



Courtesy of AltSpace

UNREAL FAST TRACK

WORKSHOP ONE

Thank you for participating in the Unreal Fast Track! For the next five weeks, you'll learn Unreal Engine through Unreal Online Learning, Epic's free learning platform. The Unreal Fast Track is designed to be completed in teams of three or four people, and it is recommended that you pick teammates who have the same level of experience in Unreal Engine and similar areas of interest. The entire Unreal Fast Track is intended to take you only one night per week for five weeks to complete. Each of the five workshops consists of three parts:



Test Drive: Learning a new part of the engine by completing an Unreal Online Learning course.



Grand Prix: Bringing your new learning and exploration to a project you'll add to each week.



Off-Roading and Discussion: Exploring, experimenting, and talking about what you just learned with your team.



Test Drive

Unreal Online Learning courses are the core of the Unreal Fast Track. Developed by Epic Games, Unreal Online Learning is a learning platform that was created at the request of professional companies seeking to upskill their employees. It provides the content that professionals use to learn in the industry, and you can and should learn with the same content.

Each part of the Unreal Fast Track has an Unreal Online Learning course associated with it for you and your team to complete. The courses teach you vital parts of Unreal Engine, including introducing engine concepts, teaching new tools within the engine, and discussing the philosophy behind many of the engine's systems. The Unreal Online Learning site can be found at www.unrealengine.com/onlinelearning.

Each course consists of videos and quizzes, and with the completion of each course, you'll earn a badge on your Unreal Online Learning Achievements page.



Grand Prix

The Grand Prix is an overarching project that you'll be adding to as you complete the Unreal Fast Track. You will add mechanics and systems to a game using what you learned during the Test Drive and what you explored in the Off Roading. After each Unreal Online Learning course, there will be something new to add to your project, and opportunities for you to modify the game to fit your design.



Off-Roading and Discussion

After completing the Unreal Online Learning course, the next step is experimentation. During the Off Roading and Discussion portion, you will be asked to alter, create, and break your projects to understand Unreal Engine better. Off Roading is an opportunity for you to flex your creativity and attempt to implement the ideas you had while going through the course. Making mistakes is completely acceptable here and even encouraged!

During and after the Off Roading process, you will be prompted to participate in a discussion with your teammates. The discussion will focus on three **Ds**:

1. **Difficulties** you encountered during the workshop, including the most confusing or difficult part
2. **Discoveries** you made while experimenting and exploring
3. **Dreams** you have about what you want to make with what you just learned, but aren't sure how to start

/// How It Works

Unreal Fast Track is completed with a virtual team.

Here is how it works:

1. Find a team of three or four people.
2. Set a date and time for your group to meet weekly for five weeks. By that time each week, each member should have completed the Test Drive and Grand Prix sections of the current workshop.
3. During the week, before you meet up, complete the current Unreal Online Learning course in the Test Drive section and the Grand Prix portion by yourself.
4. If you have issues completing the course, message your teammates or ask for help on the Discord server. If you were able to successfully complete it, assist your teammates and others who need help.
5. Get together virtually at the time your group decided and complete the Off Roading and Discussion portion.



Workshop One: Introduction to the engine and making a third-person platformer

/// Learning Points

Test Drive (1 hour)

- Creating new projects
- Navigating the Viewport
- Getting content from the Marketplace
- Placing content into a Level
- Introduction to Blueprints

Grand Prix (1 hour)

- Basic Level creation
- Further Blueprint introduction

Off-Roading and Discussion (1 hour)

- Introduction to the CharacterMovement component
- Editing variables and playtesting



Test Drive

For the first week, you will start off with an Unreal Online Learning course called "Your First Hour in Unreal Engine." This course will give you a broad overview of the engine while going through the steps for making a scene and running around inside it. You can find the course at <https://www.unrealengine.com/onlinelearning-courses/your-first-hour-with-unreal-engine>. In this document, we've included tips and changes that have been created for some of the videos since the course first came out. If needed, the tips and changes for each video are timestamped so you know when they are relevant. If there is no timestamp, you can wait until the end to read them.

1: Welcome to the Launcher

We are now up to Unreal Engine version 4.24.

Unreal Studio has been rolled into Unreal Engine now and no longer exists as a standalone.

