

プログラミング教室のテクノロ



# Pythonの道

体験用教材

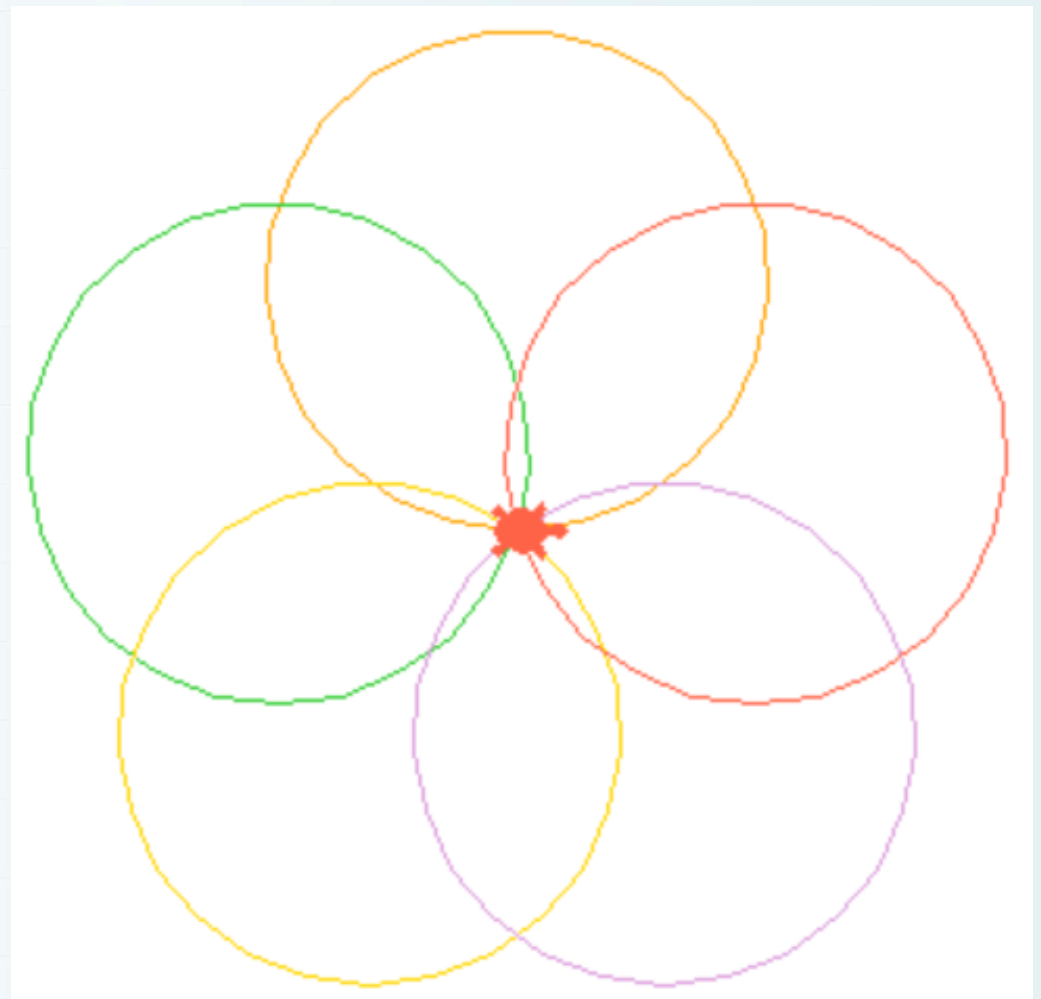
