﻿using UnityEngine;

using System.Collections;

public class Resetter : MonoBehaviour {

public Rigidbody2D projectile; // The rigidbody of the projectile

public float resetSpeed = 0.025f; // The angular velocity threshold of the projectile, below which our game will reset

private float resetSpeedSqr; // The square value of Reset Speed, for efficient calculation

private SpringJoint2D spring; // The SpringJoint2D component which is destroyed when the projectile is launched

void Start ()

{

// Calculate the Resset Speed Squared from the Reset Speed

resetSpeedSqr = resetSpeed \* resetSpeed;

// Get the SpringJoint2D component through our reference to the GameObject's Rigidbody

spring = projectile.GetComponent <SpringJoint2D>();

}

void Update () {

// If we hold down the "R" key...

if (Input.GetKeyDown (KeyCode.R)) {

// ... call the Reset() function

Reset ();

}

// If the spring had been destroyed (indicating we have launched the projectile) and our projectile's velocity is below the threshold...

if (spring == null && projectile.velocity.sqrMagnitude < resetSpeedSqr) {

// ... call the Reset() function

Reset ();

}

}

void OnTriggerExit2D (Collider2D other) {

// If the projectile leaves the Collider2D boundary...

if (other.GetComponent<Rigidbody2D>() == projectile) {

// ... call the Reset() function

Reset ();

}

}

void Reset () {

// The reset function will Reset the game by reloading the same level

Application.LoadLevel (Application.loadedLevel);

}

}