2: Creating Your First Project

1:40: The Unreal Project Browser looks a bit different to the browser seen in the video, but all of the content is still the same, just moved around. Start by clicking **Games** and then choosing the **Third Person** template from there. After clicking **Next**, the default options should be correct, but to make sure, check that the first option is set to **Blueprint**.

4: Building Your First Level

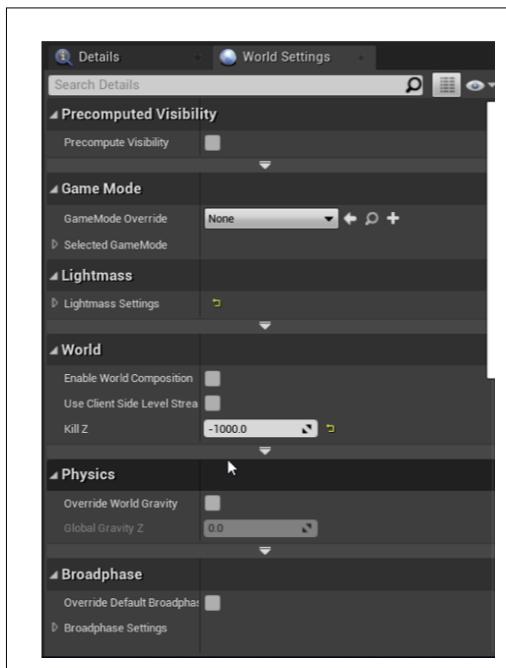
With a static mesh selected, if you hit the **End** key, the static mesh will snap to the floor.



Grand Prix

Using what you've learned so far, try creating a third-person platformer. Use the static meshes from the **Starter Content** and the **Infinity Blade** asset pack; if you want to, you could even add more of the **Infinity Blade** or **Paragon** content to your Level. As you're building it, test your Level in-engine. If you fall, simply exit the **PIE (Play in Editor)** session and press **Play** again to restart.

After you have built a third-person platformer Level, let's add the ability to die and respawn to the game. You will be introduced to new concepts here; so long as you follow the steps, you will do great.

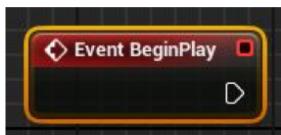


To add death to your third-person platformer, follow these instructions:

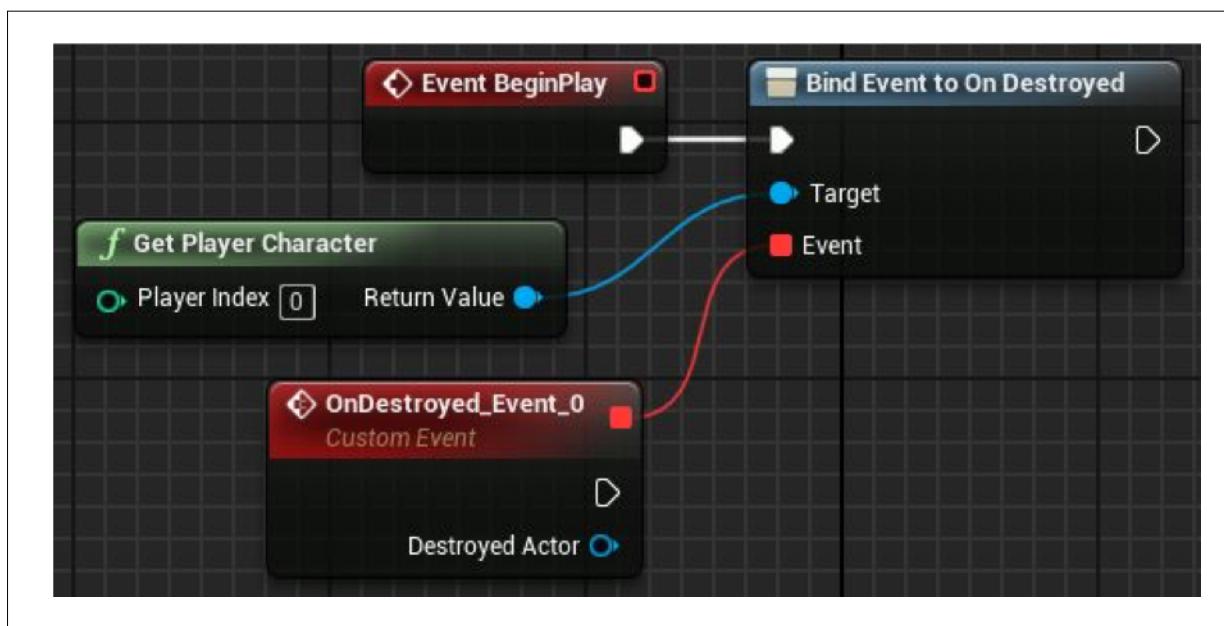
1. In the Toolbar on the top of the screen, click the Settings button and select the first option, "World Settings".
2. This will add a second tab next to your Details panel. Scroll through the World Settings panel to the World section and find the Kill Z option.
3. Set Kill Z to "-1000".
4. Now, whenever a player reaches -1000 on the Z axis, they will be immediately destroyed. You can try it for yourself right now! When you jump off a platform and reach -1000 units on the Z axis, the player will disappear because the Actor is deleted from the Level. Now that you're destroyed, how do you respawn?

