Modular Character Plugin Documentation

Welcome to the documentation for the Modular Character Plugin for Unreal Engine. This plugin empowers developers to create modular characters easily, offering a range of features such as dynamic character construction, persistent character skin data transfer across levels, and a runtime character customization environment.

F.A.Q :

My character does not show up hence i set up my character:

Ensure your character parts have at least one material.

A\All of my character parts are flickering when i move the camera:

Ensure your meshes have a physic asset assigned.

Update Notice v1.2:

- Added support for copy pose mesh; bConfigureForCopyPose
- Added bConfigureForMasterPose as option; Iin previous versions this option was default behavior except the case that bUseAnimationsPerInstance option enabled
- bUseAnimationsPerInstance has been removed (since you disabled bConfigureForMasterPose and bConfigureForMasterPose this is exact behavior of the plugin)

Update Notice v1.1:

Added Support for ALS-Refactored Plugins Default Blueprint Character:

- For using with ALS character structure close disable all checkbox except “bForceNameMatchWithComponent”
- Define CharacterParts with respect to your modular components.
- Add Skeletal Mesh Components (Component Names must be exact with each related CharacterPart)

1. Introduction

Plugin Overview

The Modular Character Plugin is a powerful tool for Unreal Engine developers looking to create modular characters with ease. Whether you're building a game with customizable characters, need dynamic character construction, or want to transfer character skin data across levels, this plugin has you covered.

2. Key Features

Dynamic Character Construction: Automatically create and manage skeletal mesh components for character parts.

Persistent Character Skin Data: Transfer character skin data across levels for a consistent character appearance.

Runtime Character Customization: Enable players to customize characters during gameplay.
Data Table Integration: Define character construction information using Unreal Engine data tables.

3. Getting Started

Before diving into the details, make sure to install the plugin correctly. Once installed, you can start using it in your Unreal Engine project.

4. Configuring Properties

Configure the plugin's properties to define your character's behavior and appearance. Key properties include:

bSelectBestOptionsForMe: Automatically selects the best construction options at runtime.
bAutoCreateSkeletalMeshes: Automatically creates skeletal mesh components.
bUseParentSkeletalMeshAsRoot: Sets the parental skeletal mesh as the root.
CharacterAnimClass: Define the animation blueprint for the character.
ConstructionTable: Use a data table to define character construction information.
bUseAnimationsPerInstance: Assigns animation blueprint for every part of character instead of just leader part

5. Runtime Character Customization

Enable players to customize characters during gameplay using the provided functions. This allows for dynamic character appearance changes based on player input.

6. API Reference

UModularCharacterComponent Class
Refer to the UModularCharacterComponent Class for an in-depth look at the class's properties and methods. This is essential for advanced usage and understanding the component's capabilities.

