



8.1 turtle库基本介绍



turtle库概述

turtle(海龟)库是turtle绘图体系的Python实现

- **turtle绘图体系：1969年诞生，主要用于程序设计入门**
- **Python语言的**标准库**之一**
- **入门级的图形绘制函数库**



标准库

Python计算生态 = 标准库 + 第三方库

- **标准库：随解释器直接安装到操作系统中的功能模块**
- **第三方库：需要经过安装才能使用的功能模块**
- **库Library、包Package、模块Module，统称模块**

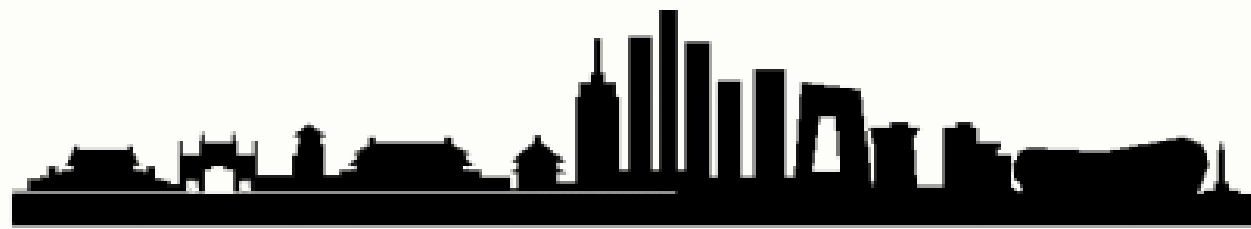
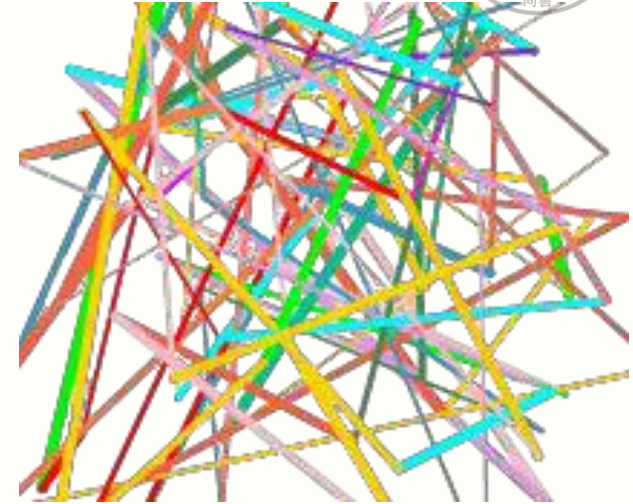
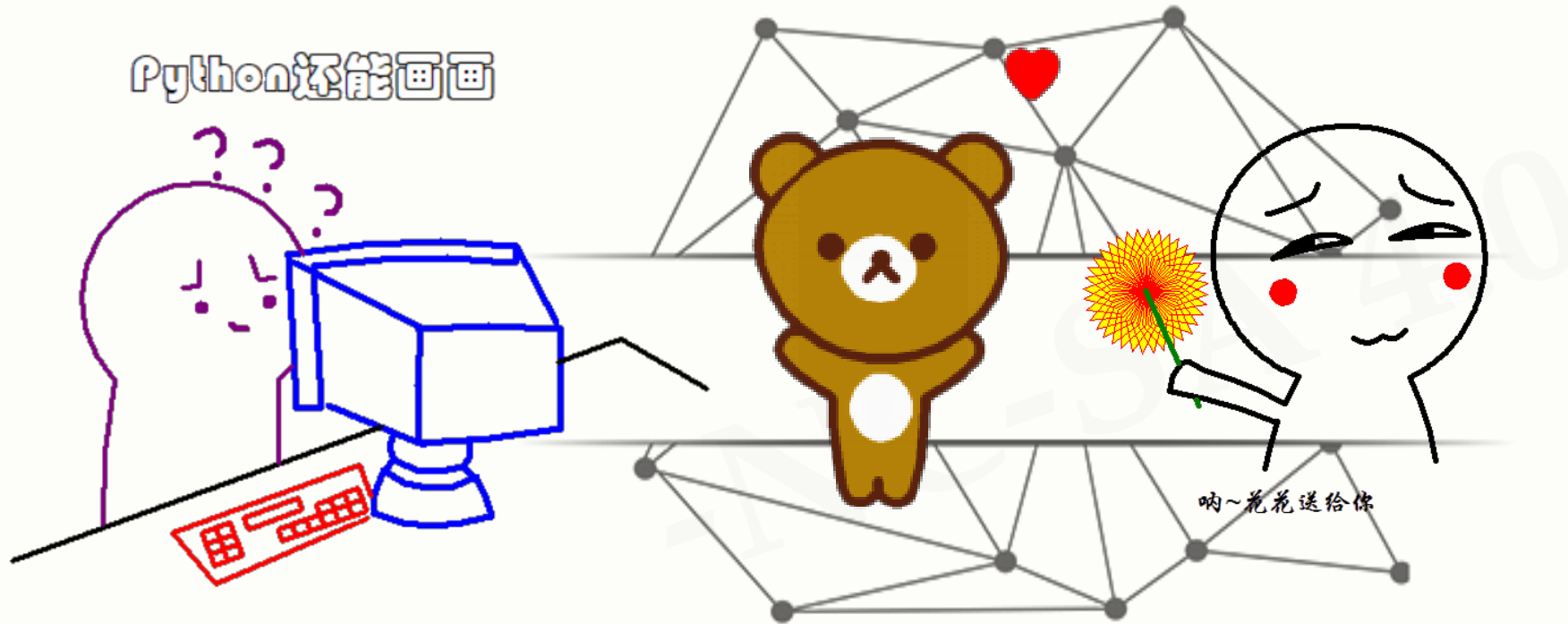


turtle的原 (wan) 理 (fa)

