

13 海龜繪圖



海龜圖是學習Python和使用程式碼創作藝術作品的有趣方式，我們將透過輸入指令用虛擬海龜游標在螢幕上畫圖。

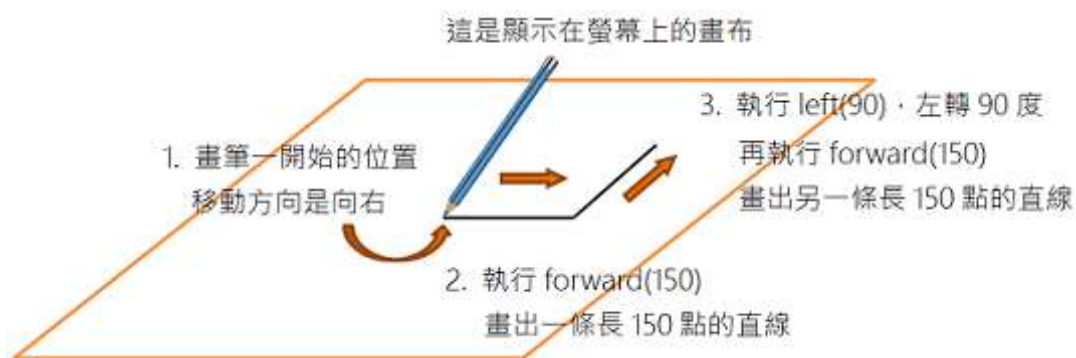
13-1. 如何在Python畫畫

- 建議用Spyder IDE(annoconda大套件)，Colab無法執行
- 使用海龜模組

要開始使用turtle模組，首先需要導入它。方法很簡單，直接輸入import，後面則接著我們想要使用的模組。

也就是，開頭輸入 `import turtle` 程式碼，表示要啟動海龜模組、使用海龜圖。此時，在螢幕上仍然看不到任何東西，但在幕後電腦已經準備好了，可以開始存取turtle模組不同片段的程式碼。

- 創造一個海龜（畫筆畫畫 => 海龜走路）



建立一個海龜物件，並且指派海龜變數名稱。

- 如 `shelly = turtle.Turtle()`
- 改變海龜形狀，輸入 `shelly.shape("turtle")`
 - 可以用 `arrow`, `circle`, `classic`, `square`, `triangle`
- 找出海龜的位置
 - `shelly.position()`

你可以看到一個新視窗跳出，海龜就在哪裡；請控制這個視窗，並用下面的指令讓海龜游標來畫圖。這些指令都是事先寫好存在海龜模組裏的程式碼。

```
1  ## 啟動模組
2  import turtle #導入海龜模組
3  shelly = turtle.Turtle() # 把模組中的螢幕游標叫出來
4  shelly.shape("turtle") # 將三角形改為烏龜
5  shelly.position() # 烏龜的位置
```

13-2. 常用指令整理

英文版：<https://docs.python.org/3/library/turtle.html>

簡中版：<https://docs.python.org/zh-cn/3/library/turtle.html>

=> 使用指令的方法：點記法(Dot Notation) - 物件.函數()

(1)運動命令:

- position():位置
- goto(x,y):將畫筆移動到座標為(x,y)的位置
- forward(d):向前移動距離d代表距離
- backward(d):向後移動距離d代表距離
- right(degree):向右轉動多少度
- left(degree):向左轉動多少度
- speed(speed):畫筆繪製的速度範圍[0,10]整數

```

1  ## 啟動模組
2  import turtle  #導入海龜模組
3  shelly = turtle.Turtle()  # 把模組中的螢幕游標叫出來
4  ## 海龜運動命令
5  shelly.goto(0,0)
6  ## 畫一條線長100
7  shelly.forward(100)
8  shelly.backward(100)
9  ## 向左轉90度
10 shelly.left(90)
11 shelly.right(90)

```

(2)畫筆控制命令:

