

ACTIVITY ONE – MOVING AROUND AND USING PYTHON TO PLACE BLOCKS!

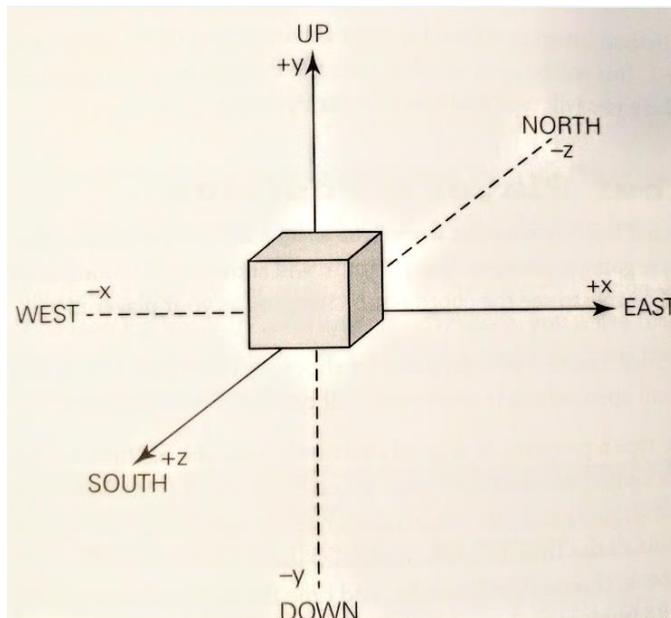
Find your location

To find your location, type:

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

pos now contains your location; access each part of the set of coordinates with `pos.x`, `pos.y` and `pos.z`.

Need a reminder of what coordinate are?



Alternatively, a nice way to get the coordinates into separate variables is to do this:

```
x, y, z = mc.player.getPos()
```

Now `x`, `y`, and `z` contain each part of your position coordinates. `x` and `z` are the walking directions (forward/back and left/right) and `y` is up/down.

Note that `getPos()` returns the location of the player at the time, and if you move position you have to call the function again or use the stored location.

Teleport!

As well as finding out your current location you can specify a particular location to teleport to.

```
x, y, z = mc.player.getPos()  
mc.player.setPos(x, y+100, z)
```

This will transport your player to 100 spaces in the air. This will mean you'll teleport to the middle of the sky and fall straight back down to where you started.

