The following exercises assume you have completed the previous worksheets.

You will need to refer back to previous examples and exercises and transfer the concepts and techniques introduced when carrying out this worksheet.

This worksheet assumes you have successfully completed worksheet eight.

The previous worksheet (worksheet eight) showed you how to control a third-person character. This worksheet will show you a technique to capture the character when it falls from the ground plane and reposition it in the World.

1. Launch Unity 3d and load the scene you created for worksheet eight. Choose **Save A**s and save the scene as **Transporter**. Your **ThirdPerson** project should now contain two scenes**, Start** and **Transporter**. The **Transporter Scene** will now be your starting point for this tutorial.
2. At the moment, if the character falls off the ground plane the only way to get it back is to restart the scene. We need to capture the character and reposition it back on the ground plane. One method is to create a **collider** below the ground plane and **trigger** an **event** that can then be used to **transform** the characters **position**.
3. Create a cube object and modify its scale so that it has an area larger than the ground plane, around, **x=30**, **z=30** and a thickness of **y=1**. Position its centre to coincide with the centre of the ground plane and around - **11 y** units below the ground plane. Name it **Capture Tray** and in the **Inspector Window** check **Is Trigger**.
4. Download the **package** **‘transporter’** from the module homepage. This is a Prefab to represent a visual object that the character will be transported to when it falls from the ground plane. It also features a particle system for visual interest and some audio effects.
5. To import the transporter package select –**Assets – Import Package** – **Custom Package** and navigate to the downloaded file.  
   In the dialogue box at the bottom right select **Import**.
6. There should now be a Prefab called **RespawnPrefab** in the **All Prefabs Search Window**.
7. From the **Project Window** drag the **RespawnPrefab** into the World and position it onto the ground plane somewhere to the right of the player’s initial position and rename it **Transporter One**.
8. Play the scene and try walking the player onto the transporter.
9. Create a new C# script and name it **SpawnPoint** .

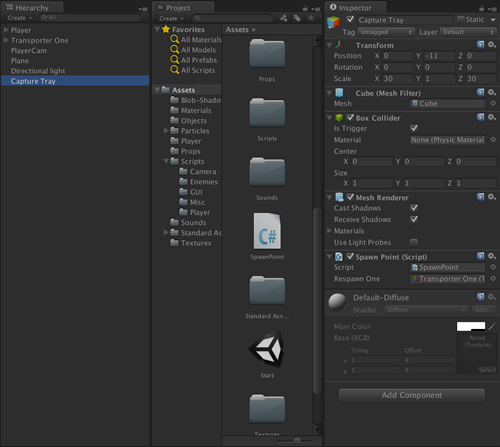
Declare a variable at top of the script after the public class declaration:

public Transform RespawnOne;

1. Next enter the following code **after the end** of the Update method:

public void OnTriggerEnter(Collider other){  
  
        if (other.tag == "Player")  
        {  
          other.transform.position = RespawnOne.position;  
        }  
    }

1. Save the script and attach it to the **Capture Tray**.
2. Select the **Capture Tray** in the **Hierarchy Window**. Now drag and drop the **Transport One** **object name** next to the **Respawn One Field** in the **Inspector Window**, (See illustration below).



This sets up a link between the **variable RespawnOne** and the **Transform (x,y,z) position** of the **Transporter One** object

1. Play the scene and make the player jump off the ground plane. The character will collide with the **Capture Tray** object that is set to **trigger** the event to **reposition** the player at the **x,y,z coordinate** **point** of the **Transporter One** object.
2. Depending on the position of the transporter one object, the repositioning of the camera may result in some camera occlusion. This can be improved with some additional techniques. Also the Capture Tray is currently visible in the scene. To hide the Capture Tray object select it in the **Hierarchy Window**, then in the **Inspector Window,** **uncheck** the box next to **Mesh Renderer**. The **mesh** is now **hidden** but the **trigger collider** is still **active**.

**Exercises:**

1. Does the player get repositioned in every case? If not you will need to increase the area of the Capture Tray.

1. Using the example above, try creating another Transporter object that transports the player from one side of the plane to the other.