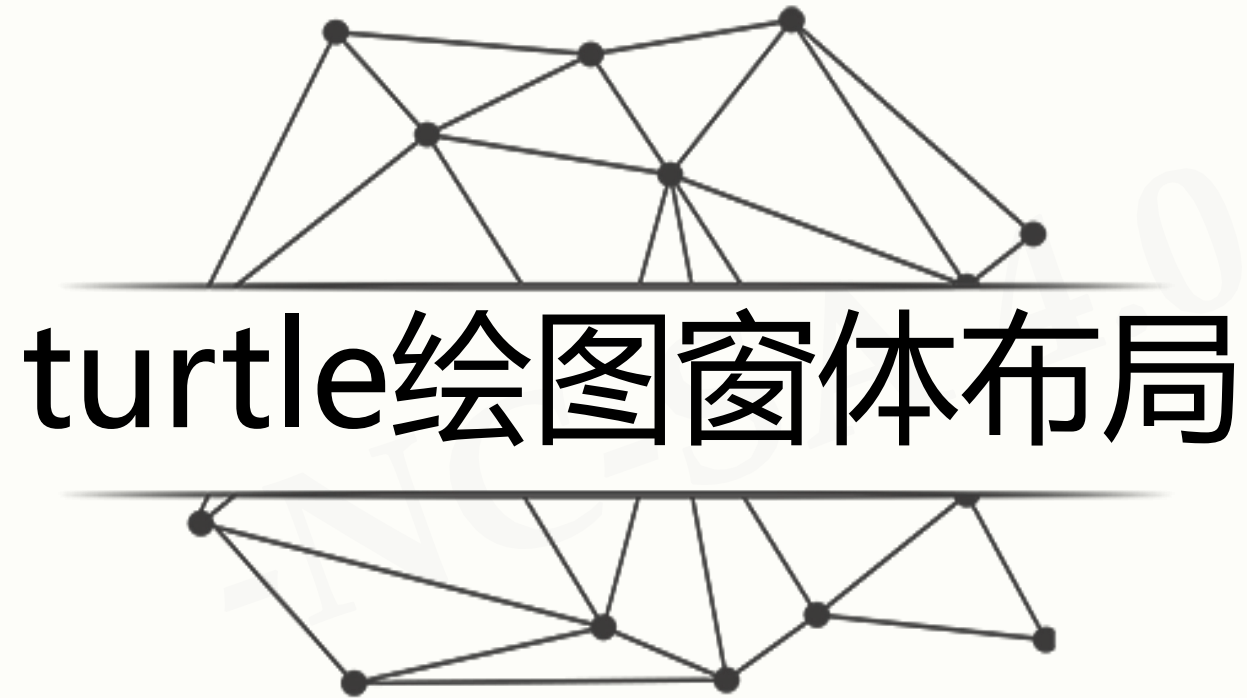
turtle(海龟)是一种真实的存在

- **有一只海龟，其实在窗体正中心，在画布上游走**
- **走过的轨迹形成了绘制的图形**
- **海龟由程序控制，可以变换颜色、改变宽度等**

turtle的魅力



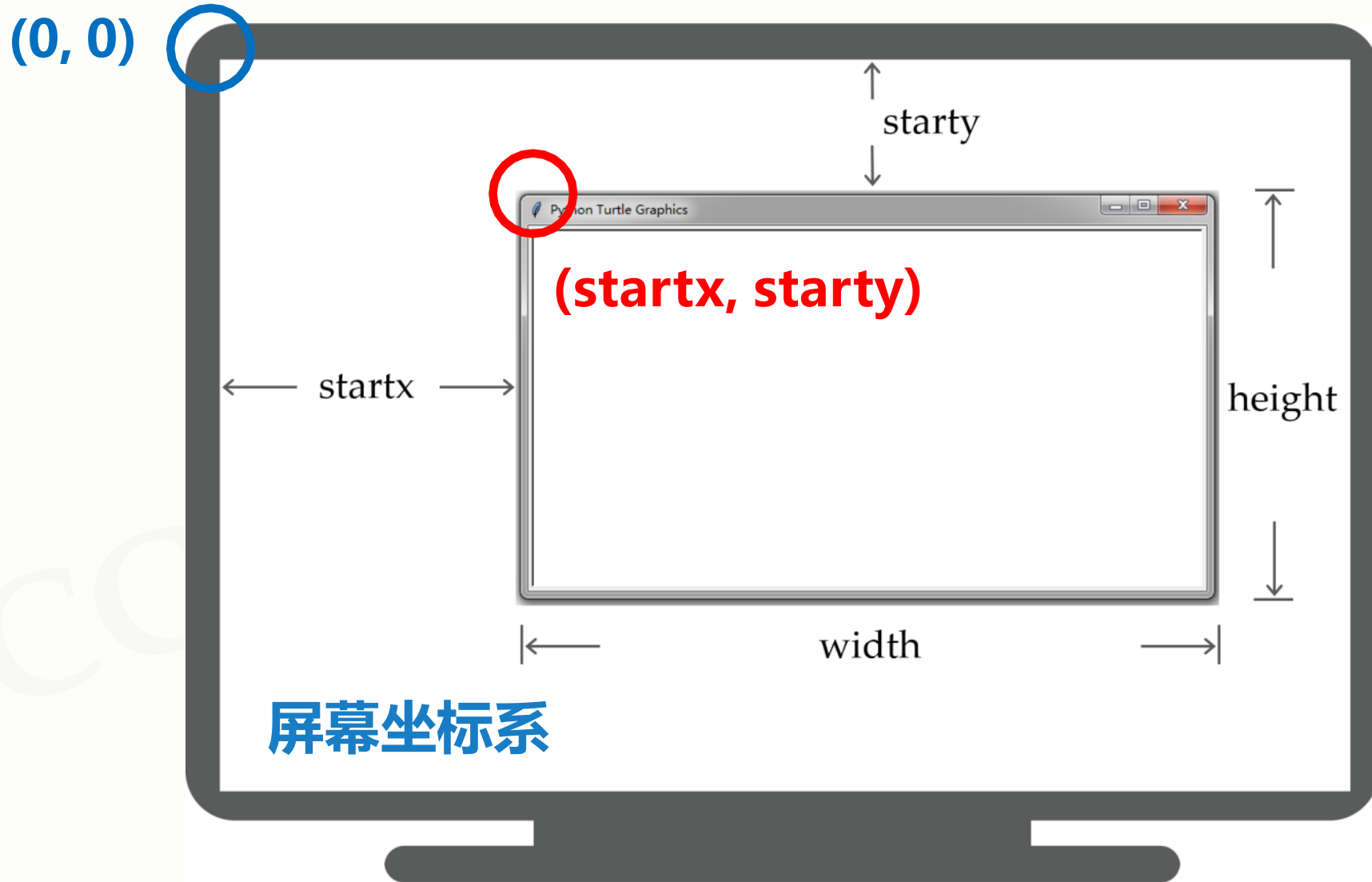
Beijing



turtle的绘图窗体

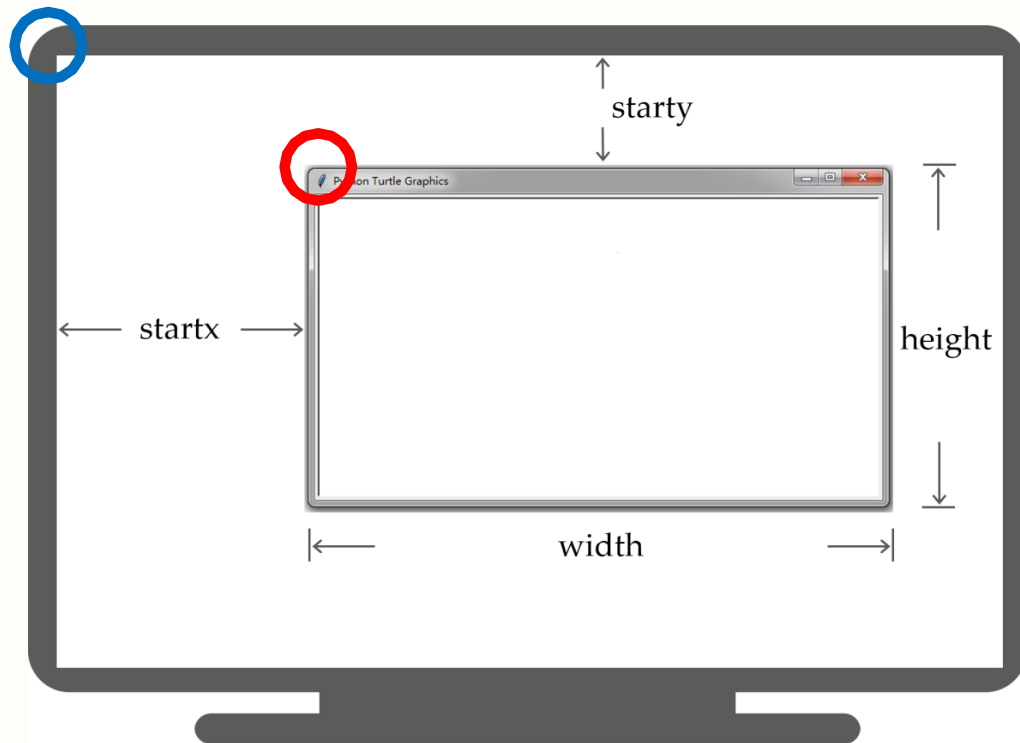


turtle的绘图窗体



turtle的绘图窗体

