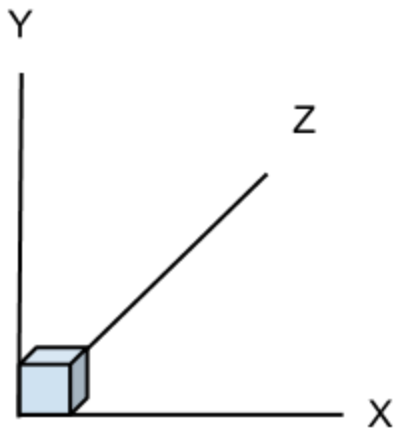
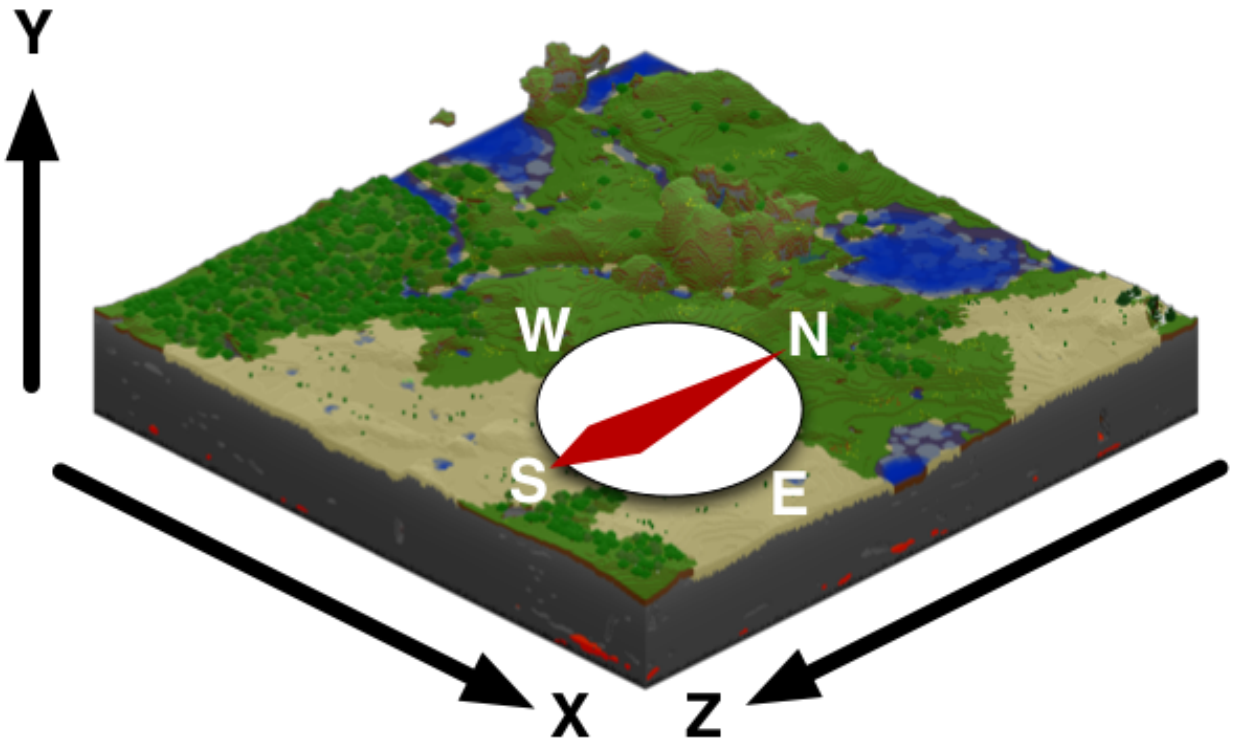
Learning Python with CoderDojo Twin Cities
Minecraft

Coordinate System

Most coordinates are in the form of a three integer vector (x,y,z) which address a specific tile in the game world. (0,0,0) is the spawn point sea level. (x,z) is the ground plane and Y is towards the sky.





Minecraft Programming Reference

World

world.getBlock(x, y, z)
Look up the type of block at the specified coordinates.

world.setBlock(x, y, z, block_type)
Set the block at the specified coordinates to the type block_type.

world.setBlocks(x1, y1, z1, x2, y2, z2, block_type)
Create a set of blocks starting at one coordinate point extending to another point with blocks of the type block_type. This can be used to make cubes or rectangles.

world.getHeight(x, z)
Look up the height (y coordinate) of the tallest brick at the specified x and y coordinates.

world.postToChat("Message")
Send a message over chat.

Player

player.getPos( )
Look up the coordinates that the player is currently positioned at.

player.setPos(xf,yf,zf)
Set the player's position to the specified coordinates.

