**I RECOMMEND VIEWING THIS DOCUMENT IN WEB LAYOUT FOR EASE OF VIEWING**

Prior Knowledge for this tutorial:

* Already have the Unity engine installed and know how to create a new project
* Already have Microsoft Visual Studio installed
* Know how to create new objects within a scene
* Know how to move between folders in Unity’s project window and create new empty scripts
* Know how to change the scale and position of game objects within the unity editor

In this tutorial I will be showing you how to use a random number generator to spawn object at random position within a scene. The aim of this tutorial is to create a script that will spawn an object within the scene at a random position within a specified range. This system can be used within a game to spawn items and pickups or even enemies at random locations within a scene.

Let’s start by creating a cube within our scene, this will act as the object we want to spawn in. Since this is just a simple object that we are using for testing we don’t actually have to do anything to it, just name it what you want and save it as a prefab so that we can spawn it in later.

To do this, create a new folder within Assets in our project window and name the folder “Prefabs”.

A screen shot of a phone

Description automatically generated

Now simply drag the cube from the hierarchy into the prefabs folder to save it as a prefab for later. You can remove the cube from the scene if you wish as we no longer need it there.

Now let’s create an empty object within the scene, this will hold our spawning script so let’s call it “Spawn Handler”.

Now we need to make the script for spawning the cubes into our scene. Let’s create a new script, call it “Random Spawn” and open it.

First things first we need the script to know what it is supposed to spawn in so let’s create a new variable for a game object by typing “public GameObject spawnItem;”



We’ll use this variable to store our cube prefab so we can spawn it in later

Since we want the spawning to occur when a button is pressed we will need to add an if statement within “Update()” to check for when the button has been pushed, we’ll use spacebar as the button for now. For this let’s write “if (Input.GetKeyDown(KeyCode.Space))”

A black screen with white text

Description automatically generated

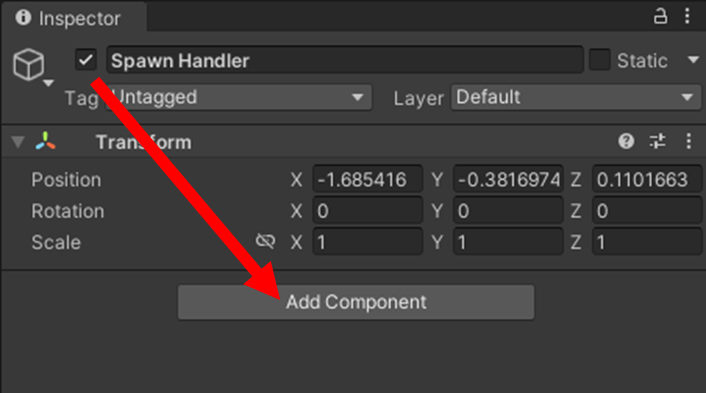
Make sure you use “GetKeyDown” and not “GetKey” as we want the item to spawn once for each press of the spacebar. “GetKey” will instead result in a new cube spawning for each frame the spacebar is held down.

Within our if statement we want to use Instantiate to spawn in the cube so let’s type “Instantiate(spawnItem)”. This code will now check for when the spacebar is pressed and spawn in a cube when it is.

A screen shot of a computer

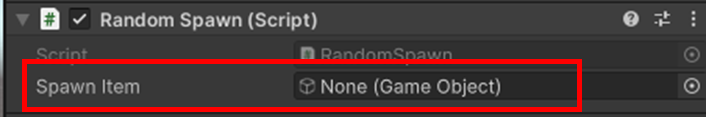
Description automatically generated

Let’s give this a quick test. Return to unity and add the script we just made to the Spawn Handler object. Click on the Spawn Handler in the hierarchy to view it in the inspector to the right. Then click on the “Add Component” button and select our script.

 A screenshot of a computer

Description automatically generated

You’ll notice that you can see the spawn item variable bit there is nothing inside it.