7. Example Videos

Video 1: Basic Setup

Video 2: Replication

Video 3: Persistent Data

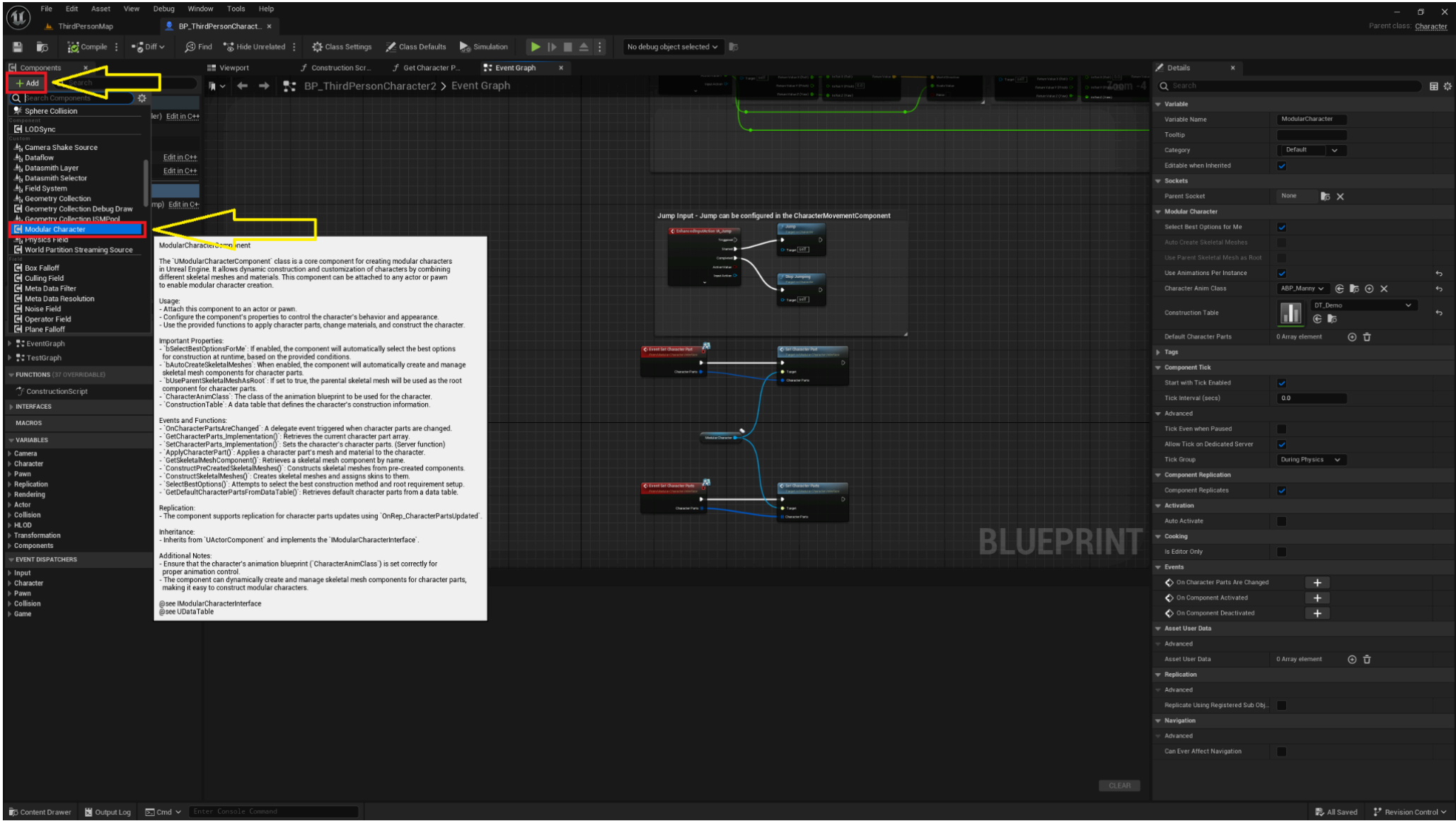
Video 4: Listen Server Support

Video 5: Gender Change To The Different Skeleton

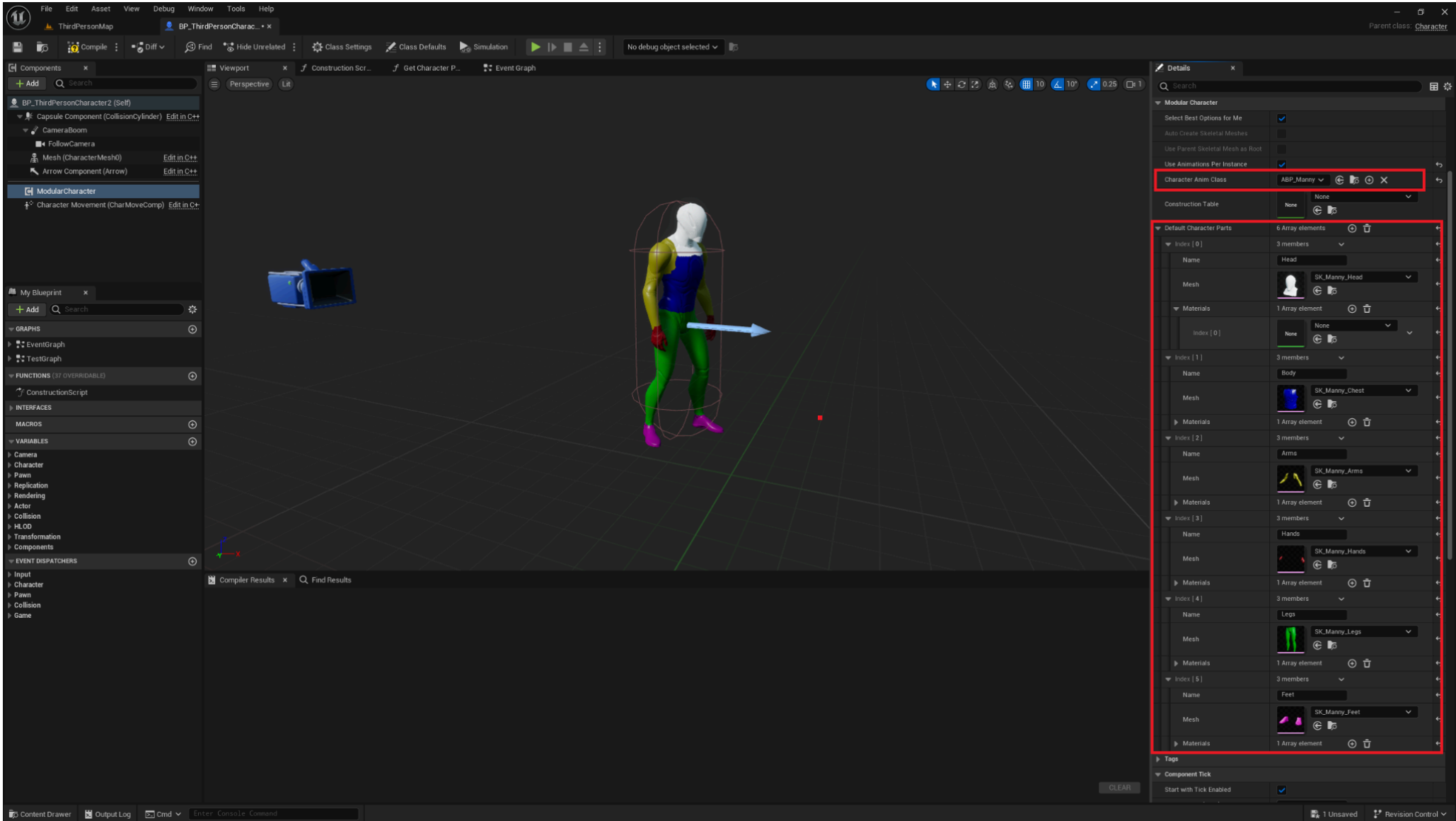
Video 6: Performance Comparison Between Conventional Method

BASIC USAGE [VIDEO]

