



“Why build when you can code”

Run the game

If your Raspberry Pi is at the command line and not the desktop, then login (user pi , password raspberry), then type `startx` and then press Enter.

From the desktop Menu, select Games - Minecraft: Pi Edition

Have a wander about!

Click Start Game, then click Create New (or choose an existing one) to enter a world.

The mouse changes where you look.

Holding left button destroys blocks.

Right button ‘places’ blocks

W, S, A, D move you forward, backward, left and right.

E opens the inventory

ESC takes you back and to the Menu

TAB releases the mouse so you can select other programs or the desktop

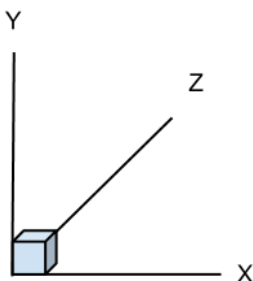
The API

The API works by changing the ‘server’, which runs underneath the game, allowing you to change the game as it is being played:

- Post messages to the player
- Get and Change (or set) the player’s position
- Get the type of a block
- Change a block to a different one

A world of blocks

Minecraft is a world of cubes or blocks, all with a relative size of 1m x 1m x 1m, and every block has a position in the world of x, y, z; x and z being the horizontal positions and y being the vertical.



Hello Minecraft World

The first program a programmer creates is called “Hello World” and the task is to get “Hello World” printed on the screen. Your first task is to get “Hello Minecraft World” displayed in the Minecraft chat window.

Leave Minecraft running (Steve must be in a world to program Minecraft) and use the TAB key to release the mouse pointer. From the desktop menu, select Programming - Python 2 (not Python 3). Then use File, New Window to create a new program and save it as myprogram.py.

If the Minecraft window gets in the way, you can move windows around by clicking and holding their title bar (at the top of each window), and dragging the window to a new location. Alternatively you can use the minimise button [_] on the title bar to shrink the window - to restore the window back again, click on Minecraft Pi Edition on the taskbar.

You are going to need 3 modules, the minecraft.py and block.py modules from the api library and the standard time module so you can introduce delays into your program.

```
import mcpi.minecraft as minecraft
import mcpi.block as block
import time
```

Use the Minecraft class in the python library to create a connection to the game’s server, this object will also be how we interact with the game and will provide access to all the functions.

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

Using the minecraft object, mc, you can interact with the game and send the player a message. Then using time put in a delay to stop the program running too fast.

```
mc.postToChat("Hello Minecraft World")
time.sleep(5)
```

Save your program, then run it by picking 'Run, Run Module' in Idle.

Switch back to Minecraft to see the results of your hard work - be quick though as its only on the screen for a few seconds, if you miss it, re-run the program and try again.

Falling down

Move Steve (the player) up in the air and let him fall to the ground, by adding the following to your program.

Find out where Steve is using `mc.player.getPos()`

```
playerPos = mc.player.getPos()
```

Change Steve's position 50 blocks up using `mc.player.setPos` and adding 50 his Y position.

```
mc.player.setPos(playerPos.x, playerPos.y + 50, playerPos.z)
```

Put a message on the screen and wait for Steve to fall to the ground.

```
mc.postToChat("Don't look down")
time.sleep(5)
```

Save and run your program.

Changing Blocks

Turn the block which Steve is standing on into Diamond. Find out which block Steve is standing on my using `mc.player.getTilePos()`

```
playerTilePos = mc.player.getTilePos()
```

Change the block below Steve using `mc.setBlock()`

```
mc.setBlock(playerTilePos.x, playerTilePos.y - 1, \
playerTilePos.z, block.DIAMOND_BLOCK.id)
```

Note that the `\` backslash just means "the next line is joined together with this one". You can omit it and continue typing all on one line if you like. Then save and run your program.

Walking on Air

Change our program so that it loops infinitely changing the block below Steve.

```
while(True):
    playerTilePos = mc.player.getTilePos()
    mc.setBlock(playerTilePos.x, playerTilePos.y - 1, \
    playerTilePos.z, block.DIAMOND_BLOCK.id)
```

Save and run, then move Steve around and watch as a path of diamonds is made under him. To stop the program, use TAB then click the mouse on the Python Shell window. Then

hold down the CTRL key and press C. This is called a "break" or a "keyboard interrupt".

Rainbow blocks

You can get coloured blocks by using WOOL, lets change our program to use random coloured WOOL by passing a number 0-15 (0="white", 1="orange", 2="magenta", etc) after the block id. To create random numbers, we will need to import python's random module at the top of the program..

```
import random
```

Change the last setBlock command to create WOOL instead of DIAMOND_BLOCK and pass a random number between 0 and 15 for the wool colour.

```
woolColour = random.randrange(0,15)
mc.setBlock(playerTilePos.x, playerTilePos.y - 1, \
playerTilePos.z, block.WOOL.id, woolColour)
```

Summary of some useful API functions

- postToChat(message) - communicate with the player(s) in the game
- getBlock(x, y, z) - get a block type for a specific position
- setBlock(x, y, z, blockType, blockData) - set (change) a block to a specific blockType
- setBlocks(x1, y1, z1, x2, y2, z2, blockType, blockData) - set lots of blocks all at the same time by providing 2 sets of co-ordinates (x, y, z) and fill the gap between with a blockType
- player.getPos() - get the precise position of a player
- player.setPos(x, y, z) - set (change) the players position
- player.getTilePos() - get the position of the block where the player current is

Have a go yourself

Use the functions above to have a go yourself!

Some Ideas

Build something of your own design (a house, a flower, a badger, anything).

Tip - you might find it easier using

```
setBlocks(x1, y1, z1, x2, y2, z2, block.STONE.id) which allows you to create lots of blocks all at the same time.
```

Create an animation - make blocks and shapes move.

Tip - to make a block disappear you need to set it to AIR, `setBlock(x, y, z, block.AIR.id)`

Martin O'Hanlon
@martinohanlon

www.stuffaboutcode.com

2015-01 new desktop update by Andrew Oakley www.cotswoldjam.org