



模块

车万翔

哈尔滨工业大学



❖ 实现一定的功能的 Python 脚本集合

❖ 引入模块

- `import module_name`

❖ 调用模块的内容

- `module_name.XXX`
- 或者：`from module_name import XXX`
 - XXX 可以直接使用
- 甚至：`from module_name import XXX, YYY`
 - XXX, YYY 可以直接使用
- 更甚至：`from module_name import *`
 - `module_name` 中的所有内容均可直接使用
 - 不推荐



❖ 创建一个 .py 文件 , 如 : hello.py

```
def hello():  
    print 'Hello, World!'  
version = '0.1'
```

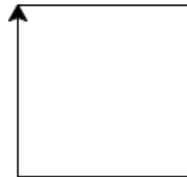
❖ 调用 hello 模块

```
import hello  
hello.hello()  
print hello.version
```



❖ Python内置的图形化模块，可以用来绘制线条、圆、文本等图形

```
>>> import turtle
>>> turtle.forward(100)
>>> turtle.right(90)
>>> turtle.forward(100)
>>> turtle.right(90)
>>> turtle.forward(100)
>>> turtle.right(90)
>>> turtle.forward(100)
>>>
```





❖ 运动

- forward(degree) #向前移动距离degree
- backward(degree) #向后移动距离degree
- right(degree) #向右移动多少度
- left(degree) #向左移动多少度
- goto(x,y) #将画笔移动到坐标为x,y的位置
- speed(speed) #画笔绘制的速度范围[0,10]整数



❖ 画笔控制

- `down()` #移动时绘制图形，缺省时也为绘制
- `up()` #移动时不绘制图形
- `pensize(width)` #绘制图形时的宽度
- `color(colorstring)` #绘制图形时的颜色
- `fillcolor(colorstring)` #绘制图形的填充颜色
- `fill(True)`
- `fill(False)`



Turtle示例 – 绘制五角星



哈爾濱工業大學
HARBIN INSTITUTE OF TECHNOLOGY



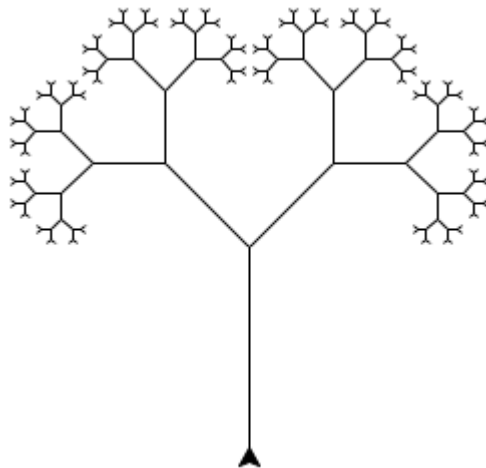
```
>>> import turtle
>>> for i in range(5):
    turtle.forward(100)
    turtle.right(144)
```



Turtle示例 – 递归



```
8 from turtle import forward, backward, left, right, up, down
9 def branch(length, level):
10     if level <= 0:
11         return
12     forward(length)
13     left(45)
14     branch(0.6 * length, level - 1)
15     right(90)
16     branch(0.6 * length, level - 1)
17     left(45)
18     up()
19     backward(length)
20     down()
21
22 left(90)
23 branch(100, 4)
```





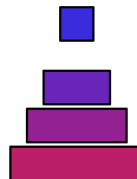
Turtle示例 – 递归



哈爾濱工業大學
HARBIN INSTITUTE OF TECHNOLOGY

❖ http://svn.python.org/projects/python/trunk/Demo/turtle/tdemo_minimal_hanoi.py

```
8 from turtle import *
9
10 class Disc(Turtle):
11     def __init__(self, n):
12         Turtle.__init__(self, shape="square", visible=False)
13         self.pu()
14         self.shapesize(1.5, n*1.5, 2) # square-->rectangle
15         self.fillcolor(n/6., 0, 1-n/6.)
16         self.st()
17
18 class Tower(list):
19     "Hanoi tower, a subclass of built-in type list"
20     def __init__(self, x):
21         "create an empty tower. x is x-position of peg"
22         self.x = x
23     def push(self, d):
24         d.setx(self.x)
25         d.sety(-150+34*len(self))
26         self.append(d)
27     def pop(self):
28         d = list.pop(self)
29         d.sety(150)
30         return d
31
32 def hanoi(n, from_, with_, to_):
33     if n > 0:
34         hanoi(n-1, from_, to_, with_)
35         to_.push(from_.pop())
36         hanoi(n-1, with_, from_, to_)
37
```





Turtle示例 – 绘制漫画



哈爾濱工業大學
HARBIN INSTITUTE OF TECHNOLOGY

```
from turtle import *  
lt(90)  
pensize(3)  
circle(66)  
rt(180)  
circle(66)  
pu()  
lt(90)  
fd(33)  
rt(90)  
fd(34)  
pd()  
fillcolor('blue')  
begin_fill()  
circle(33)  
end_fill()  
pu()  
...
```





❖ random.random

- 用于生成一个0到1的随机浮点数: $0 \leq n < 1.0$

❖ random.uniform

- `random.uniform(a, b)`，用于生成一个指定范围内的随机浮点数，两个参数其中一个是上限，一个是下限。如果 $a < b$ ，则生成的随机数 n : $a \leq n \leq b$

