1. import random随机数模块

①random.randint(a,b) a到b的随机整数

②random.uniform(a,b) a到b的浮点数

③random.choice

random.choice(seq)

从序列中随机选取一个元素。seq需要是一个序列，比如list、元组、字符串。

random.choice([1, 2, 3, 5, 8, 13]) #list

random.choice('hello') #字符串

random.choice(['hello', 'world']) #字符串组成的list

random.choice((1, 2, 3)) #元组

都是可行的用法。

④random.randrange

random.randrange(start, stop, step)

生成一个从start到stop（不包括stop），间隔为step的一个随机数。start、stop、step都要为整数，且start<stop。

比如：

random.randrange(1, 9, 2)

就是从[1, 3, 5, 7]中随机选取一个。

start和step都可以不提供参数，默认是从0开始，间隔为1。但如果需要指定step，则必须指定start。

random.randrange(4) #[0, 1, 2, 3]

random.randrange(1, 4) #[1, 2, 3]

random.randrange(start, stop, step)其实在效果上等同于

random.choice(range(start, stop, step))

以上是random中常见的几个方法。如果你在程序中需要其中某一个方法，也可以这样写：

from random import randint

randint(1, 10)

1. 计时time模块

Python中有一个time模块，它提供了一些与时间相关的方法。利用time，可以简单地计算出程序运行的时间。对于一些比较复杂、耗时较多的程序，可以通过这种方法了解程序中哪里是效率的瓶颈，从而有针对性地进行优化。

①计算程序开始和结束的时间

于是我们可以用这个方法得到程序开始和结束所用的时间，进而算出运行的时间：

import time

starttime = time.time()

print 'start:%f' % starttime

for i in range(10):

print i

endtime = time.time()

print 'end:%f' % endtime

print 'total time:%f' % (endtime-starttime)

在程序中的不同位置调用time.time()就可以得到运行到那个地方的时间，了解不同部分消耗的时间。

有了这个方法，我们还可以在Pygame课程中的打飞机游戏里，得到每一次游戏主循环刷新的时间，计算出游戏的每秒帧数，显示在屏幕上。

②让程序暂停

顺便再说下time中的另一个很有用的方法：

time.sleep(secs)

它可以让程序暂停secs秒。例如：

import time

print 1

time.sleep(3)

print 2

在抓取网页的时候，适当让程序sleep一下，可以减少短时间内的请求，提高请求的成功率。