1. Add a modular character component to a character blueprint

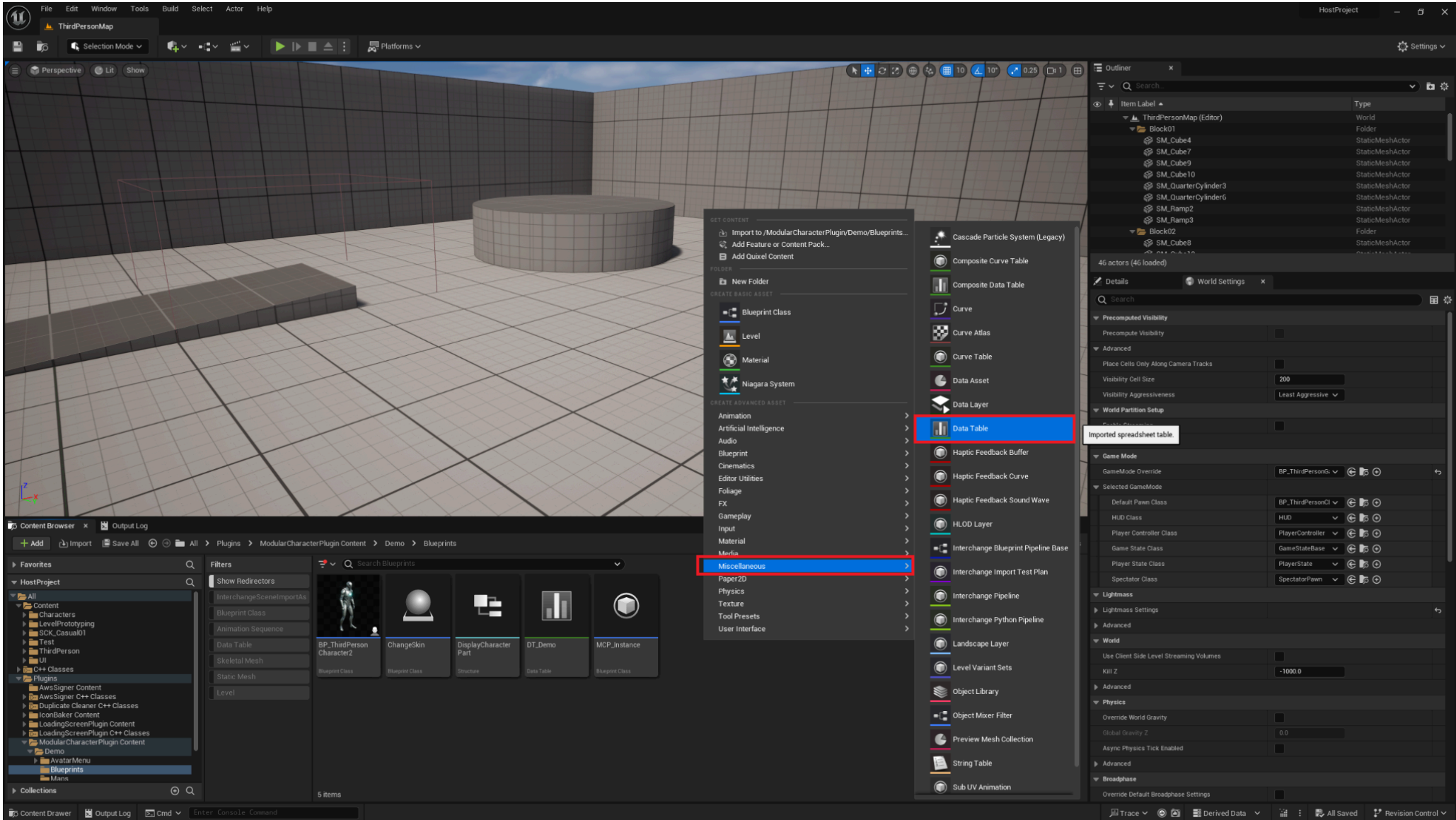


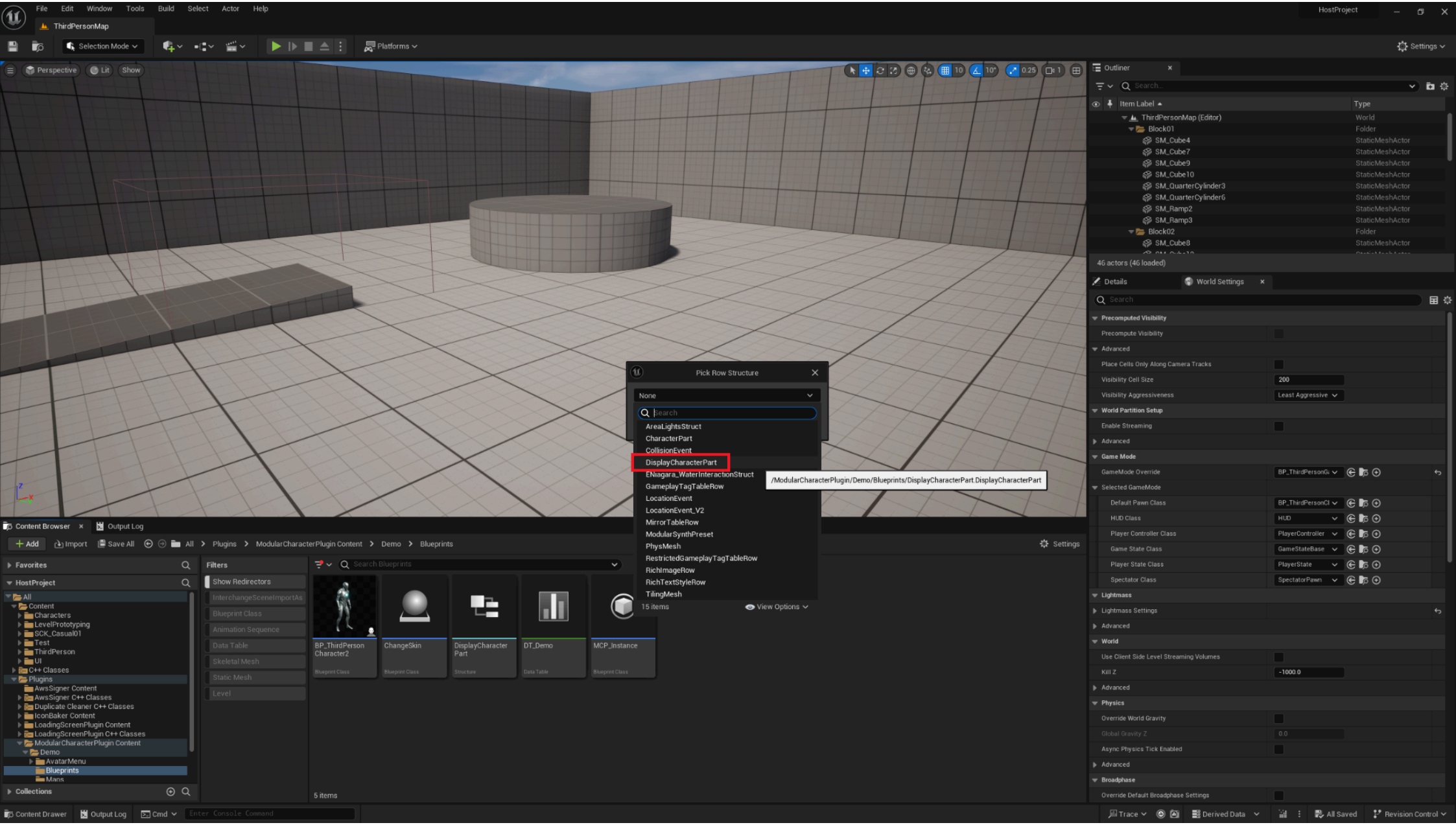
2. Select Animation class
3. Select initial meshes(each skeleton type has a unique name)
4. Each mesh has a material even if its empty