- penup():畫筆抬起，移動時不繪製圖形
- pendown():畫筆落下，移動時繪製圖形
- pensize(width):畫筆的寬度
- pencolor(colorstring):畫筆的顏色
 - white,yellow, blue, black, gold, pink, brown, purple, red, gold, seashell. tomato...
- fillcolor(colorstring):繪製圖形的填充顏色。
- color(colorstring):繪製圖形的填充顏色。
 - 例：turtle.color(color1, color2)，同時設定pencolor=color1, fillcolor=color2。
- circle(radius, extent):繪製一個圓形
 - 其中radius為半徑，extent為度數，例如若extent為180，則畫一個半圓；畫一個圓形，可不必寫第二個參數
- setheading(degree):海龜朝向，degree代表角度
- reset():恢復所有設置

```

1  ## 啟動模組
2  import turtle  #導入海龜模組
3  shelly = turtle.Turtle()  # 把模組中的螢幕游標叫出來
4  ## 畫筆控制
5  ## 放下筆貼近畫布
6  shelly.pendown()

```

```

7  ## 要填滿繪製的圖案-開始
8  shelly.begin_fill()
9  ## 用什麼顏色
10 shelly.fillcolor("orange")
11 ## 繪製圖形
12 shelly.circle(250) #半徑250的圓
13 ## 要填滿繪製的圖案-停止
14 shelly.end_fill()
15 ## 提起筆離開畫布
16 shelly.penup()

```

問題：海龜繪圖末尾會出現「沒有回應」的現象

解法：程式末尾必須把控制權交給海龜或結束海龜

問題：執行海龜繪圖時，無法每次都可以執行，會有可執行、被中斷的情形交互出現

解法：重開螢即可幕，或者在程式開頭設定特定變數 `turtle.TurtleScreen._RUNNING = True`。

```

1  import turtle
2  turtle.TurtleScreen._RUNNING = True # 加上特定變數
3  turtle.setup(width=500, height=250)
4  turtle.forward(100)
5
6  turtle.done # 交給海龜控制權

```

13-3 畫圖

13-3-1 用線條畫出文字（略）

範例1：尺寸為100的正方形

虛擬程式碼

- 向前移動100(250)步
- 向左轉90度
- 向前移動100步
- 向左轉90度
- 向前移動100步
- 向左轉90度
- 向前移動100步
- 向左轉90度

```

1  # 用線條畫出文字-口字
2  import turtle
3  shelly = turtle.Turtle()
4  shelly.forward(100)
5  shelly.right(90)
6  shelly.forward(100)
7  shelly.right(90)
8  shelly.forward(100)

```

```

8  shelly.forward(100)
9  shelly.right(90)
10 shelly.forward(100)
11 shelly.right(90)

```

加分練習題：請畫凸/凹/世/田/王等字

13-3-2 幾何繪圖

範例2：尺寸為100的正方形 - 請用for迴圈

```

1  ## 方法 (一) : for 迴圈
2  import turtle
3  shelly = turtle.Turtle()
4
5  for i in range(4):      # 畫正方形 - 1 + 2 做四次 [0,1,2,3]
6      shelly.forward(250) # 1. 畫直線
7      shelly.left(90)     # 2. 轉向
8      print (i)           # 計數，一共四次

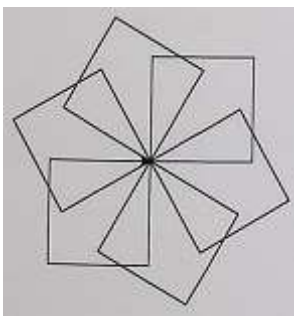
```

```

1  ## 方法 (二) : for 迴圈 + 色彩 (紅色)
2  import turtle
3  shelly = turtle.Turtle()
4  shelly.color("red") # 顏色 - 線用紅色
5
6  for i in range(4): # 形狀 - 正方形
7      shelly.forward(250)
8      shelly.left(90)