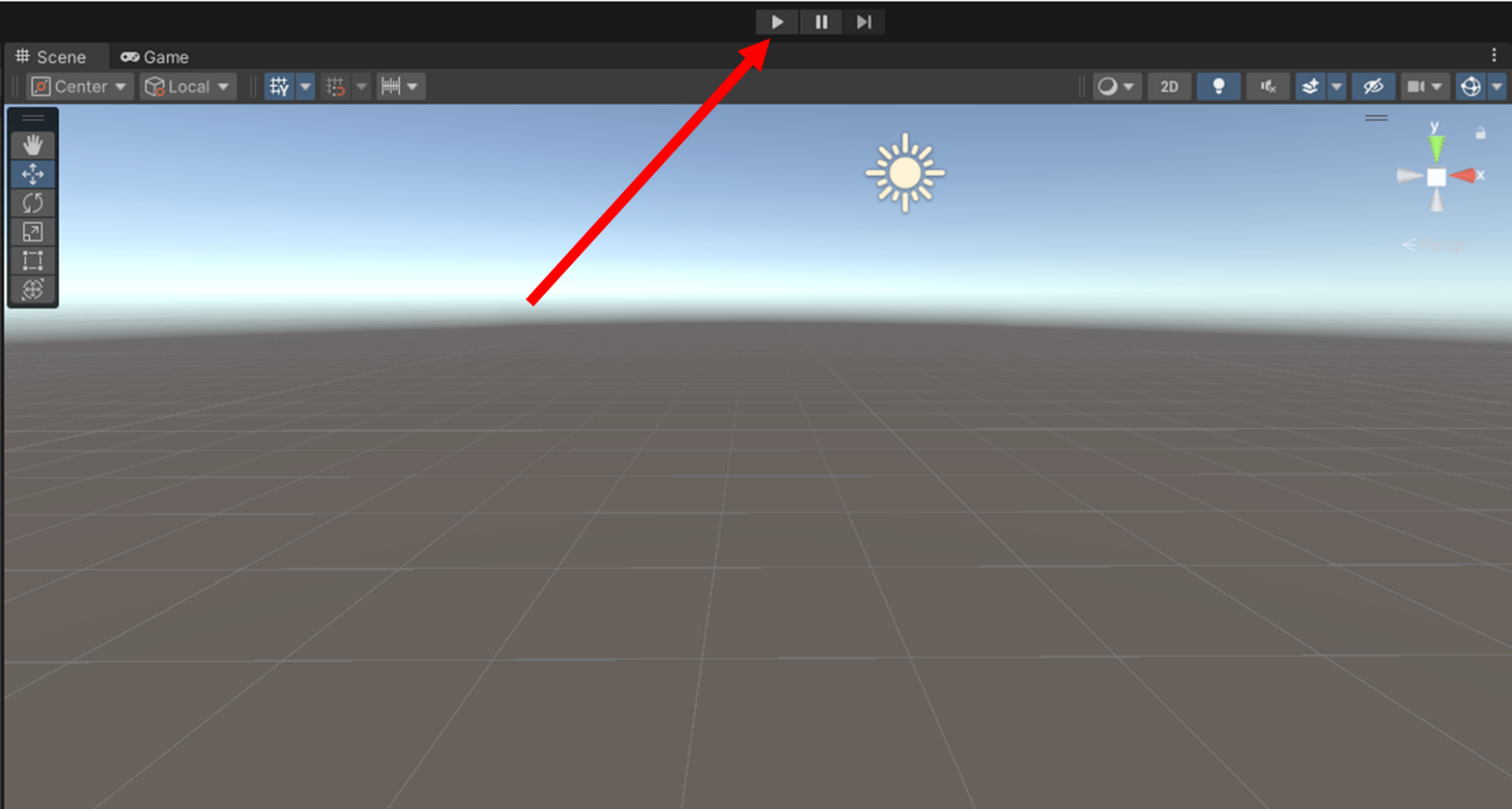


We need to assign the cube prefab that we made earlier to the variable, to do so simply drag it from the project window onto the spawn item variable in the inspector, once done is should look like this

A screenshot of a computer

Description automatically generated

Now to test our script. Enter play mode by clicking the play button above the scene.



If the code works properly then you should be able to simply press the spacebar and a cube will appear.

A white cube in a flat area

Description automatically generated

If you try pressing the spacebar again it may look like nothing happens but if you look at the hierarchy you can see that more cubes are spawning every time you press the key but they are all spawning in the same spot on top of each other making it look like there is only one cube in the scene.

A screenshot of a computer

Description automatically generated

We want these cubes to all spawn in different spots decided at random. For that we can use additional variables and Random.Range.