Respawning may be a bit trickier, but just follow along with the directions below and you'll be fine.

1. Inside your project, open up your **Game Mode** Blueprint (we are using the **ThirdPersonGameMode** Blueprint). It is located in the same folder as the **ThirdPersonCharacter** Blueprint.
2. When you open up the **Game Mode** Blueprint, there will be a note underneath **Class Defaults** telling you this is a data-only Blueprint. To get to the Blueprint Editor, click the blue words "**Open Full Blueprint Editor**".
3. Here you will see the Event Graph. Right-click in the graph and add an **Event BeginPlay** node. To add a node:
 - a. Right-click in the Event Graph.
 - b. Type in text related to the node you are looking for. Unreal Engine will show you the relevant options.
 - c. Find the node you are looking for and click on it.
 - d. Additionally, Unreal Engine will try and guess which node you want and will highlight that one in gray. If the node you are looking for is highlighted in gray, pressing **Enter** on your keyboard will select it.

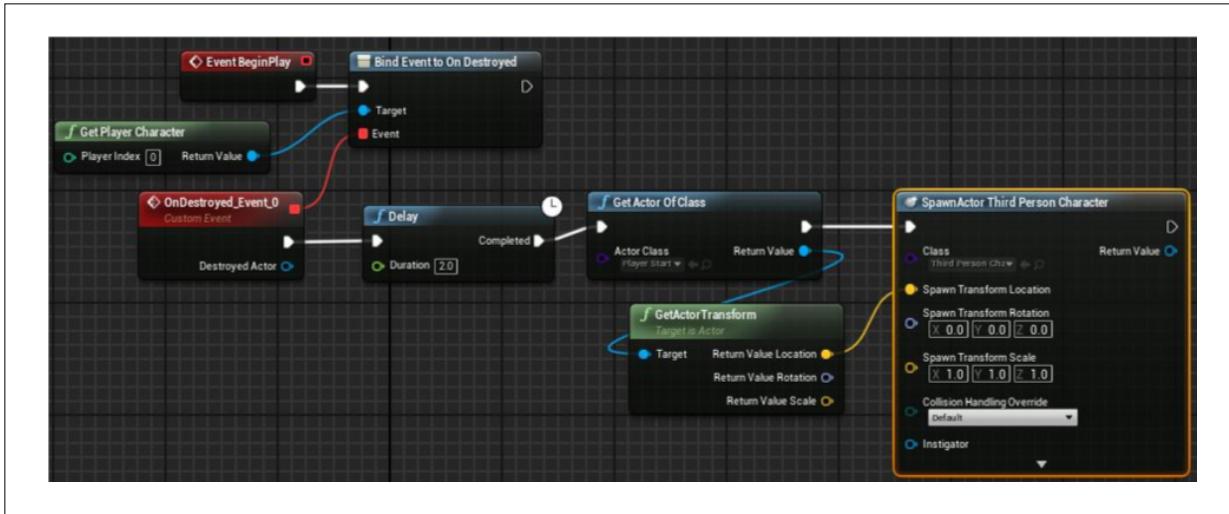


4. Right-click in the graph again and add a **Get Player Character** node.
5. Click and drag a wire from the **Get Player Character** node's **Return Value** pin and add an **Assign On Destroyed** node. (Be sure to pick the right one here! It should auto-create two nodes, **Bind Event to On Destroyed** and **OnDestroyed_Event_0**.)
6. Connect the **Event BeginPlay** node's white triangular output execution pin to the **Assign On Destroyed** node's input execution pin.

