

## e00\_hello.py

```
from mcpi.minecraft import *
mc = Minecraft.create()
mc.postToChat("Hello Minecraft World")
```

## e01\_coordinates.py

```
# Get the player's co-ordinates
# Print them to the chat

from mcpi.minecraft import *
import time

mc = Minecraft.create()

while True:    # loop will make sure your game runs forever
    time.sleep(1)
    pos = mc.player.getTilePos()
    print(pos)
    mc.postToChat(pos)
```

## e02\_teleport.py

```
# Change the player's position

from mcpi.minecraft import *

mc = Minecraft.create()

# change these to where you want to go
x = 10
y = 11
z = 12

mc.player.setTilePos(x, y, z)

# TODO
# Find the coordinates of a location in your world, either by pressing F3
# in the game, or running the e01_coordinates.py program
# enter them in this program and run it to teleport there
#
# Add
#     time.sleep(5)
# Then teleport the player somewhere else
```

## e03\_teleport2.py

```
# Before you run this, build two tiles in the game to be
# your teleporters and write down the co-ordinates of them

from mcpi.minecraft import *

mc = Minecraft.create()

# CHANGE THESE NUMBERS TO THE CO-ORDS OF YOUR TELEPORTERS

teleporter_x = 742
teleporter_z = 955

destination_x = 735
destination_z = 956

while True:
    # Get player position
    pos = mc.player.getTilePos()
    print(pos)

    # Check whether your player is standing on the teleport
    if pos.x == teleporter_x and pos.z == teleporter_z:
        mc.postToChat("teleport!")
        pos.x = destination_x
        pos.z = destination_z
        mc.player.setTilePos(pos)
```

## e04\_teleport3.py

```
# Teleport player into the air

from mcpi.minecraft import *

mc = Minecraft.create()

# CHANGE THESE NUMBERS TO THE CO-ORDS OF YOUR TELEPORTERS

teleporter_x = 9
teleporter_z = 12

height = 30

while True:
    pos = mc.player.getTilePos()

    # Check whether your player is standing on the teleport
    if pos.x == teleporter_x and pos.z == teleporter_z:
        mc.postToChat("teleport!")
        pos.y += height # up in the air!
        pos.x += 1 # move off the teleporter tile
        mc.player.setTilePos(pos)
```

## e05\_teleport4.py

```
# Do lots of teleports so it looks like a jump

from mcpi.minecraft import *
import time

mc = Minecraft.create()

# CHANGE THESE NUMBERS TO THE CO-ORDS OF YOUR TELEPORTERS

teleporter_x = 9
teleporter_z = 12

height = 20

while True:
    pos = mc.player.getTilePos()

    if pos.x == teleporter_x and pos.z == teleporter_z:
        mc.postToChat("teleport!")
        # move off the teleporter tile so we dont land on it again
        pos.x += 1
        for i in range(0, height):
            pos.y += 1 # move up a bit
            time.sleep(0.1) # short delay of 0.2 seconds
            mc.player.setTilePos(pos)

# TODO
# Change the height of the jump.
# Make the jump faster.
# Move the player in X and Z directions as well during the jump.
# Instead of one tile, use < and > to check if player is within a larger
# area.
```

## e06\_create\_block.py

```
from mcpi.minecraft import *
# If we import this we can use names instead of numbers for blocks
from mcpi.block import *

mc = Minecraft.create()
pos = mc.player.getTilePos()
x = pos.x
y = pos.y
z = pos.z
blocktype = 1
mc.setBlock(x, y, z, blocktype)

#TODO
# Make the block appear a short distance from the player
#
# Try these names instead of a number for blockType
#
# AIR, BED, BEDROCK, BEDROCK_INVISIBLE, BOOKSHELF, BRICK_BLOCK, CACTUS,
# CHEST, CLAY, COAL_ORE, COBBLESTONE, COBWEB, CRAFTING_TABLE, DIAMOND_BLOCK,
# DIAMOND_ORE, DIRT, DOOR_IRON, DOOR_WOOD, FARMLAND, FENCE, FENCE_GATE, FIRE,
# FLOWER_CYAN, FLOWER_YELLOW, FURNACE_ACTIVE, FURNACE_INACTIVE, GLASS,
# GLASS_PANE, GLOWSTONE_BLOCK, GOLD_BLOCK, GOLD_ORE, GRASS, GRASS_TALL,
# GRAVEL, ICE, IRON_BLOCK, IRON_ORE, LADDER, LAPIS_LAZULI_BLOCK,
# LAPIS_LAZULI_ORE, LAVA, LAVA_FLOWING, LAVA_STATIONARY, LEAVES, MELON,
# MOSS_STONE, MUSHROOM_BROWN, MUSHROOM_RED, OBSIDIAN,
# REDSTONE_ORE, SAND, SANDSTONE, SAPLING, SNOW, SNOW_BLOCK, STAIRS_COBBLESTONE,
# STAIRS_WOOD, STONE, STONE_BRICK, STONE_SLAB, STONE_SLAB_DOUBLE, SUGAR_CANE,
# TNT, TORCH, WATER, WATER_FLOWING, WATER_STATIONARY, WOOD, WOOD_PLANKS, WOOL
```

