

BLOCK TYPES

	AIR		CRAFTING_TABLE
	BED		DIAMOND_BLOCK
	BEDROCK		DIAMOND_ORE
	BEDROCK_INVISIBLE		DIRT
	BOOKSHELF		DOOR_IRON
	BRICK_BLOCK		DOOR_WOOD
	CACTUS		FARMLAND
	CHEST		FENCE
	CLAY		FENCE_GATE
	COAL_ORE		FIRE
	COBBLESTONE		FLOWER_CYAN
	COBWEB		FLOWER_YELLOW

BLOCK TYPES

	FURNACE_ACTIVE		IRON_BLOCK
	FURNACE_INACTIVE		IRON_ORE
	GLASS		LADDER
	GLASS_PANE		LAPIS_LAZULI_BLOCK
	GLOWING_OBSIDIAN		LAPIS_LAZULI_ORE
	GLOWSTONE_BLOCK		LAVA
	GOLD_BLOCK		LAVA_FLOWING
	GOLD_ORE		LAVA_STATIONARY
	GRASS		LEAVES
	GRASS_TALL		MELON
	GRAVEL		MOSS_STONE
	ICE		MUSHROOM_BROWN

BLOCK TYPES

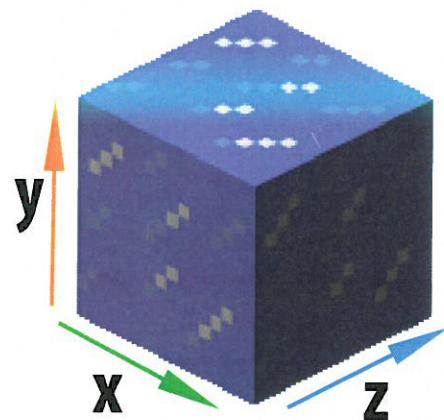
	MUSHROOM_RED		STONE_SLAB
	NETHER_REACTOR_CORE		STONE_SLAB_DOUBLE
	OBSIDIAN		SUGAR_CANE
	REDSTONE_ORE		TNT
	SAND		TORCH
	SANDSTONE		WATER
	SAPLING		WATER_FLOWING
	SNOW		WATER_STATIONARY
	SNOW_BLOCK		WOOD
	STAIRS_COBBLESTONE		WOOD_PLANKS
	STAIRS_WOOD		WOOL
	STONE		

COMMANDS

```
setblock( 20 , 20 , 20 , ICE )  
      x   y   z
```

```
teleport( 20 , 25 , 20 )  
      x   y   z
```

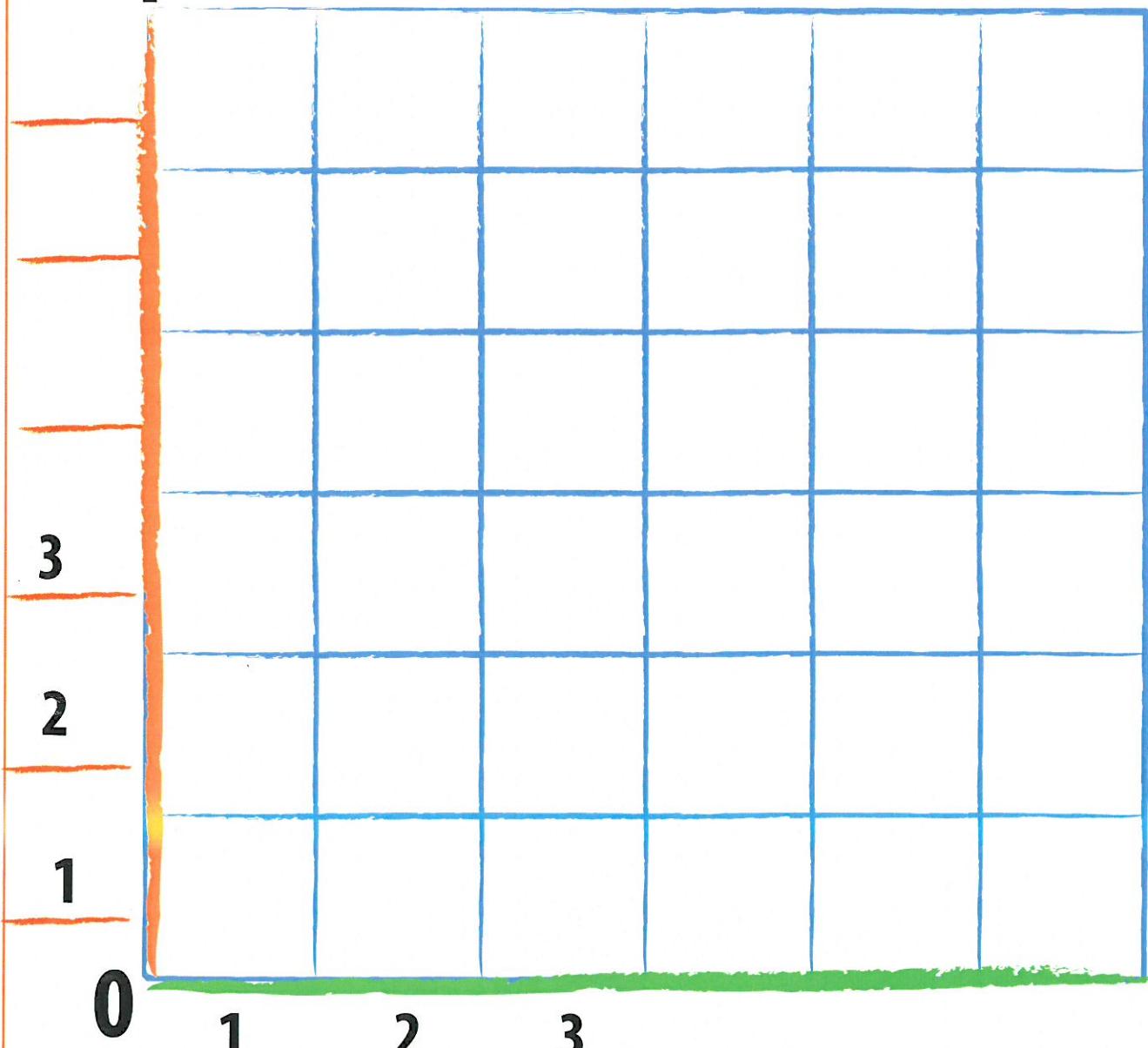
```
cube( 30 , 20 , 30 , ICE , 5 )  
      x   y   z
```



