

# 课前准备

- 1. 回顾第一讲内容
  - 1. 计算机组成
  - 2. 自然语言、编程语言
  - 3. 启动 python shell, 提示符 >>>
  - 4. print语句
  - 5. 文件：文件名、扩展名、路径、斜杠
- 2. 打字游戏
- 3. 关键词汇复习

computer	screen
keyboard	mouse
download	install
input	print

# 第2讲 数学运算与乌龟作图

## 2.1 基本数学运算

### 运算符

数学运算符	Python 运算符	描述
+	+	加法
-	-	减法
×	*	乘法
÷	/	除法
	%	求余
()	()	小括号--改变计算顺序

## 编程计算

- \* 参考计算小超市题目在python shell交互提示符 >>> 输入算式，验证python输出计算结构
- \* 参考+ - \* /混合算式
- \* 参考带括号的混合算式
- \* 求余数

结论：计算规则跟数学课堂规则一致

## 计算优先级口诀

先乘除、后加减、小括号里最优先

## 测试计算能力

编码时间

```
# 乘法口诀
5*6
8*8
81/9

# 混合运算
81 / (12 - 3)
5 + 4 * 5

# 求余数
81 % 10
9 % 4
17 % 5

# 大数计算

999*999
999*999*999*999
999*999*999*999*999*999*999*999
999*999*999*999*999*999*999*999*999*999*999*999*999

# 启动Python Shell
# 在交互提示符>>>后输入上述算式
```

```
Python 3.6.8 Shell
File Edit Shell Debug Options Window Help
Python 3.6.8 (tags/v3.6.8:3c6b436a57, Dec 24 2018, 00:16:47) [MSC v.1916 64 bit
(AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 999*999
998001
>>> 999*999*999*999
996005996001
>>> 999*999*999*999*999*999*999*999
992027944069944027992001
>>> 999*999*999*999*999*999*999*999*999*999*999*999
988065780494208923208494780065988001
>>>
```

说说你的感想？

复杂的算式、大数计算，计算机比人更擅长，比人更快

## 简易计算器

```
a = int(input("请输入a=")) #输入a值
b = int(input("请输入a=")) #输入b值
print("a+b=", a+b)
print("a-b=", a-b)
print("a*b=", a*b)
print("a/b=", a/b) # 除法，带有小数
print("a%b=", a%b)
print("a//b=", a//b) # 整除
```

