

What you've learned

Variable A variable stores a value. For example in our program we created three variables, x, y and z, which stored the values 10, 11 and 12 respectively. You can change the values 10, 11 and 12 to any number that you want. Variables can be reused. When we used setPos() on line 6 we were reusing the variables x, y and z.

Integers Integers are whole numbers, for example 10, 347 and 59. We use integers to set the values of variables when we want to use whole numbers. In this program we use the integers 10, 11 and 12.

Function A function is a reusable piece of code that performs a specific purpose. For example we used the pre-written function setPos() on line 6 to change the position of the player in the game world.

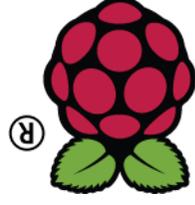
Function arguments Some functions need to be given data in order to work, this data is called an argument. For example on line 6 we gave the argument x, y and z to the setPos() in order to tell it where to teleport the player to.

Minecraft Pi API An API is a collection of pre-written functions that allow you to connect your Python code to another program. In this guide you have learned how to connect your Python programs to the Minecraft Pi API, which contains functions to control Minecraft Pi games.

Extensions

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

- Change the values of the x, y and z variables. For example you could change them to 34, 64, -12. You can also use negative numbers. If your numbers are too big you might teleport off the map.



For further exercises check out Python Programming with Minecraft Pi, the book available as a free pdf from 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.

1. Teleport

Programming with Python on the Raspberry Pi is a powerful and fun way to modify Minecraft games. Using a few lines of code you can take control of the player and the every block in the world. In this first worksheet you will use some basics of Python programming with Minecraft Pi. You'll learn how to control

teleport the player to a new position on the map using variables in Python. Before you begin you'll need to create a new folder for your programs. You'll also need to copy the set of Minecraft instructions that allow it to connect it to Python. The next page includes instructions that show you how to do this.



Setting Up

1. Switch on the Raspberry Pi:

Switch on your Raspberry Pi. Wait until the computer asks for your username.

Type in `pi` for the username and `raspberrypi` for the password.

Now start the desktop environment by typing `startx`.

2. Create the Minecraft folder:

Open the LXTerminal program, which can be found on your desktop or under the start menu.

Type in the following commands precisely:

```
mkdir -p Documents/minecraft
cd /Documents/minecraft
cp -r ~/mcp/api/python/* .
```

3. Open Minecraft

To open Minecraft Pi type in the following code into LXTerminal:

```
cd ~/mcp
./minecraft-pi
```

Start a game and create a new world.

Alert: Press tab in Minecraft to release the mouse

4. Open IDLE:

IDLE is used to run Python programs. Open IDLE using the icon on the desktop or under the programming section of the start menu.

Click File > New Window, then in the new window File > Save As. Save the file as `teleport.py` in the `Documents/minecraft` directory.

Copy the Python code below into IDLE. Press F5 in IDLE to run it.

Code

Import the API

Every Minecraft Pi program that you write in Python requires these two lines of code. The first line imports the commands that allow you to interact with a Minecraft game using Python. The second line creates a connection to the game.

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

Set the Variables

The player's position in Minecraft is represented using co-ordinates. Here we have created three variables to represent the player's position. A variable stores a value, in this case the variables `x`, `y` and `z` store the values 10, 11 and 12 respectively.

```
3 x = 10
4 y = 11
5 z = 12
```

Teleport the player

The last line teleports the player to a new position in the game. The `setPos()` function uses three number values, known as arguments, to change the player's position. In this case we're using the values of the `x`, `y` and `z` variables that we set earlier.

```
6 mc.player.setPos(x, y, z)
```