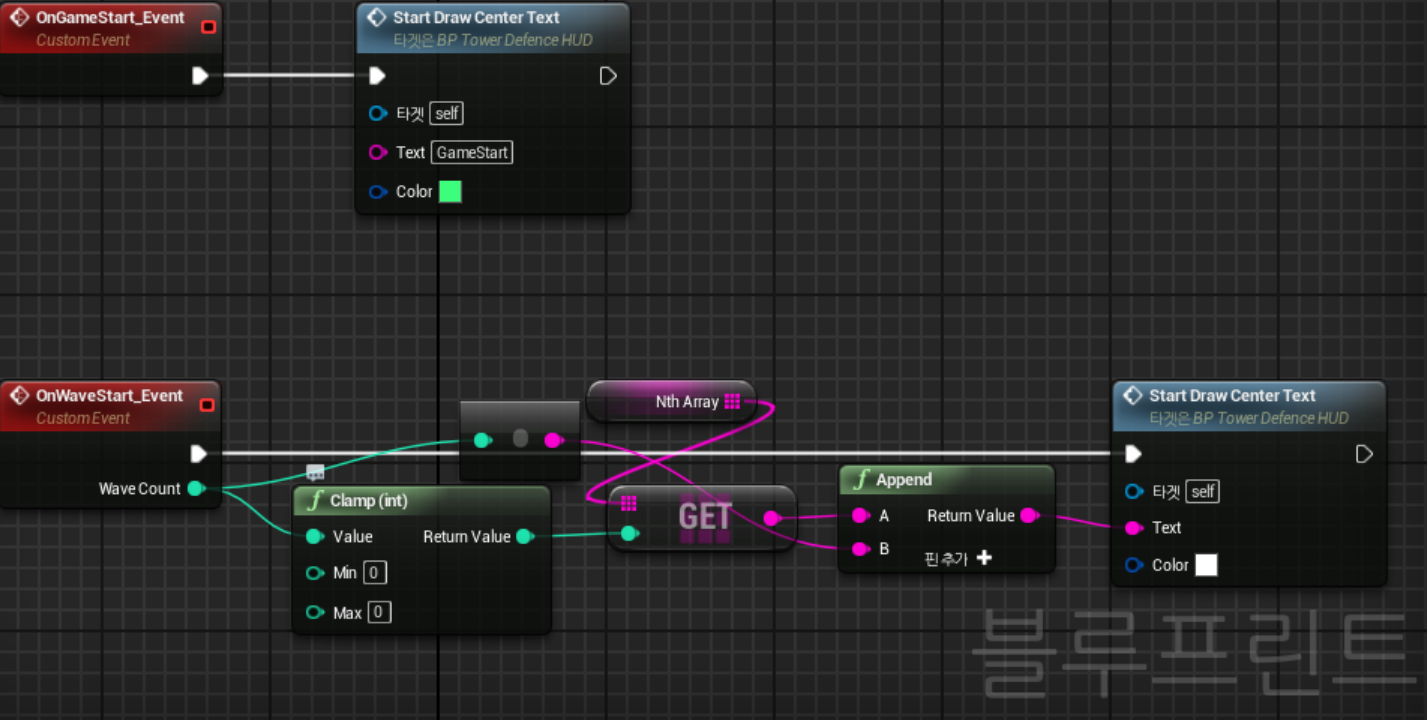
1. **HUD 클래스에 함수 추가하기**



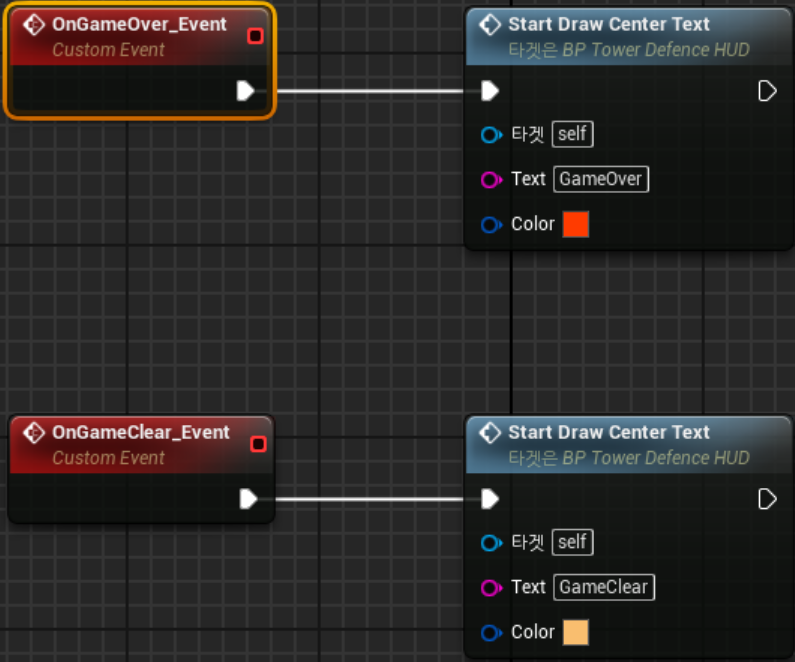
* **화면 중앙에 텍스트 출력이 시작될 때 처리하는 함수.**

**Set (Text,Color)을 하여 타이머 함수에 전달.**

1. **HUD 클래스에 이벤트 그래프 설정하기**



* **Game, Wave가 시작이 된다면 위에서 미리 만들어 두었던 스타트 함수를 사용하여 각각의 이벤트 처리를 수행.**