❖ random.randint

- `random.randint(a, b)`，用于生成一个指定范围内的整数。其中参数 a 是下限，参数 b 是上限，生成的随机数 n : $a \leq n \leq b$

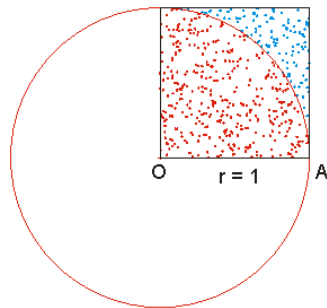
❖ 更多信息

- <https://docs.python.org/2/library/random.html>



❖ 随机算法

- 随机生成一个点
- 判断该点是否在圆内
- 根据原内外的比例，估计 π



```
8 import random
9
10 in_circle = 0
11 times = 1000000
12
13 for i in range(times):
14     x = random.random()
15     y = random.random()
16     if x**2 + y**2 <= 1:
17         in_circle += 1
18
19 print 4.0 * in_circle / times
20
```



猜数字



哈爾濱工業大學
HARBIN INSTITUTE OF TECHNOLOGY

- ❖ 先由计算机“想”一个1到100之间的数请人猜，如果猜对了，显示“正确！”；否则显示“错误！”，并提示所猜的数是大了还是小了。最多可以猜7次。如7次仍未猜中，则停止本次猜数，并羞辱一番游戏者。每次运行程序可以反复猜多个数，直到游戏者想停止时才结束。

```
8 import random
9
10 def guess_number():
11     number = random.randint(1, 100)
12     for i in range(7):
13         guess = int(raw_input('Enter your guess: '))
14         if guess == number:
15             print 'Right!'
16             return
17         elif guess > number:
18             print 'Too big!'
19         else:
20             print 'Too small!'
21
22     print 'You are stupid!'
23
24 while True:
25     guess_number()
26     ch = raw_input('Continue? (y/n): ')
27     if ch != 'y':
28         break
29
```



❖ 科学计算工具

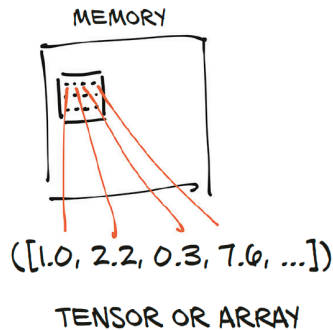
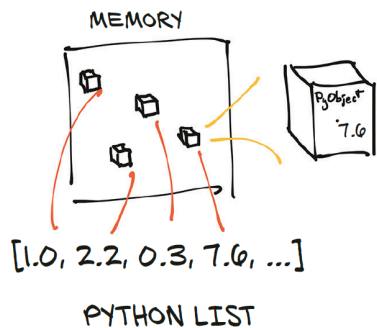
- 任意维数的数组 (ndarray , n-dimensional array)
- 通用函数 (ufunc , universal function)

❖ numpy 数组

- 下标从 0 开始
- 同一数组中所有元素的类型必须相同

❖ numpy 函数

- 如数组创建函数、三角函数等





numpy 模块使用



哈爾濱工業大學
HARBIN INSTITUTE OF TECHNOLOGY

```
import numpy as np
```

```
a = np.array([1,2,3,4]) #创建一维数组
```

```
b = np.array([[1,2,3,4], [5,6,7,8]]) #创建二维数组
```

```
c = np.zeros((3,4)) #创建 3 x 4 的 0 矩阵
```

```
d = np.linspace(-1, 0, 5) #将 -1 到 0 均分五个数字
```

```
e = np.dot(a, b.T) # a 和 b 的转置相乘
```

❖ 更多信息：<http://www.numpy.org/>



matplotlib 模块



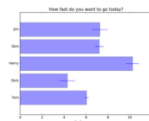
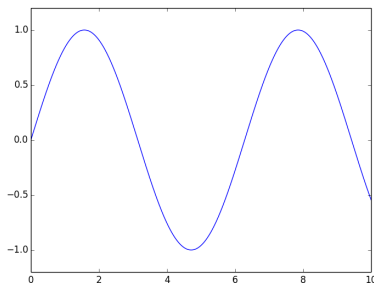
哈爾濱工業大學
HARBIN INSTITUTE OF TECHNOLOGY

❖ matplotlib 是 Python 最著名的绘图库

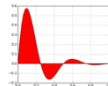
■ 样例库：<http://matplotlib.sourceforge.net/gallery.html>

❖ 简单示例

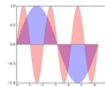
```
8 import numpy as np
9 import matplotlib.pyplot as plt
10
11 x = np.linspace(0, 10, 1000)
12 y = np.sin(x)
13
14 plt.figure()
15 plt.plot(x, y)
16 plt.ylim(-1.2, 1.2)
17 plt.show()
```



barh_demo



fill_demo



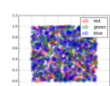
fill_demo_features



marker_reference



marker_reference



scatter_with_legend

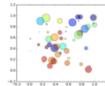
Shapes and collections



artist_reference



path_patch_demo



scatter_demo