

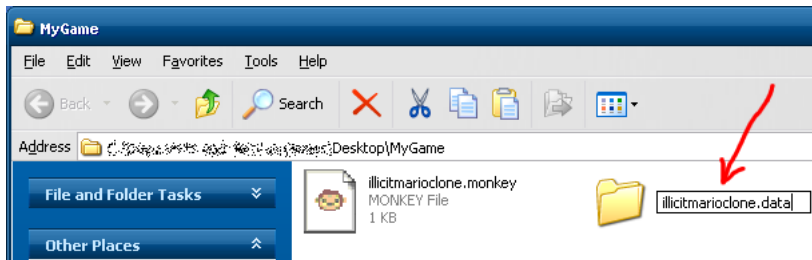
Creating a new Monkey project

There are many ways to create a new Monkey project. If you have no idea what you're doing when it comes to file management or folder creation, this article will get you started.

The screenshots below are from Windows XP, but the process is generally the same on all versions of Windows, and very similar on other operating systems such as OS X.

For experienced coders:

Simply create a new file called *illicitmarioclone.monkey* and a folder called *illicitmarioclone.data*, as below. You may wish to use a different name in place of *illicitmarioclone*...



Any data file you place in the data folder can be referred to directly from your code, using only the filename.

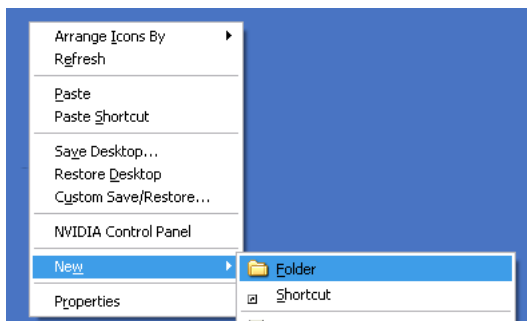
Open up *illicitmarioclone.monkey* in the Monk IDE, or your preferred code editor, and code away!

For absolute beginners:

The steps below may seem like a very involved process when fully laid out like this, but becomes trivial after a while, and you may even find your own quicker way of doing it.

Step 1

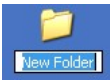
Create a new folder by right-clicking on a blank area of your desktop/backdrop. (Right-clicking means clicking with the right-hand mouse button). You'll see a pop-up menu much like the one below:



As in this screenshot, choose New, then Folder.

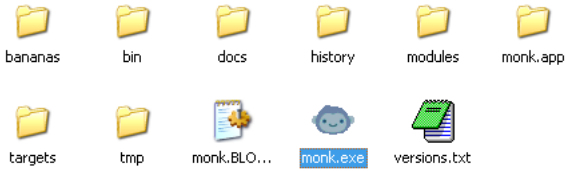
Step 2

This will create a new folder on the desktop, with the name New Folder (or similar) highlighted in blue. Just start typing to give it a new name; this will be the name of your project, and it's up to you what to call it. If you make a mistake, right-click the folder icon and choose Rename to try again.



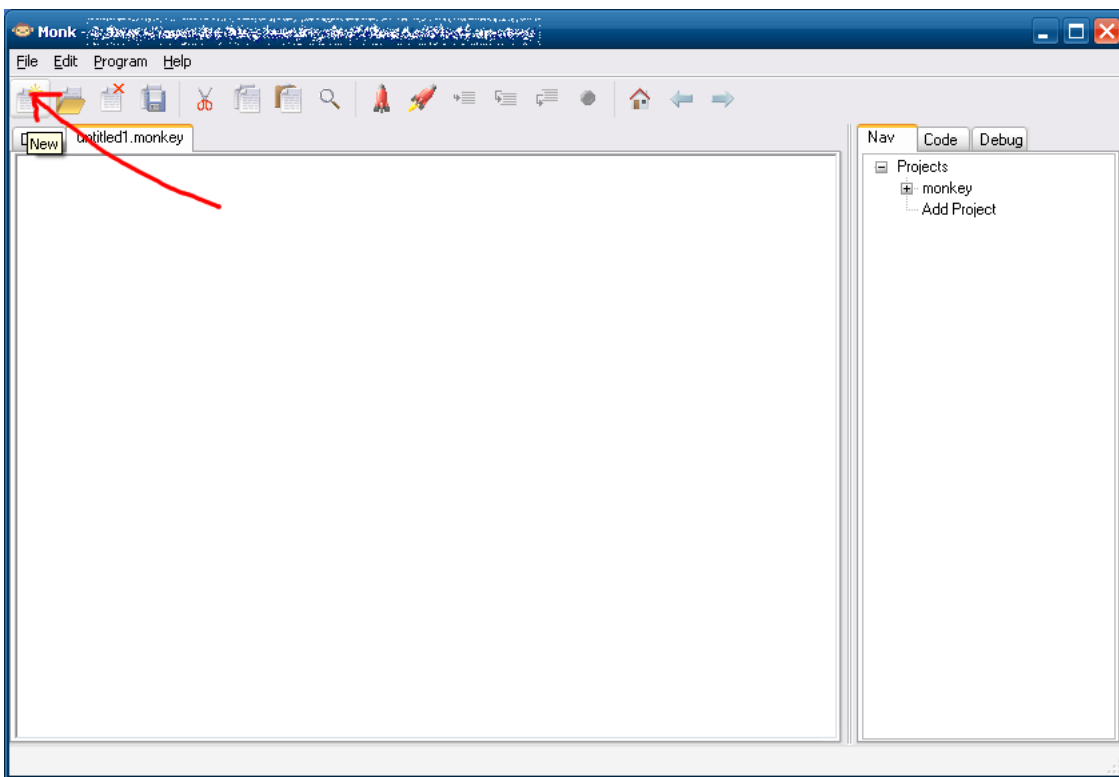
Step 3

Start the Monk IDE by clicking on the monk.exe icon within the Monkey program folder. (Depending on your settings, it may not show the .exe file extension.) The icon should look much like the one below. (Note: IDE stands for Integrated Development Environment, which just means it's a text editor that can also run your programs.)