`turtle.setup(width, height, startx, starty)`



- **setup()设置窗体大小及位置**
- **4个参数中后两个可选**
- **setup()不是必须的**

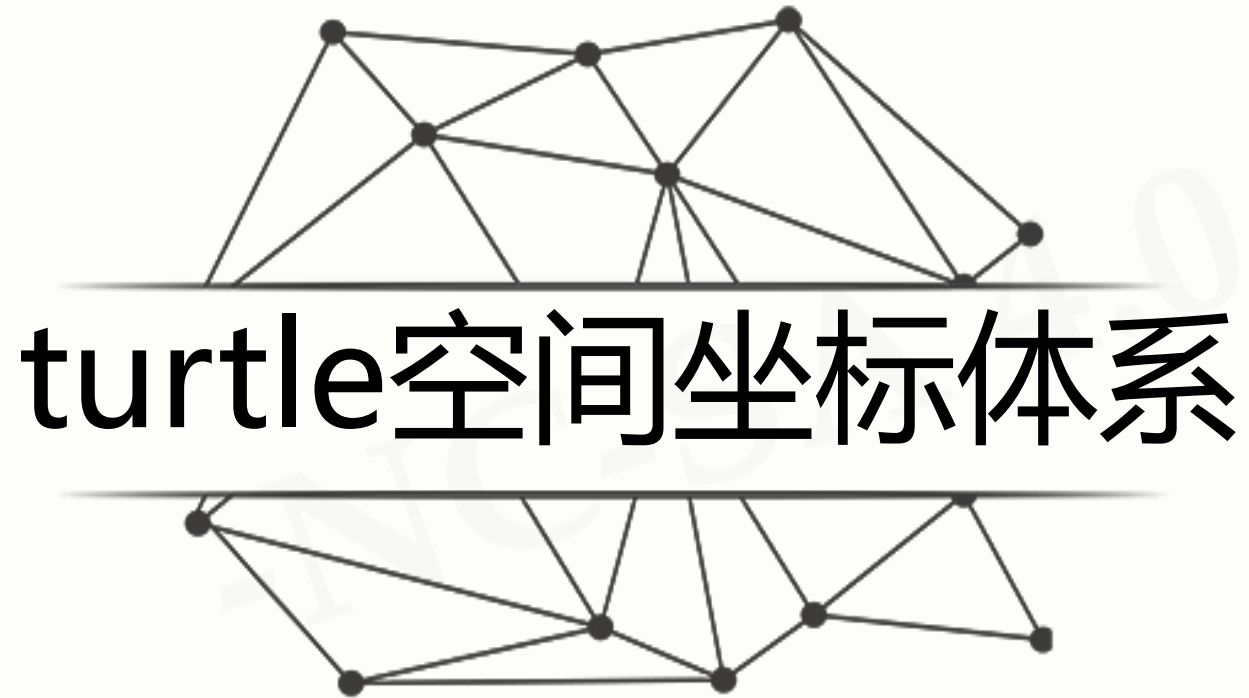
turtle的绘图窗体

`turtle.setup(800,800,0,0)`

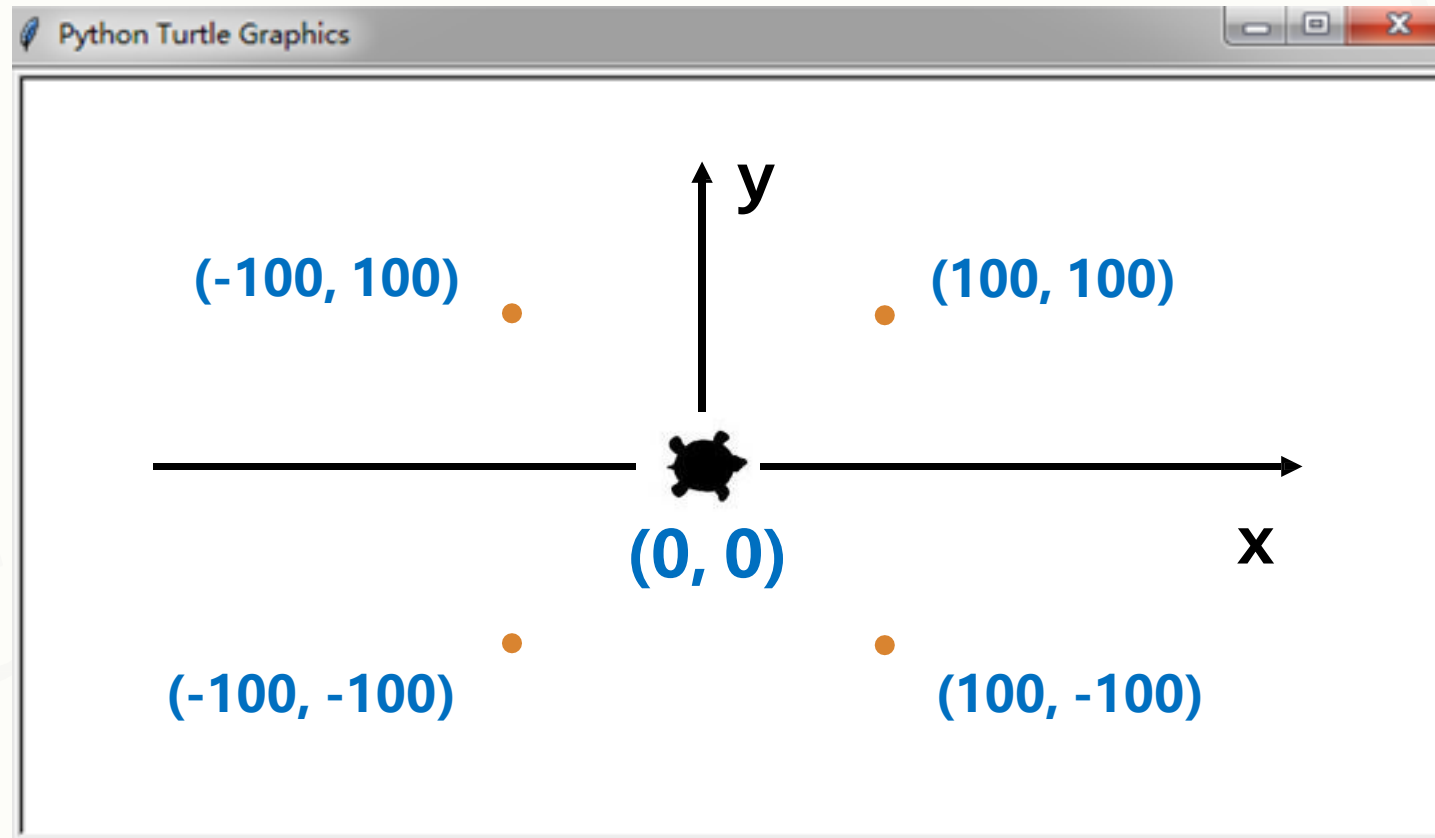


`turtle.setup(800,800)`

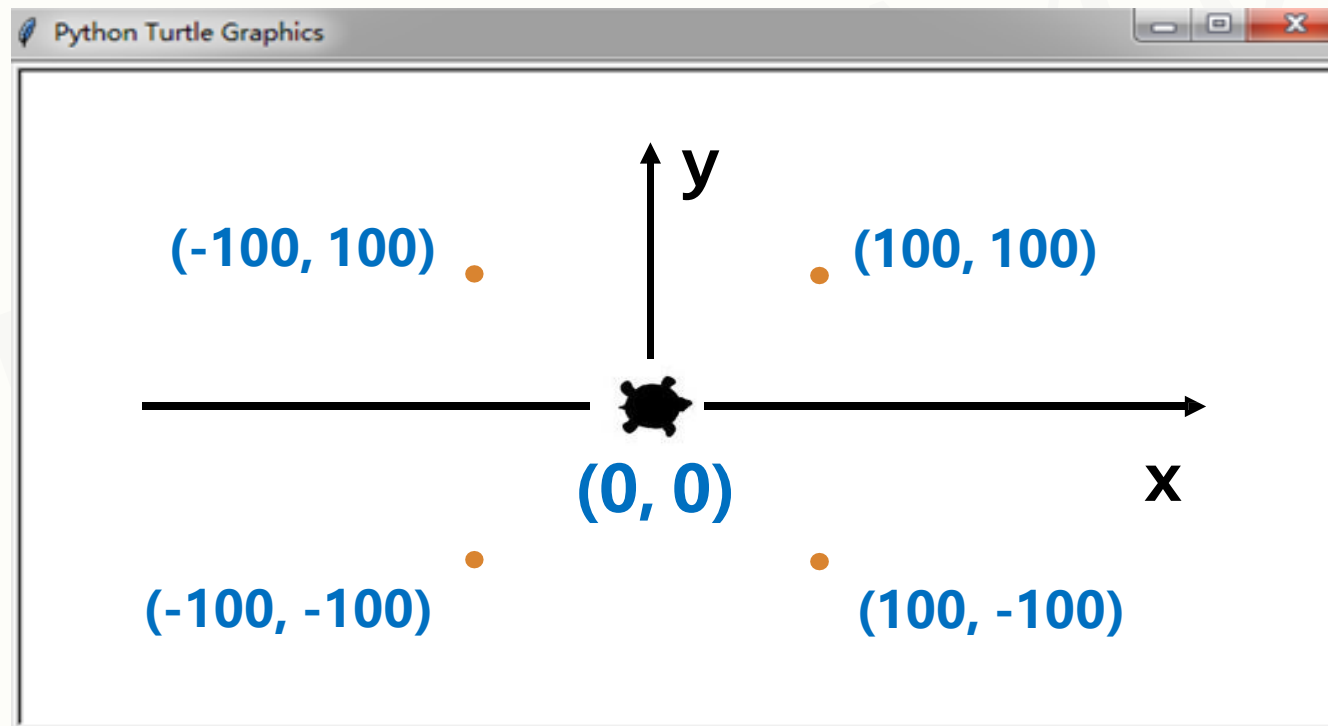




