Calculator

by Marjupi Software

About

This is a complete calculator made with the new Unity GUI for **Unity 4.6** and **Unity 5**. The package comes with complete source code for you to play with. This calculator accurately replicates the traditional Windows calculator that you would find on your desktop. You can use it as the basis for your own calculator to sell on the stores. For example you could skin it and add nice button images to it.

Learning

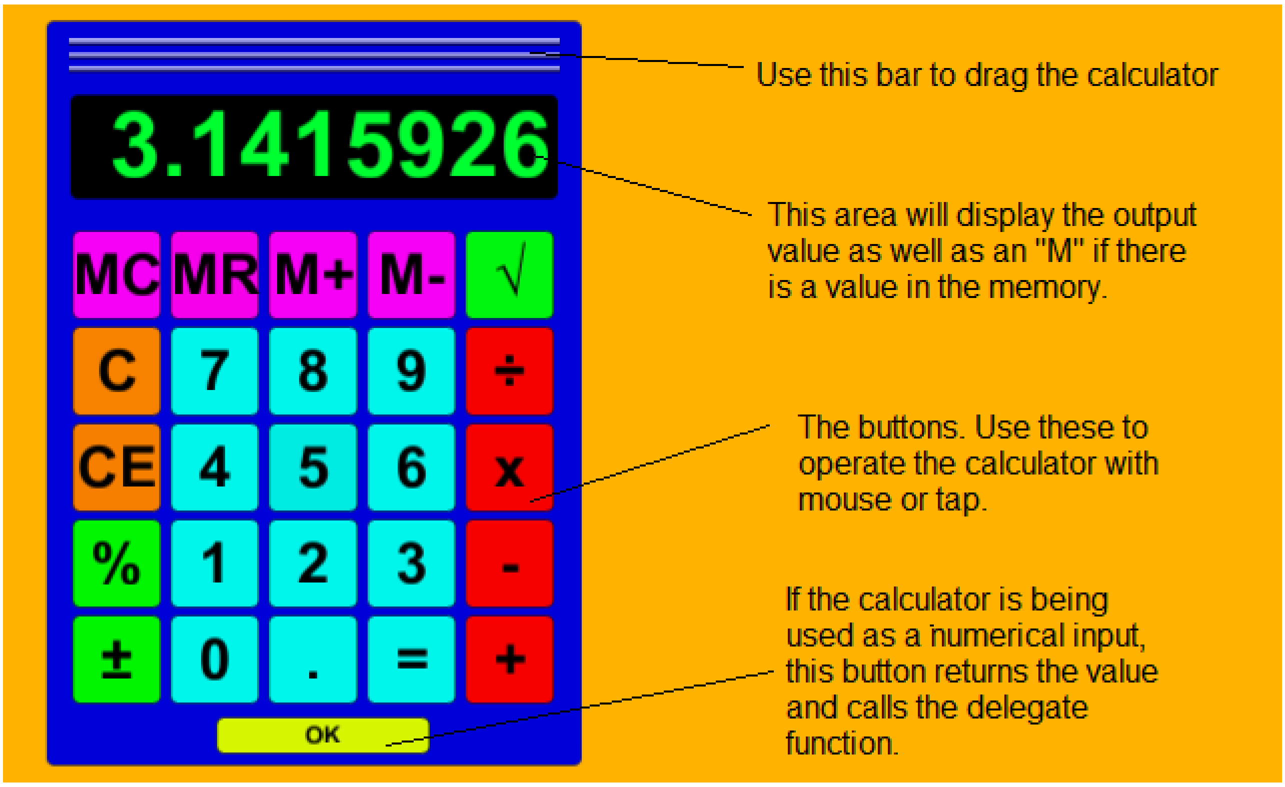
You will learn a lot about how the GUI (Graphical User Interface) system in Unity works by playing around with the scene and looking at the `calculator.as` C# script file.

Your Projects

You can use the calculator in any of your projects or as it is. Learn about how to incorporate the calculator into our projects later on.

Explanation of parts of the Calculator

The calculator consists of many parts. The buttons are all based on the same prefab which makes it easy to skin the calculator with your own button designs. All the parts are aligned with the top of the calculator and centred horizontally.



Use this bar to drag the calculator

This area will display the output value as well as an "M" if there is a value in the memory.

The buttons. Use these to operate the calculator with mouse or tap.

If the calculator is being used as a numerical input, this button returns the value and calls the delegate function.

The calculator displays a maximum of 8 digits. The smallest number it can show above zero is 0.0000001 and the largest number is 99999999. Above this number it displays NaN (not a number). When you divide by zero it displays Infinity. The normal rules apply for these “numbers”.

Using calculator in your project

You can use the calculator in your project by loading the package and then dragging the calculator asset onto the screen. If you want it to act as a numerical input you can call the calculator using the following script:

```
using UnityEngine;
using System.Collections;
using UnityEngine.UI;

public class CalculatorDisplayer: MonoBehaviour {

     //This is the place where you drag the calculator asset
     public Calculator calc;

     void Start () {
          //Set the callback for when the user clicks OK:
          calc.OKcallback = delegate(string f) {
               float value = float.Parse(f);

               /* Do something with the value here
               *
               * (Your code here)
               */

               //now close the calculator
               calc.gameObject.SetActive(false);
          };
     }

     //The calculator might be called by a button press
     public void OnCalcButtonPressed(){
          calc.gameObject.SetActive(true);
          //set the value of the calculator to something
          calc.reset(0);
     }


}
```

Ideas for Extending Calculator

For a challenge you might like to try converting it to handle complex numbers! Or try adding scientific functions to it. You might like to create a game which teaches people how to use a calculator by asking simple mathematics questions.