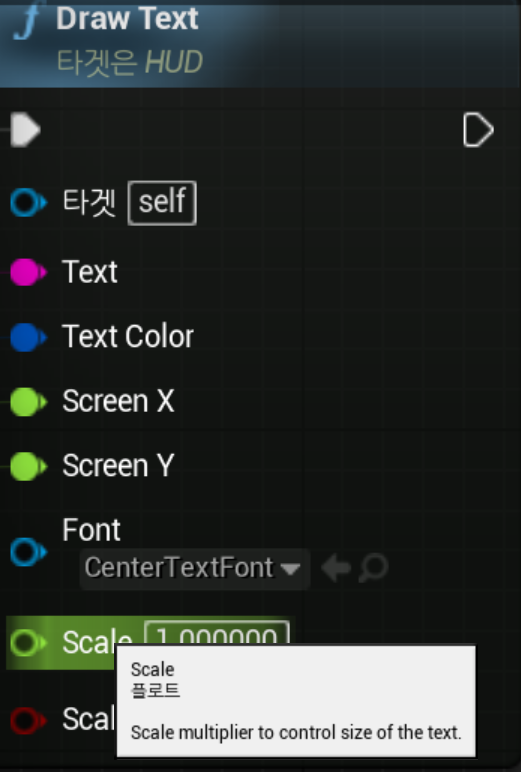


* **Game에 대하여 Clear와 Over에 대한 이벤트 처리 각각의 Color와 Text 설정.**

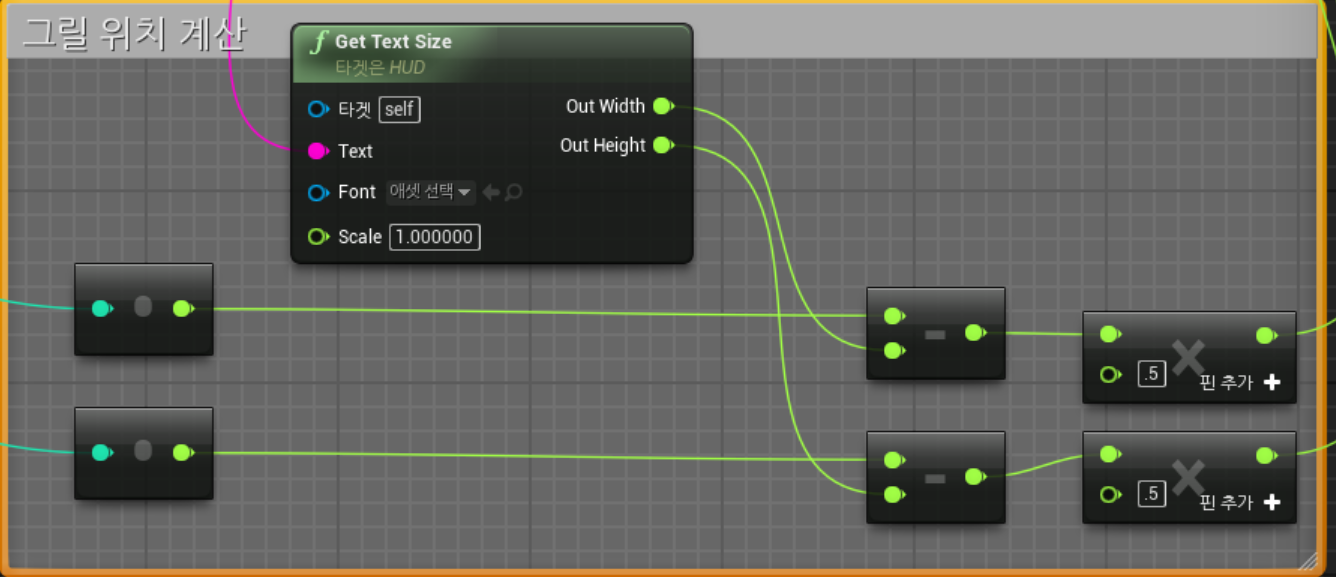
1. **이벤트 처리하기 (상세 이벤트)**



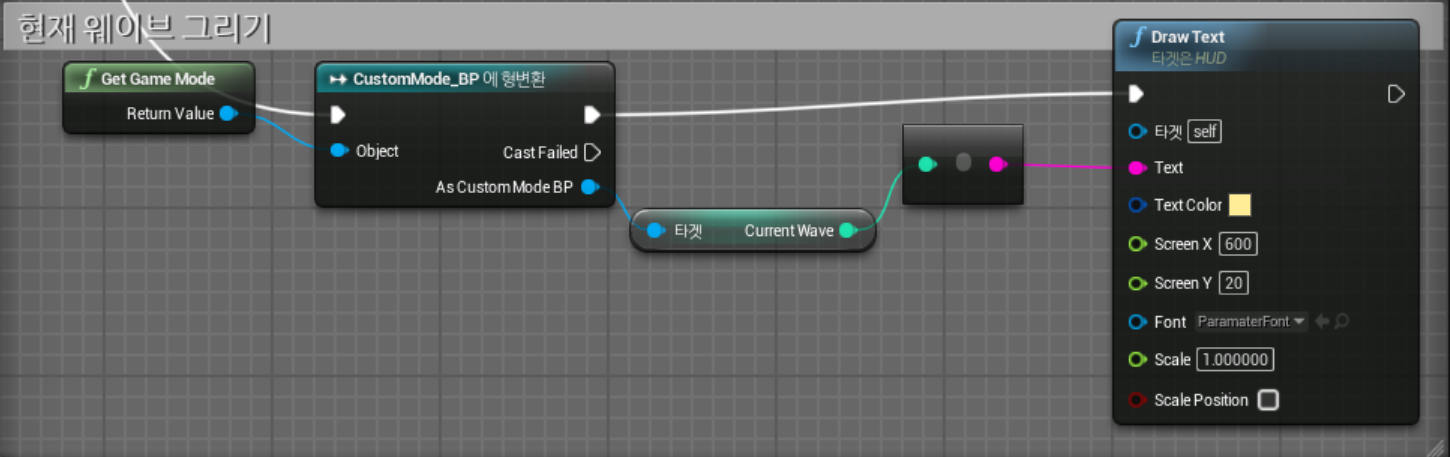
* **시퀀스 문으로 그려야 할 HUD 위치 값을 받습니다.**



* **HUD에 대하여 각각의 이벤트에 대하여 Text, Color, X, Y, Font를 받고 크기에 대하여 Scale을 정합니다**



* **텍스트에 대하여 그릴 위치를 상세하게 결정해주는 노드입니다.**



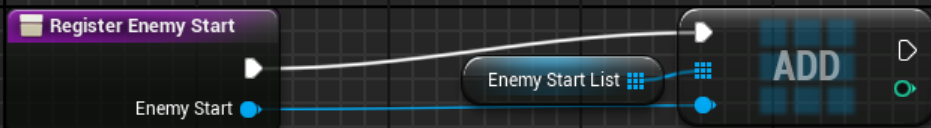
* **게임 모드로부터 현재 웨이브에 대하여 그려주는 역할을 합니다.**
* **게임 로직**