绝对坐标



`turtle.goto(x, y)`



```
import turtle
```

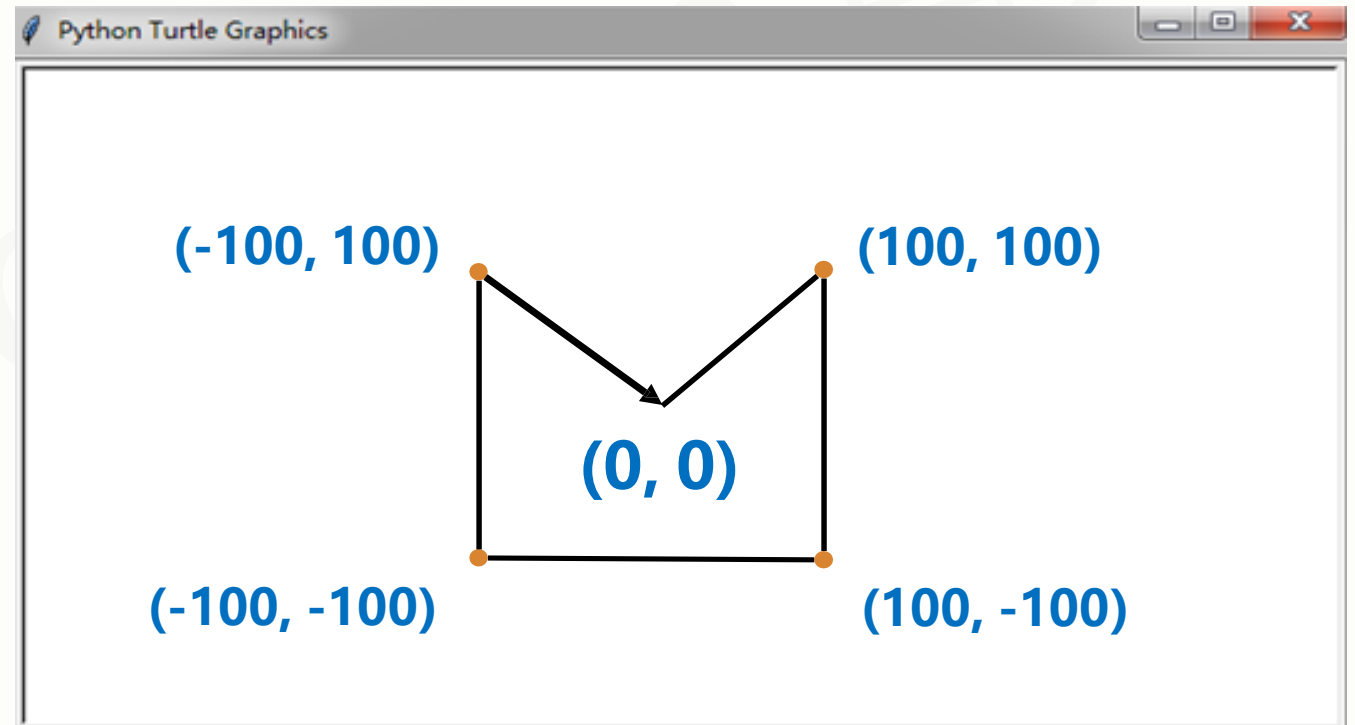
```
turtle.goto( 100, 100)
```

```
turtle.goto( 100,-100)
```

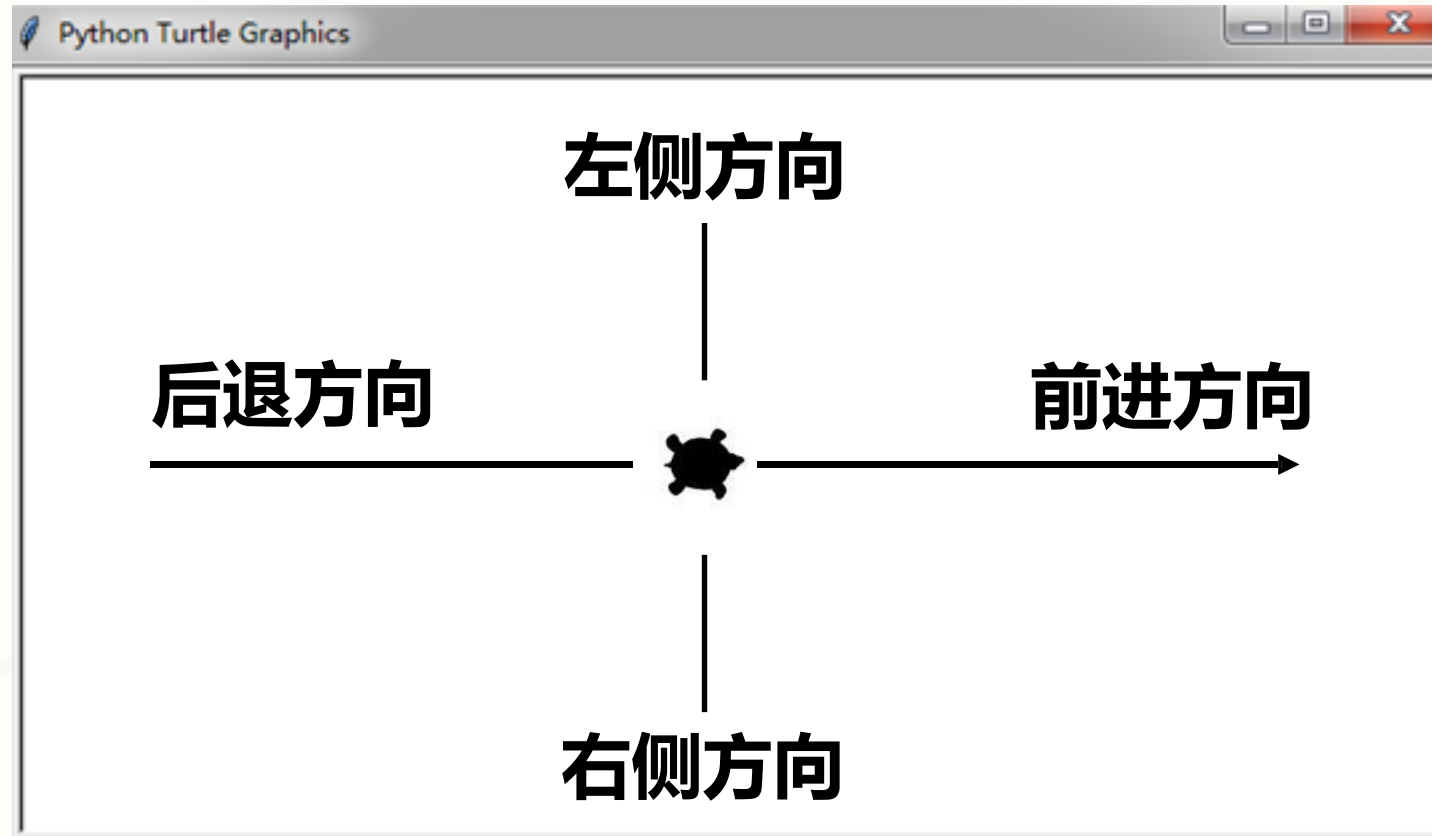
```
turtle.goto(-100,-100)
```

```
turtle.goto(-100, 100)
```

