



2. Flower Path

With Python you can achieve things that aren't normally possible in a Minecraft Pi game. In this exercise you'll create a trail of flowers that follows the player wherever they go.

You'll learn a number of new things in Python and

the Minecraft Pi API. This code can be easily modified to change the flowers to any other block type including gold and TNT.

Remember to save your program in the `minecraft` folder that you created in the last exercise.



Code

Import the API

This is the code that you always use to connect your code to a Minecraft Pi game.

Import time

This statement allows us to use `time` commands in our programs. For example we will use `time.sleep()` in our program which makes our program wait a number of seconds before continuing to the next line.

While loop and get player position

A `while` loop repeats a block of indented code. In this case it will repeat lines 5–11. The `True` part of the loop means that this loop will run forever until the user terminates the program with `ctrl+c`. Lines 5–8 find the player's position and then stores it in the `x`, `y` and `z` variables.

Set the block type

Every block type in the Minecraft world has an associated number. For example air is 0, lava is 12 and melon is 103. In order to place flowers we store the flower block's number, 38 in the `block` variable.

Set the block

The `setBlock()` function creates a block in the Minecraft world. It requires co-ordinates and a block type. Here the program is using the variables created on lines 6–9.

Slow the loop

The last line of the loop contains `sleep()`, which makes the program wait 0.2 seconds before looping again. We do this so that the loop doesn't work too quickly and use up too much processing power.

```
1 import mcpi.minecraft as minecraft
2 mc = minecraft.Minecraft.create()
```

```
3 import time
```

```
4 while True:
5     pos = mc.player.getPos()
6     x = pos.x
7     y = pos.y
8     z = pos.z
```

```
9     block = 38
```

```
10     mc.setBlock(x, y, z, block)
```

```
11     time.sleep(0.2)
```

What you've learned

while loop A **while** loop repeats a section of Python code. In this example the **while** loop repeats lines 5–11. The **True** part of the **while** loop means that it will repeat forever or until the user terminates the program.

time Using **time** in Python allows us to use functions that control time. For example the program that you've created uses the **time.sleep()** function to pause the Python program for a short type before continuing.

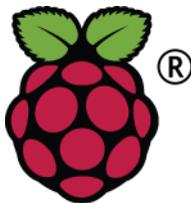
getPos() The **getPos()** function is part of the Minecraft Pi API and allows us to find the co-ordinates of the player in the game world.

setBlock() The **setBlock()** function in the Minecraft Pi API allows us to create blocks in the Minecraft Pi world. It takes four arguments, the first three of which are co-ordinates and the fourth argument is the type of block that we want to create.

Extensions

Here are some suggestions to extend your code and make it do different things. Even better if you come up with your own ideas.

- Change the block type that is placed. To do this change the value of the **block** variable on line 9. Some examples include melons (value 103), gold (value 41) and water (value 8).



For further exercises check out Python Programming with Minecraft Pi, the book available as a free pdf from [www. arg hbox. wor dpress. com](http://www.arghbox.wordpress.com)

Raspberry Pi and the Raspberry Pi logo are trademarks of the Raspberry Pi Foundation <http://www.raspberrypi.org>. Minecraft is a registered trademark of Mojang.

These resources are copyright Craig Richardson and licensed under a Creative Commons BY-NC-SA License.