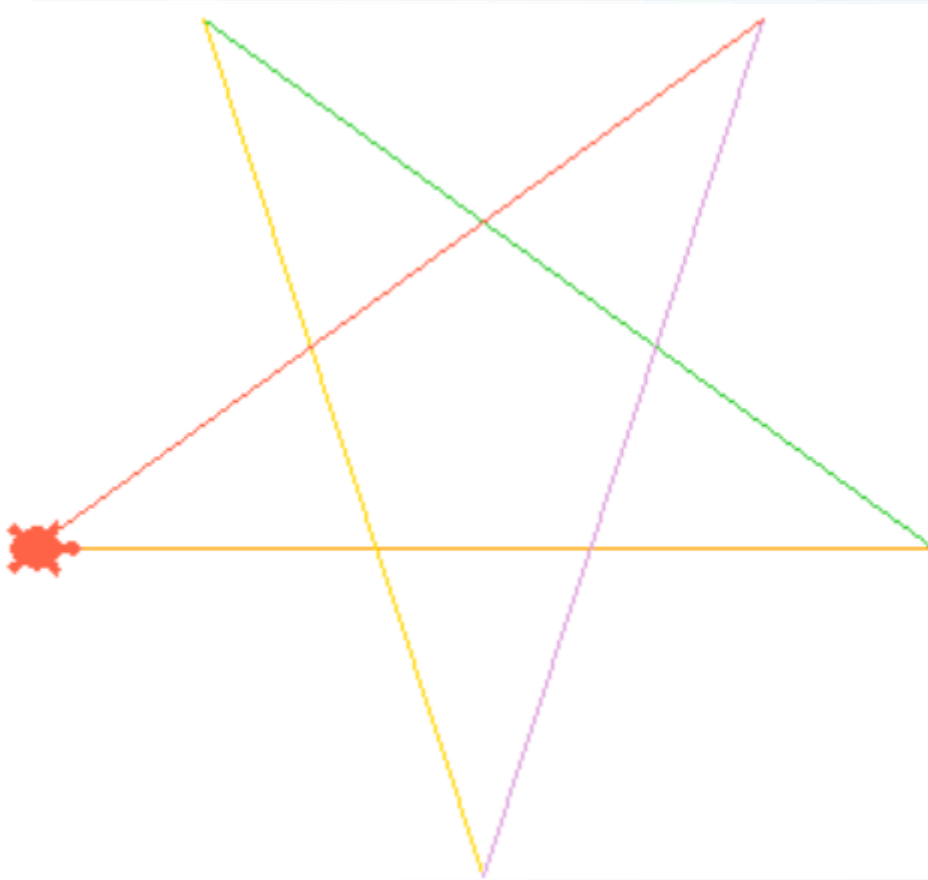
# もくじ