1. **변수 추가**



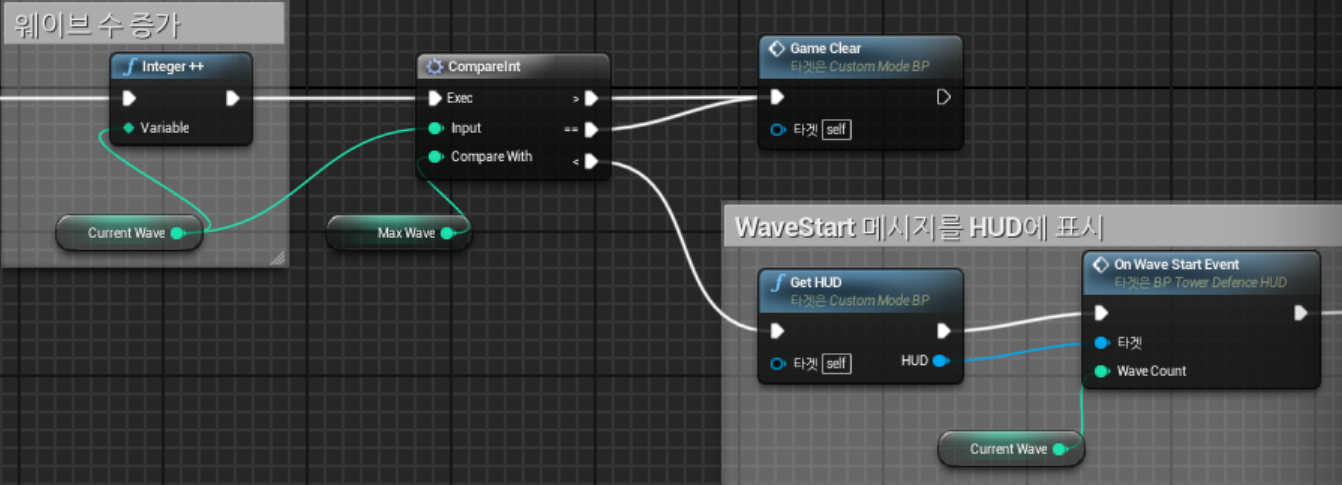
* **게임 로직 구현에 필요한 변수 추가.**

1. **게임 모드 클래스에 함수 추가하기**

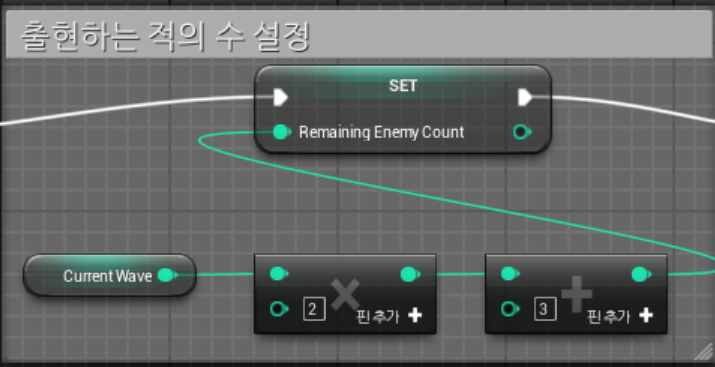