```

範例3：尺寸為100的正方形 - 巢狀迴圈



虛擬碼

以下重複六次：

 以下重複四次：

 前進250步

 左轉90度

 右轉60度

```

1  ##方法 (一) : 巢狀迴圈
2  import turtle
3  shelly = turtle.Turtle()
4  for n in range(6):      # 外迴圈重複正方形6次
5      for i in range(4):  # 內迴圈重複4次來畫正方形

```

```

6         shelly.forward(250)
7         shelly.left(90)
8         shelly.right(60)           # 畫下一個正方形前轉彎

```

```

1  ## 方法(二)：巢狀迴圈 + 加上色彩
2  import turtle
3  shelly = turtle.Turtle()
4  colors = ["red", "green", "blue", "black", "yellow", "purple"]
5
6  for n in range(6):           # 外迴圈重複正方形6次
7      shelly.color(colors[n])
8      for i in range(4):       # 內迴圈重複4次來畫正方形
9          shelly.forward(250)
10         shelly.left(90)
11         shelly.right(60)      # 畫下一個正方形前轉彎

```

```

1  ## 方法(三)：巢狀迴圈 + 填滿色彩
2  import turtle
3  shelly = turtle.Turtle()
4  colors = ["red", "green", "blue", "black", "purple", "yellow"]
5
6  ## 外迴圈重複正方形6次
7  for n in range(6):
8      shelly.color(colors[n])
9      shelly.begin_fill()
10     shelly.fillcolor(colors[n])
11
12     for i in range(4):       # 內迴圈重複4次來畫正方形
13         shelly.forward(250)
14         shelly.left(90)
15     shelly.right(60)         # 畫下一個正方形前轉彎
16     shelly.end_fill()

```

隨堂練習1：讓我們畫星星

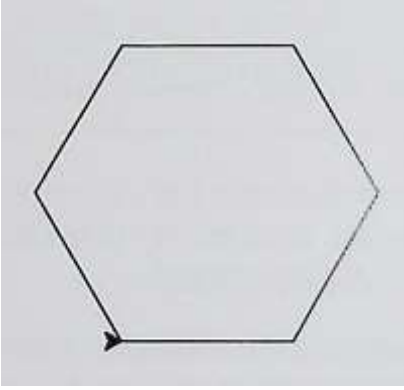
利用turtle模組，以for迴圈畫一顆星星。

13-4 專題：打造你的藝術傑作

Step 1: 畫出六角型

在任何海龜專題的開頭都要先匯入海龜模組，這樣才可以使用裡面的函數。

- 在頂端加入一行註解來提醒自己這個專題是什麼。
- 修改畫正方形的虛擬程式碼，變成六角形。
 - 數量是6
 - 轉彎角度是60



虛擬碼

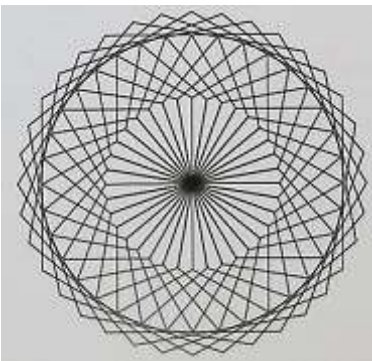
以下重複六次

前進100步

左轉60度

```
1  ## 畫出幾何圖案 (一) : 六角型
2  import turtle
3  shelly = turtle.Turtle()
4  ## 重複6次 : 前進並轉彎
5  for i in range(6):
6      shelly.forward(100)
7      shelly.left(60)
```

Step 2: 用巢狀迴圈重複六角形



用迴圈畫出六角形之後，就能把此六角形放在另一個重複的迴圈裡，畫出一個圓的多個六角形，各個稍微重疊。

- 每個六角形相對於前一個六角形只轉10度。
- 畫一個圓，一共需要 $360/10=36$ 個六角形

虛擬碼

以下重複36次

以下重複6次

前進100步

左轉60度

右轉10度

```
1  ## 畫出幾何圖案 (二) : 重複六角型
2  import turtle
3  shelly = turtle.Turtle()
4  for n in range(36):
```

```

5  ## 重複6次：前進並轉彎
6      for i in range(6):
7          shelly.forward(100)
8          shelly.left(60)
9      shelly.right(10) #加入轉彎

