

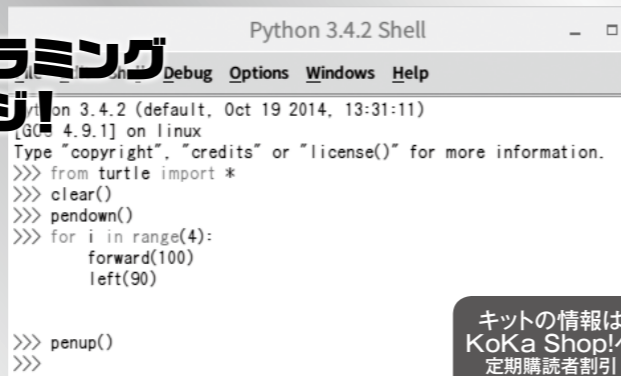
はじめよう/ ジブン専用パソコン

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19回 文字で書くプログラミング「Python」にチャレンジ!

今までこの連載では、スクラッチやスタディーノなど、ブロックを組み合わせてプログラミングをする「ビジュアルプログラミング」でプログラムを作成してきたね。今回は、キーボードから文字を入力してプログラムを書く「テキストプログラミング」に挑戦してみよう。使うのは「Python」というプログラミング言語だよ。



監修・原案 / 青山学院大学客員教授 阿部和広
構成・文 / 塩野祐樹

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

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



Pythonとは

Python (Python) は、オランダのガイド・ヴァンロッサムさんが1991年に開発したプログラミング言語だ。どんな人でも使えるように、簡単な英語を使うなどの工夫がされている。でも、他のプログラミング言語と同じくらい、いろいろな分野で使うことができるんだ。最近では、機械学習など、人工知能の分野でよく使われているよ。

さっそく、Pythonのプログラムを見てみよう。例えば、画面に「こんにちは!」と表示させる命令はこんな感じだ。

```
print("こんにちは!")
```

「print」は、「印刷」や「表示」という意味で、そのあとの「()」(カッコ)の中にあるものを画面に

表示させる命令だ。「こんにちは!」が「"」(ダブルクォーテーション)で囲まれているね。カッコの中の文字をダブルクォーテーションで囲むと、囲まれた文字がそのまま画面に表示されるよ。同じことをスクラッチでやろうとすると、こうなるね。

こんにちは! という

こんにちは!



ブロックの四角い枠に入力された文字が、ネコのセリフになって画面に表示されている。スクラッチの「~と言う」とPythonの「print」が、同じ意味で使われているんだね。

Pythonのプログラムを実行する

ジブン専用パソコンのOS、ラズビアンには、「Python 2」(2.7.9)と「Python 3」(3.4.2)という2種類のPythonが入っている。現在はPython 3の方が主流になっているので、今回はPython 3を使うことにしよう。

Pythonのプログラムを実行するには、IDLE (ア イドル)を使うのが簡単だ。画面左上のラズベリーのアイコンをクリックしてスタートメニューを開いたら、「プログラミング」から「Python 3 (IDLE)」をクリックしよう。「Python 3.4.2 Shell」というウィンドウが開いたかな? 全部英語なのでびっくりするかもしれないけど、難しいことはないので安心してね。

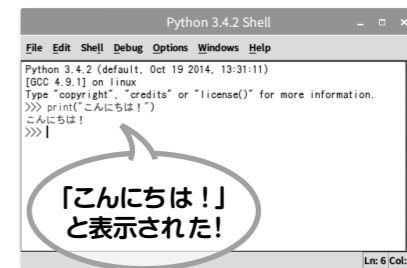


じゃあ、さっそくプログラムを入力しよう。「>>>」と表示されているところの右側に、さっきのプログラムをキーボードから打ち込もう。

```
print("こんにちは!")
```

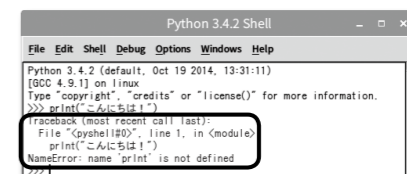
アルファベットと記号は半角モードで入力してね。「こんにちは!」は全角モードに切り替えて、ローマ字入力だ。モードの切り替えは、Ctrlキーを押しながらスペースキーを押すか、半角/全角キーを押す

んだったね。入力が終わったらEnterキーを押そう。プログラムが実行されるよ。



「こんにちは!」と表示された!

