```
// BuoyancyQuata.cs

using UnityEngine;
using System.Collections;

public class BuoyancyQuata : MonoBehaviour
{
    public float m_Speed;
    public float m_TiltAmount;
    public float m_Phase;

    Vector3 forward;

    public float buoQSwitch = 1;

    private float buoPow = 1.0f;

    private bool buoPlus = false;
    private bool buoMinus = false;
    private bool buoDelete = false;


    // Use this for initialization
    void Start()
    {
        forward = Quaternion.Euler(0f, Random.Range(0f, 360f), 0f) * transform.forward;

        m_Phase = Random.Range(0f, Mathf.PI * 2);
    }

    // Update is called once per frame
    void Update()
    {

        Vector3 v = Vector3.up + Vector3.right * Mathf.Sin(Time.time * m_Speed + m_Phase) *
m_TiltAmount * buoPow * Time.deltaTime;
```

```
        transform.LookAt(transform.position + forward, v);


        if (buoPlus == true)
        {
            buoPow += 0.05f;
            if (buoPow >= 1.7f)
            {
                buoPlus = false;
            }
        }
        if (buoMinus == true)
        {
            buoPow -= 0.03f;
            if (buoPow <= 1.0f)
            {
                buoMinus = false;
            }
        }
        if (buoDelete == true)
        {
            buoPow -= 0.08f;
            if (buoPow <= 0.0f)
            {
                buoPow = 0.0f;
                buoDelete = false;
            }
        }

    }
    public void plus()
    {
        buoPlus = true;

    }
```

```
    public void minus()
    {
        buoMinus = true;
    }

    public void resetBuo()
    {
        buoDelete = true;
    }
}
```