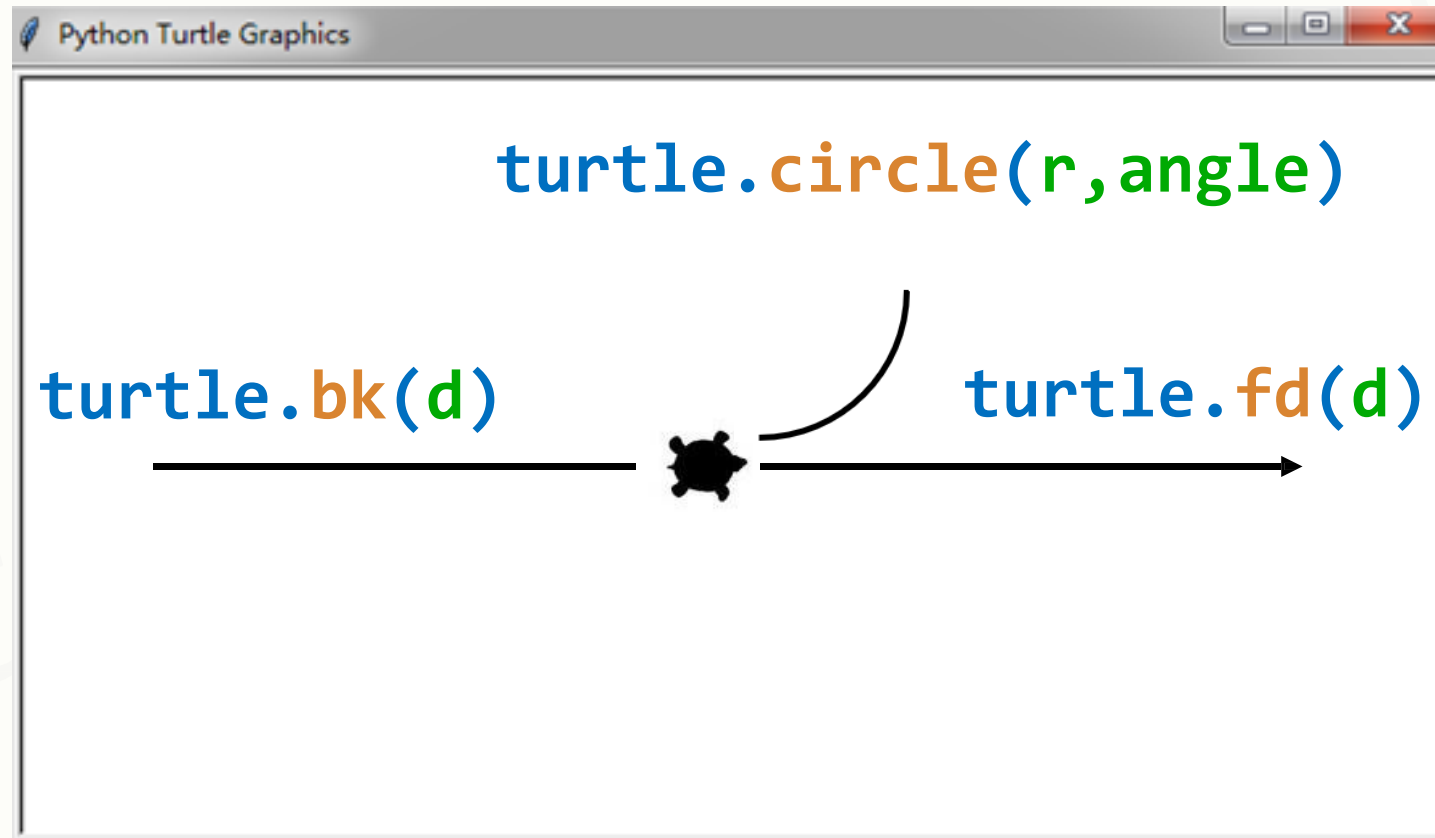
```
turtle.goto(0,0)
```

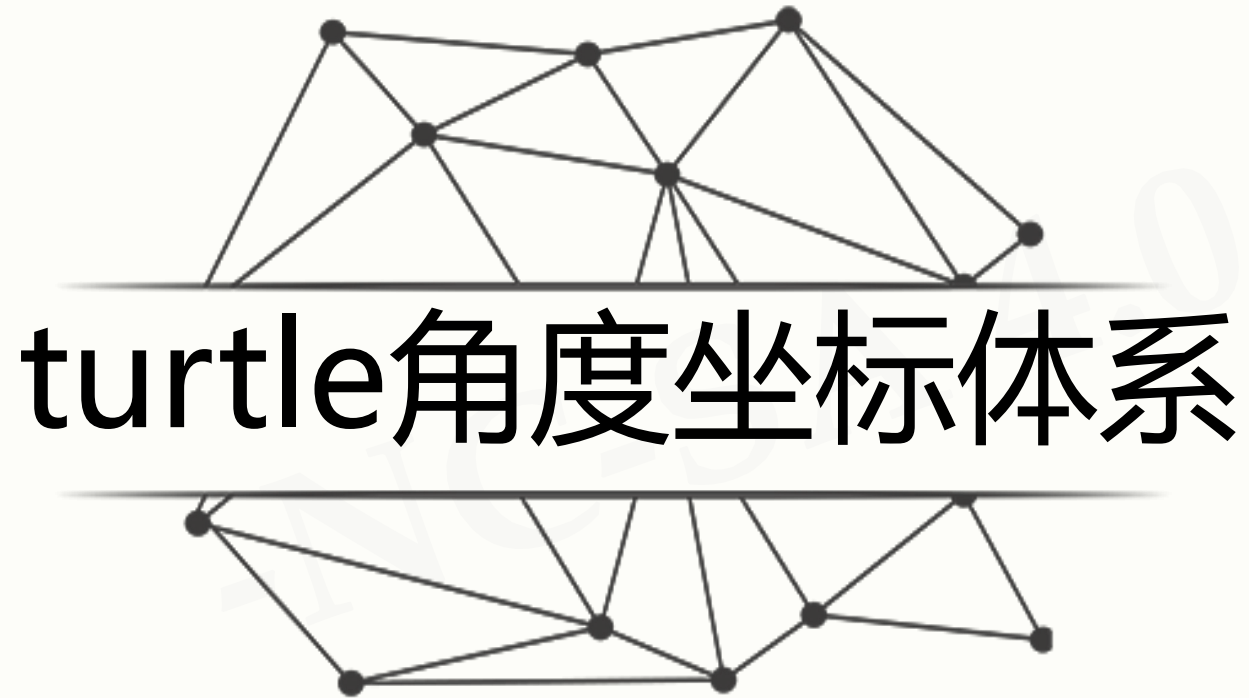


海龟坐标



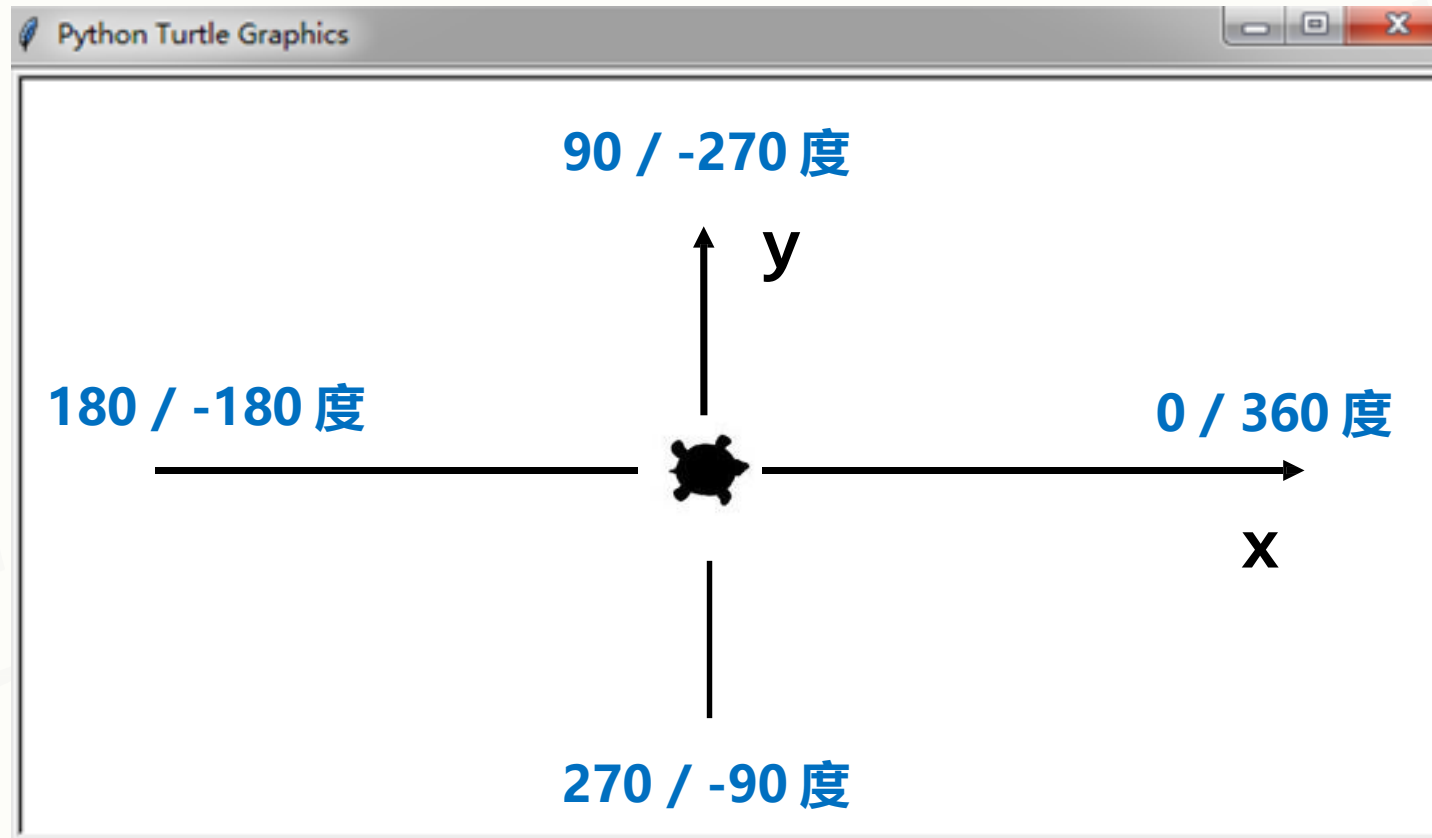
PY01B08 自然





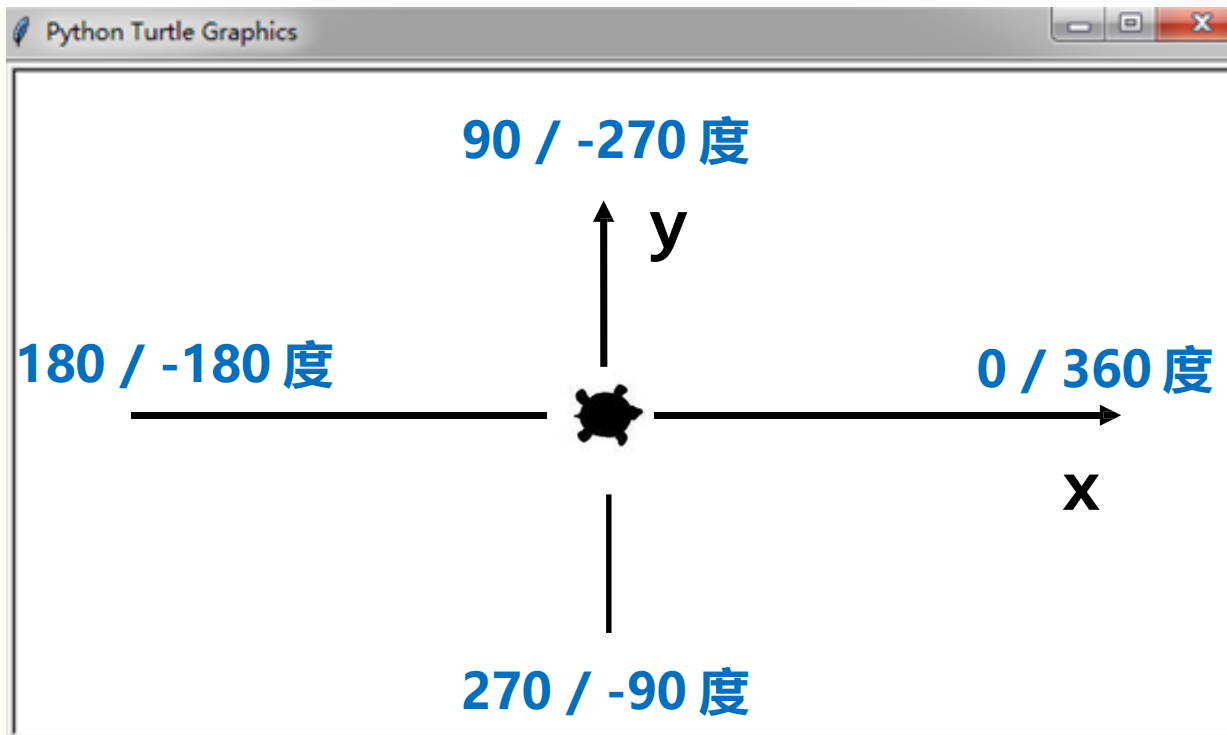
turtle角度坐标体系

绝对角度



turtle角度坐标体系

`turtle.seth(angle)`



- **seth()**改变海龟行进方向
- **angle**为绝对度数
- **seth()**只改变方向但不行进

`turtle.seth(45)`



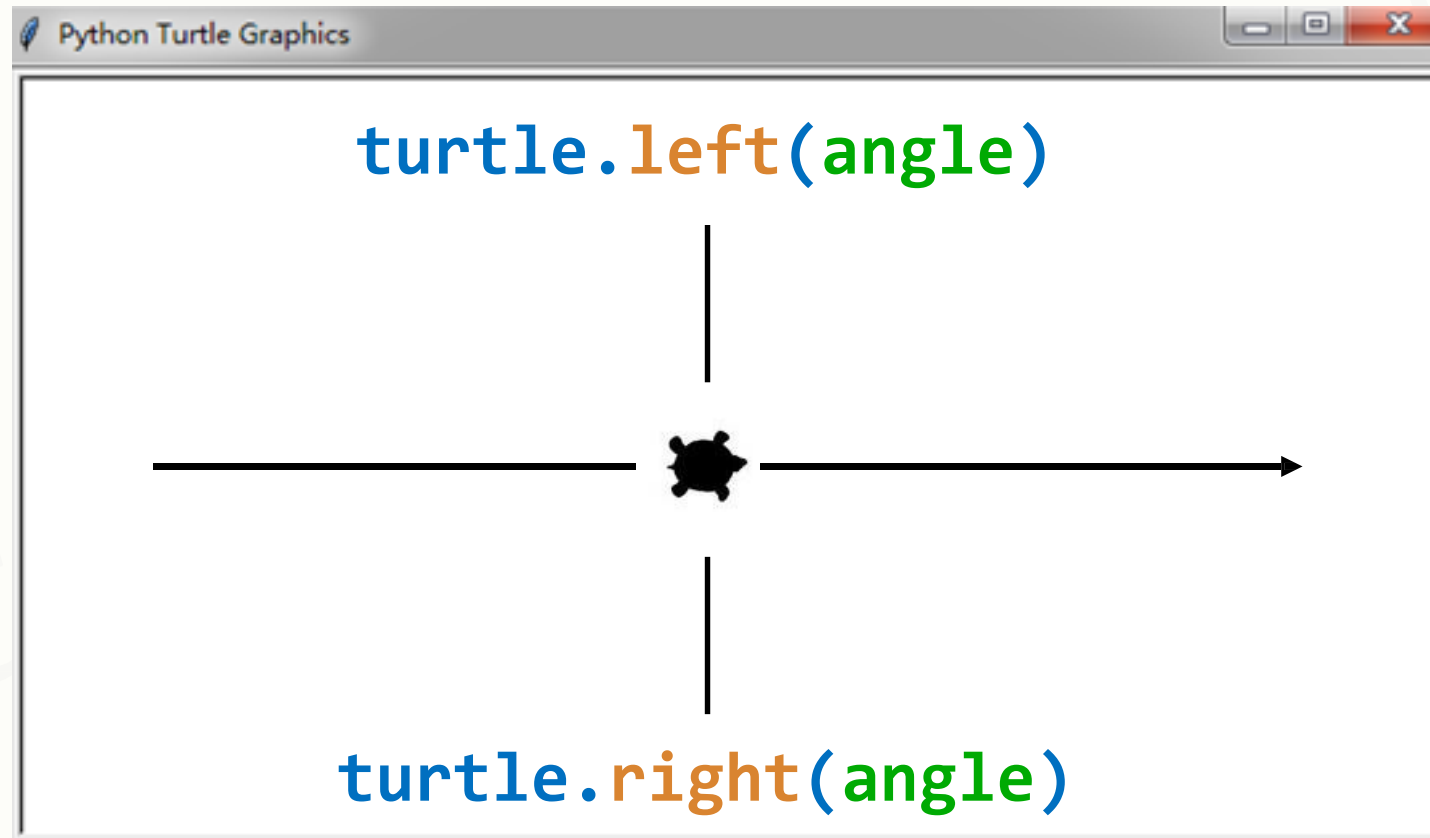
`turtle.seth(-135)`

t
三
一



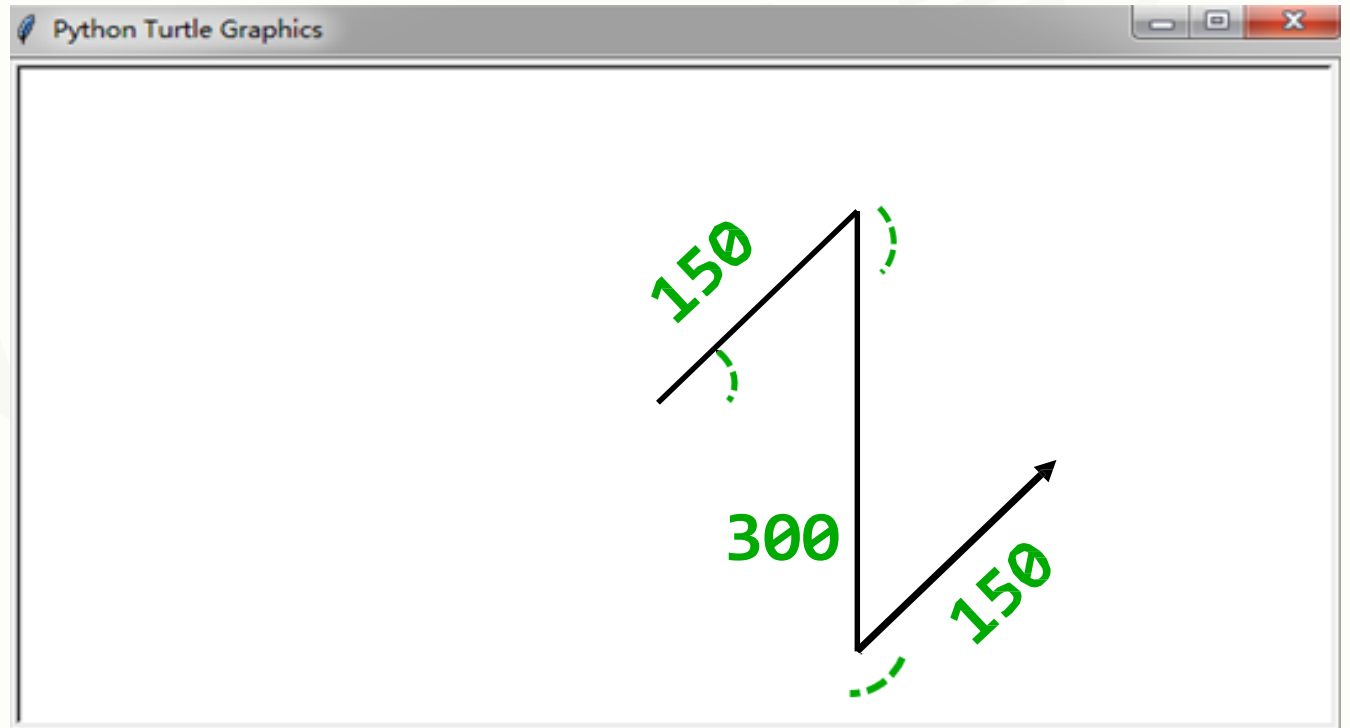
Turtle角度坐标体系

海龟角度



Turtle角度坐标体系

```
import turtle  
turtle.left(45)  
turtle.fd(150)  
turtle.right(135)  
turtle.fd(300)  
turtle.left(135)  
turtle.fd(150)
```

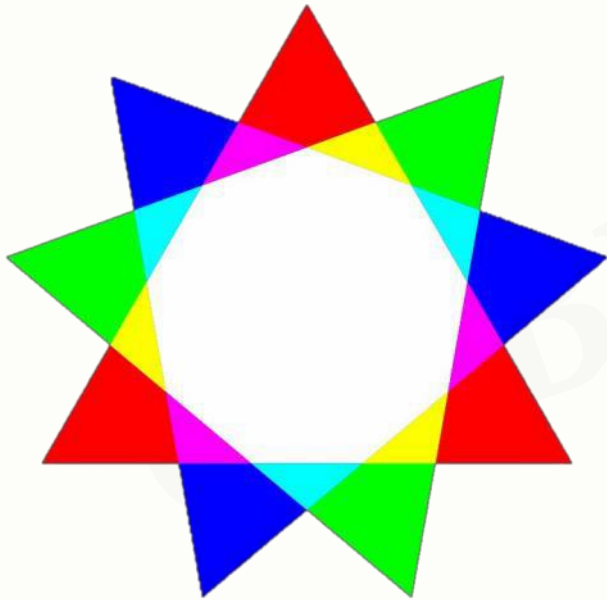




RGB色彩体系

RGB色彩模式

由三种颜色构成的万物色



- RGB指红蓝绿三个通道的颜色组合