We want to make sure a new random position is generated for each spawn so let’s create a new function that we can call when spacebar is pressed rather than directly spawning the cube that we can reroll the random number generator every time the button is pressed.

Returning to our code, lets create a function outside of Update() and call it RunSpawn

A black screen with white text

Description automatically generated

Now within this function let’s create some new variables. We want three different variables, one for the items position along each axis (whether you want to use all three axis or not is entirely up to you but for the purpose of this tutorial I will be using all three). We want these variables to be floats so that the random number generator can generate decimal numbers as well.

A black screen with white text

Description automatically generated

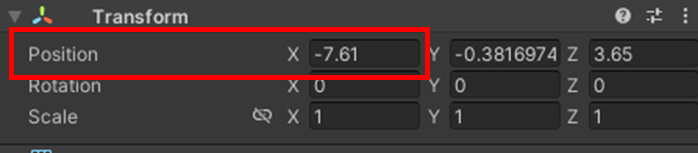
These will serve as our position variable however they are currently empty, we want them to be filled by random numbers. For that we use Random.Range which will allow us to generate a random number between two specified numbers. But first we need to decide what numbers we want to specify. First we’ll start with the X axis so let’s quickly return to our scene in unity to do that.

Let’s add another cube back to our scene and use the editor to move it along the x axis by clicking and dragging on the red arrow