## e07\_create\_block\_loop.py

```
# Same as program 06 but in a loop
```

```
from mcpi.minecraft import *
from mcpi.block import *

mc = Minecraft.create()

while True:
    pos = mc.player.getTilePos()
    x = pos.x
    y = pos.y
    z = pos.z
    blocktype = WOOL
    mc.setBlock(x, y, z, blocktype)
```

```
# TODO
# Make the block appear one meter BELOW the player's position
# Change the block to something else. Ice?
```

## e08\_tower.py

```
from mcpi.minecraft import *

mc = Minecraft.create()
pos = mc.player.getTilePos()
x = pos.x + 3
y = pos.y
z = pos.z

for i in range(10):
    mc.setBlock(x, y + i, z, 1)

#TODO
# How high can you make the tower?
# Make the program create 3 towers next to each other
```

## e09\_clear\_space.py

```
# setBlocks lets us set many blocks at once

from mcpi.minecraft import *
from mcpi.block import *

# We are defining a function so we can re-use it in other programs
# without typing it in again

def clear_space(mc, size):
    pos = mc.player.getTilePos()
    mc.setBlocks(pos.x-size, pos.y, pos.z-size, pos.x+size, pos.y+size, pos.z+size,
                AIR)

# What does setting a block to AIR do?

mc = Minecraft.create()

clear_space(mc, 10)
```

## e10\_basic\_house.py

```
from mcpi.minecraft import *
from mcpi.block import *

# This MUST be the name you gave to your clear space program!
from e09_clear_space import *

def make_house(mc, x, y, z, width, height, length):
    mc.setBlocks(x, y, z, x + width, y + height, z + length, STONE)

    # What happens if we make AIR inside the cube?
    mc.setBlocks(x + 1, y + 1, z + 1, x + width - 2, y + height - 2, z + length - 2, AIR)

mc = Minecraft.create()
pos = mc.player.getPos()
x = pos.x
y = pos.y
z = pos.z

width = 10
height = 50
length = 60

# Use the function from the other program
clear_space(mc, 10)
make_house(mc, x, y, z, width, height, length)

# TODO
# Try bashing a hole in the wall to see what is inside.
# Change the program so it automaticlaly makes a hole for a door.
# Lower the floor in your house.
# Add some furniture, torches, windows.
# Make the windows get bigger if you increase the size of the house
# Try filling a house with LAVA, or WATER, or TNT
# (Becareful with TNT, too much will crash your computer!)
```

## e11\_street.py

```
from mcpi.minecraft import *
from mcpi.block import *

# This MUST be the name you gave to your clear space program!
from e09_clear_space import *
# This MUST be the name you gave to your house program!
from e10_basic_house import *

mc = Minecraft.create()
pos = mc.player.getTilePos()

x = pos.x
y = pos.y
z = pos.z

width = 10
height = 5
length = 6

clear_space(mc, 100)

for i in range(1, 100, 20):
    print(x+i, y, z)
    make_house(mc, x+i, y, z, width, height, length)

# TODO
# How many houses are there? Make the street longer with more houses.
# Make the houses get taller as the street goes on.
# Add some towers to the street.
# Put a loop inside the loop to create multiple streets.
# Copy the make_house function from a friend so you can add a different
# kind of house to your street.
# Make some roads or fences.
# Make your houses out of TNT. Use flint tool on them.
```

## e12\_chat.py

```
# Your program can read chat messages posted by players.
# This program builds a block next to the player who
# says "build". This is the first example that will
# work for more than one player.

from mcpi.minecraft import *

mc = Minecraft.create()

while True:
    events = mc.events.pollChatPosts()
    for event in events:
        print (event)
        if event.message == "build":
            id = event.entityId
            pos = mc.entity.getTilePos(id)
            x = pos.x
            y = pos.y
            z = pos.z
            mc.setBlock(x, y, z, 1)
            mc.postToChat("done building!")

# TODO
# build a house around the player if the player says "house"
# build a lava trap if the player says "trap"
# use mc.getPlayerEntityId("fred") to get the id of a certain
# player named Fred (or whatever your friend's player name is)
```

## e13\_turtle.py

```
# THIS REQUIRES THE MINCRAFTSTUFF PACKAGE TO WORK
# You can install it in Mu by clicking in the bottom right gadget
# and adding minecraftstuff to list of third party packages

# You may have used a turtle for drawing at school.
# This is the same but in Minecraft.

from mcpi.minecraft import *
from mcpi.block import *

from minecraftstuff import MinecraftTurtle

mc = Minecraft.create()
pos = mc.player.getTilePos()
pos.y += 1

turtle = MinecraftTurtle(mc, pos)

turtle.forward(5)
turtle.right(90)
turtle.forward(5)
turtle.right(90)
turtle.forward(5)
turtle.right(90)
turtle.forward(5)

# TODO
# Draw a triangle, hexagon, etc.
# What do turtle.up(90) and turtle.down(90) do?
```