

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class Camera : MonoBehaviour
{
    public GameObject player;
    private Vector3 offset;

    public enum RotationAxes { MouseXandY = 0, MouseX = 1, MouseY = 2 };
    //Випадаюче меню для налаштування осей обертання
    public RotationAxes axes = RotationAxes.MouseXandY; //отримуємо осі з
даного елемента
    //Змінні чутливості мишки по осі X та по осі Y
    public float sensivityX = 2f;
    public float sensivityY = 2f;
    //змінні призначені для обмежень по максимальним кутам повороту для
осі X
    public float minimumX = -20f;
    public float maximumX = 20f;
    //змінні призначені для обмежень по максимальним кутам повороту для
осі Y
    public float minimumY = -20f;
    public float maximumY = 20f;

    //Змінні для визначення нинішнього кута повороту
    public float rotationX = 0f;
    public float rotationY = 0f;

    //Змінна яка містить Основний тип повороту (Quaternion)
    Quaternion originalRotation;

    void Start()
    {
        offset = transform.position -
player.transform.position; //дізнаємося який відступ від об'єкта до камери
        //заборона обертання нашого твердого тіла якщо камера не буде
рухатись
        if (GetComponent<Rigidbody>())
        {
            GetComponent<Rigidbody>().freezeRotation = true;
        }
        originalRotation = transform.localRotation;
    }
    public static float ClampAngle(float angle, float min, float max)
    {
        //умови дає змогу крутитись безкінечно навколо своєї осі
        if (angle < -360f)
            angle += 360f;
        if (angle > 360f)
            angle -= 360f;
        return Mathf.Clamp(angle, min, max); //функція обраховує мінімум
максимум і саме значення англ(кута і поверне його для подальшого
використання)
    }
    void Update()
    {
        if (axes == RotationAxes.MouseXandY) //Якщо рухати мишкою по двом
осям X і Y
        {
            rotationX += Input.GetAxis("Mouse X") * sensivityX;

```

```

        rotationY += Input.GetAxis("Mouse y") * sensityvityY;

        rotationX = ClampAngle(rotationX, minimumX, maximumX); //по
цій функції буде обраховуватись обертання по осі X
        rotationY = ClampAngle(rotationY, minimumY, maximumY); //по
цій функції буде обраховуватись обертання по осі Y
        Quaternion xQuaternion = Quaternion.AngleAxis(rotationX,
Vector3.up);
        Quaternion yQuaternion = Quaternion.AngleAxis(rotationY, -
Vector3.right);
        transform.localRotation = originalRotation * xQuaternion *
yQuaternion;
    }
    else
    if (axes == RotationAxes.MouseX) //Якщо рухати мишкою по осі X
    {
        rotationX += Input.GetAxis("Mouse X") * sensityvityX;
        rotationX = ClampAngle(rotationX, minimumX, maximumX);
        Quaternion xQuaternion = Quaternion.AngleAxis(rotationX,
Vector3.up);
        transform.localRotation = originalRotation * xQuaternion;
    }
    else
    if (axes == RotationAxes.MouseY) //Якщо рухати мишкою по осі Y
    {
        rotationY += Input.GetAxis("Mouse Y") * sensityvityY;
        rotationY = ClampAngle(rotationY, minimumY, maximumY);
        Quaternion yQuaternion = Quaternion.AngleAxis(rotationY, -
Vector3.right);
        transform.localRotation = originalRotation * yQuaternion;
    }
}
void LateUpdate() //функція визивається 1 раз в кінці кадру
{
    transform.position = player.transform.position + offset;
}
}

```