

# ゲーム 工作 プログラミング

## ラズベリーパイ Raspberry Pi 大活用

### 第21回 パイソンからマイクラをコントロール

ラズパイには、ラズパイ用のマイクラ「マイクラフトパイ」(Minecraft Pi) が最初からインストールされている。2017年10月号から4回にわたって紹介したときは、スクラッチのプログラムでブロックを積んだり、ゲームをつくりたりしたよね。今回は同じように、パイソンからマイクラを制御してみよう。マイクラが初めての人は、バックナンバー\*で予習してね。

\*2017年10月号「マイクラフトパイで遊ぼう」  
http://prog.kodomonokagaku.com/jibun/images/1710raspi.pdf

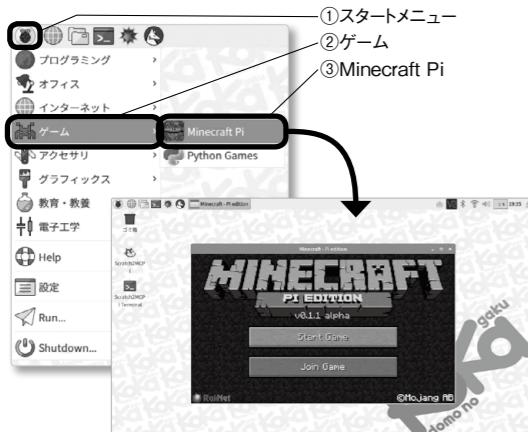


キットの情報は  
KoKa Shopへ  
定期購読者割引  
特典あり  
購入ページ

ジブン専用パソコン特設サイト  
prog.kodomonokagaku.com/jibun

### マイクラフトパイの起動

まずは、マイクラフトパイを起動しよう。



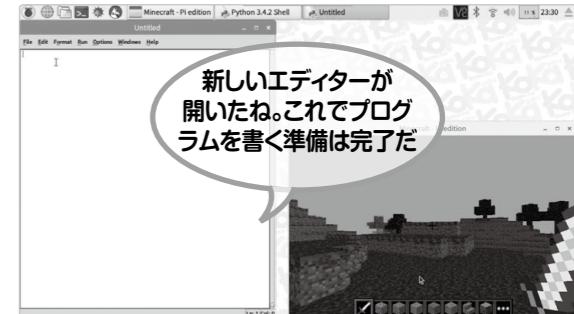
この後、他にもウィンドウを開くので、タイトルバーをドラッグして、マイクラフトパイのウィンドウを画面の右下に移動しよう。それから、ウィンドウの左上をドラッグして、少し小さくしておこう。



「Start Game」をクリックしたら、「Create new」で新しい地形をつくろう。なるべく平らな地面がある地形がいいかな。よさうだったら、Tabキーを押して、マウスカーソルがマイクラフトパイのウィンドウの外に出られるようにしておこう。



なるべく平らな地面のある地形を選ぼう。地形をつくり直したいときは、「Esc」キーを押して「Game Menu」を出し、「Quit to title」をクリックすれば、タイトル画面に戻るよ。



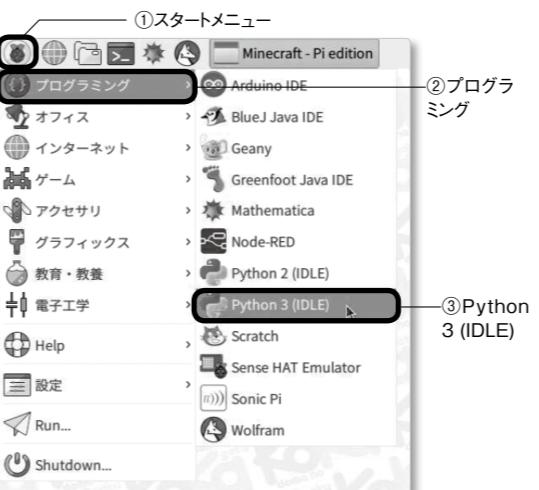
### マイクラフトパイにメッセージを表示する

ここからは、キーボードを使ってエディターにプログラムを入力していくよ。(1)などの丸数字は行番号なので入力しない。最初は、マイクラフトパイにメッセージを表示させてみよう。

