MINECRAFT - FIND A BLOCK

This workshop will guide you through programming Minecraft using the Python programming language and wiring up a LED and buzzer to create a simple game of "Find a block". As you get further away from the block the buzzer will sound and as you get closer the LED will light.

RUN MINECRAFT

Click Menu > Games > Minecraft: Pi Edition to run the game.

HAVE A WANDER ABOUT!

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

1. The mouse changes where you look
2. Holding left button destroys blocks
3. Right button ‘places’ blocks
4. W, S, A, D move you forward, backward, left and right
5. 1, 2, 3, 4, 5, 6, 7, 8 change what you are holding.
6. E opens the inventory
7. ESC takes you back and to the Menu
8. Space is jump, double tapping Space makes you fly or stop flying

GET STARTED

The first task is to start your program and get a message to appear on the Minecraft screen:

1. Press ESC to go back to the Minecraft menu but leave the game playing.
2. Open Python IDLE by clicking Menu > Programming > Python 3.
3. Use File > New Window to create a new program and save it as ‘myprogram.py’.
4. At the top of your program type the following code to import the Minecraft modules you will need.

```
from mcpi.minecraft import Minecraft
from mcpi import block
```

5. Create a connection to Minecraft using the code.

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

6. Post a message to the chat window.

```
mc.postToChat("Go find the block")
```

7. Run your program by clicking Run > Run Module.

You should see your message appear in the Minecraft chat window.

TIPS

* Any errors will be displayed in the Python Shell in red, check your code carefully.
* Capital letters are important Minecraft is different to minecraft.

MINECRAFT - FIND A BLOCK

HIDE A BLOCK

Using the random module you can generate a position near the player and use the Minecraft API to create a gold block there which the player will have to find:

1. At the top of your program import the randint function from the random module

```
from random import randint
```

2. Add the following code to the bottom of your program to find out the players position.

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

3. Generate 3 random numbers for the x, y, z of the gold block.

```
x = p.x + randint(-20, 20)
y = p.y + randint(-5, 5)
z = p.z + randint(-20, 20)
```

4. Create the gold block at the x, y, z position.

```
mc.setBlock(x, y, z, block.GOLD_BLOCK.id)
```

5. Run your program by clicking Run > Run Module or by pressing F5.

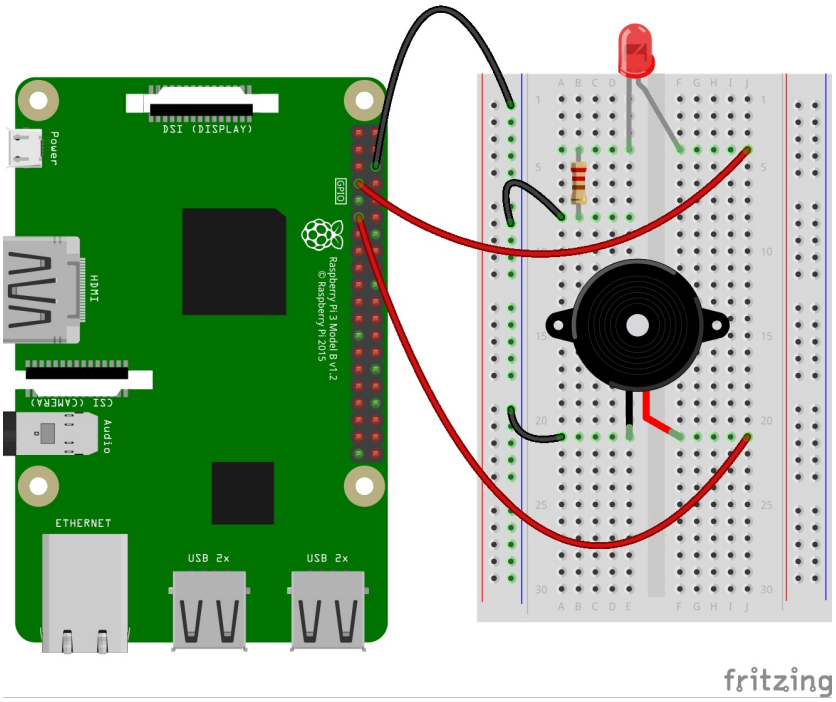
A gold block will be created within 20 blocks of the player - see if you can find it!