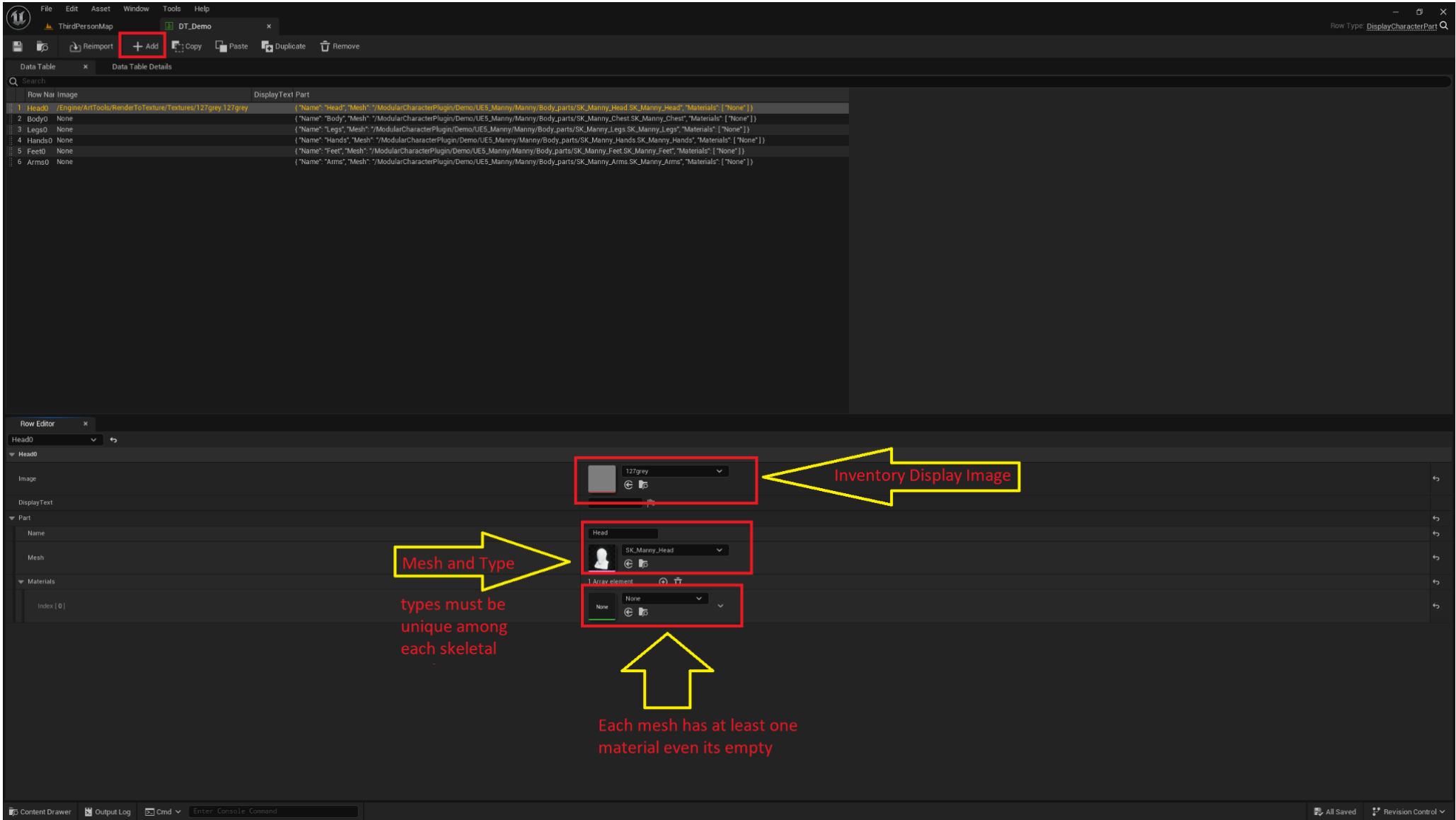
Using With Datatables

1. Create a data table from DisplayCharacterPart structure, You can use another custom structure if it has a property populated from FCharacterPart.(Not direct FCharacterPart tables are supported)



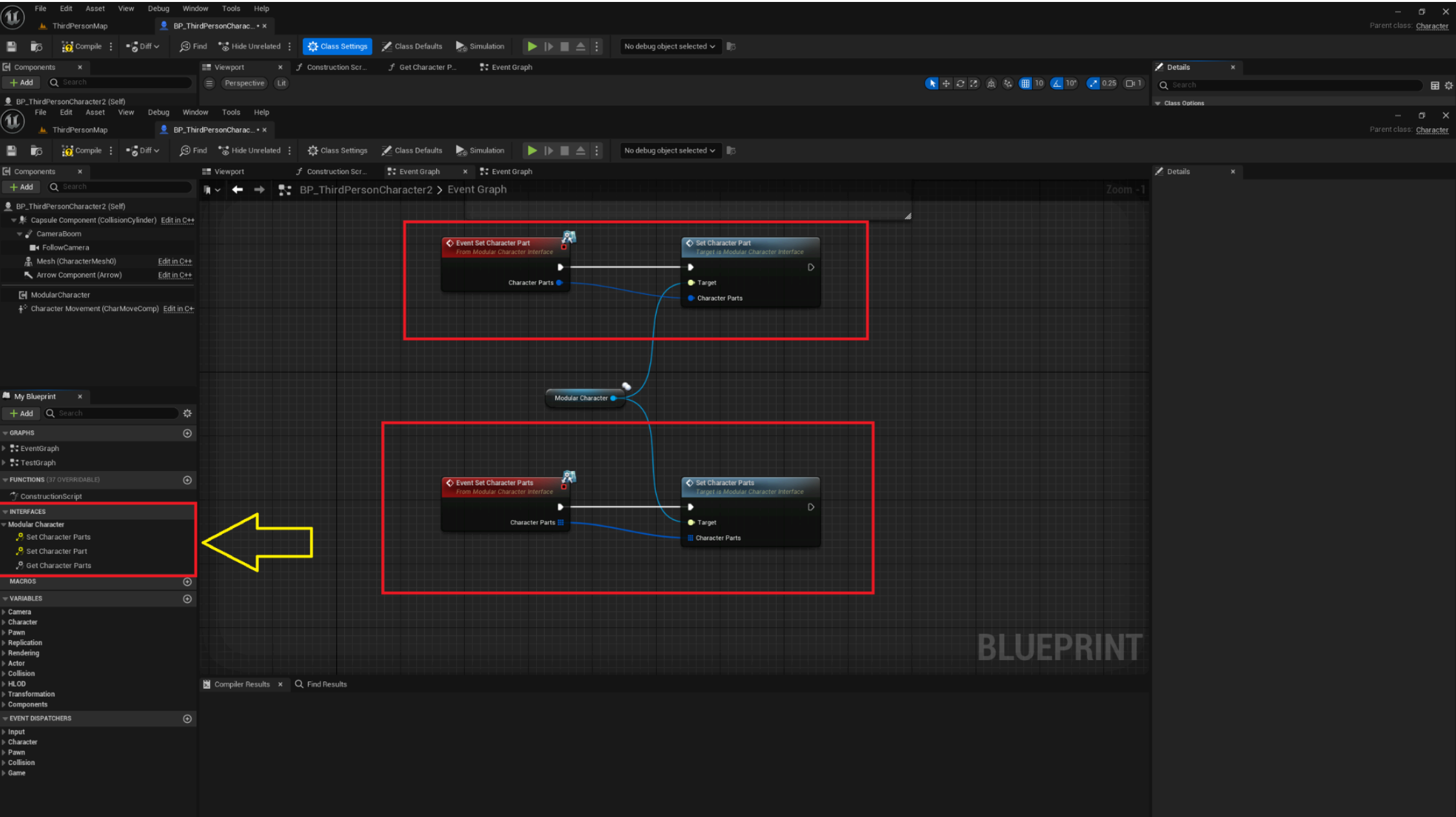

File Edit Window Tools Build Select Actor Help
HostProject
ThirdPersonMap
Selection Mode
Platforms
Settings
Perspective Lit Show
Outliner
Item Label
Type
ThirdPersonMap (Editor)
World
Block01
Folder
SM_Cube4
SM_Cube7
SM_Cube9
SM_Cube10
SM_QuarterCylinder3
SM_QuarterCylinder6
SM_Ramp2
SM_Ramp3
Block02
SM_Cube8
StaticMeshActor
46 actors (46 loaded)
Details
World Settings
Pick Row Structure
None
Search
AreaLightsStruct
CharacterPart
CollisionEvent
DisplayCharacterPart
ENiagara_WaterInteractionStruct
GameplayTagTableRow
LocationEvent
LocationEvent_V2
MirrorTableRow
ModularSynthPreset
PhysMesh
RestrictedGameplayTagTableRow
RichImageRow
RichTextStyleRow
TilingMesh
15 items
View Options
/ModularCharacterPlugin/Demo/Blueprints/DisplayCharacterPart.DisplayCharacterPart
Precomputed Visibility
Precompute Visibility
Advanced
Place Cells Only Along Camera Tracks
Visibility Cell Size
200
Visibility Aggressiveness
Least Aggressive
World Partition Setup
Enable Streaming
Game Mode
GameMode Override
Selected GameMode
Default Pawn Class
HUD Class
Player Controller Class
Game State Class
Player State Class
Spectator Class
Lightmass
Lightmass Settings
World
Use Client Side Level Streaming Volumes
Kill Z
-1000.0
Physics
Override World Gravity
Global Gravity Z
Async Physics Tick Enabled
Broadphase
Override Default Broadphase Settings
Content Browser
Output Log
Add
Import
Save All
All > Plugins > ModularCharacterPlugin Content > Demo > Blueprints
Favorites
HostProject
Content
Characters
LevelPrototyping
SCK_Casual01
Test
ThirdPerson
UI
C++ Classes
Plugins
AwsSigner Content
AwsSigner C++ Classes
Duplicate Cleaner C++ Classes
IconBaker Content
LoadingScreenPlugin Content
LoadingScreenPlugin C++ Classes
ModularCharacterPlugin Content
Demo
AvatarMenu
Blueprints
Collections
Filters
Show Redirectors
InterchangeSceneImportAs
Blueprint Class
Animation Sequence
Data Table
Skeletal Mesh
Static Mesh
Level
Search Blueprints
BP_ThirdPerson Character2
ChangeSkin
DisplayCharacter Part
DT_Demo
MCP_Instance
5 items
Content Drawer
Output Log
Cmd
Enter Console Command
Trace
Derived Data
All Saved
Revision Control

