

# PYTHON 程式設計

# Python IDE介紹(Thonny)

波羅的海三小國之一愛沙尼亞 (Estonia) 的 TARTU 大學計算機科學系所研發的 Python 編輯器, 此軟體輕盈小巧(只有10MB), 非常適合用來開發 Python 應用, 而且此編輯器內含 Python 3.6.4 版解譯器。

\*IDE 整合開發環境

Integrated Development Environment

# Python 繪圖( turtle 模組)

```
from turtle import *
```

#載入turtle 模組

```
forward(100)
```

#前進100點(pixel)

```
left(90)
```

#左轉90度

```
forward(100)
```

# Python 繪圖( turtle 模組)

## 屬性介紹

|                                                       |            |
|-------------------------------------------------------|------------|
| <code>pensize(3)</code>                               | #設定筆的大小為3  |
| <code>color('red')</code> 或 <code>color("red")</code> | #設定筆的顏色    |
| <code>forward()</code>   <code>fd()</code>            | #前進        |
| <code>backward()</code>   <code>bk()</code>           | #後退        |
| <code>left()</code>   <code>lt()</code>               | #左轉        |
| <code>right()</code>   <code>rt()</code>              | #右轉        |
| <code>speed()</code>                                  | #設定畫筆移動的速度 |

速度等級是1到10之間的整數  
數字愈大，畫筆移動愈快。

井字號(#)做為註解符號，同一列井字號後的任何字將被忽略

# Python 繪圖( turtle 模組)

## 屬性介紹

penup()：提筆之後路徑不會被畫出

pendown()：下筆之後路徑會被畫出

goto(x,y)：將游標移動到(x, y)的位置

circle ( radius )：畫出半徑為 radius 的圓

circle ( radius, extent )：畫出半徑為radius，角度  
為 extent的弧

begin\_fill()：開始填入顏色

fillcolor('red')或fillcolor("red")

end\_fill()：停止填入顏色

# Python 繪圖( turtle 模組)

EX1. 畫一個正方形

EX2. 畫藍色菱形

EX3. 畫直立的藍色菱形

```
from turtle import *  
speed(8)  
pensize('3')  
begin_fill()  
color('red')  
fillcolor("blue")  
circle(200)  
end_fill()
```

# 重覆結構介紹

For 迴圈格式

```
for i in range(次數):
```

```
=>for i in range(3):
```

總共執行**3**次，i的值為**0,1,2**

冒號(:)做為區塊敘述的開始

```
=>for i in range(3):
```

```
    print (i)
```

i是變數，存在記憶體中，其值可能會改變

# 重覆結構介紹

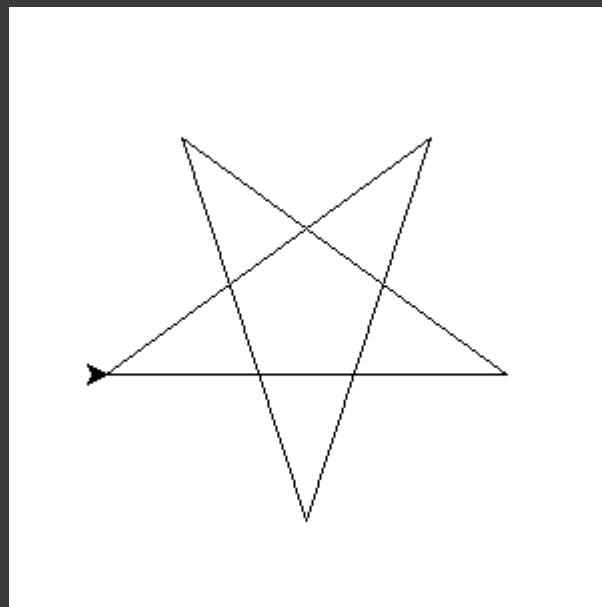
Ex. 執行下列程式

```
from turtle import *  
for i in range(36):  
    forward(200)  
    left(110)
```

Ex. 修改程式，畫出新的圖形

# 重覆結構介紹

Ex. 畫出右邊圖形



Ex. 修改程式，畫出新的圖形

# 重覆結構介紹

for i in range(初始值,終止值,間距):