- ・ タートルグラフィックスを知る
- ・ 直線を引く (Python)
- ・ 直線を引く (Scratch)
- ・ 正三角形を書く その1 (Python)
- ・ 正三角形を書く その1 (Scratch)
- ・ 正三角形を書く その2 (Python)
- ・ 正三角形を書く その2 (Scratch)

# タートルグラフィックスを知る

タートルグラフィックスとはカメを動かして絵をかくツールだよ。  
タートルグラフィックスを使っているんな図形を書いてみよう。



# 直線を引く(Python)

まずはタートルグラフィックスで直線を引いてみよう。

線を引くプログラム

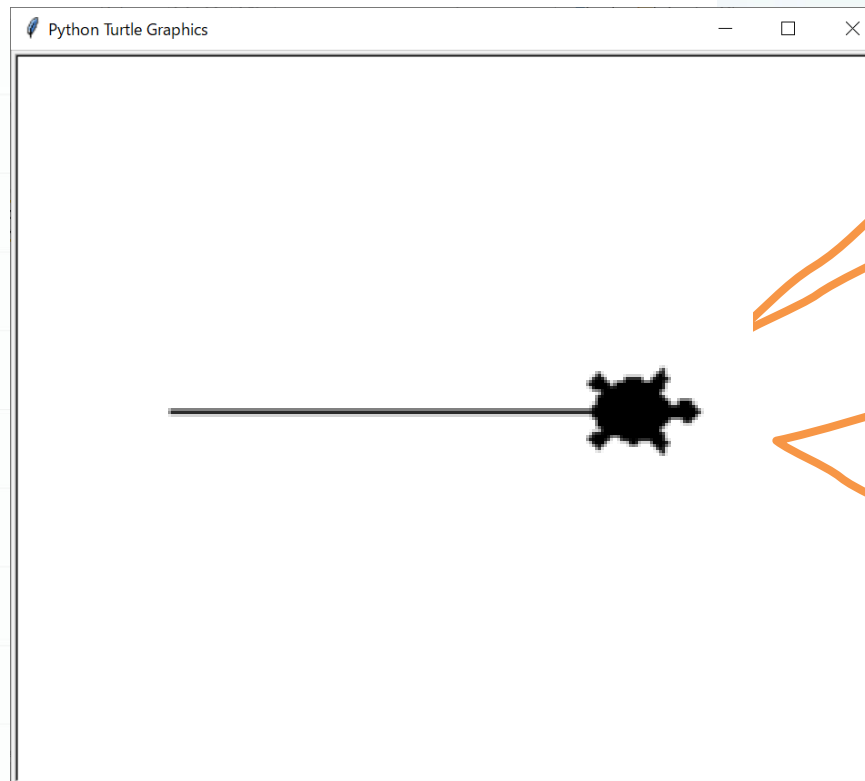
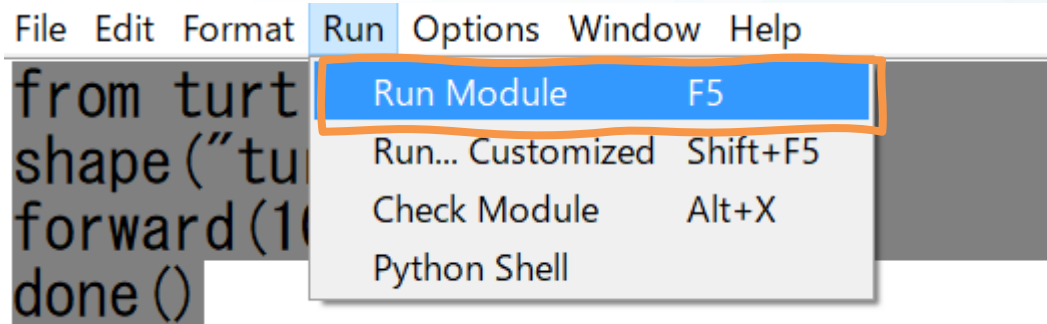
```
from turtle import * . . . . タートルグラフィックスの呼出し  
shape("turtle") . . . . カメを選択  
forward(200) . . . . 前に200進む
```

単語帳 ～英語も一緒に覚えてしまおう！～

- from: ～から
- turtle: カメ
- import: 取り込む。 **輸入する。**
- shape: 形 **姿 輪郭**
- forward: 前