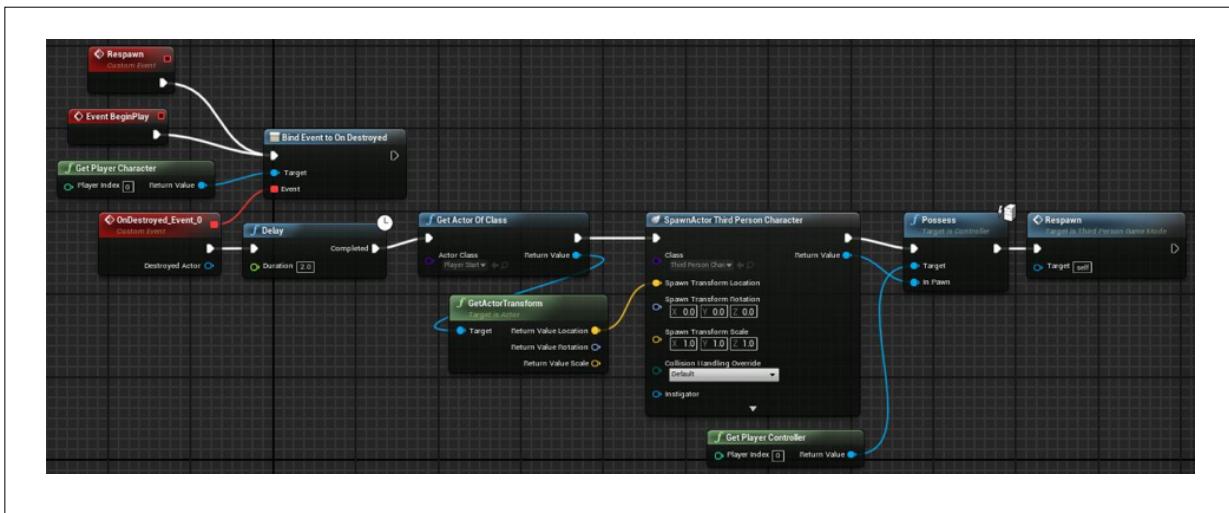


7. Off the newly created **OnDestroyed_Event_0** node, connect a **Delay** node set to 2 seconds. (Note: The **Delay** node is optional and just for aesthetic purposes.)
8. Off the **Delay** node, connect a **Get Actor Of Class** node. In the node's **Actor Class** drop-down, select "PlayerStart".
9. From the **Get Actor Of Class** node's output execution pin, connect a **SpawnActorFromClass** node. In the **SpawnActorFromClass** node's **Class** dropdown, select "**Third Person Character**".

10. From the **Get Actor Of Class** node's **Return Value** pin, connect a **GetActorTransform** node.
11. Right-click on the **GetActorTransform** node's **Return Value** pin and select "**Split Struct Pin**".