Step 4

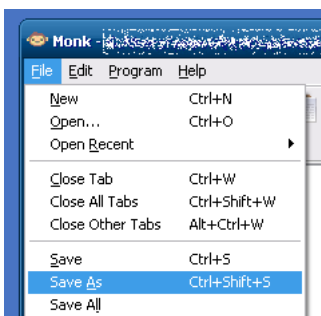
Once Monk has loaded, click on the New icon – the first icon in the toolbar at the top. (You can alternatively use the File menu and choose the New command to do the same thing.)



This creates a blank area in which you can type your program. However, it's best to save this as a Monkey source code file right away.

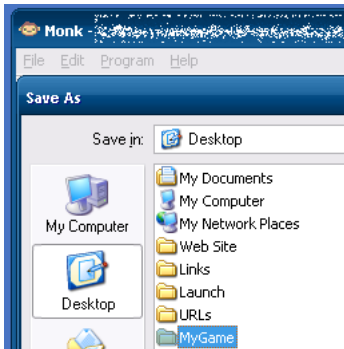
Step 5

Click on the File menu and choose the "Save as..." option.



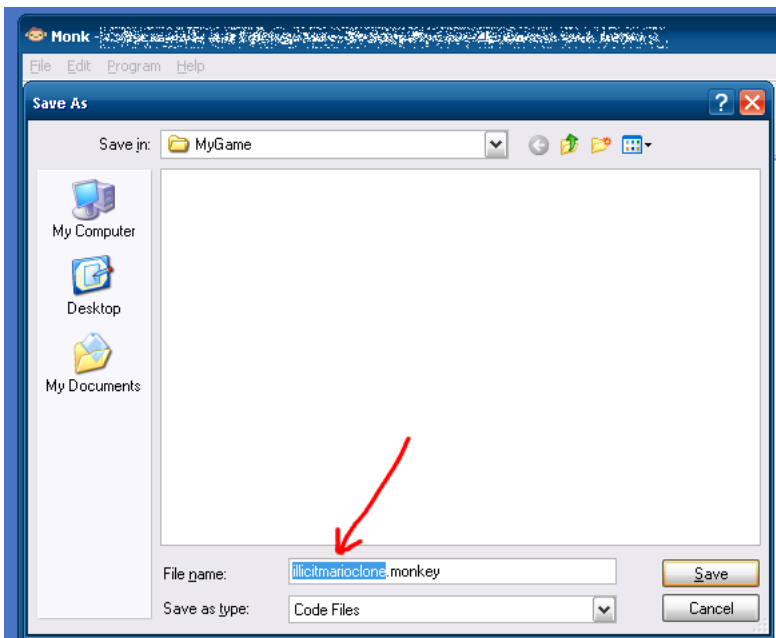
Step 6

Using the file selection dialog box, locate the folder you created earlier. In most cases, you can click on the Desktop icon in the left-hand pane (you may need to scroll to the top in some versions of Windows), then double-click (two fast clicks) on the name of your project folder in the right-hand pane.



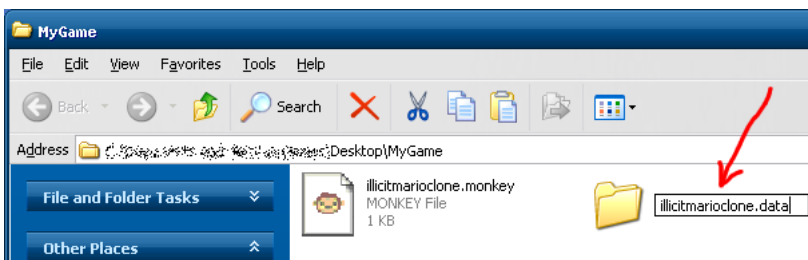
Step 7

After double-clicking on your project folder, you can type the name of your program into the box at the bottom. The name should end with `.monkey`, as below. (That's "dot-monkey".)



Step 8

Lastly, all of your game's graphics, sound or other data files need to be placed into a special folder. The name of this folder should match the name of the file you just saved, but end with `.data` ("dot-data") rather than `.monkey`, like so:



Any file you place in here can be referred to directly, using only the filename. For example, a file named `rocket.png`, stored in your data folder, would look like this when using the `LoadImage` command: `LoadImage ("rocket.png")`.

You can now switch back to the Monk IDE and start coding!

Example project

Here's an example project you can copy and paste into your Monk project to get started. Just fill in the blanks!

```
Import mojo

Function Main ()
    New Game
End

Class Game Extends App

    Method OnCreate ()
        ' Setup code goes here...
        SetUpdateRate 60
    End

    Method OnUpdate ()
        ' Game update code goes here...
    End

    Method OnRender ()
        ' Game rendering code goes here...
    End

End
```

See the [Tutorials and Articles](#) section of the [Monkey docs](#) for more information on getting started with Monkey, but if you want something to play with right away, copy and paste the code below, run it, then modify as you see fit:

```
Import mojo

Function Main ()
    New Game
End

Class Game Extends App

    Field mx:Int
    Field my:Int

    Method OnCreate ()
        SetUpdateRate 60
    End

    Method OnUpdate ()
        mx = MouseX ()
        my = MouseY ()
    End

    Method OnRender ()
        Cls 64, 128, 256
        SetColor 256, 256, 0
        DrawRect mx - 32, my - 32, 64, 64
    End

End
```

Have fun!