## 2.2 乌龟作图

### 巩固数学课上角

角的特点：有1个顶点、2条边

思考：角的大小跟2条边的长短有没有关系？

直角： $90^{\circ}$

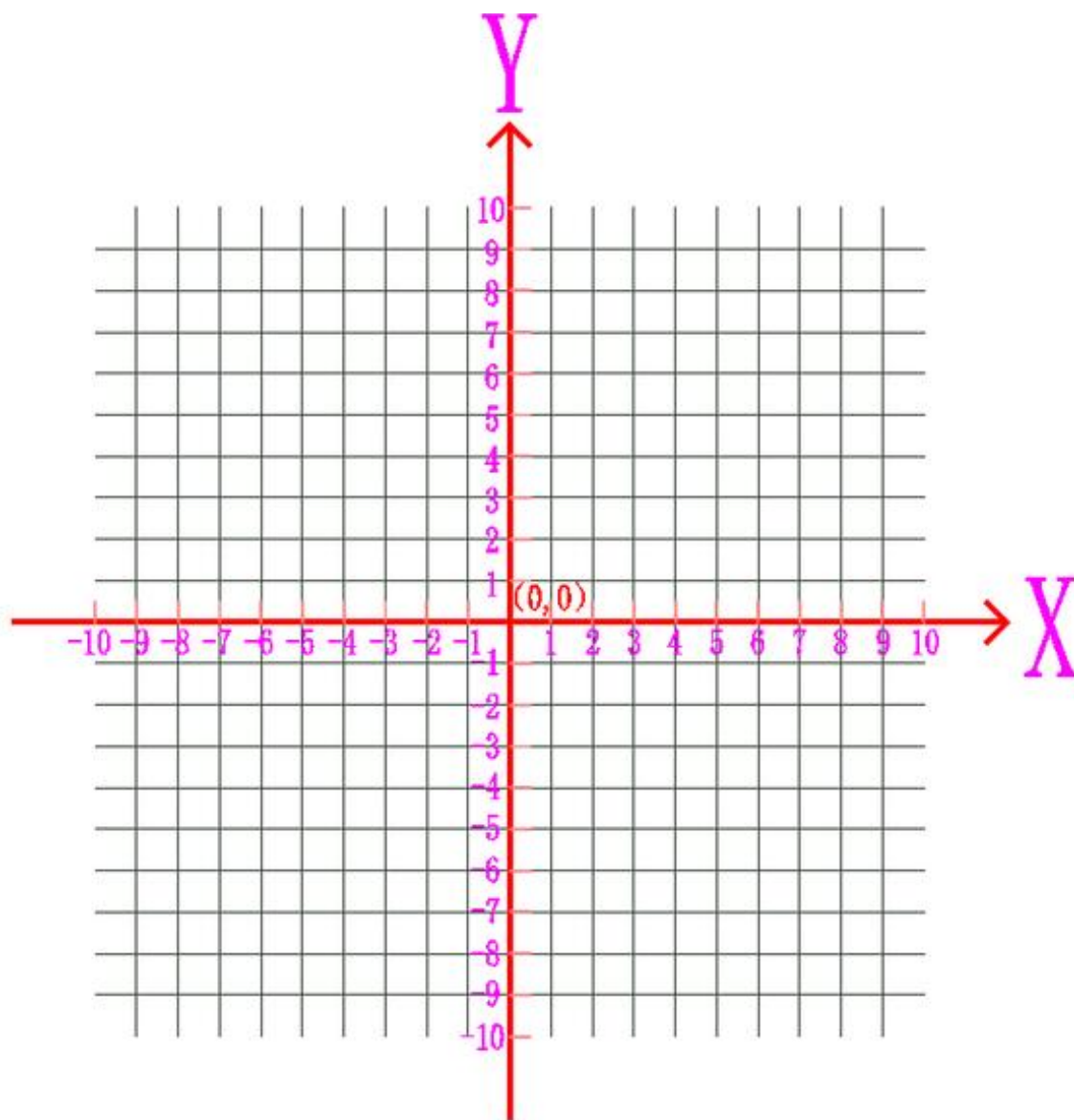
锐角： $0 < \text{锐角} < 90^{\circ}$

钝角： $90^{\circ} < \text{钝角} < 180^{\circ}$

角的度数测量：量角器使用--量角器的中心位置和角的顶点重合；角的一边和零刻度线重合，然后看角的另外一边落在量角器的哪个刻度之上，刻度是多少即该角的度数。

## 巩固坐标系知识

### 直角坐标系



两条线（X、Y）相互垂直，交叉点称为原点（0，0），即X值为0，Y值为0

通常取向右、向上（箭头方向）为正方向，即X方向，Y方向；往箭头的方向，逐渐增加，相反的方向逐渐减少

