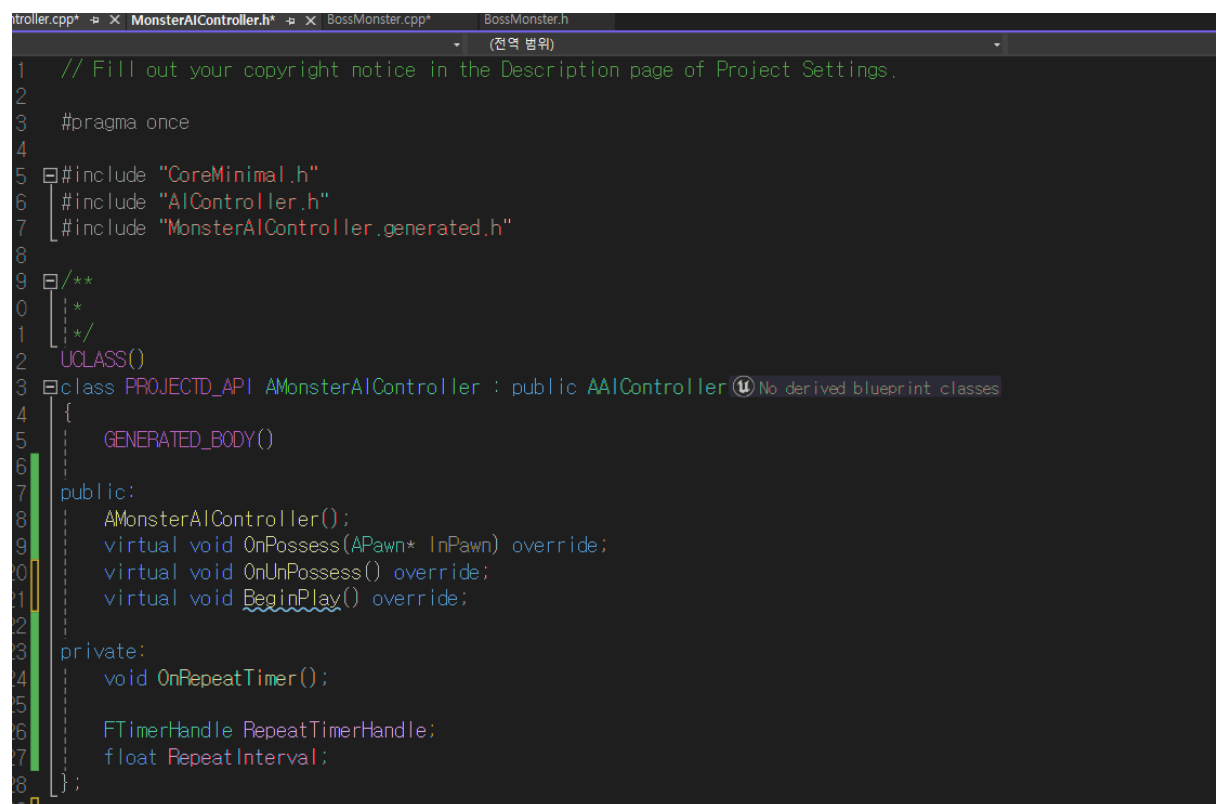


C++ : AIController 이동, 랜덤포인트

AIController를 사용하여 3초마다 네비 매쉬 위 랜덤 포인트로 이동



```
1 // Fill out your copyright notice in the Description page of Project Settings.
2
3 #pragma once
4
5 #include "CoreMinimal.h"
6 #include "AIController.h"
7 #include "MonsterAIController.generated.h"
8
9 /**
10  *
11  */
12 UCLASS()
13 class PROJECTD_API AMonsterAIController : public AAIController {
14     GENERATED_BODY()
15
16 public:
17     AMonsterAIController();
18     virtual void OnPossess(APawn* InPawn) override;
19     virtual void OnUnPossess() override;
20     virtual void BeginPlay() override;
21
22 private:
23     void OnRepeatTimer();
24
25     FTimerHandle RepeatTimerHandle;
26     float RepeatInterval;
27 };
```

```
MonsterAIController.cpp* * MonsterAIController.h* BossMonster.cpp* BossMonster.h
ProjectD (현재 범위)
4 #include "Monsters/MonsterAIController.h"
5 #include "NavigationSystem.h"
6 #include "Blueprint/AIBlueprintHelperLibrary.h"
7
8 AMonsterAIController::AMonsterAIController()
9 {
10     RepeatInterval = 3.0f;
11 }
12
13 void AMonsterAIController::OnPossess(APawn* InPawn)
14 {
15     Super::OnPossess(InPawn);
16     GetWorld()->GetTimerManager().SetTimer( RepeatTimerHandle , this , &AMonsterAIController::OnRepeatTimer , RepeatInterval , true );
17 }
18
19 void AMonsterAIController::OnUnPossess()
20 {
21     Super::OnUnPossess();
22     GetWorld()->GetTimerManager().ClearTimer( RepeatTimerHandle );
23 }
24
25
26 void AMonsterAIController::OnRepeatTimer()
27 {
28     auto CurrentPawn = GetPawn();
29     UNavigationSystemV1* NavSystem = UNavigationSystemV1::GetNavigationSystem( GetWorld() );
30     if (nullptr == NavSystem) return;
31
32     FNavLocation NextLocation;
33     if (NavSystem->GetRandomPointInNavigableRadius(FVector::ZeroVector, 500.0f, NextLocation))
34     {
35         UAIBlueprintHelperLibrary::SimpleMoveToLocation(this, NextLocation.Location );
36     }
37 }
38
```