下の画面は、プログラムが間違っていたときに表示されるエラーメッセージだ。

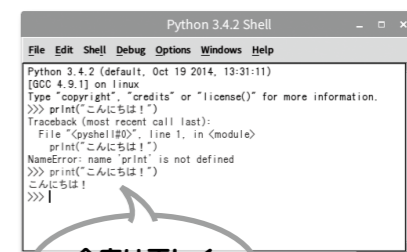


赤い文字で表示されているのがエラーメッセージだ(枠線内)。

Traceback (most recent call last):
File "<pyshell#0>", line 1, in <module>
print("こんにちは!")
NameError: name 'prlnt' is not defined

2行目にある「line 1」は、「1行目」という意味で、プログラムの1行目にエラーがあることがわかる。4行目の「NameError: name 'prlnt' is not defined」は、「prlnt」という名前が定義されていないという意味だ。

プログラムをよく確認すると「print」の「i」(アイ)が「l」(エル)になっているね。もう一度、正しく入力してからEnterキーを押すと……「こんにちは!」と表示されたぞ!



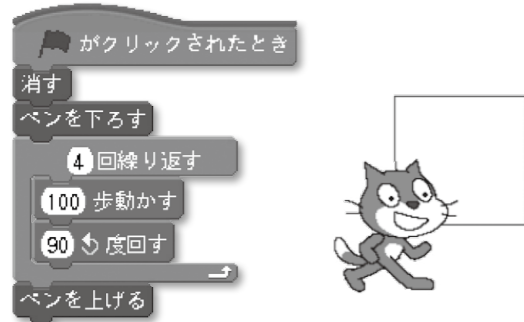
今度は正しく表示された!

テキストプログラミングでは、ほんの少しの間違いでエラーになってしまうから気をつけよう。エ

ラーが出たときは、自分では絶対に間違っていないと思っても、どこかに必ず間違いがあるよ。

タートルで絵を描こう

スクラッチに「ペン」という機能がある。ペンを使うと、スプライトが動いたあとに線を引くことができるんだ。例えば下のプログラムでは、1辺の長さが100の正方形を描いているよ。



このような機能は、一般には「タートルグラフィックス」という名前で見られている。タートル (turtle) は「カメ」という意味。昔、ロゴ (LOGO) というプログラミング言語で、スクラッチのネコのようにカメが使われていたことから、こう呼ばれているんだ。

Pythonにもタートルグラフィックスの機能があるので、それを使って絵を描いてみよう。上の例と同じように正方形を描いてみるよ。

まず、タートルを使うための準備をするよ。IDLEのウィンドウに次の命令を入力しよう。命令の最後に、Enterキーを押すのを忘れないように。

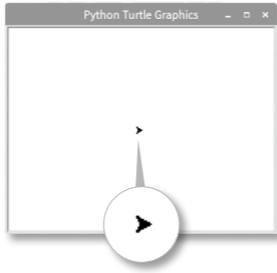
```
from turtle import *
```

「import」は読み込むという意味。画面に変化はないけど、これでタートルの機能が使えるようになったぞ。続けて、次の命令を入力しよう。

```
clear()
```