### 角度

一个直角坐标系总共有几个直角？

这四个直角加起来正好是一圈，一个直角是90度，那么一圈是多少度？**360** 所以一个人站在原地转一圈，就是转了360度

**方向：**像scratch学习，小猫移动时需要朝一个方向移动，那坐标系也需要指明方向

**约定:** X箭头（向右）方向为0，反方向180/-180；Y箭头方向为90，反方向270/-90

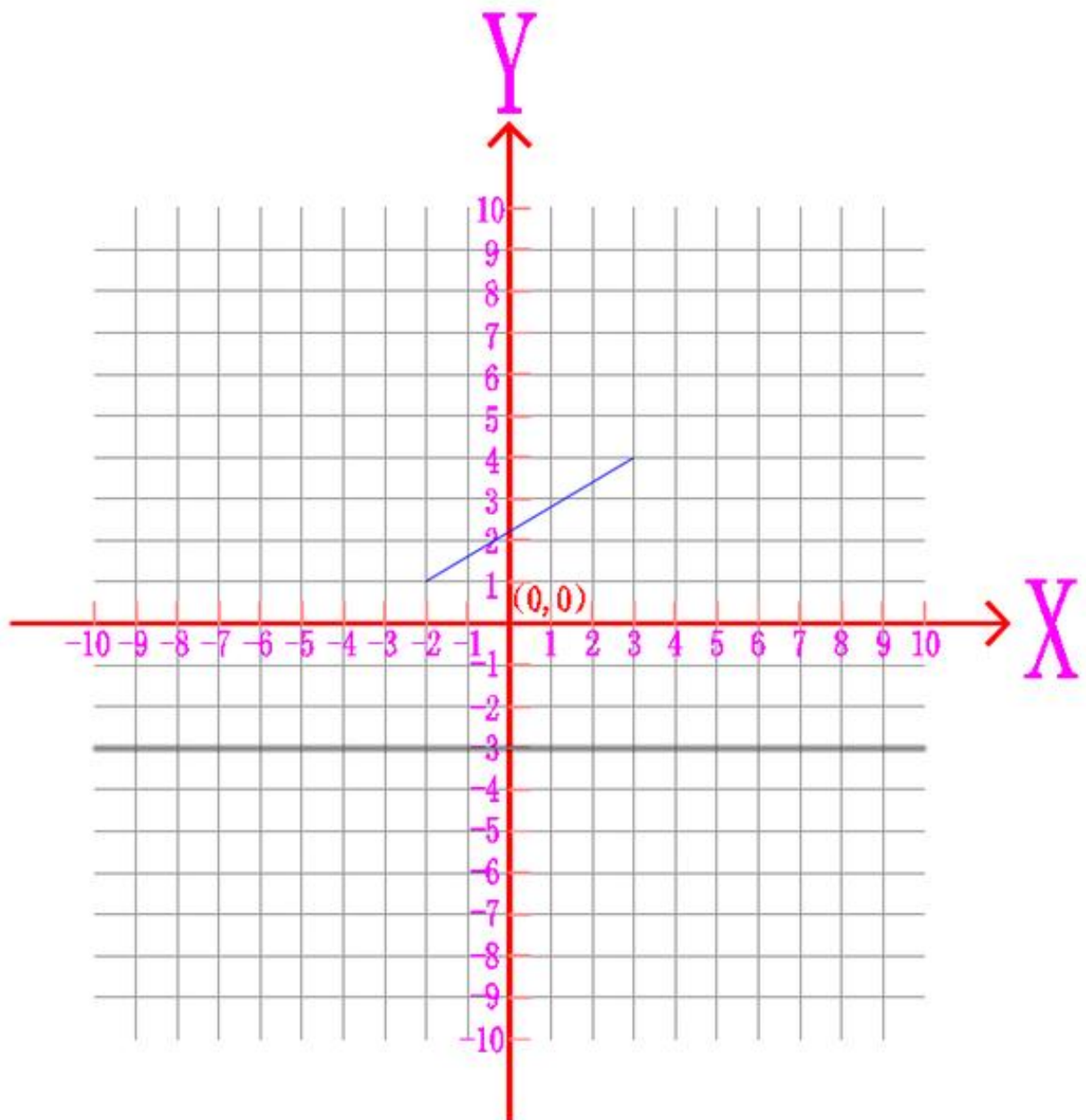
### 点的坐标

在直角坐标系中，有一个点，那么就有X, Y值，那么用  $(X, Y)$  就可以表示它的位置，举例说明

### 线段

线段有两个端点，点在坐标系中X, Y值，那么  $(X_1, Y_1)$  就可以表示一个点， $(X_2, Y_2)$  表示另一个点