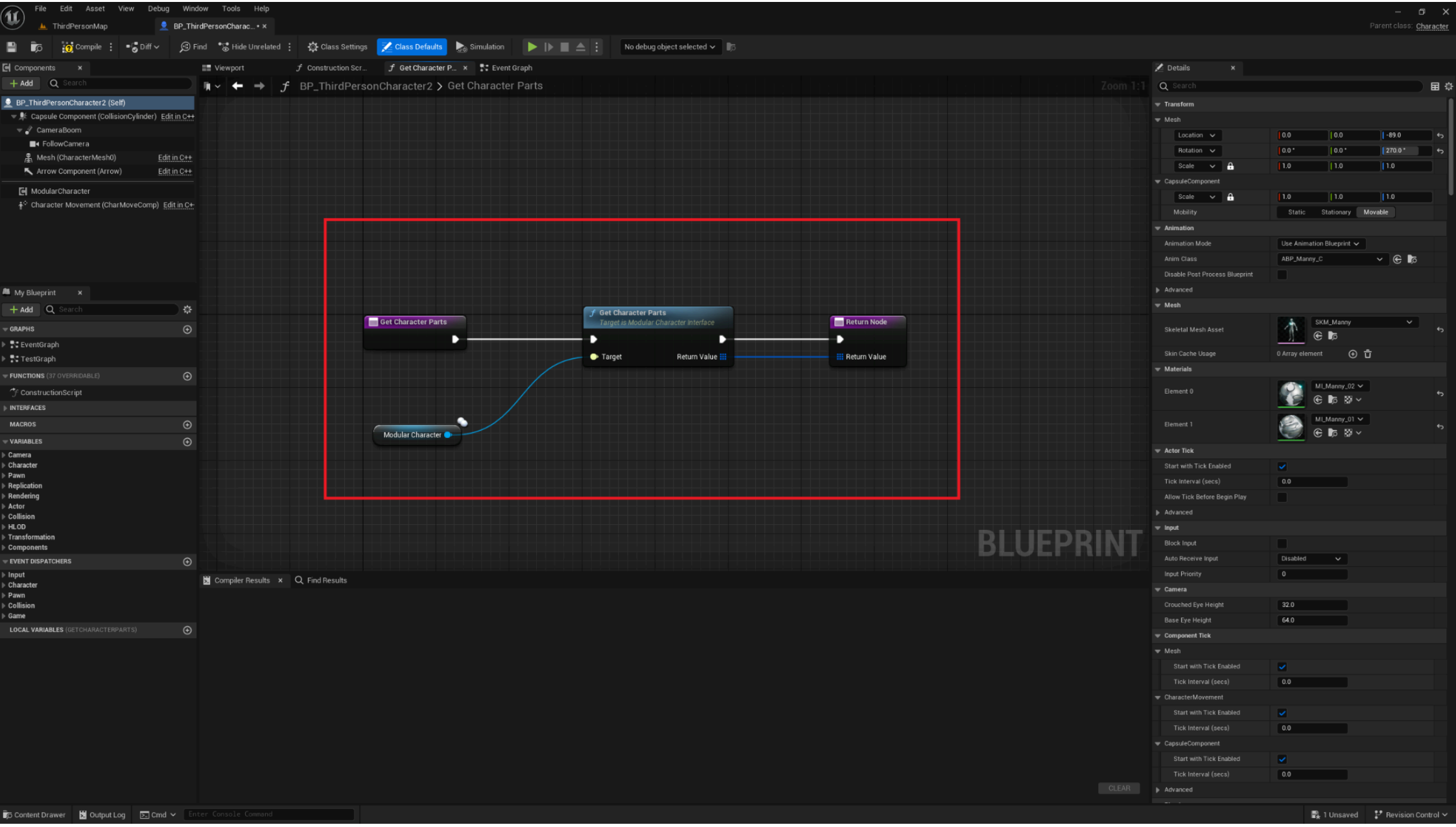
Add your character parts to the table, parts with the same types can change during runtime. But for initialization first occurred instance of a part will be used.