間距預設值為1，可省略

初始預設值為0

```
for j in range(1,4):
```

```
    print (j)
```

執行 3 次，j=1,2,3(不含4)

```
for j in range(5,1,-1):
```

```
    print (j)
```

執行 4 次，j=5,4,3,2

# 重覆結構介紹

Ex1.列印0~100的所有整數

Ex2.列印1~100中所有奇數

Ex3.列印205~1006中所有6的倍數

程式追蹤

1. for i in range(1,15,2):

2.   print(i)

| 行號 | 變數i的值 | 程式執行結果 |
|----|-------|--------|
| 1  | i=1   |        |
| 2  | 列印i的值 | 1      |
| 1  | i=3   |        |
| 2  | 列印i的值 | 3      |
| 1  | i=5   |        |
| 2  | 列印i的值 | 5      |
| 1  | i=7   |        |
| 2  | 列印i的值 | 7      |
| 1  | i=9   |        |
| 2  | 列印i的值 | 9      |
| 1  | i=11  |        |
| 2  | 列印i的值 | 11     |
| 1  | i=13  |        |
| 2  | 列印i的值 | 13     |
|    |       | 程式結束   |

# 重覆結構介紹

程式追蹤

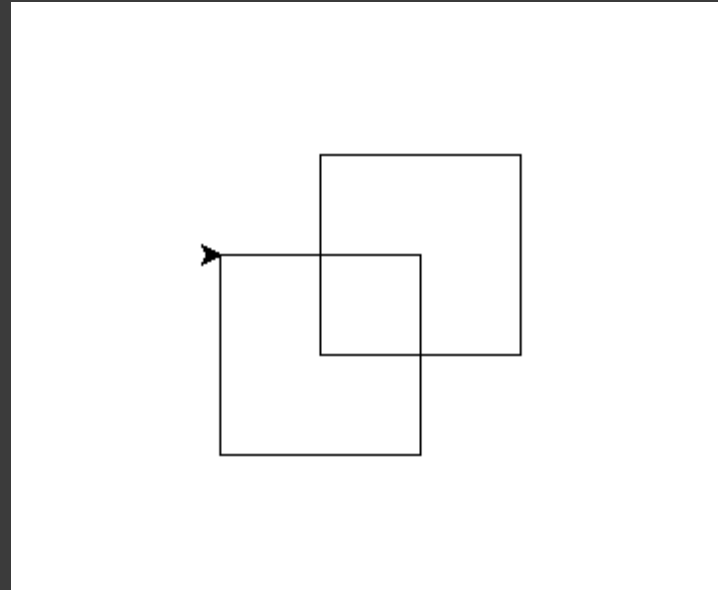
1. for i in range(1,15,2):

2.   print(i)

| 行號 | 變數( i )的值 | 程式執行結果 |
|----|-----------|--------|
| 1  | i=3       |        |
| 2  | 列印i的值     | 3      |
| 1  | i=5       |        |
| 2  | 列印i的值     | 5      |
| 1  | i=7       |        |
| 2  | 列印i的值     | 7      |
| 1  | i=9       |        |
| 2  | 列印i的值     | 9      |
| 1  | i=11      |        |
| 2  | 列印i的值     | 11     |
| 1  | i=13      |        |
| 2  | 列印i的值     | 13     |
|    |           | 程式結束   |

# 重覆結構介紹

Ex1



# 重覆結構介紹

利用函式

```
from turtle import *
```

```
def square (s):
```

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

```
        fd(s)
```

```
        rt(90)
```