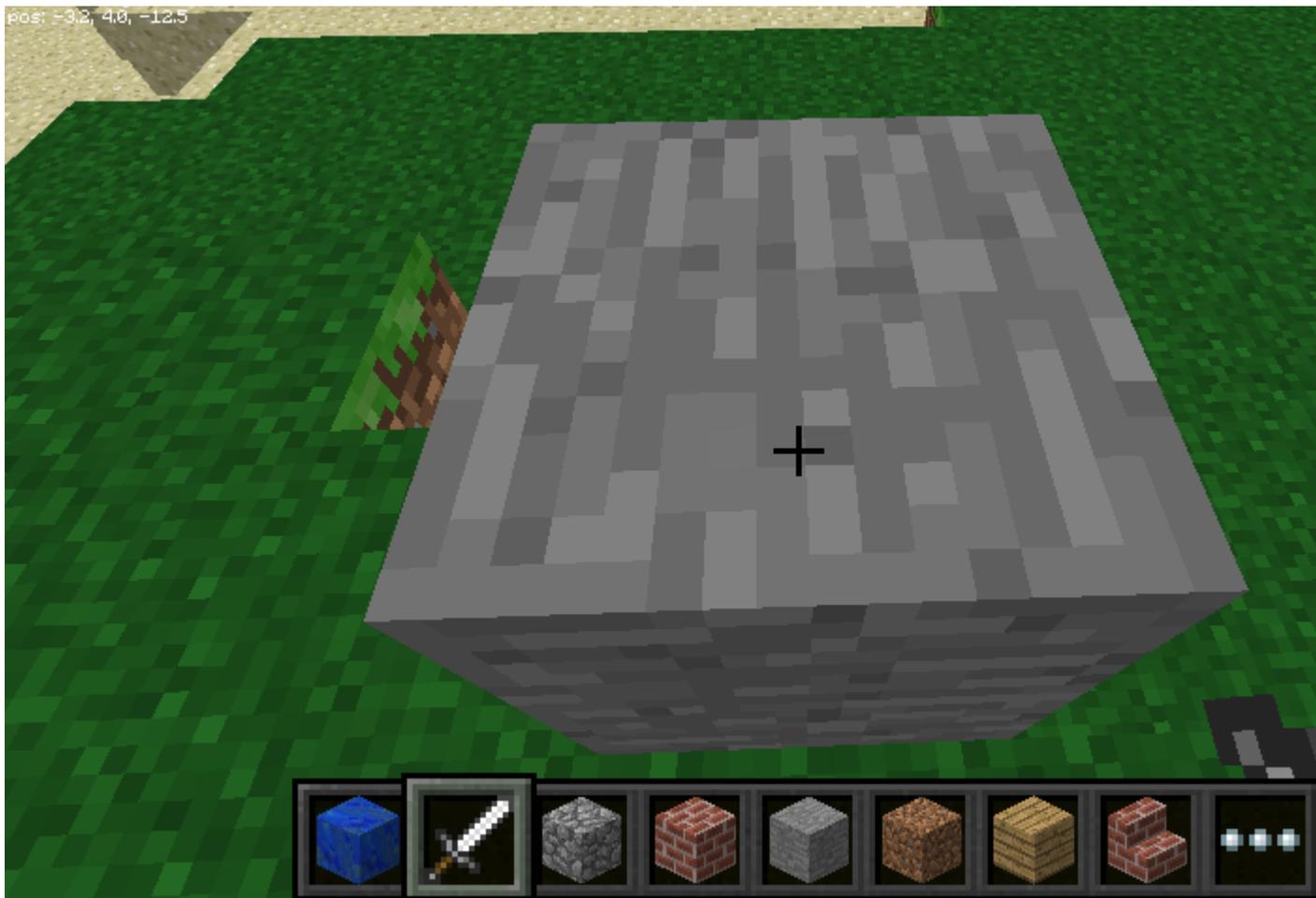
Try teleporting to somewhere else!

Set block

You can place a single block at a given set of coordinates with `mc.setBlock()`:

```
x, y, z = mc.player.getPos()  
mc.setBlock(x+1, y, z, 1)
```

Now a stone block should appear beside where you're standing. If it's not immediately in front of you it may be beside or behind you. Return to the Minecraft window and use the mouse to spin around on the spot until you see a grey block directly in front of you.



The arguments passed to `setBlock` are `x`, `y`, `z` and `id`. The `(x, y, z)` refers to the position in the world (we specified one block away from where the player is standing with `x + 1`) and the `id` refers to the type of block we'd like to place. 1 is stone.

Other blocks you can try:

Air: 0

Grass: 2

Dirt: 3

Now with the block in sight, try changing it to something else:

```
mc.setBlock(x+1, y, z, 2)
```

You should see the grey stone block change in front of your eyes!



Block constants

You can use a inbuilt block constants to set your blocks, if you know their names. You'll need another `import` line first though.

```
from mcpi import block
```

Now you can write the following to place a block:

```
mc.setBlock(x+3, y, z, block.STONE.id)
```

Block ids are pretty easy to guess, just use ALL CAPS, but here are a few examples to get you used to the way they are named.

