Semra Usanmaz

Programming tutorial – Picking up objects and gaining points from them

This tutorial teaches you how to pick up objects and gain points to the overall score from doing so.

This script follows on from the third tutorial on 'Movement and jumping' and so that tutorial should be completed first before this one.

1. Set up the scene

If you have followed the 'Movement and jumping' tutorial, the scene should already have a player and a platform on it with their respective names, tags and movement code.

Add a sphere to the scene, this will be the object that is picked up, and place it on top of the platform, name this object "pickUp" and give it the tag "Object". Remove the sphere collider on it, add a circle collider instead, and enable 'is trigger'. Enabling 'is trigger' stops the object from registering collider collision and instead it registers when another collider enters/exits its collider. This stops the object from being pushed away if the player object was to move towards and touch it.

Create a "Prefabs" folder within the project and drag in the "pickUp" object. This should turn it into a prefab. Making an object a prefab allows you to save it and duplicate it when needed with each duplicate having the exact same properties, this is better than copy/pasting the object as changing the prefab within the prefab folder affects every prefab within the scene, whereas if you were to copy/paste an object, you would need to change each of them individually.

The scene should look like this:
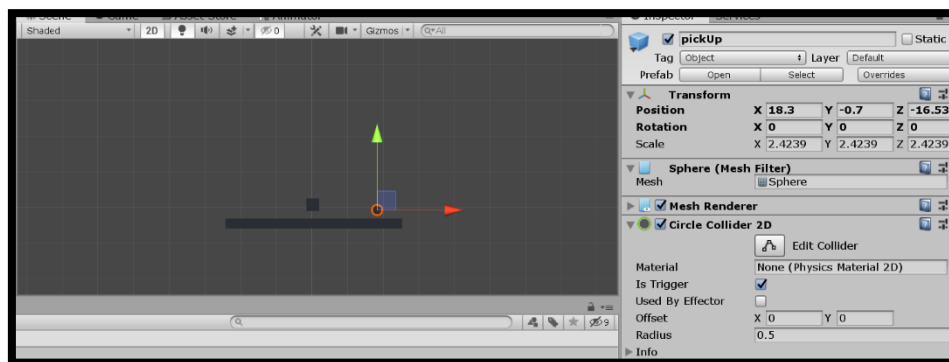
2. Picking up objects

Create a script called "pickUpObjects" and open it. First, make a reference to the object you just created by creating an array variable called "GameObject[] pickUp;", the reason as to why it is an array variable is so that the script can look for more than one of the same prefab object within the scene, if it wasn't you could only pick up one object . To do this, in the Start() function, type:

"pickUp = GameObject.FindGameObjectsWithTag("Object");"

This will search for the game objects with the tag "Object" and know that those are the 'Objects' you are referring to.

This should look like:

```
− references
public class pickUpObjects : MonoBehaviour
{
    GameObject[] pickUp;


    // Start is called before the first frame update
    − references
    void Start()
    {
        pickUp = GameObject.FindGameObjectsWithTag("Object");

    }
```

After this, you will need to make it so that when the player object enters the colliders of the pickup objects, they are removed from the scene.

To do this, below the Update() function you will need to create an 'OnTriggerEnter2D' function:

"private void OnTriggerEnter2D(Collider2D collision)

{

}

"

This checks whether something with a collider has entered another collider and if so, starts the code within the brackets.

Inside the brackets, make an 'if' statement referencing the 'pickUp' object's tag:

"if(collision.gameObject.tag == ("Object"))

{

}

"

And within these brackets, write:

"Destroy(collision.gameObject);"

The "Destroy(collision.gameObject)" function removes the object within the brackets (in this case the object the player collides with) from the scene.

This code overall should look like:

```
− references
private void OnTriggerEnter2D(Collider2D collision)
{
    if(collision.gameObject.tag == "Object")
    {
        Destroy(collision.gameObject);

    }
}
```

Save the script, go back to the Unity project, and drag this script onto the 'Player' game object.

If you click on the play button now and move the player character onto the 'pickUp' object, it should disappear as soon as you touch it and so the object has been 'picked up'.