* **적의 시작 지점을 등록하는 함수를 만듭니다.**

1. **웨이브 시작하는 함수 만들기**

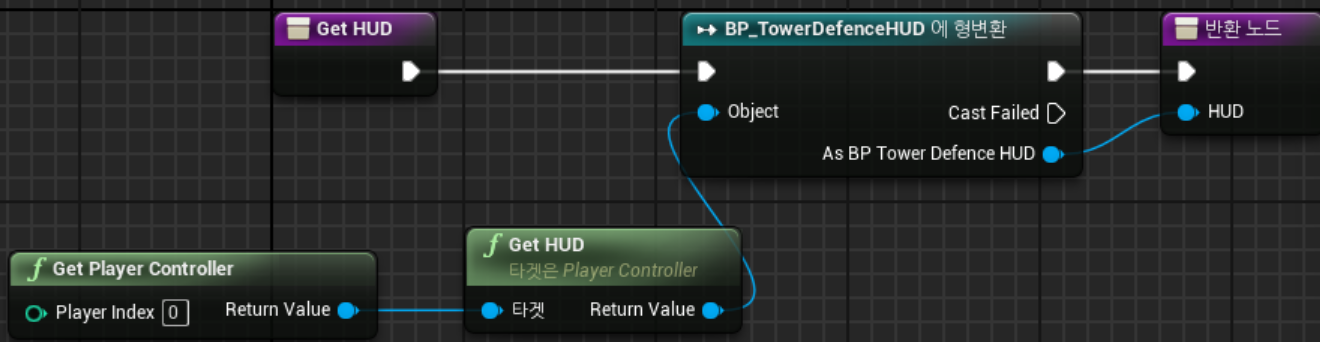


* **웨이브 수 증가에 따른 웨이브 시작 메시지를 HUD에 표시해줍니다.**



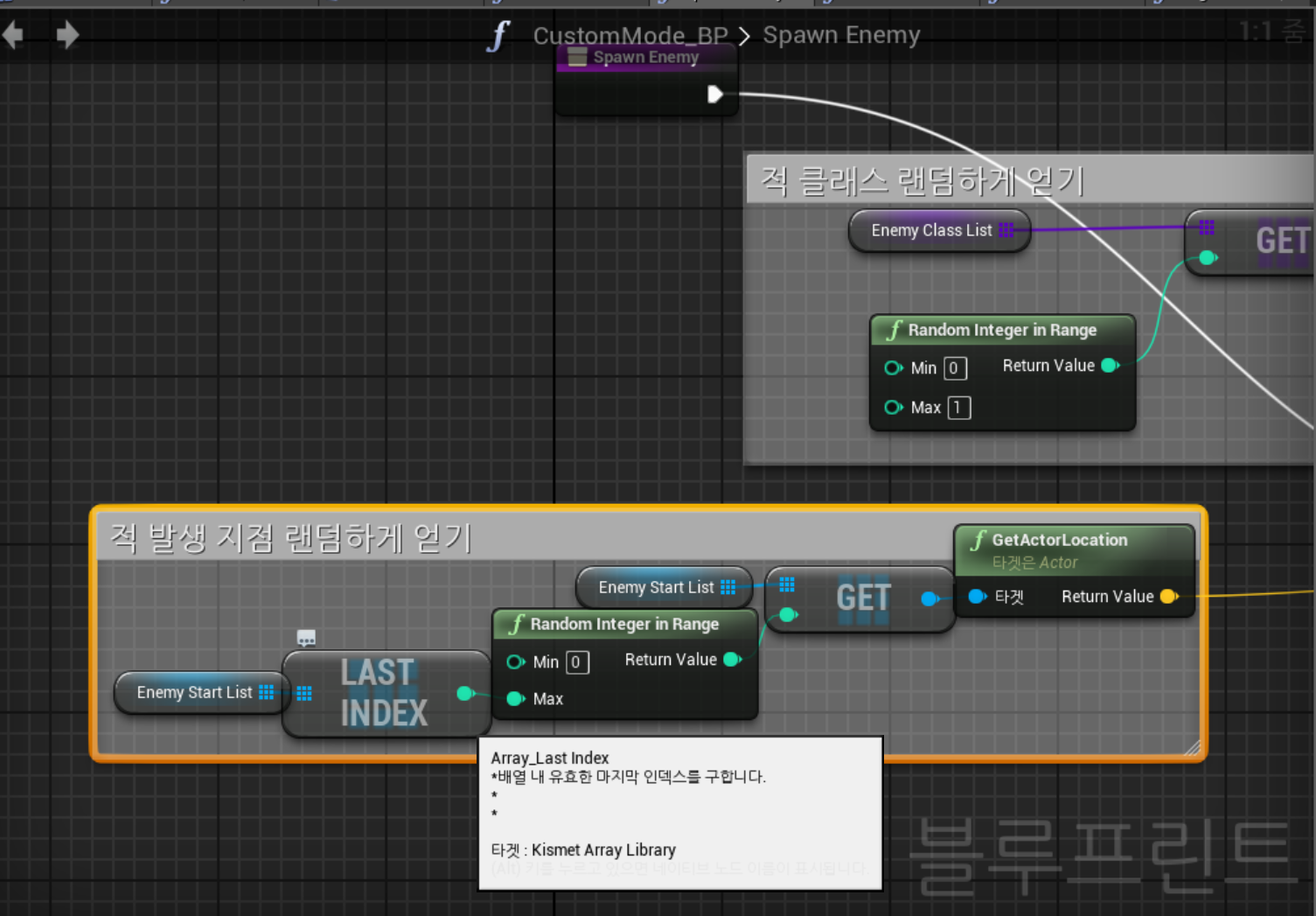