比如从  $(-2, 1)$  到  $(3, 4)$  的线段，下图蓝色的线段



## 关键词汇

left	right
forward	back
left	right
up	down
turtle	shape

## 乌龟作图

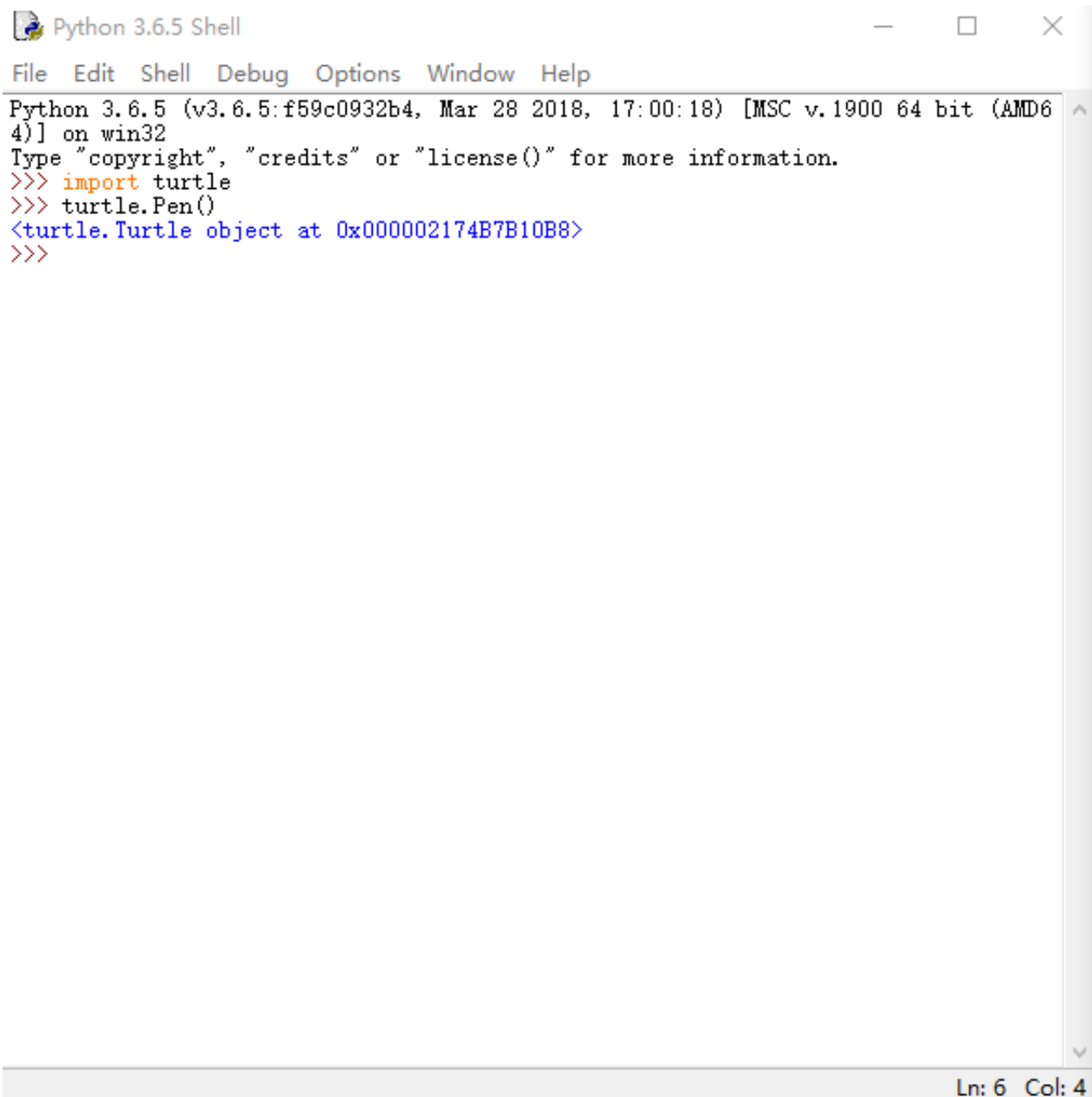
- 在python交互提示符 >>> 逐句执行，观察小乌龟的变化

### 画线

```
import turtle # 导入小乌龟
import turtle as tt

tt.Pen()
turtle.Pen() # 创建画笔
```

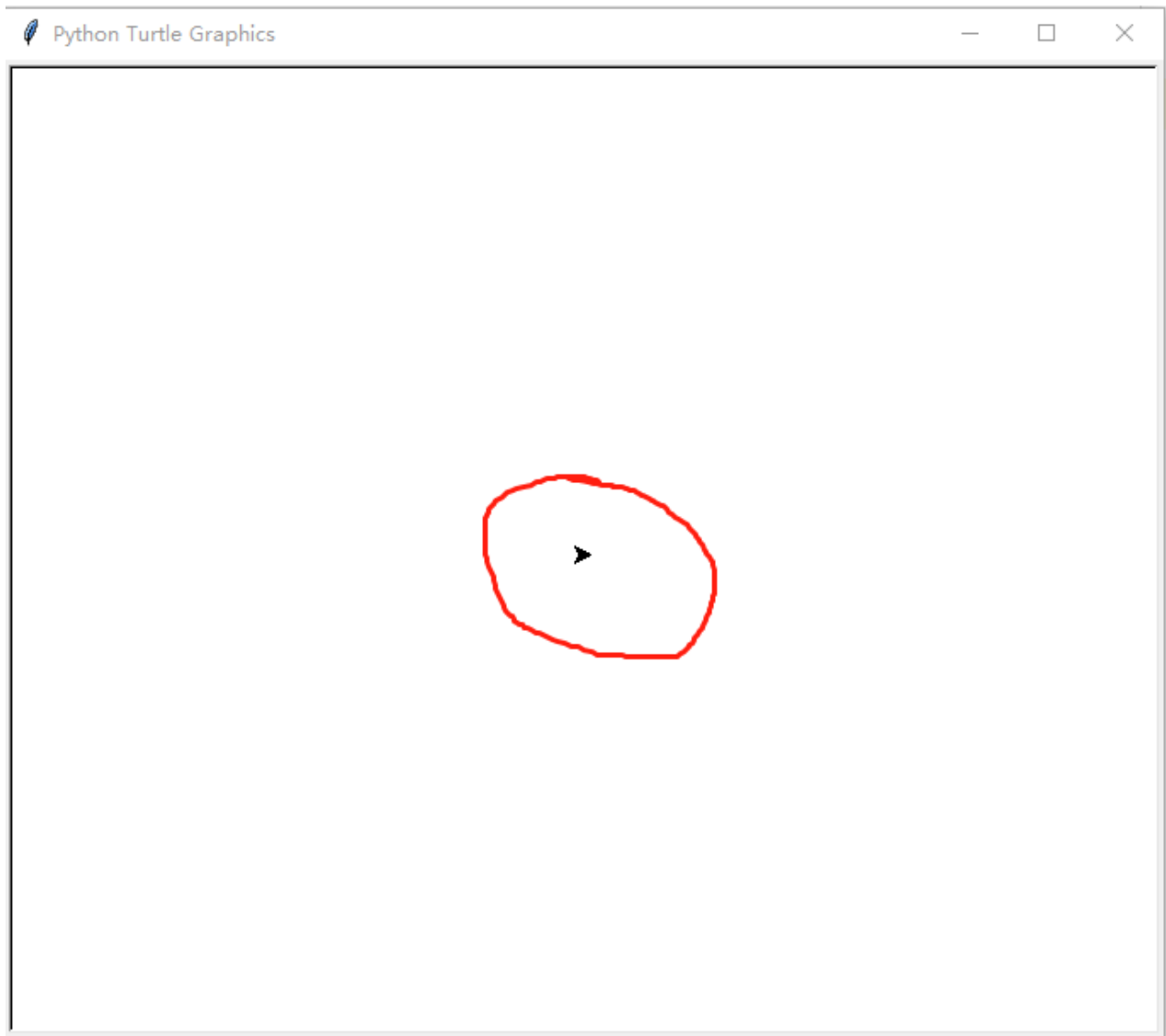
turtle 是python的一个库，里面包含了小乌龟各种操作（函数），比如移动，旋转，设置颜色，画笔粗细等等

