1.类和对象

2.正则表达式

3.re模块

4.datetime模块学习

5.http请求

类是抽象的模板，比如Student类，而实例是根据类创建出来的一个个具体的“对象”，每个对象都拥有相同的方法，但各自的数据可能不同。

使用class关键字创建一个最简单的类

class Student:

stuCount = 0 # 类变量 值会被所有实例共享

def \_\_init\_\_(self, name, salary): # 初始化函数/构造函数

#self 代表类的实例，在定义类的方法时必须有，虽然在调用时不必传入相应的参数

self.name = name

self.age= age

Student.stuCount += 1

def printInfo(self): # 类方法

print("Student Name:{}, Age:{}".format(self.name, self.age))

正则表达式用的最多的两个函数是re.match()和re.search()

使用re.match()

match()视图从字符串的起始位置对模式进行匹配.

如果匹配成功, 就返回一个**匹配对象**, 如果匹配失败, 就返回 None.

匹配对象的group()方法能够显示那个成功的匹配.

使用语法: re.match(pattern, string[, flags=0])

re.match()总是从字符串的开始位置开始匹配, 所以很多情况下能够匹配的概率不大.

re.search()是在字符串中搜索正则表达式模式, 任何位置都可以. 返回的是第一次出现的匹配情况(因为正则字符串中可能会多处匹配)

他的参数和re.match()是一样的, 返回值也是一个匹配对象.

datetime模块

datetime模块提供了处理日期和时间的类，既有简单的方式，又有复杂的方式。它虽然支持日期和时间算法，但其实现的重点是为输出格式化和操作提供高效的属性提取功能。

实现http请求有三种方式：

1、urllib-是标准库  
2、urllib3-与urllib类似，但属于扩展库，需要安装

from urllib import request

with request.urlopen('https://blog.csdn.net/hxy199421/article/details/81033229') as f:

data = f.read()

3、requests

import requests

data = requests.get('https://blog.csdn.net/duozhishidai/article/details/89095517',timeout=10)

print(data.content) #返回字节形式