VIRTUAL KBOARD

```
W <=> {Any other key}

        Switch keycodes, use
        name for access!
public void UpdateKeys()    One name - any keycode!
{
    w = VKBoard.GetKey("W").code;
    a = VKBoard.GetKey("A").code;
    s = VKBoard.GetKey("S").code;
    d = VKBoard.GetKey("D").code;
}
```

Hi there! There is a Virtual KBoard tutorial. Here`s shown methods of work with it.

There`s no so much information, look the source code to get how it works.

Sorry for my English, if I made some mistakes.

1. Demo scene

The demo scene contains a virtual keyboard (keyboard layout with UI buttons). If you will press a key, key will go to changing code state. Next, you need press a new key for it. Then key saves, and next time it will use a new keycode. You can see how it works using WASD keys for navigation of camera on the scene. If you will change W to, for example, Q, next time you need to press Q key, instead of W.

2. Rules & Using

If you try to change key, name of which is already busy, and code also, keys will be switched.

If key doesn`t exists in the dictionary, it will be created.

If keycode already exists, but it hasn`t a name, it will be named.

If you want to get key by name using VKBoard.GetKey("keyname"); you will get a struct like this:

{

Int id;   //not used

String name;   //name of key

KeyCode code;   //code of key

KeyCode lastCode;   //not used

}

Then you can use it:

KeyCode hotkey = VKBoard.GetKey("hotkey name").code;

And use it in Input:

If(Input.GetKey(hotkey))

{

        DoSmth();

}

Look the Demo Scene and source code to get how it works.