Minecraft Pi

Minecraft on the Raspberry Pi

Minecraft: Pi Edition is a version of Minecraft that's designed to work on the Raspberry Pi.

Based on the "Pocket Edition", Minecraft Pi is smaller and faster as well as containing support for multiple programming languages. Minecraft Pi runs on the Raspbian Operating System for the Pi. This is the most common operating system used on the Pi.

Minecraft Pi is now loaded directly with the LX desktop, no messing around with downloads or installation files. Click on the "Menu" icon and then head down to "Games" and over to "Minecraft Pi". At this point you can play Minecraft as a single user or over a local network.

For those of you who are unfamiliar with Minecraft, a simple way to get started is to:
Click on "Start Game", then "Create New" (World). After your world is created you will be placed in and you can begin moving around. Basic controls are as follows- Mouse Left / Right / Up / Down will change your point of view. Keyboard Keys A and D will move you Left / Right and W / S will move you forward and back. A click of the left mouse button will cause your sword to strike the ground and dig a hole; if you hold the left button you will dig a tunnel. That's the basics; get familiar with it because what comes next is better; programming Minecraft to perform certain functions for you.

■ Minecraft Pi.

Programming Minecraft Pi

Once you have launched the game then you can select Start Game followed by Create New. This should take you into a randomly generated world that you can start to modify with code.

You should still have your LX Terminal windows open. If not then click on the LX Terminal window again to open a command line prompt. To do this you will need to press escape to pause the game and allow you to move the mouse out of the game window. You can then launch another terminal from the desktop. One way is to have the programming terminal window and the Minecraft Window side by side

Once you have your terminal launched you need to go to the Python API directory (programming tools) by typing the following at the terminal window:

```
cd mcpi/api/python/mcpi
```

and then:

```
python
```

and pressing return.

Your terminal prompt should change to **>>>**.

Now we are ready to start programming. The first thing we need to do is to import the Minecraft library.

At the command prompt in the terminal windows type:

```
import minecraft
```

Now we need to initialize it which will create a connection to your game and allow you to start modifying the game world. So now type:

```
mc = minecraft.Minecraft.create()
```

You can now use the mc object you have created to control the world. Let's say hello to your Minecraft world:

So type at the terminal window:

```
mc.postToChat("Hello Minecraft World!")
```

You should then see your message come up on the chat window in your game. Feel free to change this message to what ever you want. You can use the postToChat method to post all sorts of information you want the player to see.

Congratulations, you have learned to simply program Minecraft Pi.

Now on to a more advanced project, that is Placing Block within your game. To place a block let's type the following at the command prompt:

```
import block
mc.setBlock(0, 10, 0, block.STONE)
```

The first line which imports block simply allows us to use the block names when we position them. The name "block. STONE" can be changed to block. LAVA, block.MELON or block.WATER. The Minecraft Wikipage gives you many other types of blocks to use. minecraft.wikia.com/wiki/Category:Blocks

The second line calls setBlock and passes in 3 numbers and the block type. These 3 numbers are the X, Y and Z coordinates in the game world. Just like you would find in maths, this coordinate system, called Cartesian Coordinates, allows you to map your blocks in a 3D world. From your player start position X is left and right, with numbers going down as you head left and up if you head right. The Y coordinates are up and down, with up increasing the number and down decreasing the number. This leaves the Z axis being forwards and backwards, with forward increasing the number and backwards decreasing. It's worth noting that decreasing numbers go to a negative value.

Have a go at setting different coordinates for blocks and see how they come out. If you don't see a block then it means you either put it very far away or it's underground. To avoid blocks going too far away try and keep the X, Y and Z values within 20 and -20.

You can view the source code for this tutorial here: github.com/piprogramming/ Minecraft/blob/master/tutorial1.py

If you have the git-core package installed then you can download it using: git clone git://github.com/ piprogramming/Minecraft.git