1월 5일

기존 구조에는 같은 코드를 여러 번 작성 했었어야 했고 관리가 어려운 단점이 있었음.

예로 데이터 테이블이 필요한 클래스들 (차량 및 아이템)들은 각각 한 번씩 생성해줘야했고 위젯컴포넌트 또한 각각 한번 씩 초기화 해줬었야 됐음.

장점) 구조를 싱글톤 구조로 만들었으며 작성해야할 코드 줄임과 동시에 직관성과 관리 측면에서 향상시킴.

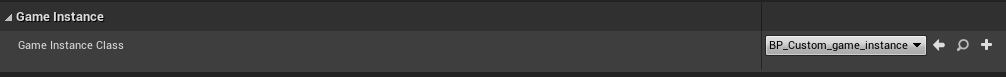
단점) 그 어떤 블루프린트 컴포넌트가 빠지면 안되고, 하나라도 빠진다면 프로그램이 뻗어버림 그리고 널 포인터 접근 가능성이 높아졌으므로 메모리 릭에 대해 보다 더 노출시킴.

그 말은 즉슨 안전성을 버리는 대신 재활용성을 살렸음.

텍스트, 풀볼, 검은색, 스포츠이(가) 표시된 사진

자동 생성된 설명

게임 인스턴스는 전역변수 및 함수용이며 게임 저장 같은 기능 구현할 때 유용하나 게임 시작 직후 생성되기 때문에 생성자에서 자원이 필요한 매니저들 관리하게끔 못함.

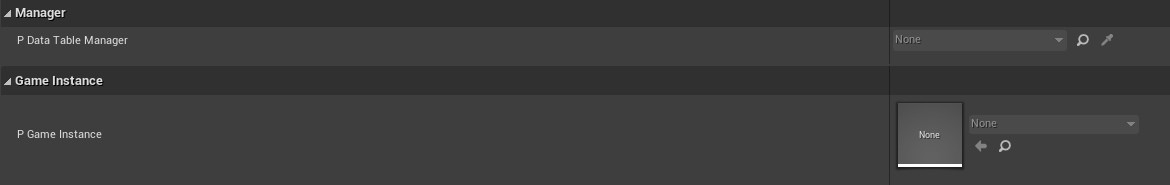


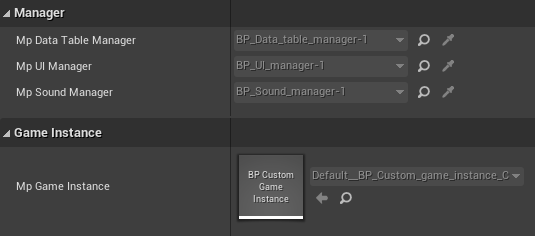
데이터 테이블 정보는 게임 시작 전 필요하기 때문에 매니저로 생성x Global 클래스 내 생성 및 게임 인스턴스에 대한 포인터도 생성

Global 클래스는 주로 생성자에서 즉 게임 시작하기 전에 필요한 데이터를 다루기 위해 사용될거임.

특이사항)

월드에 배치시켜야 자세하게 보임





게임 인스턴스 내에 사용되는 클래스 또한 게임 시작시 소환

Global 클래스 블루프린트는 무조건 하나라도 월드에 배치 되어있어야됨



텍스트이(가) 표시된 사진

자동 생성된 설명

매니저를 이용한 작업은 ConstructorHelpers 특성상 모두 생성자 내에 이루어져야됨 즉 생성자가 아닌 곳에는 클래스 멤버 변수로 두고 해야됨.