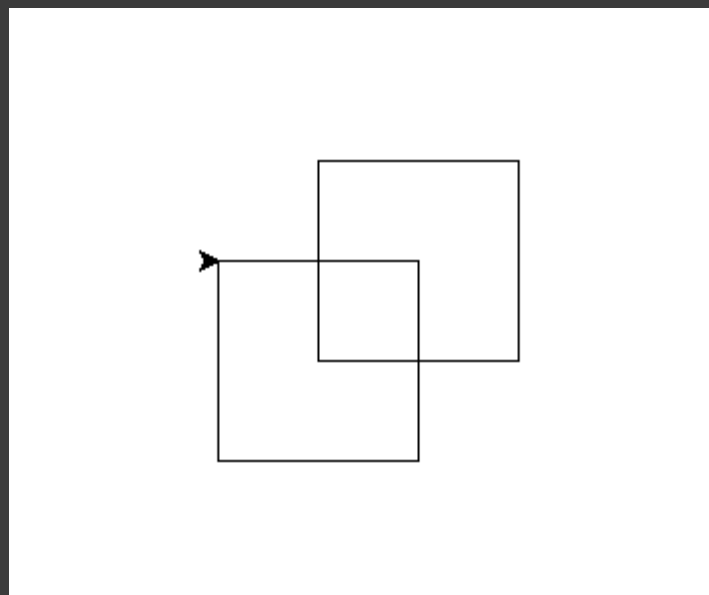
```
square (100)
```

```
penup()
```

```
goto (-50,-50)
```