Blocks

AIR	COAL_ORE	BRICK_BLOCK	REDSTONE_ORE
STONE	WOOD	TNT	SNOW
GRASS	LEAVES	BOOKSHELF	ICE
DIRT	GLASS	MOSS_STONE	SNOW_BLOCK
COBBLESTONE	LAPIS_LAZULI_ORE	OBSIDIAN	CACTUS
WOOD_PLANKS	LAPIS_LAZULI_BLOCK	TORCH	CLAY
SAPLING	SANDSTONE	FIRE	SUGAR_CANE
BEDROCK	COBWEB	STAIRS_WOOD	FENCE
WATER_FLOWING	GRASS_TALL	CHEST	PUMPKIN
WATER	WOOL	DIAMOND_ORE	NETHERRACK
WATER_STATIONARY	FLOWER_YELLOW	DIAMOND_BLOCK	GLOWSTONE_BLOCK
LAVA_FLOWING	FLOWER_CYAN	CRAFTING_TABLE	BEDROCK_INVISIBLE
LAVA	MUSHROOM_BROWN	FARMLAND	STONE_BRICK
LAVA_STATIONARY	MUSHROOM_RED	FURNACE_INACTIVE	GLASS_PANE
SAND	GOLD_BLOCK	FURNACE_ACTIVE	MELON
GRAVEL	IRON_BLOCK	DOOR_WOOD	GLOWING_OBSIDIAN
GOLD_ORE	STONE_SLAB_DOUBLE	LADDER	NETHER_REACTOR_C
IRON_ORE	STONE_SLAB	DOOR_IRON	ORE

Minecraft Controls

Keyboard

W,A,S,D - Move (navigate inventory)
SPACE - Jump, double tap to start/stop flying, hold to fly higher
SHIFT - Sneak, hold to fly lower
E - Open inventory
1-8 - Select inventory slot item to use
ESC - Show/hide menu
TAB - Release mouse without showing menu
ENTER - Confirm menu selection

Mouse

Steer - Look/turn around
Left button - Remove block (hold)
Right button - Place block, hit block with sword
Mouse wheel - Select inventory slot item to use

Resources / Credits

Online Classes
https://www.udacity.com/
http://www.codecademy.com/
http://www.learnpython.org/

<Stuff about="code" />
http://www.stuffaboutcode.com/2013/02/raspberry-pi-minecraft-install.html
Lots of really cool examples!

Get Your Own Raspberry Pi

A Raspberry Pi is a little computer that you used while learning how to program today! To get your own, here's what you'll need:

1. Raspberry Pi model B - http://www.adafruit.com/products/998
2. Power supply - http://www.adafruit.com/products/501
3. SD card - Preinstalled with Raspbian - http://www.adafruit.com/products/1121
   Blank (Raspbian is free, and we can help you) - http://www.adafruit.com/products/102
4. HDMI cable to connect to your Raspberry Pi TV or monitor (or appropriate adapters)
5. USB keyboard and mouse

Optional Parts:

1. Case - http://www.adafruit.com/products/859

If Statements

```
#!/usr/bin/python
import mcpi.minecraft as minecraft
import mcpi.block as block

# Connect to the Minecraft server
world  =  minecraft.Minecraft.create()

# Get the player's current position and store the coordinates
[x,y,z] =  world.player.getPos()

# If the player's y coordinate position is greater than 0, then they must be
# flying or walking on stilts!
flying  =  y > 0

if flying:
      print "You are flying high!"
else:
      print "You are on the ground!"
```

While Loops

```
# Connect to the Minecraft server
world  =  minecraft.Minecraft.create()

# Get the player's current position and store the coordinates
[x,y,z] =  world.player.getPos()

# Set some variables to customize your tower
height    =  3
material  =  block.GLASS

level  =  0
keep_building  =  True

# Execute the loop, building from the bottom up
while keep_building:
      world.setBlock( x, level, z, material )
      level = level + 1

      if level > height:
            keep_building  =  False

# Put the player on top of the tower
world.player.setPos( x, height, z )
```

For Loops

```
# Connect to the Minecraft server
world  =  minecraft.Minecraft.create()

# Get the player's current position and store the coordinates
[x,y,z] =  world.player.getPos()

# Set some variables to customize your tower
height    =  3
material  =  block.GLASS

# Execute the loop, building from the bottom up
for level in range( 0, height ):
      world.setBlock( x, level, z, material )

# Put the player on top of the tower
world.player.setPos( x, height, z )
```

Building a Pyramid

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

# Connect to the Minecraft server
world  =  minecraft.Minecraft.create()

# Get the player's current position and store the coordinates
[x,y,z] =  world.player.getPos()

# Set some variables to customize your pyramid
height    =  10
material  =  block.GLASS

# This variable will track the current level being created inside the loop
level  =  1

# Execute the loop, building from the top down
while level <= height:
      print level
      world.setBlocks( x - level, height - level, z - level,
                       x + level, height - level, z + level, material )
      level  =  level + 1;

# Put the player on top of the pyramid!
world.player.setPos( x, height, z )
```