# Unit 1- Lesson 8. Comparing OnCollisionEnter, OnTriggerEnter and More Player Controls

**Aim:**

* What are the differences between OnCollisionEnter() and OnTriggerEnter?
* When shall we use OnTriggerEnter()?
* How do we move the player controlled character forward and backward?

**Objectives:** After the lesson, students should be able to:

* Obtain basic understanding of OnCollisionEnter() and OnTriggerEnter()
* Write program in Unity and C# to simulate collision detection in 3-D
* Write simple program to control player movements

**Motivation:**

* 3-D computer graphics simulation / video games

***Common Core:***[***CCSS.MATH.PRACTICE.MP4***](http://www.corestandards.org/Math/Practice/MP4/)***Model with mathematics.***

Mathematically proficient students can apply the mathematics they know to solve problems arising in everyday life, society, and the workplace.

**CLASS PROCEDURE:**

***Do Now:***

Open your Rolling Sphere project in Unity. Choose a game object that you would like to have the sphere to “pick up”, for an example, a cube. Highlight the cube, and under the Inspector panel, **check the “Is triggered” option** under the collider. Go to the script, replace the OnCollisionEnter (Collision other) with the OnTriggerEnter(Collider other), and test the game.

void OnTriggerEnter(Collider other) {

if (other.gameObject.CompareTag ("Pick Up"))

{

other.gameObject.SetActive (false);

}

}

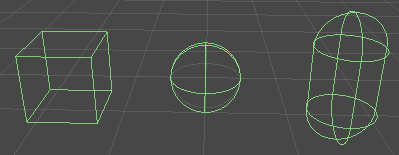
***Class Presentation / Discussion [10 mins]:***

1. What are the differences between OnCollisionEnter() and OnTriggerEnter()? When shall we use OnCollisionEnter and when shall we use OnTriggerEnter?

* In Unity, we can use the Collider as a trigger! One of the advantage of using OnTriggerEnter() is that we can turn it on and off very easily.
* When you use the OnTriggerEnter(), the game object doesn’t need to have a rigidbody, but the “Is triggered” option must be selected. Also, the game character (like the sphere) must have a rigidbody!

1. What is Unity collider and how do we use the collider component in Unity?

* Collider components define the shape of an object for the purposes of physical collisions. A collider, which is invisible, need not be the exact same shape as the object’s mesh and in fact, a rough approximation is often more efficient and indistinguishable in gameplay.
* The simplest (and least processor-intensive) colliders are the so-called primitive collider types. In 3D, these are the [Box Collider](http://docs.unity3d.com/Manual/class-BoxCollider.html), [Sphere Collider](http://docs.unity3d.com/Manual/class-SphereCollider.html) and [Capsule Collider](http://docs.unity3d.com/Manual/class-CapsuleCollider.html). In 2D, you can use the [Box Collider 2D](http://docs.unity3d.com/Manual/class-BoxCollider2D.html) and [Circle Collider 2D](http://docs.unity3d.com/Manual/class-CircleCollider2D.html). Any number of these can be added to a single object to create compound colliders.



* With careful positioning and sizing, compound colliders can often approximate the shape of an object quite well while keeping a low processor overhead. Further flexibility can be gained by having additional colliders on child objects (eg, boxes can be rotated relative to the local axes of the parent object). When creating a compound collider like this, there should only be one Rigidbody component, placed on the root object in the hierarchy.

***Discussion: How can we make a player controlled character moving forward and backward without rolling? What if we want to define the key control in the program?***

You may use Input.GetKey(KeyCode.xxxxxx) to take the user’s key input. The syntax is provided as follows.

void Update () {

if([Input.GetKey](http://docs.unity3d.com/Documentation/ScriptReference/Input.GetKey.html" \t "_blank)([KeyCode.UpArrow](http://docs.unity3d.com/Documentation/ScriptReference/KeyCode.UpArrow.html" \t "_blank)))

transform.Translate([Vector3.forward](http://docs.unity3d.com/Documentation/ScriptReference/Vector3-forward.html) \* 10f \* [Time.deltaTime](http://docs.unity3d.com/Documentation/ScriptReference/Time-deltaTime.html));

if([Input.GetKey](http://docs.unity3d.com/Documentation/ScriptReference/Input.GetKey.html" \t "_blank)([KeyCode.DownArrow](http://docs.unity3d.com/Documentation/ScriptReference/KeyCode.DownArrow.html" \t "_blank)))

transform.Translate(Vector3.back \* 10f \* [Time.deltaTime](http://docs.unity3d.com/Documentation/ScriptReference/Time-deltaTime.html));

}

***Pair – sharing Activity:***

Continue working on your project. Due: Friday, Sept. 27th.