# 直線を引く(Python)

「Run Module」またはF5キーを押してプログラムを実行する。



カメが前に向かって200  
進むことを確認する。

カメ以外にも  
arrow  
circle  
square  
triangle  
classic  
のアイコンも用意されている



# 直線を引く(Scratch)

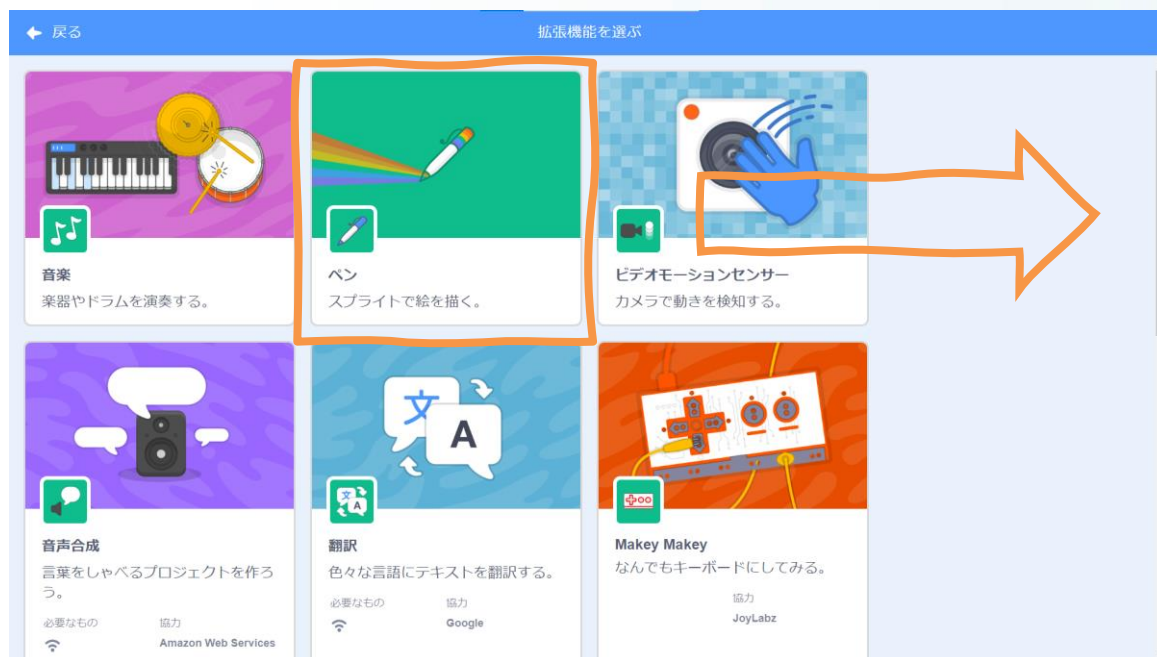
まずはタートルグラフィックスで直線を引いてみよう。

## 線を引くプログラム



．．．．．ペンを下ろして線がひける状態にする

．．．．．カメが200歩、移動する



．．絵を描くにはscratchの拡張機能呼び出す必要がある

# 直線を引く (Scratch)

ブロックをクリックしてプログラムを実行する。



# 【参考】直線を引く(Scratch)

何度も繰り返し実行するには以下のように書く

The image shows the Scratch web interface. The 'Code' tab is selected. A loop of five blocks is highlighted with an orange box:

- 全部消す (Erase all)
- x座標を 0、y座標を 0 にする (Set x to 0, y to 0)
- ペンを下ろす (Put pen down)
- 1 秒待つ (Wait 1 sec)
- 200 歩動かす (Move 200 steps)
- ペンを上げる (Put pen up)

An orange arrow points from this loop to a preview window on the right. The preview window shows a turtle character at the bottom right of a stage, with a horizontal blue line drawn to its left. The stage's coordinate system is visible on the right, with x and y axes and a direction of 90 degrees.



# from文を理解する(Python)

from文を使えば拡張機能を取り込むことができる。つまり、Pythonにない機能を使うことができるようになる。



拡張機能の取り込みはscratchの「拡張機能を選ぶ」ボタンと同じ

## コードの書き方

```
from 拡張機能 import *
```

fromの後ろに呼び出したい拡張機能の名前を書く。  
importの後ろの\*（アスタリスク）は拡張機能がもつすべての機能を呼び出すという意味。

# 正三角形を書く その1 (Python)

次に正三角形を書いてみよう。

## 正三角形を描くプログラム

|                                   |                         |
|-----------------------------------|-------------------------|
| <code>from turtle import *</code> | ・ ・ ・ ・ タートルグラフィックスの呼出し |
| <code>shape("turtle")</code>      | ・ ・ ・ ・ カメを選択           |
| <code>forward(200)</code>         | ・ ・ ・ ・ まっすぐ200進む       |
| <code>left(120)</code>            | ・ ・ ・ ・ 120度曲がる         |
| <code>forward(200)</code>         | ・ ・ ・ ・ まっすぐ200進む       |
| <code>left(120)</code>            | ・ ・ ・ ・ 120度曲がる         |
| <code>forward(200)</code>         | ・ ・ ・ ・ まっすぐ200進む       |
| <code>left(120)</code>            | ・ ・ ・ ・ 120度曲がる         |

# 正三角形を書く その1 (Python)

単語帳 ～英語も一緒に覚えてしまおう！～

■from:～から

■turtle:カメ

■import:取り込む 輸入する。

■shape:形 姿 輪郭

■forward:前

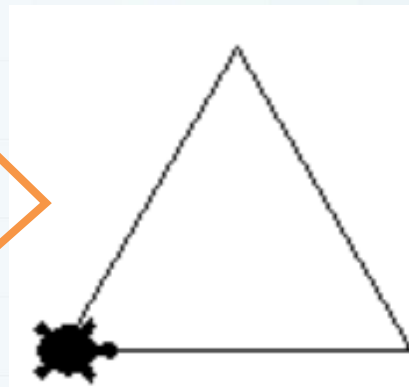
■left:左

「Run Module」またはF5キーを押してプログラムを実行する。

File Edit Format Run Options Window Help

```
from turtle import *\nshape("turtle")\nforward(100)\ndone()
```

