Introduction to C# programming and Unity

Week 3 - Exercise 16

Shrink and grow

The **Time** class in the unity is an important aspect in designing the games since it involves properties to fetch the various timing details. This exercise makes use of **Time.deltatime** which returns the completion time in seconds for the last frame i.e., the time between the current frame and previous frame.

1. Grow out of control

The 'resizer' script attached to the object is in such a way that for every second, the object doubles the size. After a point the console window says that **transform.localScale assign attempt for 'goldball' is not valid. Input localScale is { Infinity, Infinity, 1.000000 }.** This means that the game object has grown infinitely large.

2. Shrink and grow

Minor changes are made to the previous script to scale the resizing of the object for a fixed time which in turn decides the speed of the shrink and grow.

The TotalResizeSeconds decides the time interval for which the object should shrink and grow
The ScaleFactorPerSecond decides how large the object can grow and shrink for a given amount of time
The ScaleFactorSignMultiplier is responsible for reversing the action of grow and shrink for every time interval as specified by the TotalResizeSeconds.

```
void Update()
  {
     Vector3 newscale = transform.localScale;
     newscale.x += ScaleFactorSignMultiplier * ScaleFactorPerSecond * Time.deltaTime ;
     newscale.y += ScaleFactorSignMultiplier * ScaleFactorPerSecond * Time.deltaTime ;
     transform.localScale = newscale;

     elapsedResizeSeconds += Time.deltaTime;
     if(elapsedResizeSeconds >= TotalResizeSeconds)
     {
        elapsedResizeSeconds = 0;
        ScaleFactorSignMultiplier *= -1; //reversing the direction alternates the action of the object to shrink and grow
alternatively

     }
  }
```