

# 趣玩 Python 之绘制基本图形

原创 萌较瘦 Python技术 3月2日

Python中的类库极其丰富，数据科学中经常会用到可视化技术。今天我们来一学习一下Python中基本图形的绘制方法，本文我们将主要基于 `turtle` (小乌龟)库来画图~

为了方便后面进行交互性演示，这里我建议大家安装好 `Anaconda`，传送门是 <https://www.anaconda.com/distribution/>，请根据自己电脑的操作系统(Windows/Mac/Linux)自行下载和安装，记得要选 `Python3.7` 的版本，因为Python 2.7官方不打算维护了。

Anaconda-download

后面我们需要用到其中的一个强大工具箱 `jupyter notebook`。

jupyter-notebook

Anaconda安装完毕之后呢，接下来，在电脑的命令行(终端)中输入如下命令:

`jupyter notebook` 来启动 `notebook`，这时会打开浏览器，进入网址<http://localhost:8888/tree>，其界面如下~

接下来，我们需要创建一个 `notebook` 文件，按下图操作，点 `Files` -> `New` -> `Python 3` 即可。

create-python-notebook

创建好之后，可以按下图对 `notebook` 文件重命名:

rename

## 画正方形

现在，我们先画一个正方形试试水~

想象一下，我们现在一个起点o，如何从这个起点o画一个正方形呢？好啦，考虑一下后发现是这样的:

- 先水平向右画一条直线，长度比如就为 `100` 像素吧
- 画完上一条边后，顺时针旋转90°角，再画一条等长度的边
- 循环上一步骤几次，正方形就画出来了
- 最后停止画笔

draw-triangle-delay

具体代码如下:

```
1 # 画正方形
2 import turtle as t # 调用turtle库, 并给它一个别名t
3 t.pensize(2) # 设置线的大小
4 for i in range(4): # 画四条边
5     t.fd(100) # 每一次画100个像素
6     t.left(90) # 画100个像素之后转动90°
7 t.done() # 绘图结束后停止画笔
```

## 画三角形

同理, 我们现在一个起点o, 如何从这个起点o画一个三角形呢? 好啦, 考虑一下后发现是这样的:

- 先水平向右画一条直线, 长度比如就为 100 像素吧
- 画完上一条边后, 逆时针旋转120°角, 再画一条等长度的边
- 接着, 顺时针旋转60°角, 再画一条等长度的边
- 最后停止画笔

draw-triangle-delay

具体代码是:

```
1 # 画三角形
2 import turtle as t
3 import time
4 t.forward(100)
5 time.sleep(0.2)
6 t.left(120)
7 t.forward(100)
8 time.sleep(0.2)
9 t.right(60)
10 t.backward(100)
11 time.sleep(0.2)
12
13 time.sleep(0.3)
14 t.done()
```

## 画六边形

画六边形和正方形有点像, 我们只需把之前每次画新边时旋转的90°改为60°, 并把循环次数改为6即可。

six-bian-xing

具体代码是:

```
1 # 画六边形
2 import turtle
3 turtle.pensize(2) #设置线的大小
4 for i in range(6): #因为有六条边, 所以我们画六次
5     turtle.fd(100) #前进100个像素单位
6     turtle.left(60) #向左旋转60度 (每一个内角的外角都为60°)
7 turtle.done() # 画布停留
```

## 画两个六边形的叠边图

想象一下两个六边形对称性地错位, 其特点是, 有**九条边**, 小乌龟最后会回到自己的出发点, 所以角度是 $360^\circ$ 的倍数, 又因为有九条边所以我们可以得出每次转角为 $80^\circ$ 。

于是, 画这个叠边图与画六边形有点像, 我们只需把之前每次画新边时旋转的 $60^\circ$ 改为 $80^\circ$ , 并把循环次数改为9即可。

six-bian-xing-overlap

其具体的代码为:

```
1 # 画 两个六边形叠边图
2 import turtle
3 turtle.pensize(2)
4 for i in range(9):# 因为有九条边, 所以我们选择画九次
5     turtle.fd(150)
6     turtle.left(80)# 每次转角为80°
7 turtle.done()
```

## 画同切圆

首先, 同切圆是什么? 直接给个图吧, 就是这样(多个圆都切于同一条线, 比如我们就用水平线):

tong-qie-yuan

怎么画一个同切圆呢? 观察后发现可以这么干:

- 先以一定长度为半径, 画一个圆, 画完后默认会回到原起点
- 增大半径, 再重复上一步骤
- 重复上一步骤
- 最后停止画笔

我们这就画有4个圆同切吧~

tong-qie-yuan-draw

具体代码为:

```
1 # 画同切圆
2 import turtle
3 turtle.pensize(2)#以左侧30像素处为圆心绘制360°即绘制一个圆（不给出弧度值则表示默认画一个圆）
4 turtle.circle(30)
5 turtle.circle(40)
6 turtle.circle(50)
7 turtle.circle(60)
8 turtle.done()
```

## 画五角星

在草稿纸上画了画之后，我们会发现五角星和正方形的画法类似，旋转角度为 $144^\circ$ ，边数为5。

wu-jiao-xin

其具体代码为:

```
1 # 画五角星
2
3 import turtle
4 p = turtle
5 p.pensize(3)
6 for i in range(5):
7     p.forward(100)
8     p.left(144)#左转144°
9 turtle.done()
```

## 画奥运五环

奥运五环正式版我们就不要完全一样了，我们只需要画出如下近似的即可~

wu-huan

观察之后，我们大概可以这样做:

- 画一个完整的圆，此时画笔回到起点

但我们需要将画笔搬到该点关于圆心对称的地方继续画，怎么办呢？这时我们可以使用 `turtle` 中的 `goto(x,y)` 函数，直接将画笔移动到坐标(x,y)

- 从新的起点继续画完整的圆，画完后移动画笔，直到画完最开始的3个圆，并按要求为边上色
- 用同样的方式画完最后两个圆，并按要求为边上色

wu-huan-draw

- 其具体代码为:

```
1 # 画奥运五环
2 import turtle
3 p = turtle
4 p.pensize(3)
5 p.color("blue")
6 p.circle(30,360)
7 p.pu()
8 p.goto(60,0)
9 p.pd()
10 p.color("black")
11 p.circle(30,360)
12 p.pu()
13 p.goto(120,0)
14 p.pd()
15 p.color("red")
16 p.circle(30,360)
17 p.pu()
18 p.goto(90,-30)
19 p.pd()
20 p.color("green")
21 p.circle(30,360)
22 p.pu()
23 p.goto(30,-30)
24 p.pd()
25 p.color("yellow")
26 p.circle(30,360)
27 p.done()
```

## 画风轮

最后呢，我们来画一个风轮，其中要求每个风轮内角为45度，风轮边长150像素。

feng-ye

结合画上一个图的经验，其实我们结合 `goto(x,y)` 函数来画4个45°的扇形即可~

其具体的代码是:

```
1 # 画风轮
2 import turtle
3 for i in range(4):
4     turtle.fd(100)
5     turtle.right(90)
6     turtle.circle(-100,45)
7     turtle.goto(0,0)
8     turtle.left(45)
9 turtle.done()
```

今天, 这些知识你都学会了嘛~

## 总结

今天我们用 `turtle` 库画了很多基础的图形, 都比较简单, 大家可以发挥自己的想象力多画一些更好玩的。

## 代码地址

示例代码: <https://github.com/JustDoPython/python-100-day/>

**PS:** 公号内回复: Python, 即可进入Python 新手学习交流群, 一起**100天计划!**

-END-

**Python 技术**

关于 Python 都在这里