Run Module F5  
Run... Customized Shift+F5  
Check Module Alt+X  
Python Shell



カメが前に100進む。  
120度向きを変える。  
前に100進む。  
120度向きを変える。  
前に100進む。

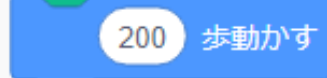
# 正三角形を書く その1 (Scratch)

次に正三角形を書いてみよう。

## 正三角形を描くプログラム



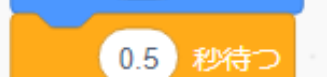
．．．．ペンを下ろして線がひける状態にする



．．．．カメが200歩、移動する



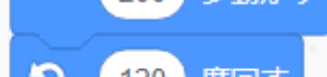
．．．．カメが半時計回りに120度回転する



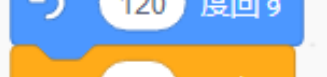
．．．．0.5秒待つ



．．．．カメが200歩、移動する



．．．．カメが半時計回りに120度回転する



．．．．0.5秒待つ



．．．．カメが200歩、移動する



．．．．カメが半時計回りに120度回転する

# 正三角形を書く その1 (Scratch)

ブロックをクリックしてプログラムを実行する。

The image shows the Scratch web interface. The 'Code' tab is selected. On the left, the 'Motion' (動き) category is expanded. A sequence of blocks is highlighted with an orange box and a hand cursor, indicating the steps to draw a triangle:

- ペンを下ろす (Pen down)
- 200 歩動かす (Move 200 steps)
- 120 度回す (Turn 120 degrees)
- 0.5 秒待つ (Wait 0.5 seconds)
- 200 歩動かす (Move 200 steps)
- 120 度回す (Turn 120 degrees)
- 0.5 秒待つ (Wait 0.5 seconds)
- 200 歩動かす (Move 200 steps)
- 120 度回す (Turn 120 degrees)

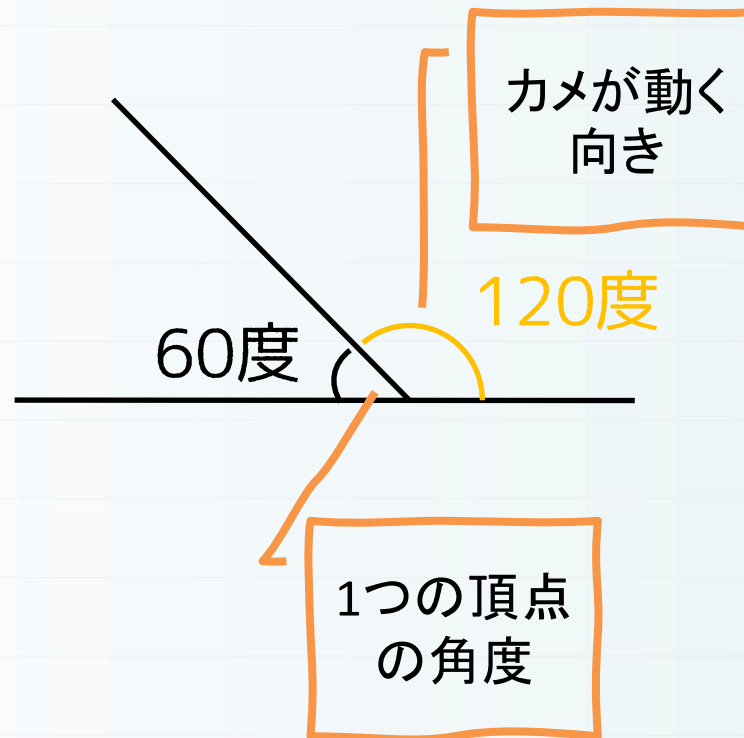
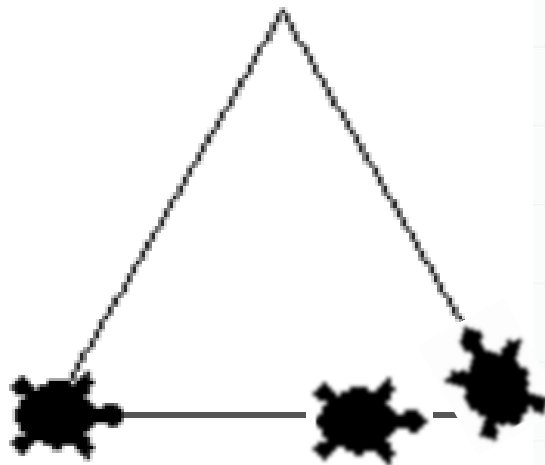
The stage area shows a turtle cursor at the bottom-left vertex of a blue equilateral triangle. A small inset window shows the turtle's current position and orientation: x: -107, y: -78, size: 60, direction: 90. The bottom status bar says 'バックバック' (Back Back).