COMMAND # 1

setblock(_____, _____, 0, _____)

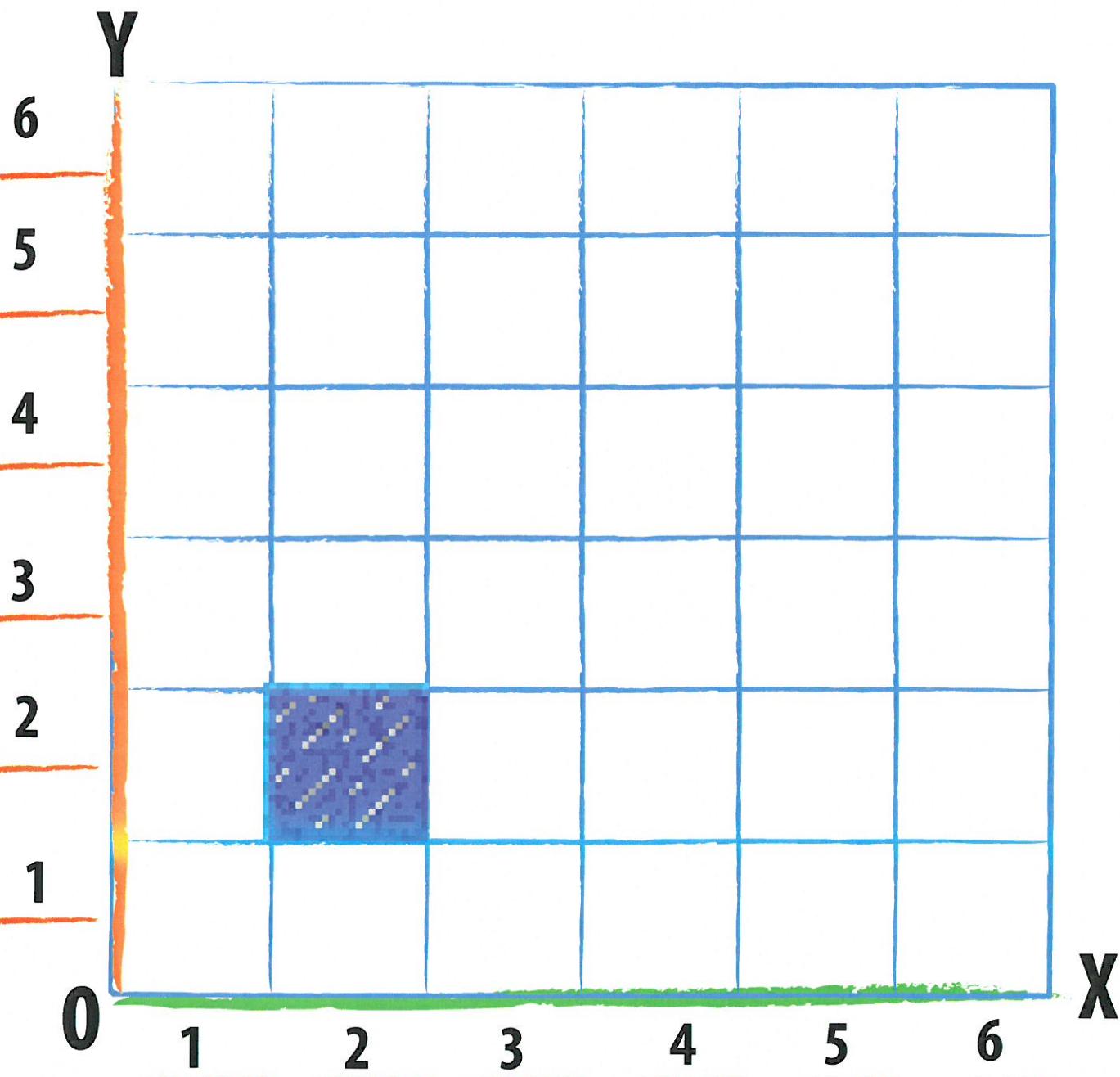
Y



X

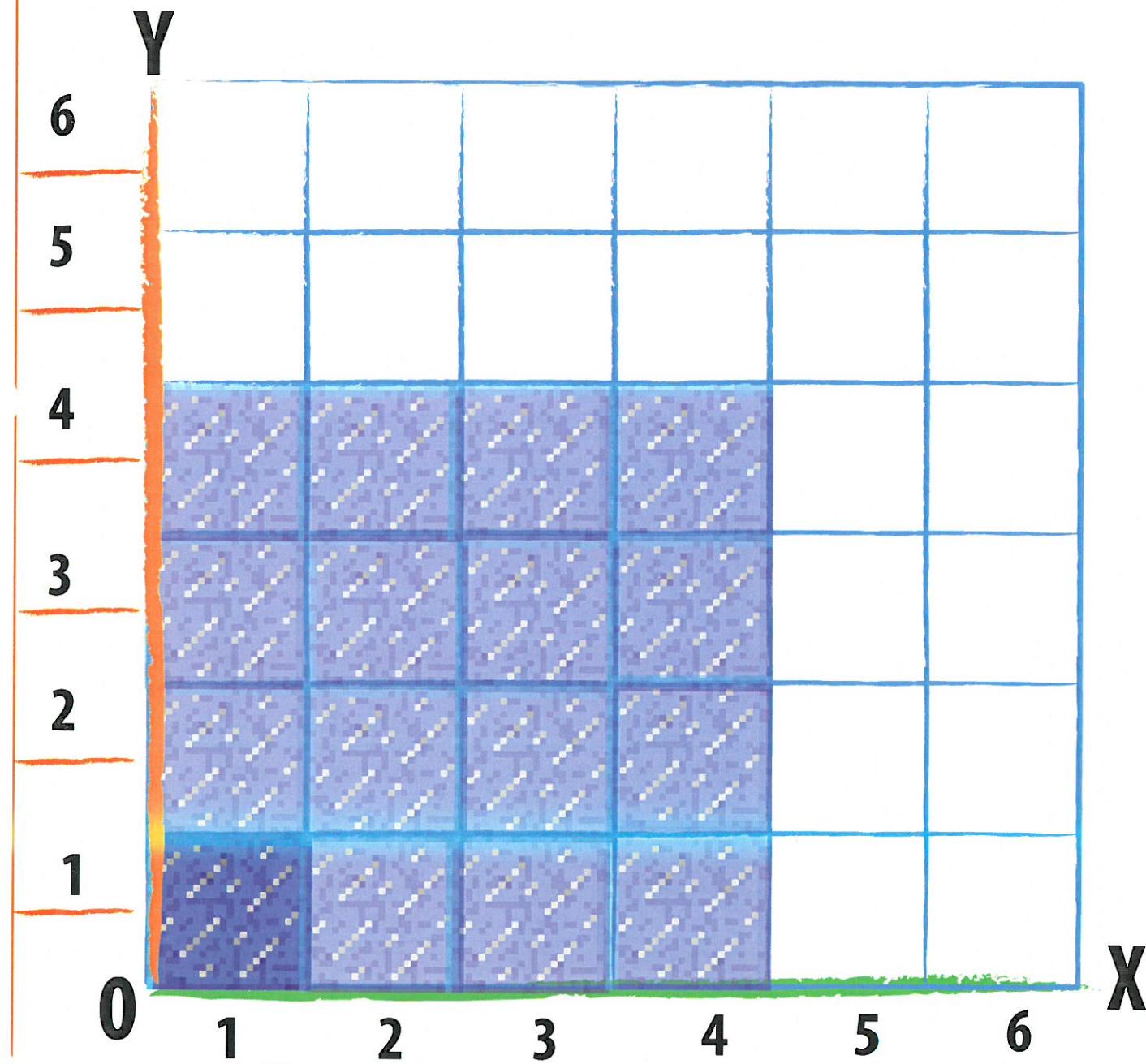
COMMAND # 2

teleport(, , 0)



COMMAND # 3

cube(, , 0, ICE,)



FOR LOOP

Programmers use a 'for loop' to repeat a block of work. It looks like this:

Reminder:

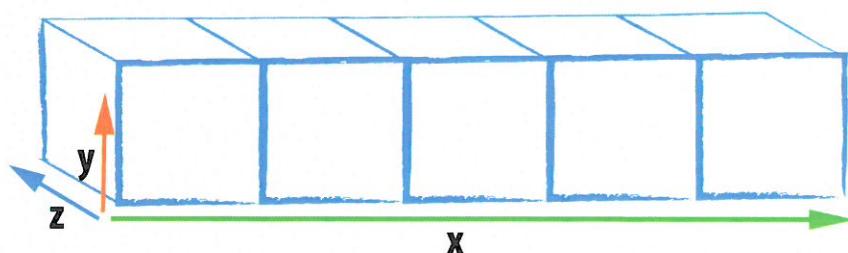
Only commands that are indented will be included in the for loop!

```
for n in range( 5 ):
```

```
    setblock( n , 0 , 0 )
```

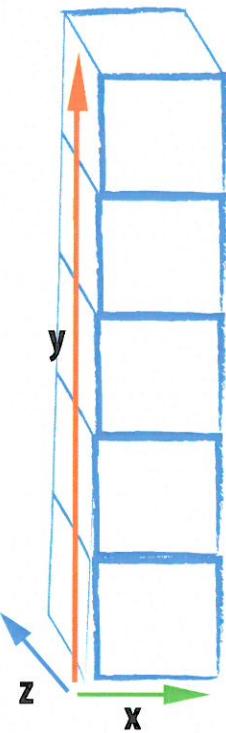
x y z

In this example, **n** represents **x**. Setblock is called 5 times. Each time, the **n** value increases by 1.



To get the same result without a loop, the commands would need to be written individually.

```
setblock(0,0,0)  
setblock(1,0,0)  
setblock(2,0,0)  
setblock(3,0,0)  
setblock(4,0,0)
```



for n in range(5):

setblock(0 , n , 0)
 x y z

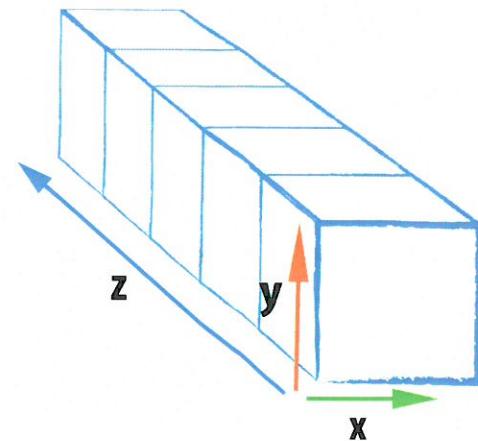
When n represents y, the blocks are placed higher and higher as n increases.

Reminder:
n can represent
ANY variable!

for n in range(5):