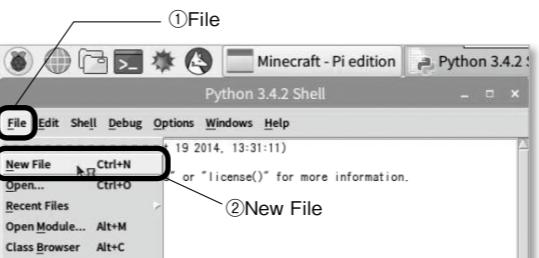
① `from mcpi import minecraft`  
② `mc = minecraft.Minecraft.`  
    `create()`  
③ `mc.postToChat("Hello")`

### IDLEの起動

次に、パイソンでプログラムを書くための準備をしよう。スタートメニューの「プログラミング」から「Python 3 (IDLE)」を選ぼう。



IDLEのシェルが開いたら、プログラムを入力するエディターを開こう。「File」メニューから、「New File」を選ぶ。

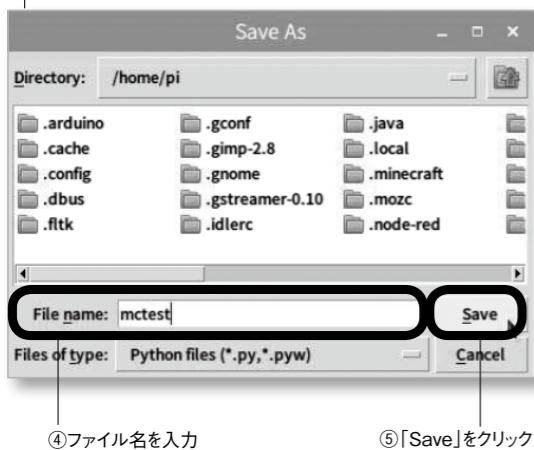
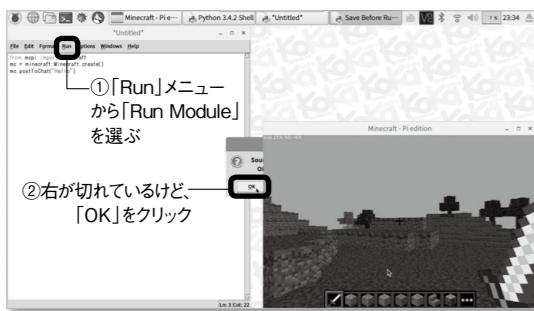


1行目は、マイクラフトパイを使うために必要な機能の読み込み(インポート)だ。スクラッチで「Scratch2MCPI」を使ったのと一緒にだよ。

2行目は、マイクラフトパイをパイソンで操作するためのオブジェクトをつくる、「mc」という名前の変数に入れている。スクラッチでクローンをつくるのにちょっと似ているね。

3行目は、「Hello」という文字をマイクラフトパイに表示(チャットにポスト)しているんだ。

ここまで入力したら、「Run」メニューから「Run Module」して、実際に動かしてみよう。このとき、ソースファイルをまだ保存していないので、保存してから実行することを確認するダイアログが表示される。ちょっと右が切れているけど、「OK」をクリックしよう。すると、ファイルを保存(セーブ)するためのダイアログが開くので、左にドラッグして全体が見えるようにする。ファイル名はなんでもいいけど、今回は「mctest」にしたよ。拡張子の「.py」は自動的にに入るから入力しなくても大丈夫。入れ終わったら、「save」ボタンを押そう。