TIPS

* You might find it easier to find the gold block if you post the x, y & z values to the chat!
* Change the values passed to randint to adjust how far away the block might be created.

LEDS & BUZZERS

To help the player find the block you are going to use an LED which will indicate when the player is close by and a buzzer to let the player know they are walking towards the block.
You are going to need a breadboard, LED, resistor, buzzer and jumper cables, they should be connected as shown in the diagram:



MINECRAFT - FIND A BLOCK

When the circuit is complete, update your program to flash the LED and buzz the buzzer to let the player know the game has started:

1. Import LED, Buzzer and sleep from the gpiozero and time python modules.

```
from gpiozero import LED, Buzzer
from time import sleep
```

2. Create the LED, which is connected to GPIO 24.

```
led = LED(24)
```

3. Create the Buzzer, which is connected to GPIO 17.

```
buzz = Buzzer(17)
```

4. Turn the LED and buzzer on, sleep for 1 second and then turn them back off.

```
led.on()
buzz.on()
sleep(1)
led.off()
buzz.off()
```

5. Run your program, the LED and buzzer will turn on for 1 second.

TIPS

* If your LED doesn't light up, check it's the right way round, the long leg should go to positive, the short leg to ground.
* You can change how long the LED and buzzer are on for by changing the sleep time.

MAKING THE BUZZER 'BUZZZZZZ'

The buzzer should buzz when the player is getting further away from the gold block, so if the buzzer remains silent you are getting closer.

Update your program so that it works out the distance between the player and to the block and turns the buzz on or off:

1. To calculate the distance you will use the maths square root (sqrt) function, so import it now.

```
from math import sqrt
```

2. Create a variable called dist (for distance) and set it to 0.

```
dist = 0
```

3. Create a variable called gameover and set it to False - it will be set to True at the end of the game when the player has found the block.

```
gameover = False
```

4. Create a loop which will continue until the game is over.

```
while gameover == False:
```

5. Get the players position.

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

MINECRAFT - FIND A BLOCK

6. Work out the distance between the player and the gold block.

```
xd = p.x - x
yd = p.y - y
zd = p.z - z
dist_now = sqrt((xd*xd) + (yd*yd) + (zd*zd))
```

7. If the distance is going up, turn the buzzer on, else turn it off.

```
if dist_now > dist:
        buzz.on()
    else:
        buzz.off()
```

8. Set the variable dist to dist_now so it can compared next time around the loop.

```
dist = dist_now
```

9. Run the program, the buzzer should buzz when the player is getting further away from the gold.

NEARLY THERE LED

To give the player a chance to find the block when he is close by.

1. Turn the led on when the distance to the gold block is less than 5:

```
if dist_now < 5:
    led.on()
else:
    led.off()
```

2. Run your program and when you get close the LED should light up.

GAME OVER

Once the player find the block the game is over.

1. When the distance between the player is less than 1.5, set the gameover variable to True and post a message to let the player know.

```
if dist_now < 1.5:
    gameover = True
    mc.postToChat("You got GOLD")
```

2. Finally clean up by turning off your led and buzzer.

```
    led.off()
    buzz.off()
```

3. Run your program and find the gold block!

MINECRAFT - FIND A BLOCK

CHALLENGES

This game is just the start, can you finish it? Here are some challenges:

1. Create a timer so that you can have races trying to find the gold block.
2. Add more LED's (i.e. red, yellow and green) which show when you are getting nearer the gold.
3. Hide several blocks which the player has to find in order.

TIPS

* The time() function in the time module will give the time in seconds
* You can add as many LEDs as you want using led2 = LED(gpio pin number)

FIND OUT MORE

If you have enjoyed this check out the book "Adventures in Minecraft" and the Minecraft resources on the Raspberry Pi website www.raspberrypi.org/resources

CREATED BY

Martin O'Hanlon
@martinohanlon
www.stuffaboutcode.com