```

Step 3: 變更背景：加入彩虹顏色

我們可以加入色彩和背景讓這個圖更有趣。



- 背景變黑
 - `turtle.bgcolor("black")`
- 建立顏色清單，然後用迴圈取顏色。
 - `colors = ["red", "yellow", "blue", "orange", "green", "red"]`
 - `shelly.color(colors[i])`

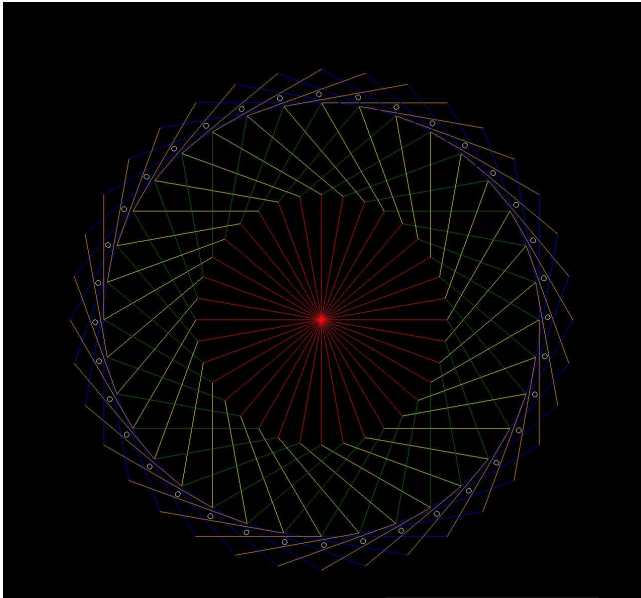
```

1  ## 畫出幾何圖案 (三)：彩虹重複六角型
2  import turtle
3  shelly = turtle.Turtle()
4
5  turtle.bgcolor("black") #把背景變黑
6  ## 畫出36個六角形，每隔10度
7  for n in range(36):
8      ## 選擇六角形顏色順序
9          colors = ["red", "yellow", "blue", "orange", "green", "purple"]
10     ## 重複6次來畫六角形
11         for i in range(6):
12             shelly.color(colors[i]) #選擇i位置的顏色
13             shelly.forward(100)
14             shelly.left(60)
15     ## 在畫下一個六角形前轉彎
16     shelly.right(10)

```

Step 4: 在圖案周圍加上小白圓圈

在海龜圖案的邊緣，畫一個小圓圈。



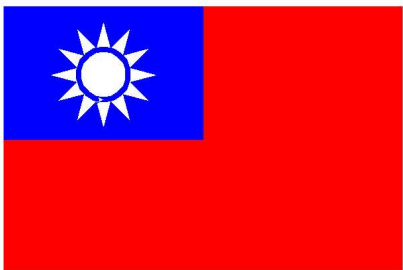
- 因為有36個六角形，所以會有36個小白圓圈。
- 畫一個小白圓圈，然後退回原點，然後轉10度；再出去畫一個小白圓圈，反覆執行。

```

1  ## 畫出幾何圖案 ( 四 ) : 小圓圈彩虹重複六角型
2  ## 畫36個小圓圈
3  shelly.penup()
4  shelly.color("white")
5  ## 重複36次，找到對應的六角形
6  for i in range(36):
7      shelly.forward(220)
8      shelly.pendown()
9      shelly.circle(5)
10     shelly.penup()
11     shelly.backward(220)
12     shelly.right(10)
13  ## 隱藏海龜
14  shelly.hideturtle()

```

加分練習題：中華民國國旗



```

1  import turtle
2  turtle.TurtleScreen._RUNNING = True
3  turtle.speed(0)
4
5  turtle.penup()
6  turtle.goto(-300, 200)
7  turtle.pendown()
8
9  turtle.color('blue', 'red')