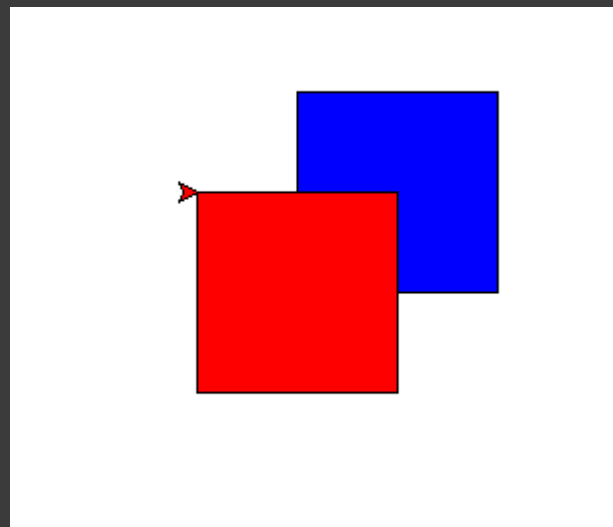
```
pendown()
```

```
square (100)
```



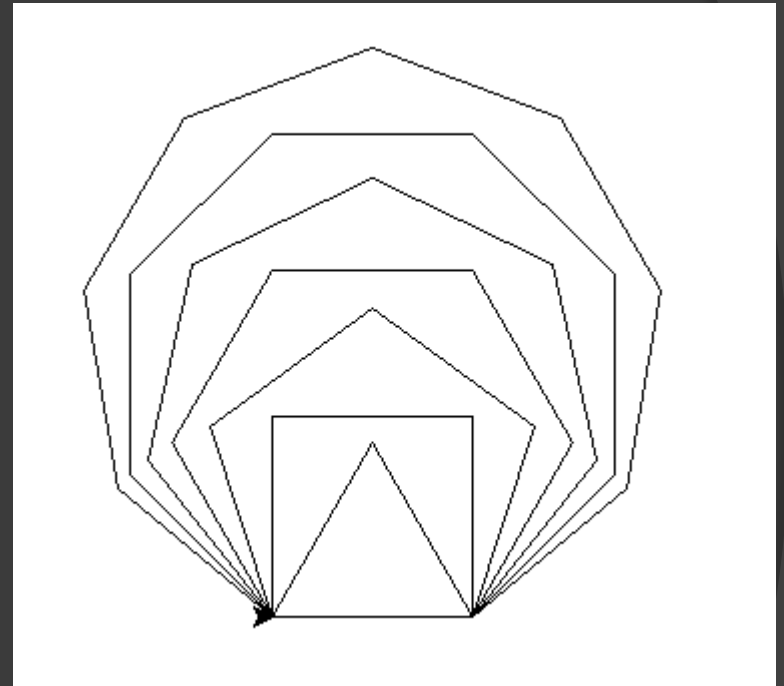
# 重覆結構介紹

利用函式



# 重覆結構介紹

Ex. 畫出右邊的圖形

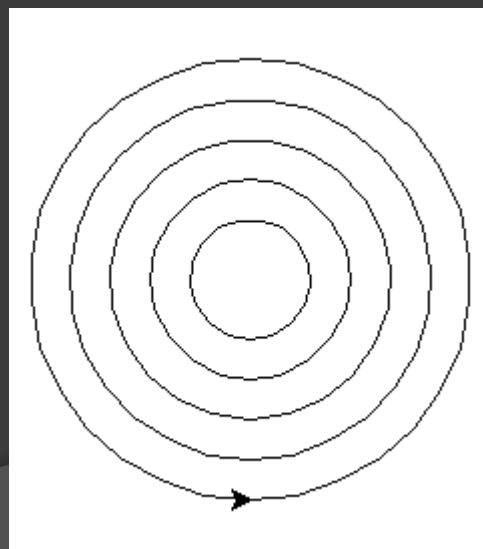


# 重覆結構介紹

Ex. 畫出圖形

```
from turtle import *  
speed(10)  
for i in range(50,170,20):  
    circle(i)
```

Ex. 畫出右圖



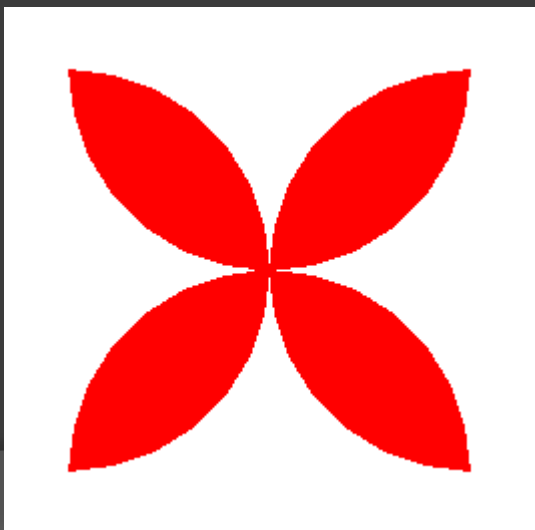
# 重覆結構介紹

Ex. 利用函式畫弧(花朵)

```
def petal(s):  
    circle(s,90)  
    left(90)  
    circle(s,90)
```

```
def flower(s):  
    for i in range(4):  
        petal(s)
```

Ex. 畫出



```
from turtle import *  
speed(10)  
def petal(s):  
    circle(s,90)  
    left(90)  
    circle(s,90)
```

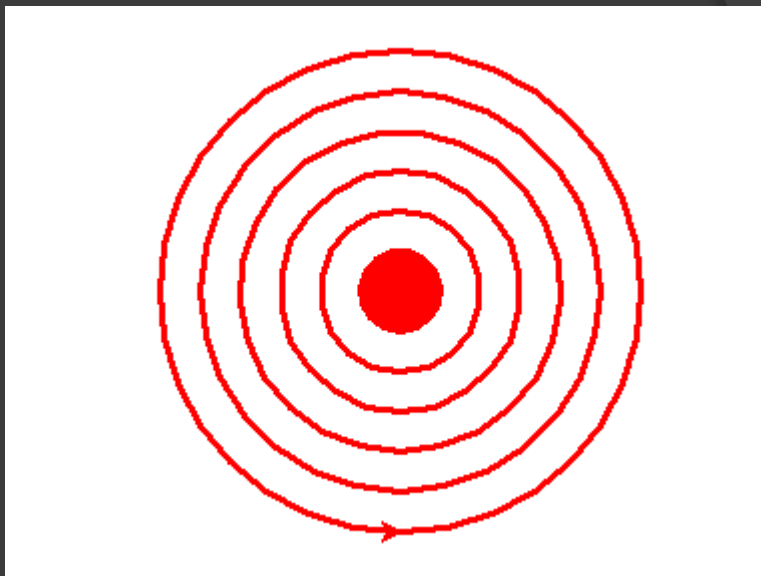
```
def flower(s):  
    for i in range(4):  
        petal(s)  
begin_fill()  
color('red')  
fillcolor('red')  
flower(100)  
end_fill()
```

# 重覆結構介紹

Ex. 畫出圖形

Ex. 畫出

```
from turtle import *  
speed(8)  
pencolor('red')  
pensize(3)  
begin_fill()  
color('red')  
circle(20)  
end_fill()  
for i in range(40,140,20):  
    penup()  
    goto(0,20-i)  
    pendown()  
    circle(i)
```