A screenshot of a Python 3.6.5 Shell window. The window has a title bar with the text "Python 3.6.5 Shell" and standard window controls (minimize, maximize, close). Below the title bar is a menu bar with the following items: File, Edit, Shell, Debug, Options, Window, and Help. The main area of the window contains the following text:

```
Python 3.6.5 (v3.6.5:f59c0932b4, Mar 28 2018, 17:00:18) [MSC v.1900 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> import turtle
>>> turtle.Pen()
<turtle.Turtle object at 0x0000002174B7B10B8>
>>>
```

The text is color-coded: "import" is orange, "turtle.Pen()" is blue, and the object representation is blue. The status bar at the bottom right of the window shows "Ln: 6 Col: 4".

此时乌龟画图界面启动

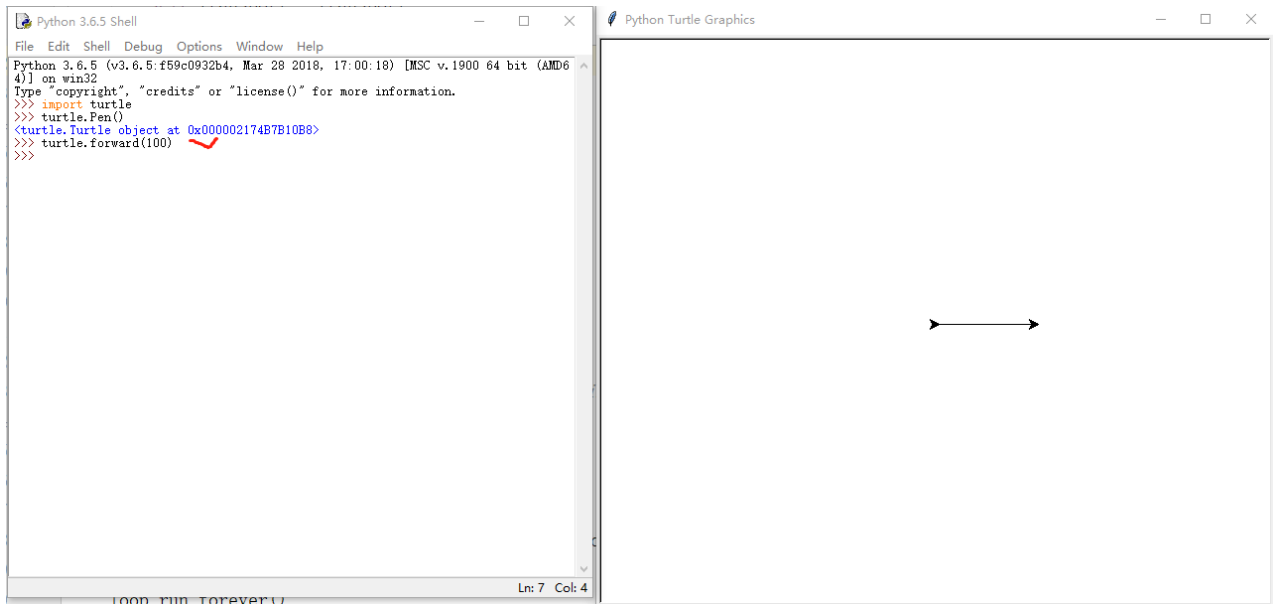


请观察上图黑色箭头，位置位于界面正中心，即  $(0, 0)$ ；方向朝着直角坐标系X方向，角度为0度

从中心位置  $(0, 0)$ ，朝着X方向（0度方向），前进100

```
turtle.forward(100) # 沿着0度方向，向前走了100
```





逐句执行如下代码，观察乌龟画图界面的变化

```
turtle.left(90) # 左转90度

turtle.forward(100) # 沿着90度方向，向前走了100

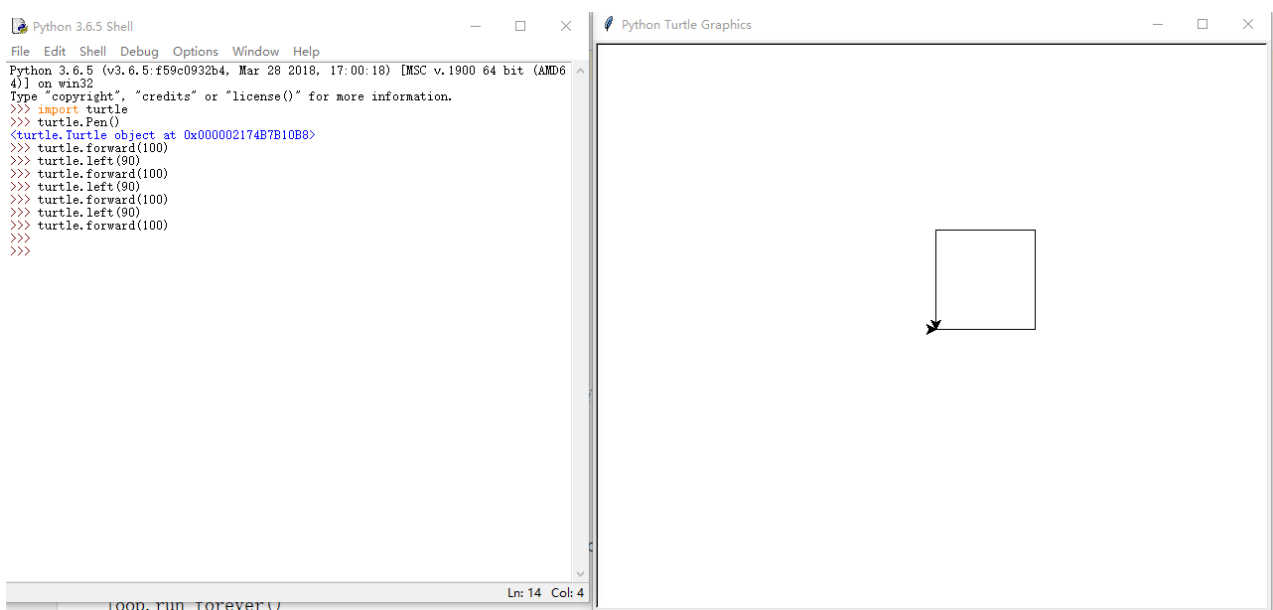
turtle.left(90) # 再左转90度

turtle.forward(100) # 沿着180度方向，向前走了100

turtle.left(90) # 再左转90度

turtle.forward(100) # 沿着270度方向，向前走了100
```

执行完上述python代码，运行效果如下



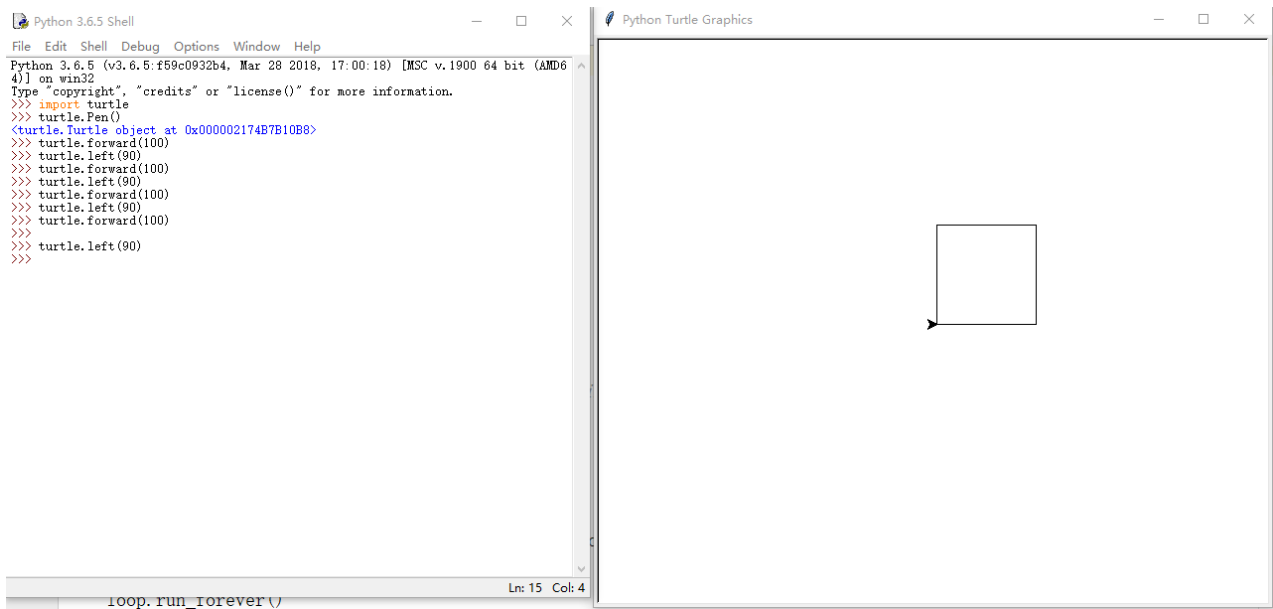
绘制出了一个正方形，箭头回到了中心位置，但是箭头的方向是指向270度，而不是0度