```



```
9  turtle.color('red', 'red')
10 turtle.begin_fill()
11 turtle.forward(600)
12 turtle.right(90)
13 turtle.forward(400)
14 turtle.right(90)
15 turtle.forward(600)
16 turtle.right(90)
17 turtle.forward(400)
18 turtle.right(90)
19 turtle.end_fill()
20
21 turtle.color('blue', 'blue')
22 turtle.begin_fill()
23 turtle.forward(300)
24 turtle.right(90)
25 turtle.forward(200)
26 turtle.right(90)
27 turtle.forward(300)
28 turtle.right(90)
29 turtle.forward(200)
30 turtle.right(90)
31 turtle.end_fill()
32
33 turtle.penup()
34 turtle.goto(-150, 175)
35 turtle.pendown()
36
37 turtle.color('white', 'white')
38 turtle.begin_fill()
39 turtle.right(75)
40 for i in range(12):
41     turtle.forward(150)
42     turtle.right(150)
43 turtle.end_fill()
44
45 turtle.penup()
46 turtle.goto(-150, 60)
47 turtle.pendown()
48
49 turtle.setheading(0)
50 turtle.pencolor('blue')
51 turtle.pensize(8)
52 turtle.circle(38)
53
54 turtle.done()
```

學習愉快 - 烏龜皮卡秋



```

1  ## 皮卡秋 - 函數function
2
3  import turtle as t
4  def nose():
5      t.penup()
6      t.seth(90)
7      t.fd(100)
8      t.pendown()
9      t.begin_fill()
10     t.fillcolor("black")
11     t.seth(45)
12     t.fd(25)
13     t.seth(135)
14     t.circle(25,90)
15     t.seth(315)
16     t.fd(25)
17     t.end_fill()
18
19  def eyes(seth,fd,c):
20     t.penup()
21     t.seth(seth)
22     t.fd(fd)
23     t.pendown()
24     t.begin_fill()
25     t.fillcolor('black')
26     t.circle(50)
27     t.end_fill()
28     t.penup()
29     t.circle(50,c)
30     t.pendown()
31     t.begin_fill()
32     t.fillcolor('white')
33     t.circle(20)
34     t.end_fill()
35
36  def face(seth,fd):
37     t.penup()
38     t.seth(seth)
39     t.fd(fd)
40     t.pendown()
41     t.begin_fill()
42     t.fillcolor('red')
43     t.circle(70)

```

```
44     t.end_fill()
45
46 def lip():
47     t.penup()
48     t.seth(135)
49     t.fd(250)
50     t.pendown()
51     t.seth(-300)
52     t.circle(30,-65)
53     t.begin_fill()
54     t.fillcolor('Firebrick')
55     t.seth(165)
56     t.fd(140)
57     t.seth(195)
58     t.fd(140)
59     t.seth(-360)
60     t.circle(30,-65)
61     t.penup()
62     t.seth(-60)
63     t.circle(30,65)
64     t.pendown()
65     t.seth(-70)
66     t.fd(240)
67     t.circle(55,140)
68     t.seth(70)
69     t.fd(240)
70     t.end_fill()
71     t.seth(-110)
72     t.fd(80)
73     t.begin_fill()
74     t.fillcolor('Firebrick')
75     t.seth(120)
76     t.circle(120,123)
77     t.seth(-70)
78     t.fd(165)
79     t.circle(55,140)
80     t.seth(72)
81     t.fd(165)
82     t.end_fill()
83
84 def setting():
85     t.pensize(4)
86     t.hideturtle()
87     t.setup(1000,600)
88     t.speed(10)
89     t.screensize(bg='yellow')
90
91 def main():
92     setting()
93     nose()
94     eyes(160,250,60)
95     eyes(-9.5,530,230)
96     face(195,600)
97     face(-11,720)
```

```
97         tcc, 11, 10,  
98         lip()  
99         t.done()  
100  
101 if __name__ == '__main__':  
102     main()
```