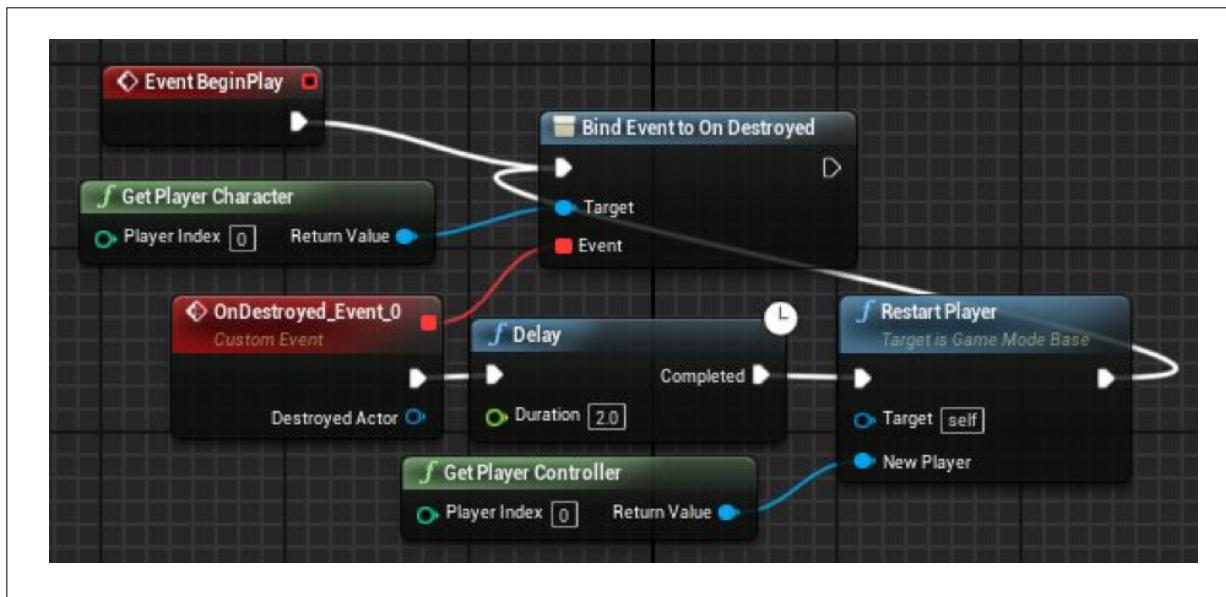


12. On the **SpawnActorFromClass** node you created, right-click on the **Spawn Transform** pin and select "**Split Struct Pin**". Connect the **GetActorTransform** node's **Return Value Location** pin to the **SpawnActorFromClass** node's **Spawn Transform Location** pin.
13. Right-click in the graph to add a **Get Player Controller** node. (Note: This does **NOT** say "**Get Player Character** node" as in step four.)
14. Drag a wire from the **Get Player Controller** node's **Return Value** pin and add a **Possess** node.
15. Connect the **SpawnActor** node's **Return Value** pin to the **Possess** node's **In Pawn** pin and connect the execution wires.
16. Near the **Event BeginPlay** node, right-click in the graph and add a custom event and name it "**Respawn**".
17. Connect the custom event to the **Bind Event to On Destroyed** node.
18. Back at the **Possess** node, drag a wire from its output execution pin and type "**respawn**" in the search box to call the newly made **Respawn** event.
19. Compile and save the Blueprint. Your full graph should look similar to this:



20. Congratulations! You now have dying and respawning in your game. Be sure to save this project so we can come back to it.

This exercise is intended to help you learn and explore Blueprints just a small amount. If you were to do it the Epic way, you could change the Blueprint graph to look like this:



Off-Roading and Discussion

Experiment with the various Actors you've been introduced to in this course. Try messing around with the settings in the Details panel for the lights and post-processing effects; see what new kinds of lighting you can create by adjusting various settings. Explore and share with your teammates some of the interesting settings you've found. Next, in the **ThirdPersonBP** folder, you'll find another folder labeled "**Blueprints**". There, double-click **ThirdPersonCharacter**, and it will open up the ThirdPersonCharacter Blueprint. In the Blueprint Editor, on the left side, find the **Components** panel. There, select the **CharacterMovement** component. After selecting it, you will see the Details panel on the right side show you all of the character movement settings.

Try changing settings in the **Character Movement (General Settings)**, **Character Movement: Walking**, and **Character Movement: Jumping/Falling** sections. See how changing the numbers and checkboxes affects your character in-game. Think about the game feeling you might want. For instance:

- Do you want a high-jumping, fast-running character?
- Do you want a very slow, methodical character?
- Do you want a character that can walk up steep surfaces?

Play around with the settings and create an experience for your teammates. Play each other's Levels and discuss the following:

- Difficulties: What was the most difficult or confusing part of this week's workshop?
- Discoveries: What were some interesting, fascinating, or exciting things you discovered while making the Level and adjusting the character movement?
- Dreams: What ideas for mechanics and systems have popped into your mind while doing this week's workshop? Discuss how you might be able to create those with your current knowledge.

CONGRATULATIONS!



You've finished Workshop
One of the Fast Track!