# 角度について

## カメが向く角度の考え方

カメが向く角度は  
 $180\text{度} - 1\text{つの頂点の角度} = \text{カメが動く向き}$



正三角形の場合、1つの頂点の角度は60度のため、カメが向く角度は120度になる

# 正三角形を書く その2 (Python)

もっとシンプルなプログラムで正三角形を書いてみよう。

正三角形を描くプログラム

```
from turtle import *      . . . . タートルグラフィックスの呼出し
shape("turtle")           . . . . カメを選択
for i in range(3):         . . . . 以下の2行を3回くり返す
    forward(200)           . . . . まっすぐ200進む
    left(120)              . . . . 120度曲がる
```

単語帳 ～英語も一緒に覚えてしまおう！～

■from:～から   ■turtle:カメ   ■import:取り込む。 **輸入する。**  
■shape:形   **姿   輪郭**   ■for:～のために   ■in:～の中に  
■range:範囲   ■forward:前   ■left:左

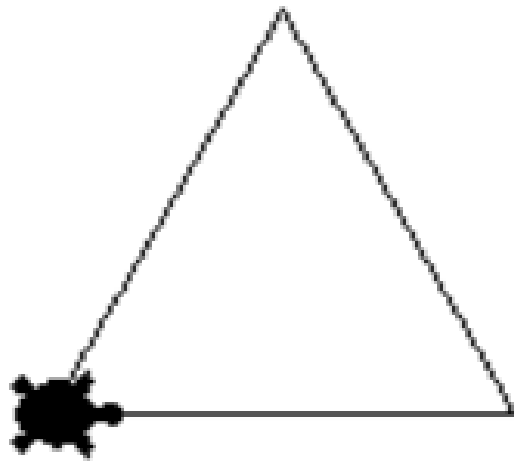
# 正三角形を書く その2

「Run Module」またはF5キーを押してプログラムを実行する。

File Edit Format Run Options Window Help

```
from turtle  
shape("turtle")  
forward(100)  
done()
```

|                   |          |
|-------------------|----------|
| Run Module        | F5       |
| Run... Customized | Shift+F5 |
| Check Module      | Alt+X    |
| Python Shell      |          |



カメラが前に200進む。  
120度向きを変える。  
前に200進む。  
120度向きを変える。  
前に200進む。

# 正三角形を書く その2 (Scratch)

もっとシンプルなプログラムで正三角形を書いてみよう。

正三角形を描くプログラム



．．．ペンを下ろして線がひける状態にする

．．．①～③の処理を3回繰り返す

．．．①カメが200歩、移動する

．．．②カメが半時計回りに120度回転す

．．．③0.5秒待つ

# 正三角形を書く その2 (Scratch)

ブロックをクリックしてプログラムを実行する。





# for文を理解する

For文を使えば繰り返し処理を実行することができる。



繰り返し処理とはscratchの「〇回繰り返す」ブロックと同じ

5回繰り返す処理を書くコード（変数を使わない場合）

```
for 変数 in range(5): . . . . 関数を使うときは最後に「:」をつける  
    forward(200)  
    left(90) } . . . . 繰り返しの中で行う処理
```

※forとinの間には変数が入る。for文で使用する変数のことをカウンタ変数という。カウンタ変数を使用しない場合でもforとinの間に変数を書く必要がある。

5回繰り返す処理を書くコード（変数を使う場合）

```
for 変数 in range(5):  
    print(変数)
```



```
0  
1  
2  
3  
4  
>>> |
```

# メモ

---



# プログラミング教室の テクノロ

なまえ：