- 覆盖视力所能感知的所有颜色
- RGB每色取值范围0-255整数或0-1小数



常用RGB色彩

英文名称	RGB整数值	RGB小数值	中文名称
white	255, 255, 255	1, 1, 1	白色
yellow	255, 255, 0	1, 1, 0	黄色
magenta	255, 0, 255	1, 0, 1	洋红
cyan	0, 255, 255	0, 1, 1	青色
blue	0, 0, 255	0, 0, 1	蓝色
black	0, 0, 0	0, 0, 0	黑色



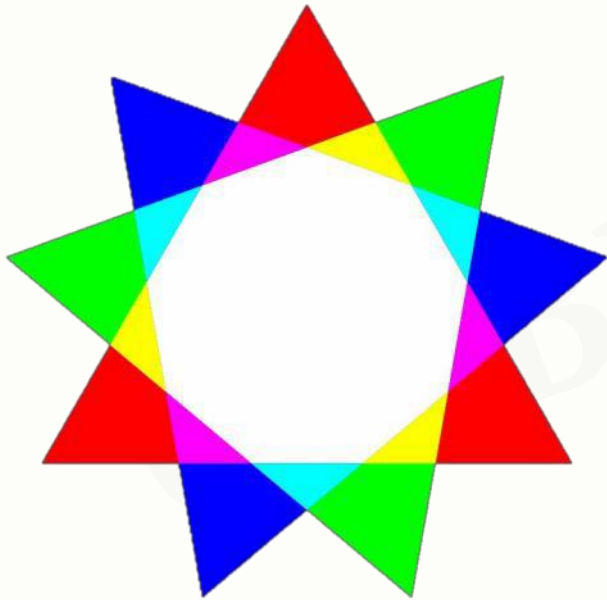
常用RGB色彩

英文名称	RGB整数值	RGB小数值	中文名称
seashell	255, 245, 238	1, 0.96, 0.93	海贝色
gold	255, 215, 0	1, 0.84, 0	金色
pink	255, 192, 203	1, 0.75, 0.80	粉红色
brown	165, 42, 42	0.65, 0.16, 0.16	棕色
purple	160, 32, 240	0.63, 0.13, 0.94	紫色
tomato	255, 99, 71	1, 0.39, 0.28	番茄色

turtle的RGB色彩模式

默认采用小数值 可切换为整数值

`turtle.colormode(mode)`



- **1.0**: RGB小数值模式
- **255**: RGB整数值模式

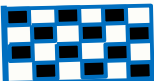


单元小结

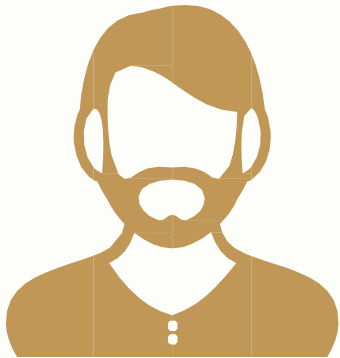


turtle库的使用

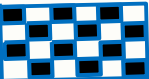
- **turtle库的海龟绘图法**
- **turtle.setup()**调整绘图窗体在电脑屏幕中的布局
- **画布上以中心为原点的空间坐标系: 绝对坐标&海龟坐标**
- **画布上以空间x轴为0度的角度坐标系: 绝对角度&海龟角度**
- **RGB色彩体系, 整数值&小数值, 色彩模式切换**



turtle程序语法元素分析



- 库引用与import
- turtle画笔控制函数
- turtle运动控制函数
- turtle方向控制函数
- 基本循环语句
- "Python蟒蛇绘制"代码分析





turtle画笔控制函数



```
import turtle
turtle.setup(650, 350, 200, 200)
turtle.penup()
turtle.fd(-250)
turtle.pendown()
turtle.pensize(25)
turtle.pencolor("purple")
turtle.seth(-40)
for i in range(4):
    turtle.circle(40, 80)
    turtle.circle(-40, 80)
turtle.circle(40, 80/2)
turtle.fd(40)
turtle.circle(16, 180)
turtle.fd(40 * 2/3)
turtle.done()
```

penup(), pendown()

pensize(), pencolor()