# 基本輸入輸出

Print(變數,sep="分隔符號",end="結束符號")

print ('Hello World')或print ("Hello World")

print (1,1+2,'1+2')

print(1,1+2,'1+2',sep=' / ')=>1/3/1+2

print(1,1+2,'1+2',sep=" ' ")=>1'3'1+2

Ex. 請列印出1"3"1+2

print(1,1+2,'1+2',sep=' ' ) 預設結束符號為\n(換行)

print(1,1+2,'1+2',sep=' ' ,end=' ')設定結束符號就不會換行

print(1,1+2,'1+2',sep=' ' ,end='p')

# 基本輸入輸出

Ex.

a=15

b=35

print ( )

印列結果為a+b=50

# 基本輸入輸出

```
name=input('輸入你的帳號：')
```

```
pw=input('輸入你的密碼：')
```

```
print(name,'歡迎你：')
```

```
print('你的密碼是：',pw)
```

# 基本輸入輸出

```
a=input('請輸入第一個數：')
```

```
b=input('請輸入第二個數：')
```

```
print ( )
```

印列結果為兩數的和=

input輸入的內容為字串

```
a=int(input('請輸入第一個數：'))
```

```
b=int(input('請輸入第二個數：'))
```

```
print ('a+b=',a+b,sep="")
```

印列結果為兩數的和=

int(input())輸入的內容為整數

# 基本輸入輸出及運算

## 1. 算術運算元

這些運算元用於基本的數學計算。

| 運算元 | 說明        | 範例                 |
|-----|-----------|--------------------|
| +   | 加法        | 5 + 3 結果為 8        |
| -   | 減法        | 5 - 3 結果為 2        |
| *   | 乘法        | 5 * 3 結果為 15       |
| /   | 除法        | 5 / 3 結果為 1.666... |
| //  | 整數除法      | 5 // 3 結果為 1       |
| %   | 取餘數       | 5 % 3 結果為 2        |
| **  | 指數 ( 次方 ) | 5 ** 3 結果為 125     |

# 基本輸入輸出及運算

## 數字格式化

| Number     | Format  | Output    | Description           |
|------------|---------|-----------|-----------------------|
| 3.1415926  | {:.2f}  | 3.14      | 浮點數，顯示到小數點第2位         |
| 3.1415926  | {:+.2f} | +3.14     | 浮點數，顯示到小數點第2位，顯示正負符號  |
| -1         | {:+.2f} | -1.00     | 浮點數，顯示到小數點第2位，顯示正負符號  |
| 2.71828    | {:.0f}  | 3         | 顯示到小數點第0位，四捨五入        |
| 5          | {:0>2d} | 05        | 將數字用0補全，從左邊開始補全，補到長度2 |
| 5          | {:x<4d} | 5xxx      | 用x補全，從右邊開始補全，補到長度4    |
| 1000000    | {:,}    | 1,000,000 | 用逗號顯示數字格式             |
| 0.25       | {:.2%}  | 25.00%    | 百分比表示法                |
| 1000000000 | {:.2e}  | 1.00e+09  | 指數表示法，科學記號表示法         |
| 13         | {:10d}  | 13        | 靠右對齊，默認長度10           |
| 13         | {:<10d} | 13        | 靠左對齊，默認長度10           |
| 13         | {:^10d} | 13        | 置中對齊，默認長度10           |

# 基本輸入輸出及運算

```
x = int(input("Enter first number: "))  
y = int(input("Enter second number: "))  
print(x+y)  
print(x-y)  
print(x*y)  
print(x/y)  
print(x%y)
```

%取餘數

# 基本輸入輸出及運算

Ex. 寫一個攝氏轉華氏，華氏轉攝氏的程式

$$^{\circ}\text{F} = \frac{9}{5} (^{\circ}\text{C}) + 32$$
$$^{\circ}\text{C} = \frac{5}{9} (^{\circ}\text{F} - 32)$$