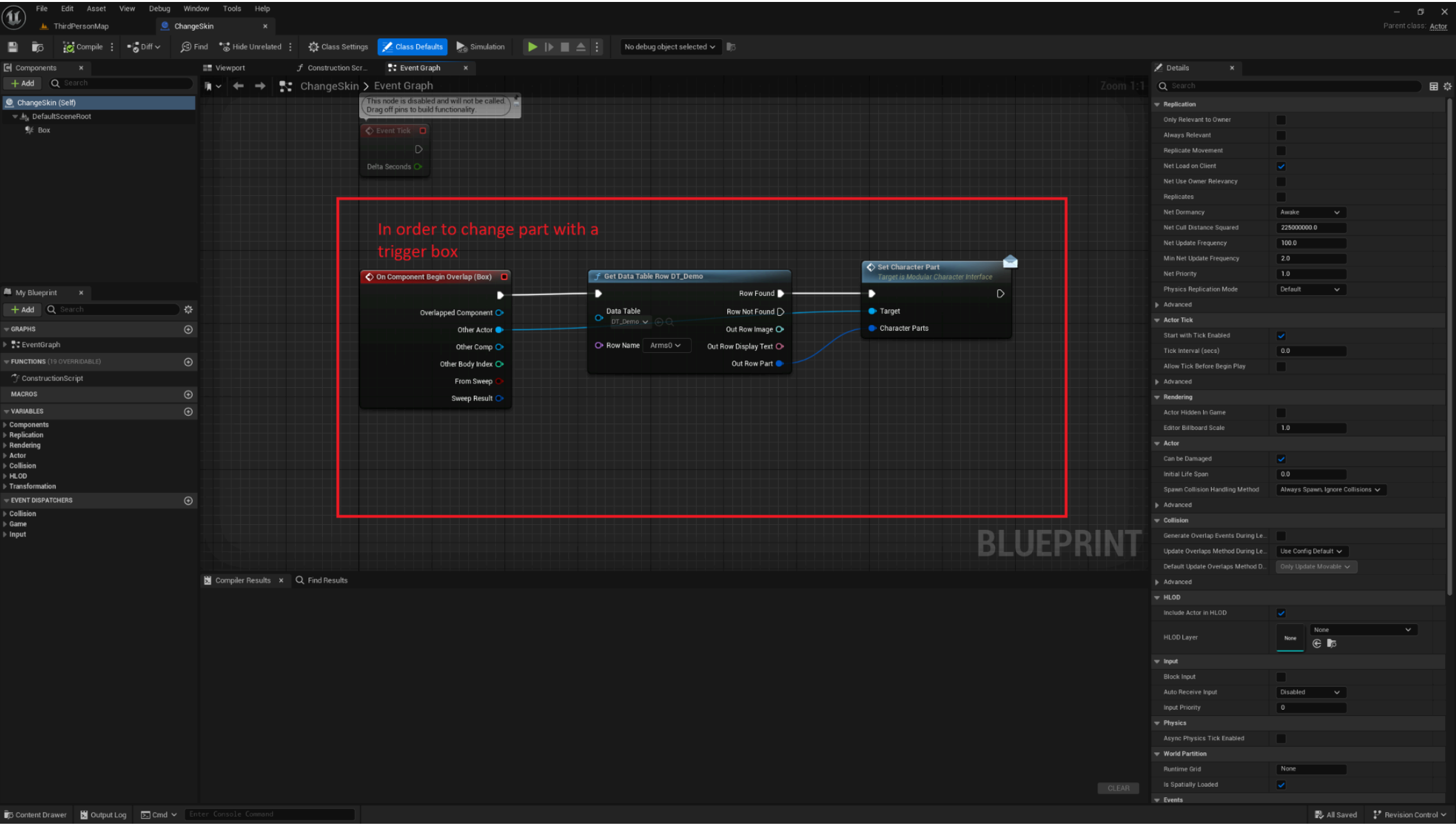
Changing Parts at Runtime [VIDEO]

1. Implement the ModularCharacterInterface to the Character Blueprint with clicking Class Settings in Character Blueprint