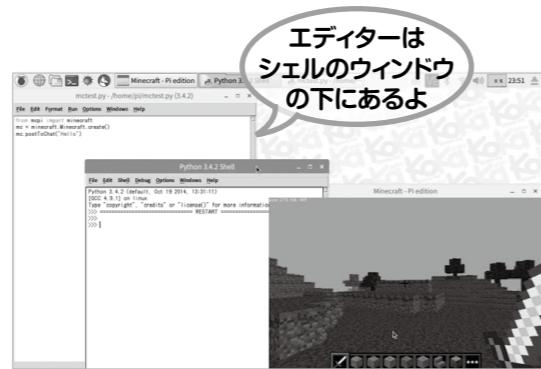


シェルが前面に出て、マインクラフトパイのウィンドウに「Hello」と表示されれば成功だ。もし、シェルに赤い字でエラーメッセージが表示されたときは、プログラムを間違えていないかよく確認してから、もう一度「Run Module」しよう。



## ブロックを積む

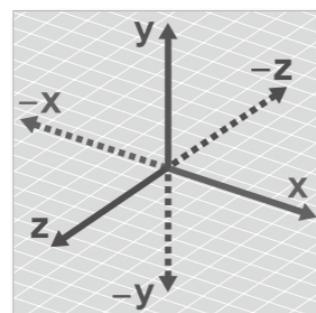
次はブロックを積むプログラムをつくろう。あれ、エディターがなくなっちゃった? シェルの下に隠れているだけだから、シェルのウィンドウを少しずらして出でてくるよ。



じゃあ、さっきのプログラムに続けて入力していくよ。

```
④ x,y,z = mc.player.getPos()
⑤ mc.setBlock(x,y,z,1)
```

4行目は、マインクラフトパイ(mc)のワールド上で自分(player)がいる現在の位置を取得して(getPos)、その座標をx、y、zのそれぞれの変数に入れている。3次元だから3つの座標があるんだったね。



マインクラフトパイの世界は3次元なので、左右(X)、上下(Y)、前後(Z)の3つの座標があるんだ。

そして、5行目で、その座標にブロックを置いている(setBlock)。最後の数字はブロックの種類(ID\*)だよ。「1」は石だったね。

\*[https://minecraft-ja.gamepedia.com/Java\\_Edition](https://minecraft-ja.gamepedia.com/Java_Edition) のデータ値/ブロックID

ここまでできたら「Run Module」しよう。保存の確認は「OK」で大丈夫だよ。

何も置かれてないように見えるときは、マインクラフトパイのウィンドウをクリックして、Sキーで少し後ろに下がってみよう。石のブロックが見つかったかな。足元で見えなかったんだね。



## 塔を建てよう

次は「繰り返し」を使って塔を建ててみよう。繰り返しには、2018年10月号で紹介した「for」文を使うよ。さっきのプログラムの「mc.setBlock(x,y,z,1)」の代わりに次の2行を入力しよう。

```
⑤ for i in range(10):
⑥     mc.setBlock(x,y+i,z,1)
```

5行目は、forの下にある字下げ(インデント)した部分を10回繰り返すという意味だ。このとき、変数の「i」には、繰り返すたびに、0、1、2、……9の数が入る。すると6行目の「y+i」は、「y+0」、「y+1」、……「y+9」と、1ずつ値が増えていく。Y座標は上下、つまり高さを表しているんだったよね。これによって、繰り返すたびに「1」高いところにブロックが積まれるので、塔が高くなるんだ。

入力できたら「Run Module」してみよう。後ろに下がって見てみると、ちゃんと塔ができるはずだ。



## TNTの立方体を爆発させよう

ここまでプログラムを応用して、1辺が10個のブロックでできた立方体をつくるプログラムを書いてみたよ。使ったブロックは全部で1000個! ブロックのIDは「46」、つまりTNTだ。その後ろの数字の「1」は、「叩くと爆発する」という意味だよ。

```
from mcpi import minecraft
mc = minecraft.Minecraft.create()
mc.postToChat("Hello")
x,y,z = mc.player.getPos()
for k in range(10):
    for j in range(10):
        for i in range(10):
            mc.setBlock(x+j, y+i, z+k, 46, 1)
```



どうなるかは、実行してのお楽しみ。2017年12月号\*のスクラッチで書いたプログラムと比較してみてもおもしろいよ。

\*<http://prog.kodomonokagaku.com/jibun/images/1712raspi.pdf>