WOOD_PLANKS

WATER_STATIONARY

GOLD_ORE

GOLD_BLOCK

DIAMOND_BLOCK

NETHER_REACTOR_CORE

PLACING MULTIPLE BLOCKS

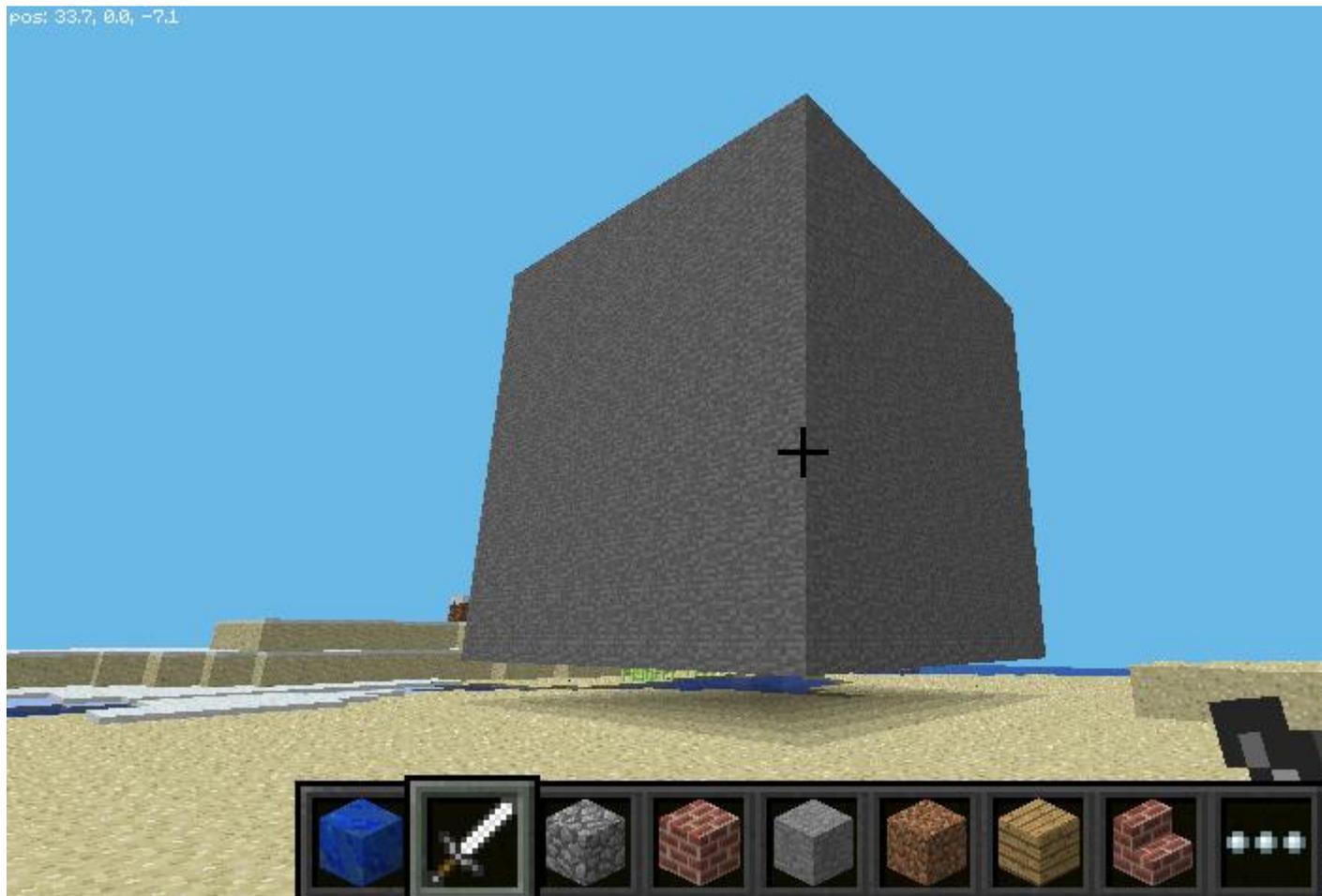
As well as setting a single block with `setBlock` you can fill in a volume of space in one go with `setBlocks`:

```
stone = 1
```

```
x, y, z = mc.player.getPos()
```

```
mc.setBlocks(x+1, y+1, z+1, x+11, y+11, z+11, stone)
```

This will fill in a 10 x 10 x 10 cube of solid stone.



You can create bigger volumes with the `setBlocks` function but it may take longer to generate!

Special blocks

There are some blocks which have extra properties, such as Wool which has an extra setting you can specify the colour. To set this use the optional fourth parameter in `setBlock`:

```
wool = 35  
mc.setBlock(x, y, z, wool, 1)
```

Here the fourth parameter `1` sets the wool colour to orange. Without the fourth parameter it is set to the default (`0`) which is white. Some more colours are:

- 2: Magenta
- 3: Light Blue
- 4: Yellow

Try some more numbers and watch the block change!

Other blocks which have extra properties are wood (`17`): oak, spruce, birch, etc; tall grass (`31`): shrub, grass, fern; torch (`50`): pointing east, west, north, south; and more.