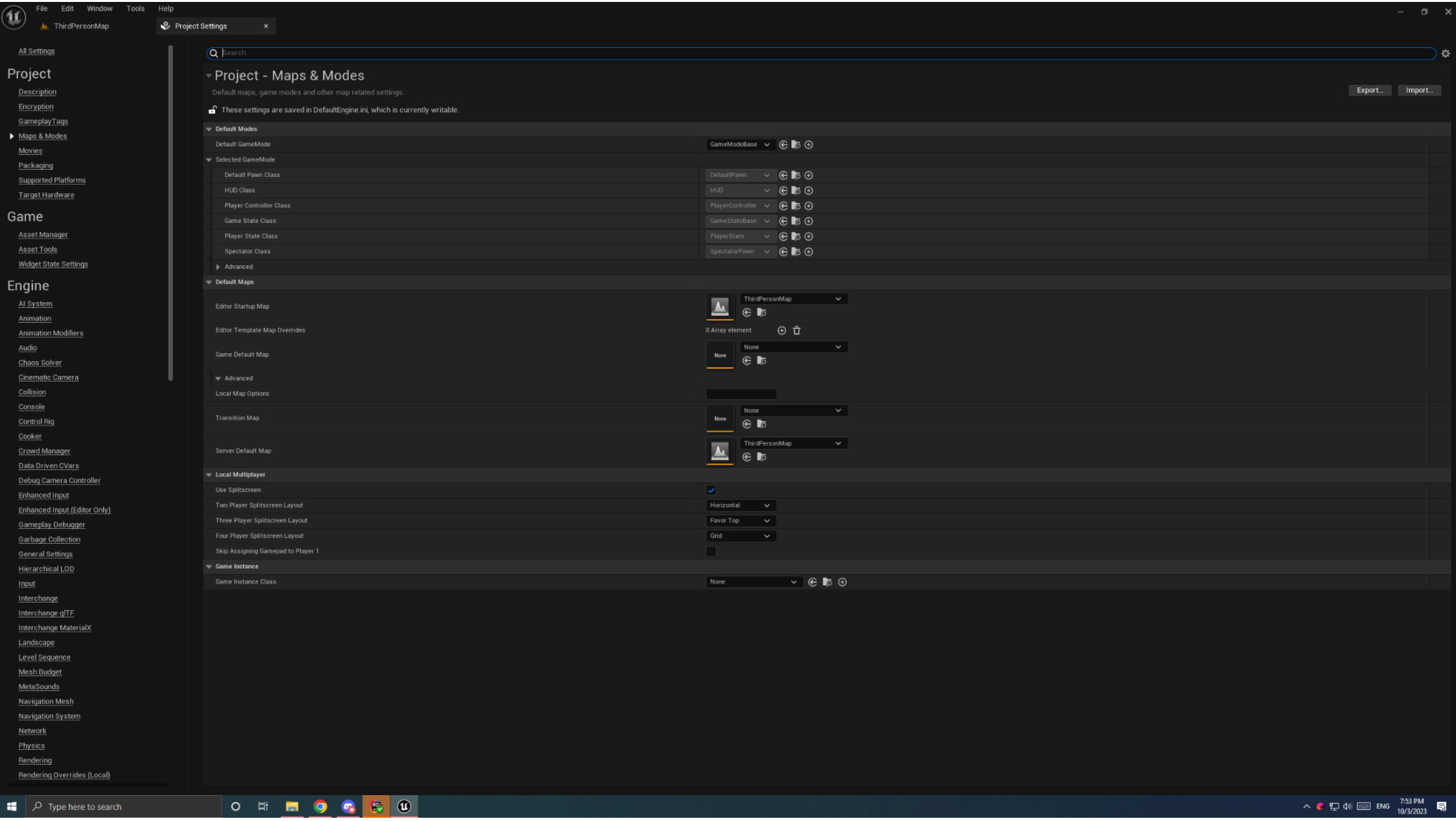

File Edit Asset View Debug Window Tools Help
ThirdPersonMap
BP_ThirdPersonCharac...
Parent class: Character
Compile
Diff
Find
Hide Unrelated
Class Settings
Class Defaults
Simulation
No debug object selected
Components
Add
Search
BP_ThirdPersonCharacter2 (Self)
Capsule Component (CollisionCylinder) Edit in C++
CameraBoom
FollowCamera
Mesh (CharacterMesh0) Edit in C++
Arrow Component (Arrow) Edit in C++
ModularCharacter
Character Movement (CharMoveComp) Edit in C++
Viewport
Construction Scr...
Get Character P...
Event Graph
BP_ThirdPersonCharacter2 > Get Character Parts
Zoom 1:1
Get Character Parts
Get Character Parts
Target is Modular Character Interface
Target
Return Value
Return Node
Return Value
Modular Character
BLUEPRINT
My Blueprint
Add
Search
GRAPHS
EventGraph
TestGraph
FUNCTIONS (37 OVERRIDABLE)
ConstructionScript
INTERFACES
MACROS
VARIABLES
Camera
Character
Pawn
Replication
Rendering
Actor
Collision
HLOD
Transformation
Components
EVENT DISPATCHERS
Input
Character
Pawn
Collision
Game
LOCAL VARIABLES (GETCHARACTERPARTS)
Compiler Results
Find Results
CLEAR
Details
Search
Transform
Mesh
Location 0.0 0.0 -89.0
Rotation 0.0 * 0.0 * 270.0 *
Scale 1.0 1.0 1.0
CapsuleComponent
Scale 1.0 1.0 1.0
Mobility Static Stationary Movable
Animation
Animation Mode Use Animation Blueprint
Anim Class ABP_Manny_C
Disable Post Process Blueprint
Advanced
Mesh
Skeletal Mesh Asset SKM_Manny
Skin Cache Usage 0 Array element
Materials
Element 0 MI_Manny_02
Element 1 MI_Manny_01
Actor Tick
Start with Tick Enabled
Tick Interval (secs) 0.0
Allow Tick Before Begin Play
Advanced
Input
Block Input
Auto Receive Input Disabled
Input Priority 0
Camera
Crouched Eye Height 32.0
Base Eye Height 64.0
Component Tick
Mesh
Start with Tick Enabled
Tick Interval (secs) 0.0
CharacterMovement
Start with Tick Enabled
Tick Interval (secs) 0.0
CapsuleComponent
Start with Tick Enabled
Tick Interval (secs) 0.0
Advanced
Content Drawer
Output Log
Cmd
Enter Console Command
1 Unsaved
Revision Control

Create a actor for manipulate character, in this example basic actor used, with Boxtrigger

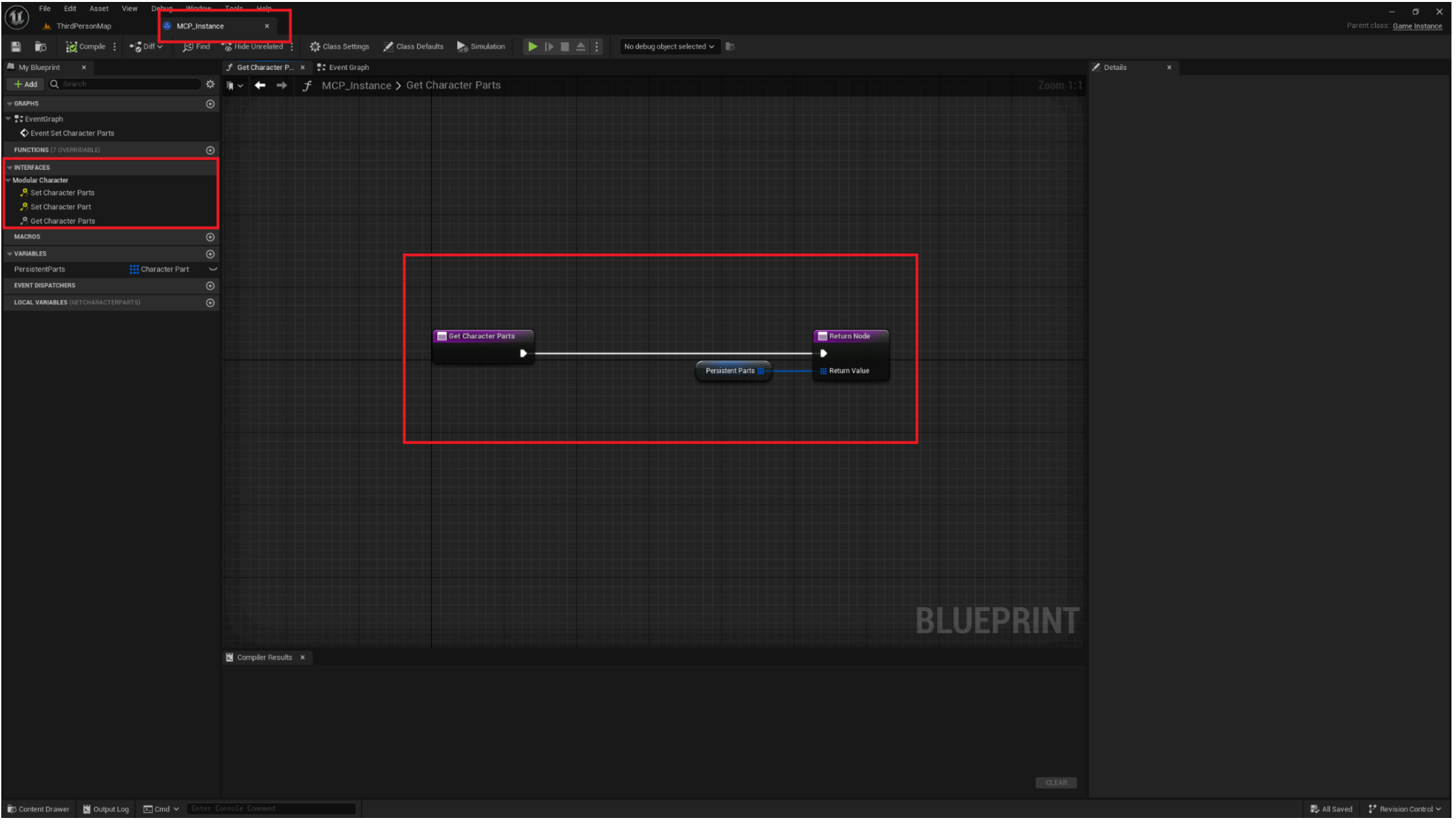


Persistent Part Data Storing [VIDEO]