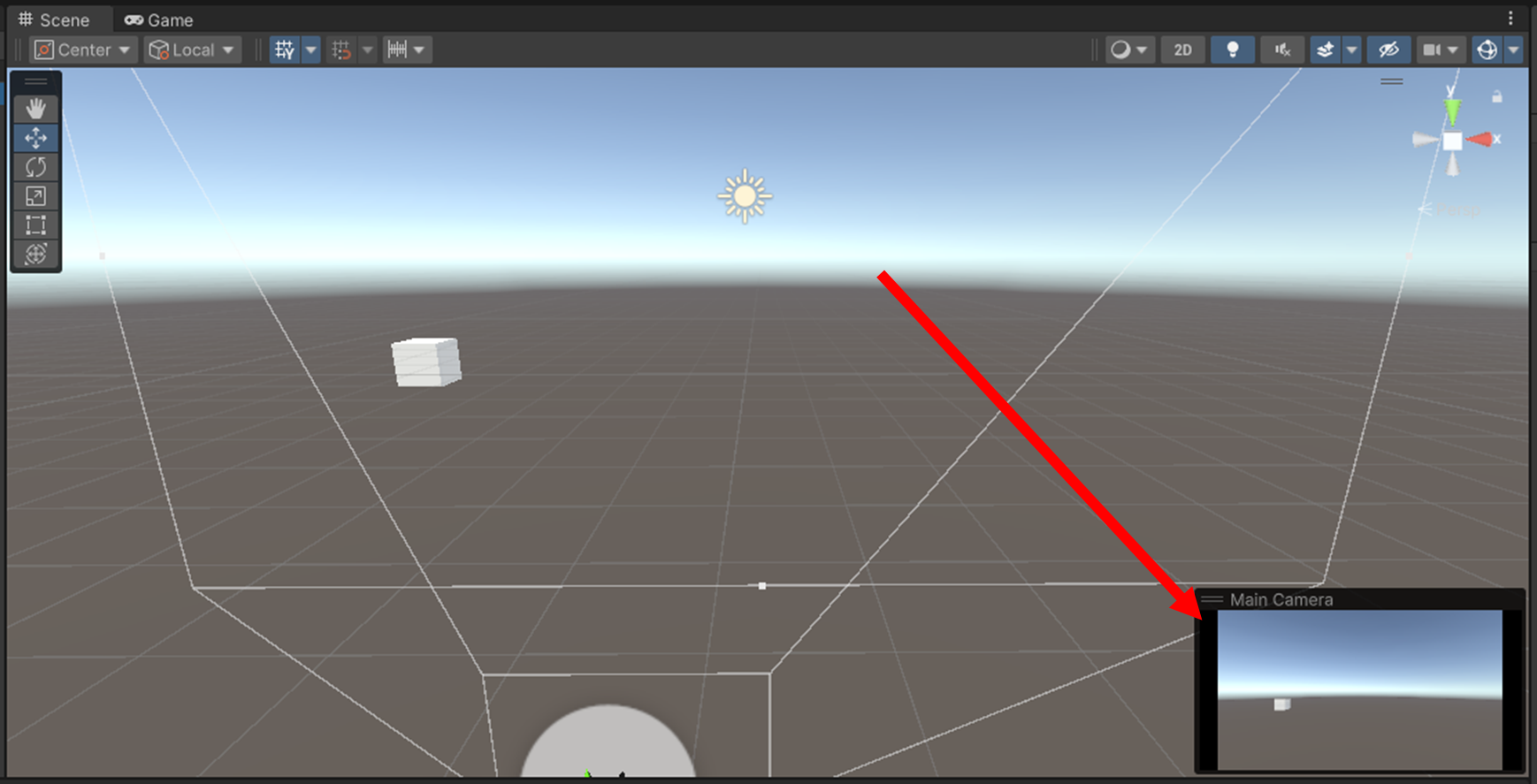
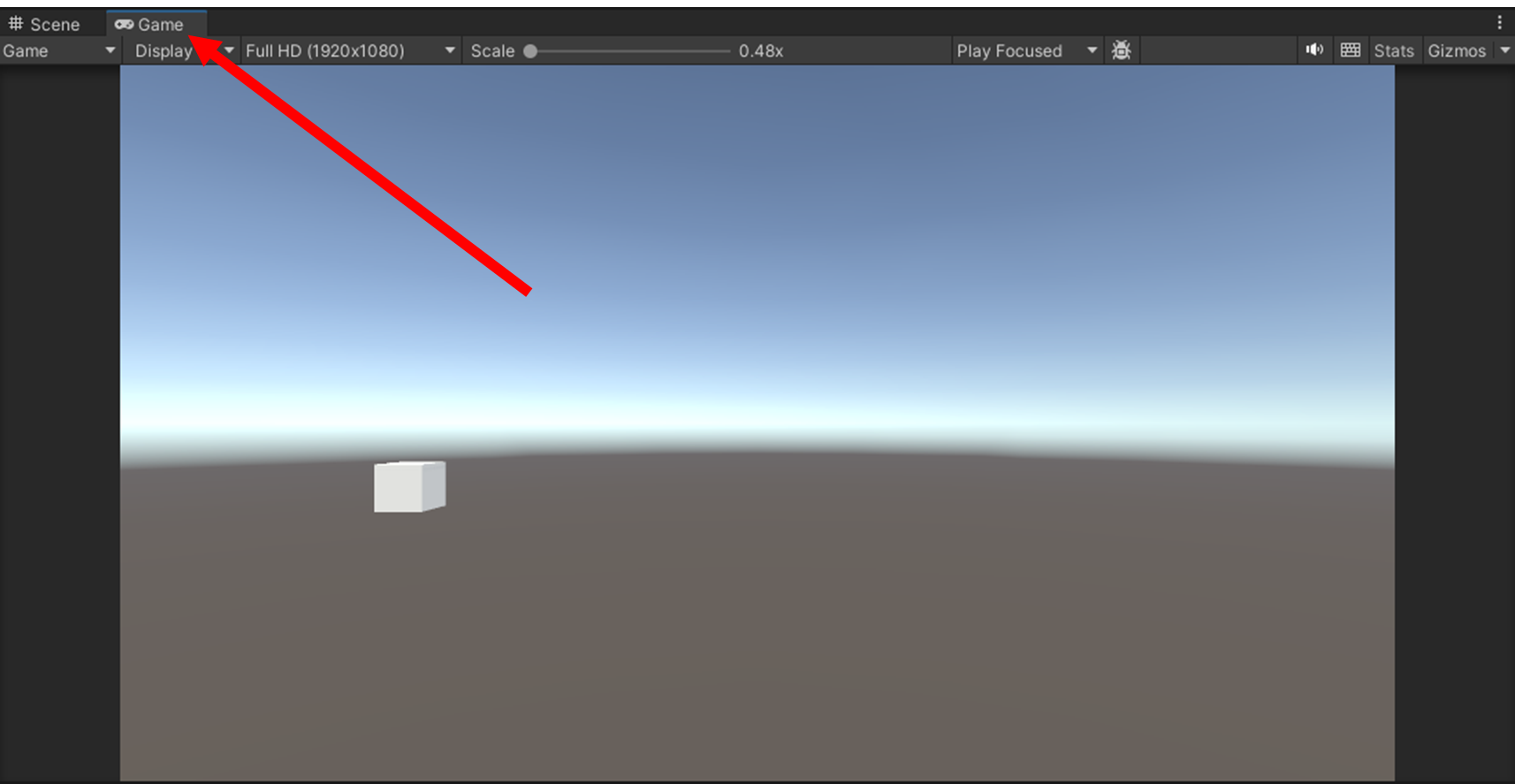
A computer generated image of a cube with arrows and a red square

