**FPS Objective Project**

Show the finished project

Create a C++ class PickupActor

Create a blueprint class of the same by PickupActor.

We will add components through C++

So in the header file add following to protected

***UPROPERTY(VisibleAnyWhere, Category = "Component")***

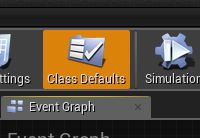
***UStaticMeshComponent\* MeshComp;***

***UPROPERTY(VisibleAnyWhere, Category = "Component")***

***USphereComponent\* SphereComp;***

forward declaration required so add at start

***class USphereComponent;***

effects visible only in class defaults. Look for static mesh and sphere components 

Now to C++ file add following to the constructor

***MeshComp = CreateDefaultSubobject<UStaticMeshComponent>(TEXT("MeshComp"));***

***RootComponent = MeshComp;***

***SphereComp = CreateDefaultSubobject<USphereComponent>(TEXT("SphereComp"));***

***SphereComp->SetupAttachment(MeshComp);***

need to include following

***#include "Components/SphereComponent.h"***

***#include "Components/StaticMeshComponent.h"***

Add actor fpsobjective to level and see the difference.

Go to level BP and add a sphere to the StaticMesh component. May have to scale to match.

Add following to constructor for Setting up Collisions

***MeshComp->SetCollisionEnabled(ECollisionEnabled::NoCollision);***

set collision on spatial queries (raycasts, sweeps, and overlaps)

***SphereComp->SetCollisionEnabled(ECollisionEnabled::QueryOnly);***

***SphereComp->SetCollisionResponseToAllChannels(ECR\_Ignore);***

***SphereComp->SetCollisionResponseToChannel(ECC\_Pawn, ECR\_Overlap );***

Collision should be working

**Adding Particle Effects**

Create a variable for FX

***UPROPERTY(EditDefaultsOnly, Category = "Effects")***

***UParticleSystem\* PicksupFX;***

Create function in header file

***void PlayEffects();***

Go to implementation in C++ file

***void AFPSObjectiveActor::PlayEffects()***

***{***

***UGameplayStatics::SpawnEmitterAtLocation(this, PicksupFX, GetActorLocation());***

***}***

Need include file

***#include "Kismet/GameplayStatics.h"***

add following to begin play to test it.

***PlayEffects();***

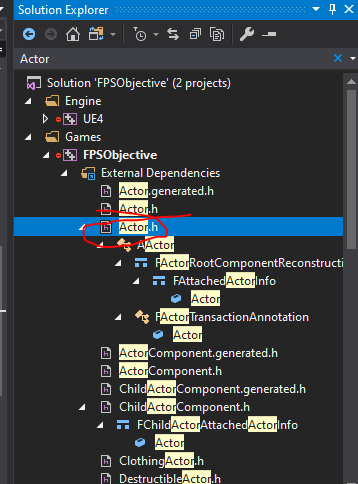
We should be able to get the effects in editor. (pick a particle effect in class defaults)

**Play effects on Overlap**

add following method declaration to header file (can comment out the playeffect() in beginplay)

***virtual void NotifyActorBeginOverlap(AActor\* OtherActor);***

Method can be found in the following class



then add following definition to C++ file.

***void AFPSObjectiveActor::NotifyActorBeginOverlap(AActor\* OtherActor)***

***{***

***Super::NotifyActorBeginOverlap(OtherActor);***

***PlayEffects();***

***}***

Exercise:

When player moves away destroy the actor.

(Note: Emitter will note be destroyed.)

Answer

***void APickupActor::NotifyActorEndOverlap(AActor \* OtherActor)***

***{***

***Destroy();***

***}***