Based off Verison 1.00

Table of Contents:

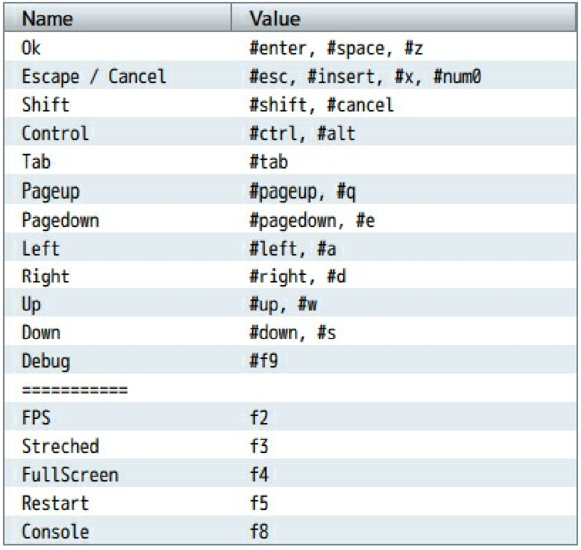
Input

Scripts that Require this

Remapping Default Keys

Remapping keys allows you to set which keyboard keys relate to default inputs. You can set them to multiple keys by seperating each key with a comma. You must use a Quasi Key (Full available keys below) Keys are case sensative!!! Means #C is not the same as #c !!IMPORTANT!!

WASD Setup Example:



Name	Value
Ok	#enter, #space, #z
Escape / Cancel	#esc, #insert, #x, #num0
Shift	#shift, #cancel
Control	#ctrl, #alt
Tab	#tab
Pageup	#pageup, #q
Pagedown	#pagedown, #e
Left	#left, #a
Right	#right, #d
Up	#up, #w
Down	#down, #s
Debug	#f9
===========	
FPS	f2
Streched	f3
FullScreen	f4
Restart	f5
Console	f8

* You can notice I also changed Pagedown to e. I Also left the old, left, right, up, down because it felt more natural using arrow keys in menus.

When remapping FPS, Streched, FullScreen, Restart, Console you can only put 1 key! Do not use the # idetifier when remapping these. If you want to disable one of those, set it to 0E, if you put in an incorrect key, it will use default MV key

Quasi Key List

Modifier, Editing, and Navigation keys

#backspace #tab #enter #shift #ctrl
#alt #esc #space #pageup #pagedown
#left #up #right #down #escap

Number keys, above letters

#0 #1 #2 #3 #4 #5 #6 #7 #8 #9

Numberpad

#num0 #num1 #num2 #num3 #num4
#num5 #num6 #num7 #num8 #num9

Letter keys

#a #b #c ... #z
(All letters between a-z US keyboard, just add a # infront)

F keys

#f1 #f2 #f3 #f4 #f5 #f6 #f7 #f8 #f9 #f10 #f11 #f12

Using Quasi Keys

If you haven't noticed by now, Quasi keys have a # identifier. So if you want to run a specific key, all you need to do is type in any of the Quasi Keys in the input field.

Examples

```
Input.isTriggered("#left"); // Will check for left arrow key
Input.isTriggered("left");  // Will check for remapped left, default is #left and #num4
```

There is a new Input function **.anyTriggered()** which will return true when any of the keys you pass in the parameter are pressed.

The Script Call:

```
Input.anyTriggered(keylist);
```

keylist: This can be set as a-z, a-z0-9, 0-9, sym or a list of keys seperated by commas.
a-z will check all letters between a and z.
0-9 will check all numbers between 0 and 9.
a-z0-9 will check both a-z and 0-9.
sym will check for symbol keys (, . / ; ' [ ] and \ ).

Input Text Window

To put this script to better use, I created a new window called Window_TextInput. This window works pretty similar to a selectable window. It needs to be activated before you can type. You can set a handler for **#enter** and **#esc**. I would recommend you to look at scripts that use this if you want to see how to use it.

Name Input

Requires: Name Input

This plugin will modify the name input scene, to allow for keyboard inputting.

> Just a few keys not working; quotes, braces and backslash. I think I broke gamepad, but I don't have one to test it pic.twitter.com/Gx0XxJt6h7
>
> — Quasi (@QuasiXi) November 20, 2015

You can also remove the old input window to make it look like:

Test

Easy Json

Requires: Easy JSON

This plugin is used to help set up the JSON files that are used in some of my other plugins.

Almost finished this editor thingy. pic.twitter.com/ViqEhE6BLZ

— Quasi (@QuasiXi) November 20, 2015