Description automatically generated

While you move your cube around keep an eye on its position in the inspector, we’ll use this to determine what numbers we will specify for random.



We still want our cube to visible when it spawns in so check the camera view to make sure the positions you decide on are still visible by the camera. Do this by either switching to game view at the top or clicking on main camera in the hierarchy to display the main cameras vision in the scene view.



Once you have decided on you positions return to your code. I have decided that I want my cube to spawn along the x axis between -12 and 12. So let’s add our Random.Range. When specifying what numbers within a Random.Range command make sure that the higher number is actually a little more than the number you want to generate(this is because the second number is no actually included in the possible outcomes so by putting a number above the one you want to generate will insure that your desired maximum is included in the possible outcomes).

A black screen with white text

Description automatically generated

Repeat this process with the other two axis.

A black screen with white text

Description automatically generated

With our random generators complete we now need to instantiate our cube at the positions stored within these variables. Let’s add an instantiate command below our variables.

A computer screen shot of a code

Description automatically generated

Our Instantiate command now takes a Vector3 value to which we pass the values stored in the variables that contain our random numbers and spawns the cube at that position. “Quaternion.identity” affects the item’s rotation, it is not important to understand for this tutorial however make sure you include it in your command.

Lastly, we need to edit our if statement from before. Instead of instantiating a cube within that if statement. We need to call out new command to lets replace our original instantiate with “RunSpawn();”

A computer screen shot of a black screen

Description automatically generated

All code within RunSpawn() will run each time it is called, and since it is called each time the space bar is pressed this mean a new set of random numbers will generate every button press resulting in a completely random position for the spawned cube.

Now if you return to Unity and test our code again, you should find that the cubes now spawn in different positions each time you press the spacebar.

A screenshot of a computer

Description automatically generated

You now have cubes that spawn in random locations each time.

But what if you want to take this a little further? What if you want the item that spawns to also be random? Well you can do that with a similar method. Let’s create a couple more objects, this time let’s use a sphere and a cylinder and like before let’s make them prefabs.

A white cylinder and a white ball

Description automatically generated

Let’s go back to our code and create some more variables. Below our spawnItem variable we will create two more GameObject variables to store our other two objects.

A black background with white letters

Description automatically generated

We need to create another empty GameObject variable to be used later. We want this one empty for now but we also don’t want it to be edited through the inspector so let’s make this one private instead of public.

A black screen with white text

Description automatically generated

Now we need to determine which item will spawn when we press spacebar so let’s return to our RunSpawn() function. We need to determine the item before we spawn it so let’s add out new lines of code above the other lines within the function.

Fire we will need to generate a new random number. This time we aren’t working with positions so let’s generate an integer instead of a float this time so that each item has a 1/3 chance to be spawned. Remember to put the higher number at 4 so that 3 can be generated.

A computer code on a black background

Description automatically generated

Now we will need to assign out empty GameObject variable one of the object from the other variables depending on which number is generated. Let’s spawn our cube if the number is one so I’ll add an if statement below our itemNo variable.

A computer screen shot of a black screen

Description automatically generated

Within here we want to assign itemToSpawn to match the variable our cube is in, which is spawnItem.

A computer screen shot of white text

Description automatically generated

Repeat this for the other two items and possible numbers and make sure to change the item your are instantiated to itemToSpawn otherwise you will still spawn a cube everytime. When you are done it should look like this.

A computer screen shot of a black screen

Description automatically generated

Now we just need to assign our prefabs for the other two objects same as before.

A screenshot of a computer script

Description automatically generated

Now we can test one final time.

A group of white objects floating in the air

Description automatically generated

Congratulations. You can now spawn random objects at random positions within a scene.

Thanks for reading.