「clear」は「消す」という意味。これで、線が

消えた状態のタートルのウィンドウが表示されるよ。Pythonのタートルはカメじゃなくて、矢印の先っぽのような形で表されているよ。



次は、ペンを下ろす命令だ。

```
pendown()
```

これも画面に変化はないけど、ペンを下ろしたよ。さらに次の命令を入力しよう。

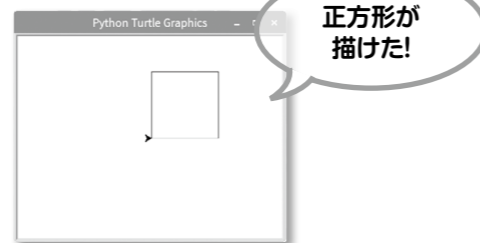
```
for i in range(4):
```

これは、「4回繰り返す」という命令だ。なお、この命令にある「i」は変数で、繰り返すたびに0、1、2、3と数字が入る。ここでは使っていないけど、今後使うこともあるので覚えておいてほしい。最後の記号はコロン(:)だ。

Enterキーを押すと、自動的にカーソルが右に移動する(字下げされる)から、繰り返しの途中で実行したい命令を入力しよう。

```
forward(100)
left(90)
```

これは、「100歩動かす」と「左に90度回す」という意味だ。入力できたら、Enterキーを2回押すと命令が実行されるよ。



最後にペンを上げる命令を入力しよう。

```
penup()
```

正方形が描けたかな? スクラッチとPythonを比べると、言語は違っても、命令の内容や順番はほとんど同じだってことがわかるよね。

プログラムをファイルから実行する

タートルで絵を描けるようにはなったけど、命令を1行ずつ入力するのは面倒だね。ちょっとでも打ち間違えたら入力し直しだし、実行するたびに入力しなければいけないのも大変だ。実は、プログラムをファイルに保存して、まとめて実行することができるんだ。

「File」メニューから「New File」を選んで、プログラムを入力するための新しいウィンドウを開こう。



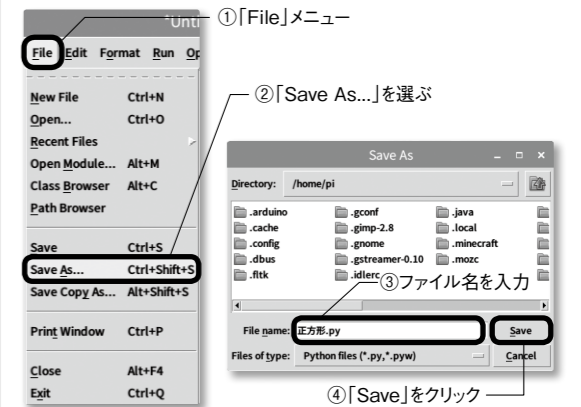
ウィンドウが開いたら、さっきのプログラムをまとめて入力しよう。

```
from turtle import *
clear()
pendown()
for i in range(4):
    forward(100)
    left(90)
penup()
```

このとき、「forward(100)」と「left(90)」の前に空白を入れる(字下げする)ことを忘れないようにしよう。これは、半角スペースでもいいし、Tabキーでタブを入れてもいいよ。

入力が終わったら、「File」メニューから「Save As...」を選んで保存しよう。

ファイル名を入力するダイアログが開いたら、わかりやすいファイル名をつけよう。このとき、ファイル名の最後に半角で「.py」という拡張子をつける必要がある。ここでは「正方形.py」とした。名前をつけたら「Save」ボタンをクリックしよう。



プログラムを実行したいときは、「Run」メニューから「Run Module」を選ぶ。「Run」は「実行」という意味だ。

さっきと同じように正方形が描けたかな? うまくいったら、「range」の繰り返し回数と、「forward」の辺の長さ、「left」の曲がる角度を変えて、正三角形や正五角形など、他の正多角形も描いてみよう。工夫すれば星型や円も描けるかも。

テキストプログラミングは、ビジュアルプログラミングに比べて、難しいと感じたかもしれないね。でも、複雑で大きなプログラムを作成するには、こちらの方が便利なおもいんだ。言語にはそれぞれ特徴があるので、目的に合わせて使い分けられるようになるといいね。