PY01B09 无识无是





画笔控制函数

画笔操作后一直有效，一般成对出现

- **turtle.penup()** 别名 **turtle.pu()**

抬起画笔，海龟在飞行

- **turtle.pendown()** 别名 **turtle.pd()**

落下画笔，海龟在爬行



画笔控制函数

画笔设置后一直有效，直至下次重新设置

- `turtle.pensize(width)` 别名 `turtle.width(width)`

画笔宽度，海龟的腰围

- `turtle.pencolor(color)` `color`为颜色字符串或r,g,b值

画笔颜色，海龟在涂装



画笔控制函数

pencolor(color)的color参与可以有三种形式

- **颜色字符串** : `turtle.pencolor("purple")`
- **RGB的小数值**: `turtle.pencolor(0.63, 0.13, 0.94)`
- **RGB的元组值**: `turtle.pencolor((0.63,0.13,0.94))`



```
import turtle
turtle.setup(650, 350, 200, 200)
turtle.penup()
turtle.fd(-250)
turtle.pendown()
turtle.pensize(25)
turtle.pencolor("purple")
turtle.seth(-40)
for i in range(4):
    turtle.circle(40, 80)
    turtle.circle(-40, 80)
turtle.circle(40, 80/2)
turtle.fd(40)
turtle.circle(16, 180)
turtle.fd(40 * 2/3)
turtle.done()
```

penup()

pendown()

pensize(width)

pencolor(colorstring)

pencolor(r,g,b)

pencolor((r,g,b))





turtle运动控制函数





```
import turtle
turtle.setup(650, 350, 200, 200)
turtle.penup()
turtle.fd(-250)
turtle.pendown()
turtle.pensize(25)
turtle.pencolor("purple")
turtle.seth(-40)
for i in range(4):
    turtle.circle(40, 80)
    turtle.circle(-40, 80)
turtle.circle(40, 80/2)
turtle.fd(40)
turtle.circle(16, 180)
turtle.fd(40 * 2/3)
turtle.done()
```

fd()

circle()



运动控制函数

控制海龟行进：走直线 & 走曲线

- `turtle.forward(d)` 别名 `turtle.fd(d)`

向前行进，海龟走直线

- `d`: 行进距离，可以为负数



运动控制函数

控制海龟行进：走直线 & 走曲线

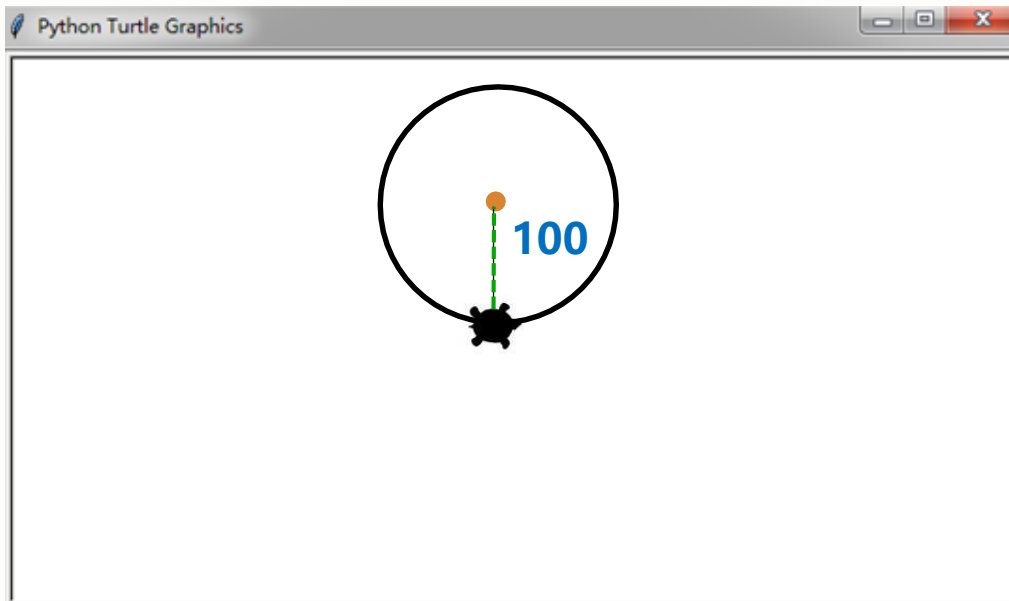
- `turtle.circle(r, extent=None)`

根据半径`r`绘制`extent`角度的弧形

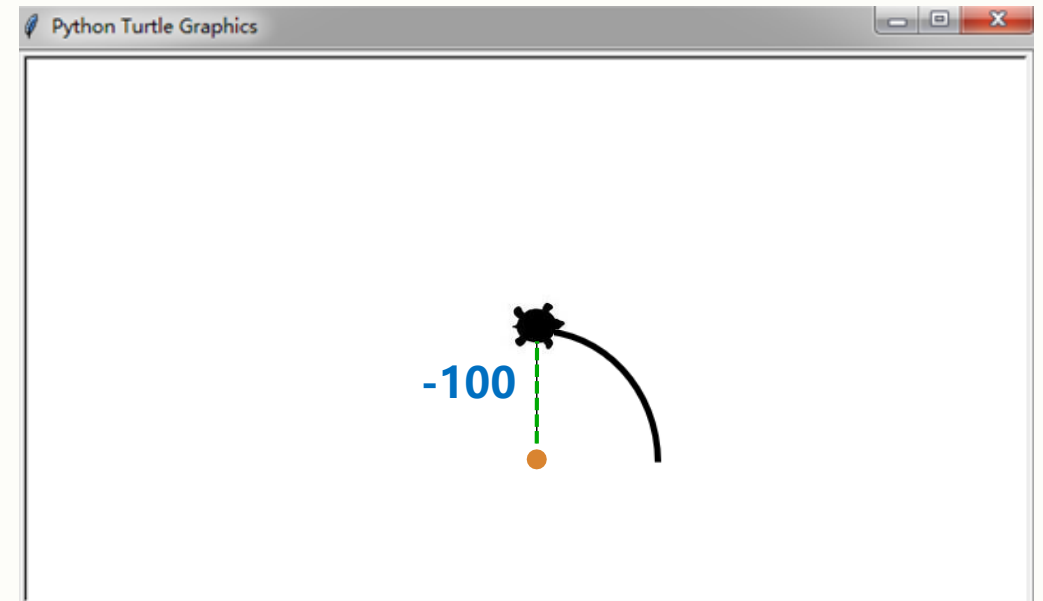
- `r`: 默认圆心在海龟左侧`r`距离的位置
- `extent`: 绘制角度，默认是360度整圆

运动控制函数

`turtle.circle(100)`



`turtle.circle(-100,90)`





```
import turtle
turtle.setup(650, 350, 200, 200)
turtle.penup()
turtle.fd(-250)
turtle.pendown()
turtle.pensize(25)
turtle.pencolor("purple")
turtle.seth(-40)
for i in range(4):
    turtle.circle(40, 80)
    turtle.circle(-40, 80)
turtle.circle(40, 80/2)
turtle.fd(40)
turtle.circle(16, 180)
turtle.fd(40 * 2/3)
turtle.done()
```

fd(d)

circle(r, extent=None)



运动控制函数

画笔设置后一直有效，直至下次重新设置

- `turtle.forward(d)` 别名 `turtle.fd(d)`

向前行进，海龟走直线

- `d`: 行进距离，可以为负数



turtle方向控制函数



```
import turtle
turtle.setup(650, 350, 200, 200)
turtle.penup()
turtle.fd(-250)
turtle.pendown()
turtle.pensize(25)
turtle.pencolor("purple")
turtle.seth(-40)
for i in range(4):
    turtle.circle(40, 80)
    turtle.circle(-40, 80)
turtle.circle(40, 80/2)
turtle.fd(40)
turtle.circle(16, 180)
turtle.fd(40 * 2/3)
turtle.done()
```

seth()



方向控制函数

控制海龟面对方向: 绝对角度 & 海龟角度

- `turtle.setheading(angle)` 别名 `turtle.seth(angle)`

改变行进方向，海龟走角度

- **angle**: 行进方向的绝对角度

方向控制函数

`turtle.seth(45)`



`turtle.seth(-135)`





方向控制函数

控制海龟面对方向: 绝对角度 & 海龟角度

- `turtle.left(angle)` 海龟向左转
- `turtle.right(angle)` 海龟向右转
- **angle**: 在海龟当前行进方向上旋转的角度



```
import turtle
turtle.setup(650, 350, 200, 200)
turtle.penup()
turtle.fd(-250)
turtle.pendown()
turtle.pensize(25)
turtle.pencolor("purple")
turtle.seth(-40)
for i in range(4):
    turtle.circle(40, 80)
    turtle.circle(-40, 80)
turtle.circle(40, 80/2)
turtle.fd(40)
turtle.circle(16, 180)
turtle.fd(40 * 2/3)
turtle.done()
```

seth(angle)



PY01B10 不受物性