setblock(0 , 0 , n)
 x y z

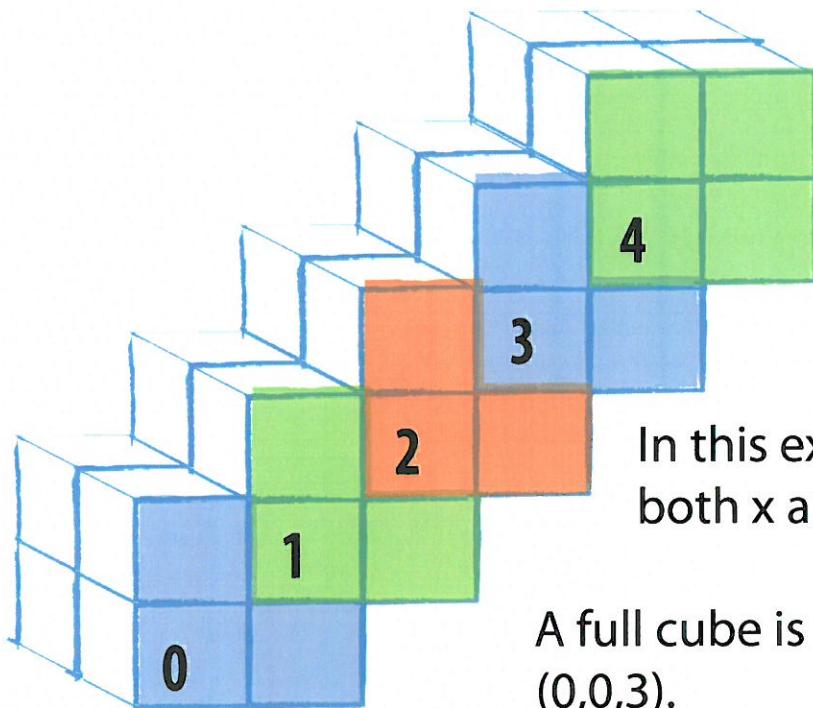
...or the z value can increase. In this example, n represents z.



Exciting news everyone! **n** can be used more than once :)

for **n** in range(5):

cube(**n**, **n**, 3, **GLASS**, 2)
 x y z



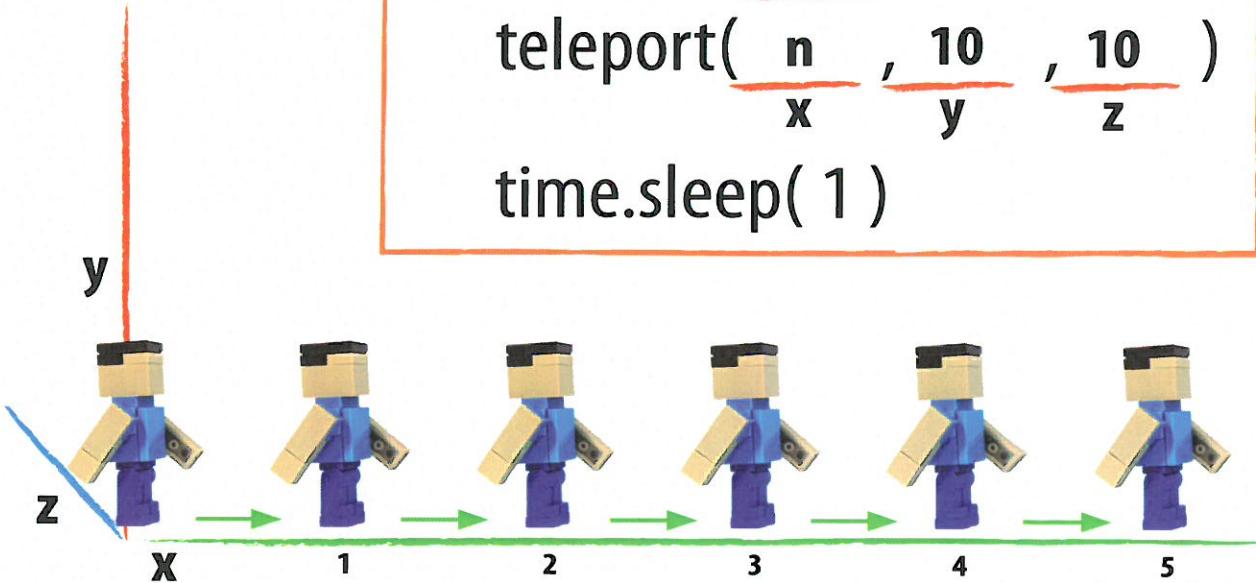
Reminder:
The value of **n** starts at 0

In this example, **n** represents both x and y.

A full cube is drawn at position (0,0,3).

n increases by 1, and the next cube is drawn at (1,1,3). This causes an overlap with the first cube. The pattern will repeat until the loop ends.

```
for n in range( 5 ):  
    teleport( n , 10 , 10 )  
    time.sleep( 1 )
```



→ This loop will teleport the player along the x axis.

After teleporting, the code will “sleep” for 1 second. In computer code, sleep means to wait.



So this code will cause the player to teleport, sleep, teleport, sleep, teleport sleep... 5 times.

Can you guess how to make the code sleep for 3 seconds?