請輸入攝氏溫度： 40

請輸入感氏溫度： 30

40 C= 104.0 F

30 F= -1.1111111111111112 C

# 基本輸入輸出及運算

Ex. 寫求函數值的程式

$$f(x)=2x^3+5x^2-3x-5$$

請輸入  $x$  的值: 2  
這個函數代入 2 的值等於 25

# 基本輸入輸出及運算

Ex. 寫求根號值的程式

$$\sqrt{a + b}$$

請輸入a的值: 4

請輸入b的值: 5

4 + 5 開根號的值等於 3.0

# 比較運算元

Python 的比較運算子如下：

1. `==` : 等於 ( 檢查兩個值是否相等 )
2. `!=` : 不等於 ( 檢查兩個值是否不相等 )
3. `>` : 大於
4. `<` : 小於
5. `>=` : 大於或等於
6. `<=` : 小於或等於

這些運算子可以用於比較數值、字串等資料型態，並且回傳布林值 ( `True` 或 `False` )。

# 選擇結構

if判斷式

if條件式：

條件成立時，執行的程式區塊(程式要縮排)

1.if後面要加分號

2.如果條件式不成立則不會執行if區塊的程式，  
會繼續往下執行

# 選擇結構

## if-else判斷式

If條件式：

條件成立時，執行的程式區塊(程式要縮排)

else：

條件不成立時，執行的程式區塊(程式要縮排)

1.if和else後面要加分號

2.if和else要對齊

# 選擇結構

Ex. 判斷a是否為b的倍數

```
請輸入a的值: 15  
請輸入b的值: 3  
15 是 3 的倍數  
3 是 15 的因數
```

```
請輸入a的值: 5  
請輸入b的值: 3  
5 不是 3 的倍數
```

# 選擇結構

## if-elif-else判斷式

if條件式：

條件成立時，執行的程式區塊(程式要縮排)

elif條件式：

條件成立時，執行的程式區塊(程式要縮排)

else：

條件不成立時，執行的程式區塊(程式要縮排)

1.if、elif和else後面要加分號

2.if、elif和else要對齊

# 選擇結構

Ex.判斷成績等第

(分數>100：請勿亂輸入，90~100：A+，80~90：B+，70~80：B，70以下：C)

請輸入考試分數 (1-100) : 85  
成績 : B+

# 巢狀迴圈

Ex. 列印九九乘法表

|   |   |   |   |   |
|---|---|---|---|---|
| 1 | * | 7 | = | 7 |
| 1 | * | 8 | = | 8 |
| 1 | * | 9 | = | 9 |
|   |   |   |   |   |
| 2 | * | 1 | = | 2 |
| 2 | * | 2 | = | 4 |
| 2 | * | 3 | = | 6 |
| 2 | * | 4 | = | 8 |

# 巢狀迴圈

Ex. 列印

★

★ ★

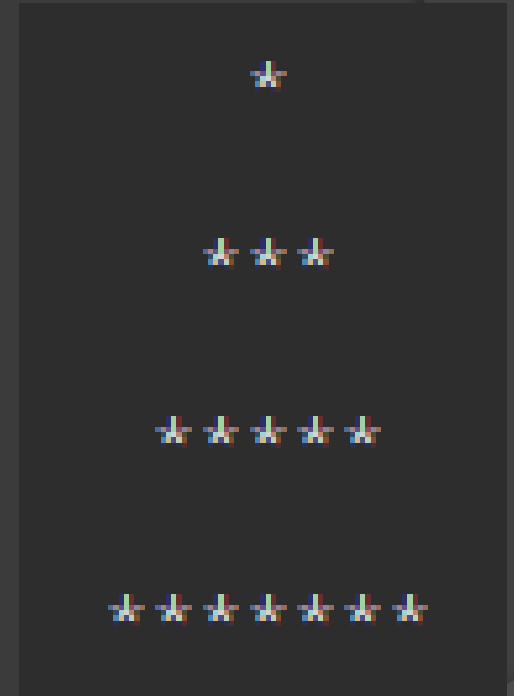
★ ★ ★

★ ★ ★ ★

★ ★ ★ ★ ★

# 巢狀迴圈

Ex. 列印



# 巢狀迴圈

Ex. 輸入行數，列印右圖

請輸入印列行數：4

★

★ ★ ★

★ ★ ★ ★ ★

★ ★ ★ ★ ★ ★ ★

# 巢狀迴圈

Ex. 輸入行數，列印右圖

請輸入印列行數：4

★

★ ★

★ ★ ★

★ ★ ★ ★