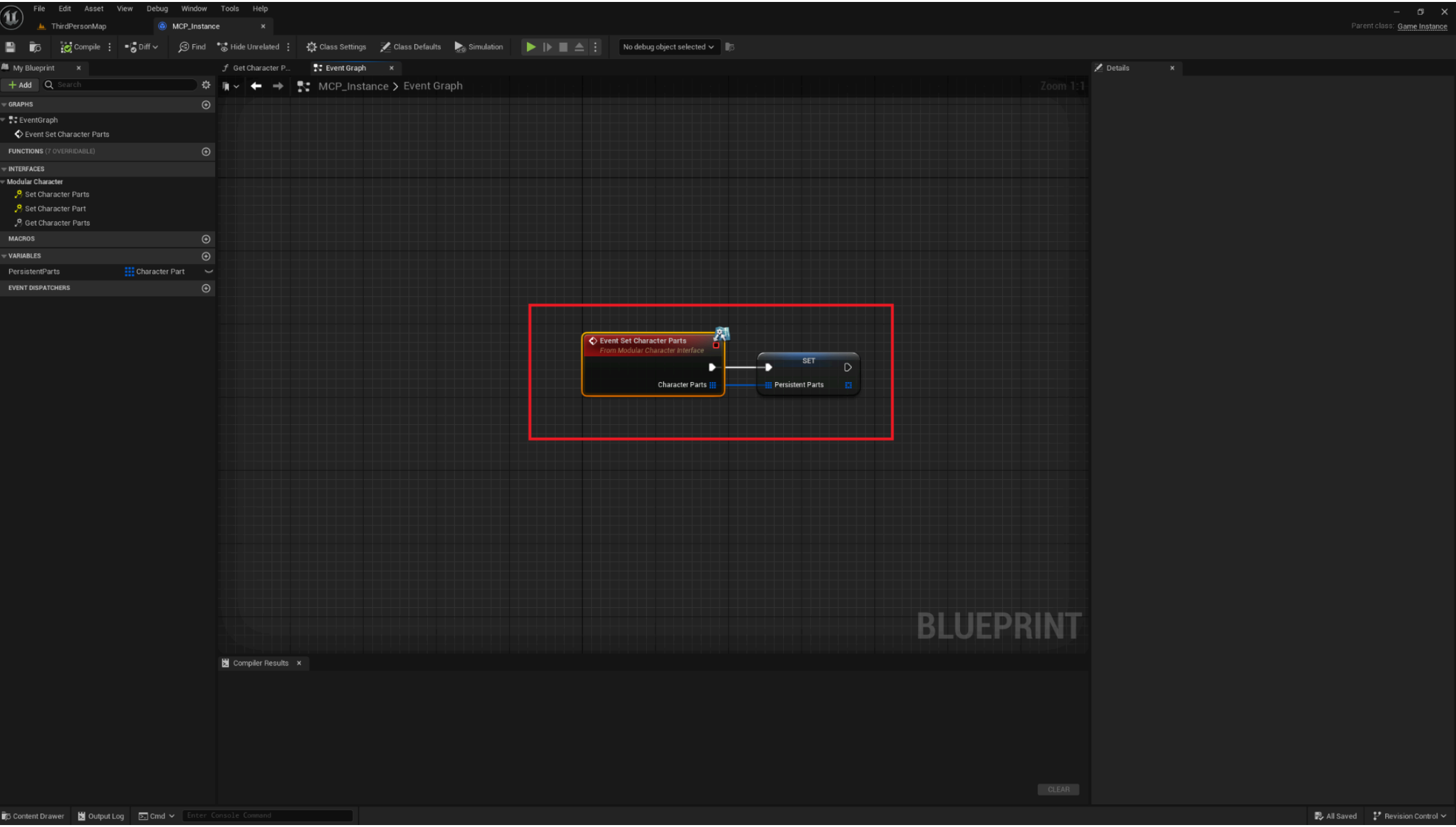
Create A Game Instance and set it to Default Game Instance from Project Settings -> Maps&Modes-> Default Game Instance



Implement The Modular Character Interface to created GameInstance from Class Settings - > Interfaces -> Add

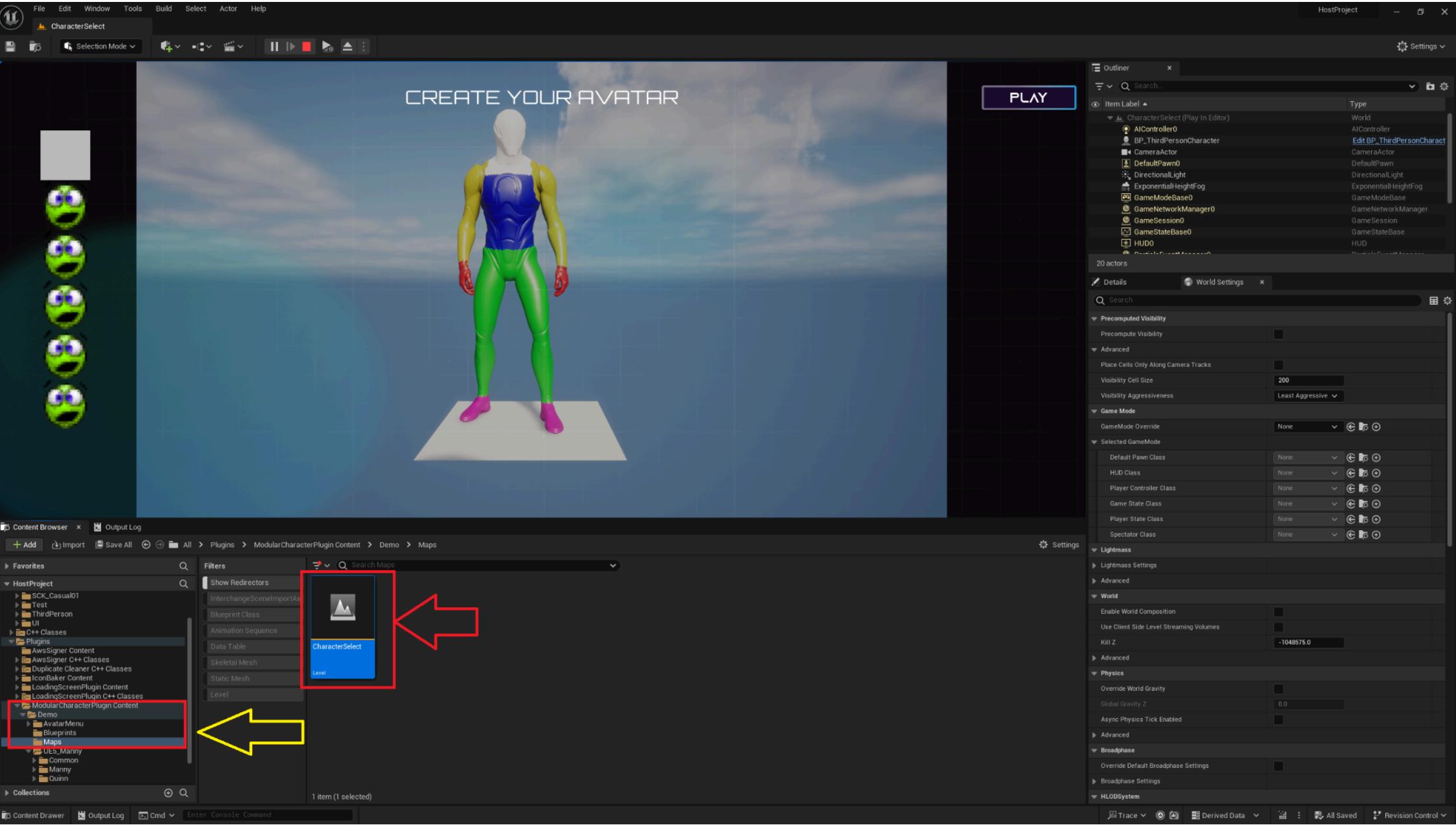


Create an Array of FCharacterPart variable (TArray<FCharacterPart>) and implement set character parts & get character parts with using this variable



Using Demo Character Selection Screen [VIDEO]

Open CharacterSelect level from Plugins->ModularCharacterPlugin's Content -> Maps



From Level Blueprint Select DataTable that you created before
If you select another blueprint instead of BP_demo_thirdperson you must replace character in this level and also level blueprint
Widget's part selection screen will be created for your data table dynamically.