* **이에 따른 출현하는 적의 수 설정해줍니다.**

1. **HUD를 얻는 클래스 만들기**



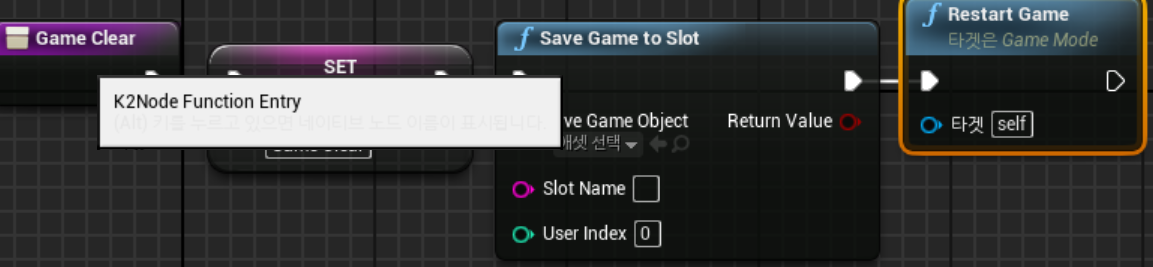
* **함수를 새로 만들고 앞에서 미리 만들어 두었던 HUD를 가져옵니다.**