3. The Score Canvas

Create a new canvas within the Unity project and name it "scoreCanvas" and add text to it called "Score", change the text within it to "Score:" as well. Move the "Score" text to an area on the screen where you want the score to be (e.g. one of the corners of the screen) and add another text to it which will represent the actual point value, change the text from "New Text" to "0" and name this object "scoreText". Move this to the right side of the "Score:" text. Resize both to any size you want.

The objects in the scene should be:
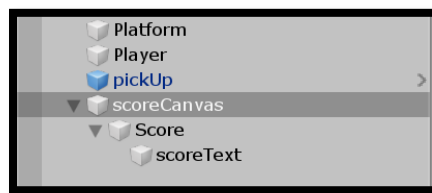
The current in-game scene should look something like:

Once you have done this, go back to the "pickUpObjects" script and type "Using UnityEngine.UI;" beneath "using UnityEngine;" in order to allow the script to work with UI components such as 'Text', 'Canvas' etc.

Make a reference to the score text by creating a "public Text Score;" variable under the "GameObject pickUp;" variable. When this is done, save, go back to the Unity project, and drag the "scoreText" text into the new variable box on the 'Player' object - This will be the one that will increase in value once an object is picked up.
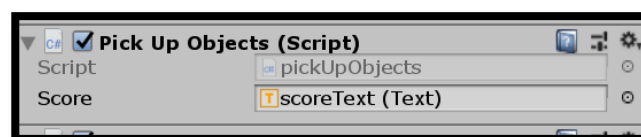
Beneath the "public Text Score;" variable, create a private int called "index" which will be holding the points/score – type: "private int index". Within the Start() function, to ensure that at the start of the game the index/the score is 0, type "index = 0".

After this, within the Update() function, add: "Score.text = index.ToString();", this will ensure that the index number (in string format instead of int) is equal to the score text and that the text updates if the index number changes.

The script at this point, without the "OnTriggerEnter2D" section, should look like this:

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI;

references
public class pickUpObjects : MonoBehaviour
{
    GameObject[] pickUp;

    public Text Score;
    private int index;


    // Start is called before the first frame update
    references
    void Start()
    {
        pickUp = GameObject.FindGameObjectsWithTag("Object");
        index = 0;

    }

    // Update is called once per frame
    references
    void Update()
    {
        Score.text = index.ToString();

    }
```

4. Adding points to the score when an object is picked up

In the same script, create a new public variable called "public int Points;" and make it equal to the number of points you want the object to add to your current score once you pick it up. In this example, the number of points will be 20 – "public int Points = 20;". You cannot use decimal places for the number of points in this case as they are not integers.

Then, within the 'if' function in the 'OnTriggerEnter2D" function, make it so that when the collision between the objects is detected, the index raises by the "Points" value you just created. To do this type:

"index =+ Points;".

The final look of the script should be similar to this:

```
references
public class pickUpObjects : MonoBehaviour
{
    GameObject[] pickUp;

    public Text Score;
    private int index;
    public int Points = 20;

    // Start is called before the first frame update
    references
    void Start()
    {
        pickUp = GameObject.FindGameObjectsWithTag("Object");
        index = 0;

    }

    // Update is called once per frame
    references
    void Update()
    {
        Score.text = index.ToString();

    }
    references
    private void OnTriggerEnter2D(Collider2D collision)
    {
        if(collision.gameObject.tag == "Object")
        {
            Destroy(collision.gameObject);

            index += Points;

        }
    }
```
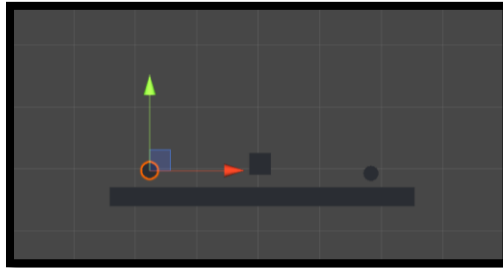
Go back to the Unity project and press play, the score should increase by 20 when you pick up the object.

To test the script out more, drag another one of the pickUp prefabs into the scene and place it on the other side of the platform, like this:

Click play once again, when you pick up both objects, the score should go from 0, to 20, to 40.