通过代码看一看，总共左转了几个90度，算一算，目前箭头的方向是否正确？

思考，再转多少度，箭头的反向正好是0度？

再左转90度

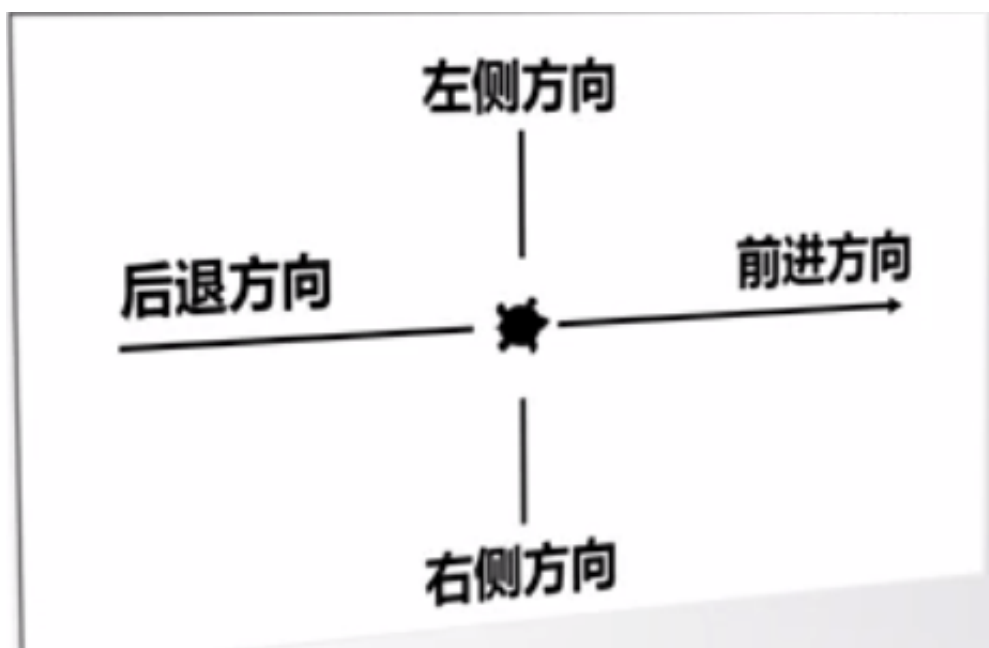
```
turtle.left(90)
```

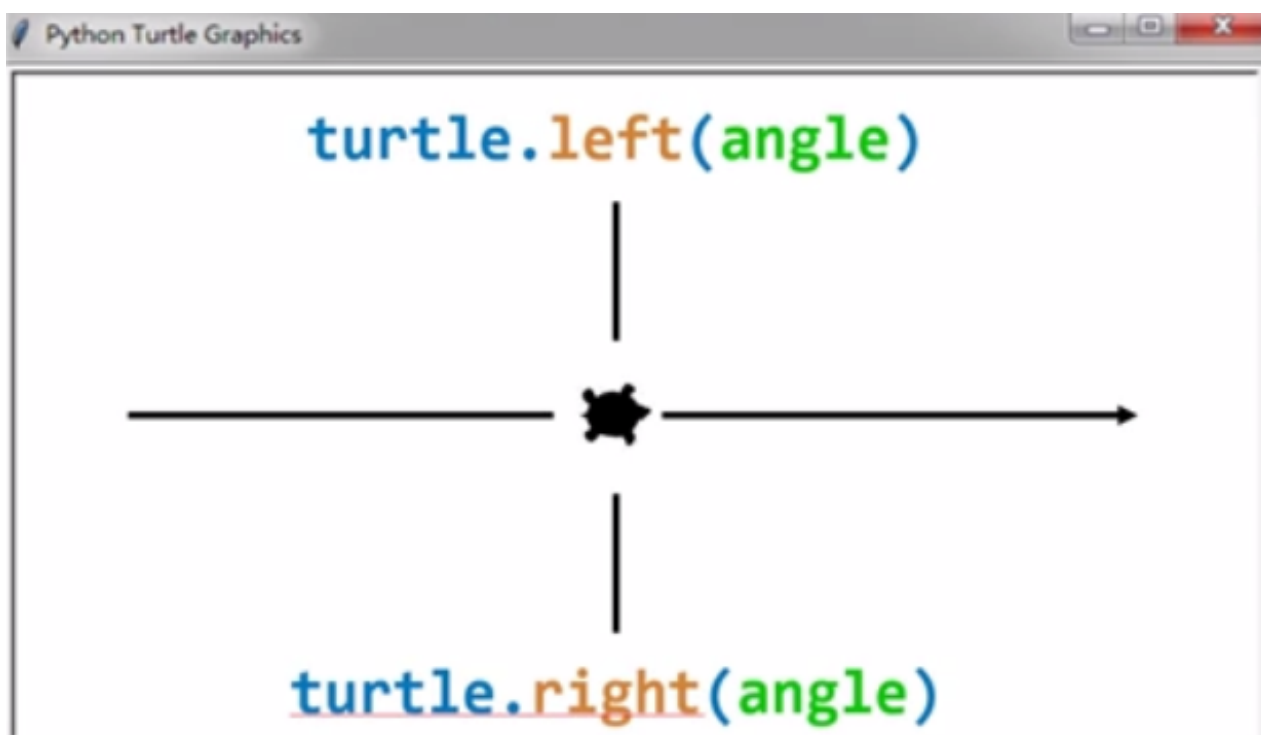
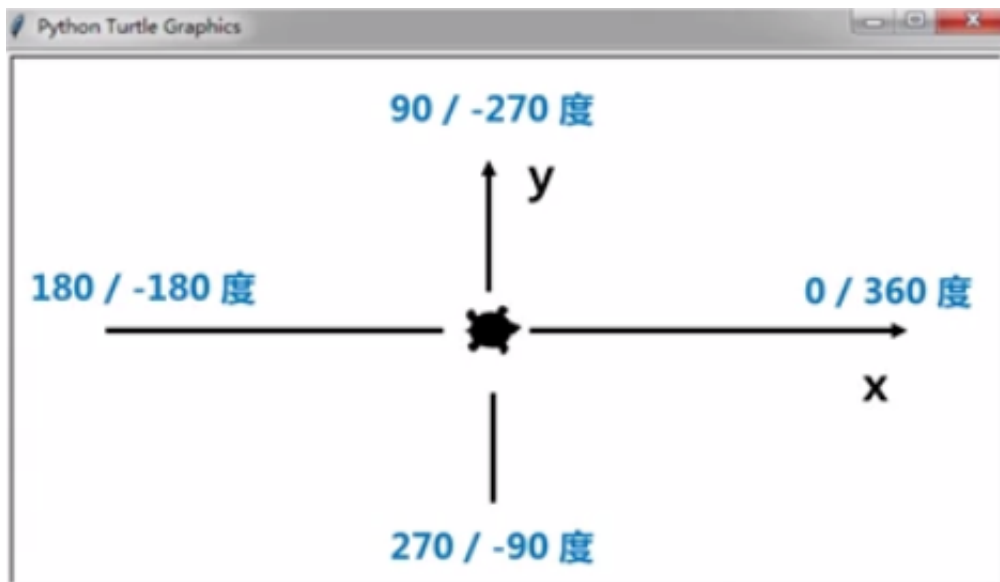


此时，转了4次，刚好转了一圈（360度），箭头又指向了X的方向（0度）

```
turtle.shape('turtle') # 箭头变为小乌龟的形状
```

方向、左转、右转





```
turtle.shape("turtle") # 设置画图小乌龟，默认为箭头
turtle.left(45)
turtle.right(45)
turtle.left(90)
turtle.left(90)
turtle.right(45)
turtle.right(45)
```

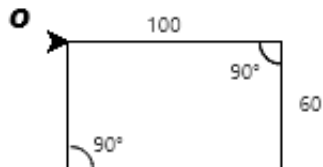
## 2.3 编程实践

## 画线

- 画线

## 画四边形

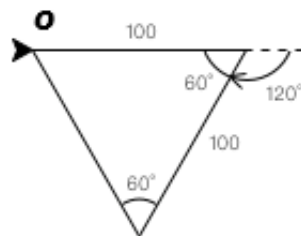
分析：



- 与正方形区别
  - 正方形四条边一样长，四个角都是直角
  - 长方形四条边，对边的长度一样，四个角都是直角
- 关键代码
  - `forward (100)` , `forward (60)` , `forward (100)` , `forward (60)`
  - `left(90)` 或者 `right(90)`

## 画等边三角形

分析：



- 有三条边，三条边一样长
- 边与边的夹角是60度，需要转 $180-60=120$ 度
- 设置颜色`pencolor("red")`

- forward(200)
- left(120) 或者 right(120)

### 打字练习

在金山打字通里自定义今天关键词汇，进行指法训练

## 2.4 你学到了什么？

---