***TOON TANKS PROJECT:***

Property specifiers:

When declaring properties, **Property Specifiers** can be added to the declaration to control how the property behaves with various aspects of the Engine and Editor

<https://docs.unrealengine.com/en-US/ProgrammingAndScripting/GameplayArchitecture/Properties/Specifiers/index.html>

|  |  |
| --- | --- |
| BlueprintReadOnly | This property can be read by Blueprints, but not modified. This Specifier is incompatible with the BlueprintReadWrite Specifier. |
| EditAnywhere | Indicates that this property can be edited by property windows, on archetypes and instances. This Specifier is incompatible with any of the the "Visible" Specifiers. |

UPROPERTY([specifier, specifier, ...], [meta(key=value, key=value, ...)])

Type VariableName;

StaticMeshComponent is used to create an instance of a [UStaticMesh](https://docs.unrealengine.com/en-US/API/Runtime/Engine/Engine/UStaticMesh/index.html) . A static mesh is a piece of geometry that consists of a static set of polygons.

A StaticMesh is a piece of geometry that consists of a static set of polygons. Static Meshes can be translated, rotated, and scaled, but they cannot have their vertices animated in any way. As such, they are more efficient to render than other types of geometry such as [USkeletalMesh](https://docs.unrealengine.com/en-US/API/Runtime/Engine/Engine/USkeletalMesh/index.html) , and they are often the basic building block of levels created in the engine.

Constructors used:

| **Prefix** | **Meaning** |
| --- | --- |
| A | Extends from the base class of spawnable gameplay objects. These are Actors, and can be spawned directly into the world. |
| U | Extend from the base class of all gameplay objects. These cannot be directly instanced into the world; they must belong to an Actor. These are generally objects like [Components](https://docs.unrealengine.com/en-US/ProgrammingAndScripting/ProgrammingWithCPP/UnrealArchitecture/Actors/Components/index.html). |

Text

Description automatically generated

Here **GetHitResultUnderCursor(…)** does the LineTrace and gets the TraceHitResult which you can assign it to HitLocation using TraceHitResult.**ImpactPoint**