1. **적을 생성하는 함수 만들기.**



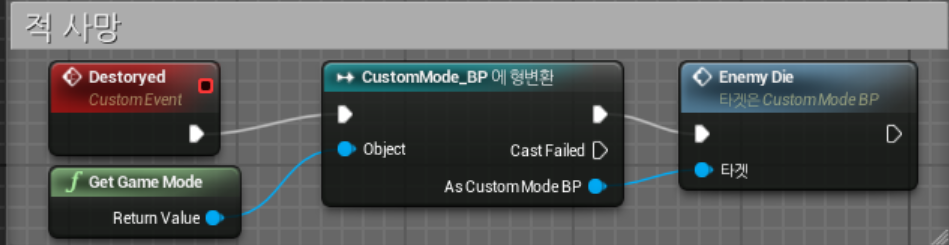
* **적을 랜덤하게 발생 시켜줍니다. 그에 따른 클래스를 얻고 발생 지점을 얻습니다.**

1. **게임 클리어 이벤트**

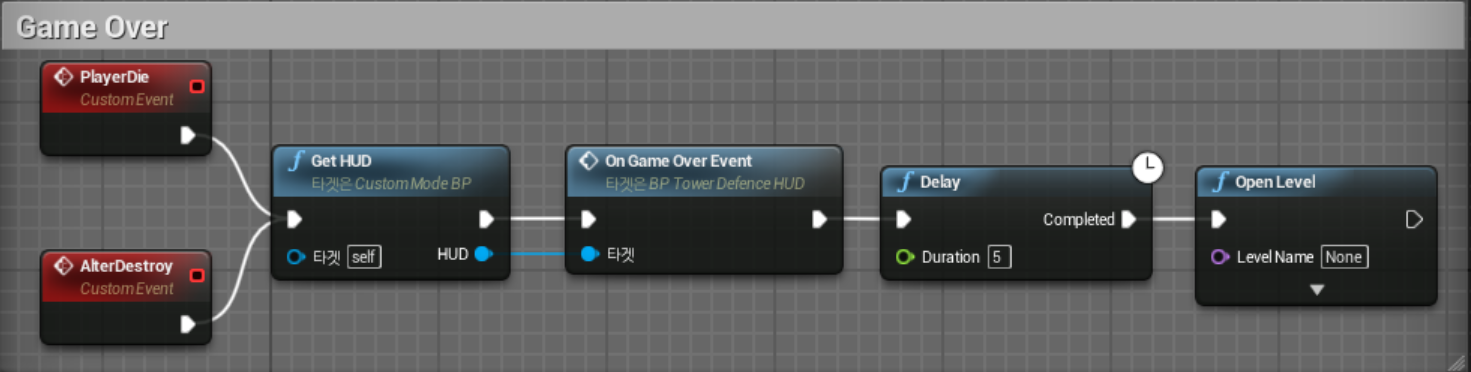


* **게임을 클리어 했다면 그에 해당하는 블루프린트 노드입니다.**

1. **사망 처리 및 게임 오버 처리**

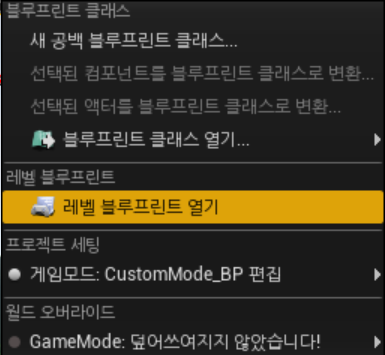


* **적이 죽었다면 그에 따른 사망 처리를 해줍니다.**



* **플레이어가 죽거나 마을이 파괴되면 게임 오버처리를 해줍니다.**

1. **레벨 블루 프린트 설정하기**



* **레벨 블루프린트에서 배경 음악이나 제단, 적 시작위치 등을 등록합니다. 즉 앞에서 구현을 했다면 실제로 적용시킬 맵에서 이 작업을 수행해야만 합니다.**



* **레벨 블루프린트에 대한 노드 값이며 각각 사운드를 받고 게임모드에 전달하는 방식을 가집니다. 그 게임모드에서는 시퀀스 문을 통해 앞에서 만들어 두었던 Register Enemy Start에 타겟으로 보내지게 됩니다.**