```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
/* Handles camera movement
 *
 * NOTE: zoomSpeed  - affects how fast we can zoom in and out
 *       zoomOutMin - affects how close we can zoom in
 *       zoomoutMax - affects how far out we can zoom
 *
 * Also we may want to rework our inputs(GetMouseButtonDown) if it becomes a problem.
 * Add a slide to our camera if a player swipes ~~~~
 */
public class CameraController : MonoBehaviour
{
    Vector3 touchStart;

    const float zoomSpeed = 0.01f;
    public float zoomOutMin = 1;
    public float zoomOutMax = 8;

    void Update()
    {
        if (Input.GetMouseButtonDown(0))
        {
            touchStart = Camera.main.ScreenToWorldPoint(Input.mousePosition);           // Stores our first touch.
        }
        if(Input.touchCount == 2)                                                                                   // Handles condition for zoom
        {
            Touch touchZero = Input.GetTouch(0);
            Touch touchOne = Input.GetTouch(1);

            Vector2 touchZeroPrevPos = touchZero.position - touchZero.deltaPosition;    // Stores position of previous touch locations.
            Vector2 touchOnePrevPos = touchOne.position - touchOne.deltaPosition;

            float prevMagnitude = (touchZeroPrevPos - touchOnePrevPos).magnitude;       // Converts our coordinates into a magnitude.
            float currentMagnitude = (touchZero.position - touchOne.position).magnitude;

            float difference = currentMagnitude - prevMagnitude;                        // Takes the difference of our two magnitudes.

            zoom(difference * zoomSpeed);                                                               // Zoom in
```

```
       or out depending on our magnitude.
    }
    else if (Input.GetMouseButton
     (0))                        // Handles condition
     for moving camera.
    {
   // Finds the difference of previous touchPos to current touchPos.
      Vector3 direction = touchStart - Camera.main.ScreenToWorldPoint
       (Input.mousePosition);
      direction.z =
       0;                                  //
       Removes z from our vector.
      Camera.main.transform.position +=
       direction;                    // We move our camera
       accordingly.
    }
    zoom(Input.GetAxis("Mouse ScrollWheel"));
  }
  void zoom(float increment)
{
 // Allows us to change camera size while as long as we are within our limits
   // (zoomOutMin and zoomOutMax).
    Camera.main.orthographicSize = Mathf.Clamp(Camera.main.orthographicSize -
     increment, zoomOutMin, zoomOutMax);
  }


}

```