数据存储：

NoSQL:非关系型数据库

执行mongod.exe -f ..\mongod.conf的命令行不能关闭。

MongoDB测试：

例：程序crawler\_test/mongodb\_test.py

import pymongo  
  
mongo\_host = 'localhost'  
mongo\_port = 27017  
  
if \_\_name\_\_ == '\_\_main\_\_':  
 client = pymongo.MongoClient(mongo\_host, mongo\_port)  
 mydb = client['mydb'] # 如果没有就创建mydb数据库  
 test = mydb['test'] # 如果没有就创建test数据集合  
 test.insert\_one({'name': 'Jan', 'sex': '男', 'grade': 89})

爬取数据并存入MongoDB

例：程序crawler\_test/crawler\_test9.py

import pymongo  
import requests  
from lxml import etree  
import crawler\_test1  
import re  
import time  
import mongodb\_test  
  
client = pymongo.MongoClient(mongodb\_test.mongo\_host, mongodb\_test.mongo\_port)  
mydb = client['mydb']  
music\_top = mydb['musictop']  
  
  
def get\_url\_music(url):  
 html = requests.get(url, headers=crawler\_test1.headers)  
 selector = etree.HTML(html.text)  
 music\_hrefs = selector.xpath('//a[@class="nbg"]/@href')  
 for music\_href in music\_hrefs:  
 get\_music\_info(music\_href)  
  
  
def get\_music\_info(url):  
 html = requests.get(url, headers=crawler\_test1.headers)  
 selector = etree.HTML(html.text)  
 name = selector.xpath('//\*[@id="wrapper"]/h1/span/text()')[0]  
 author = re.findall('表演者:.\*?>(.\*?)</a>', html.text, re.S)[0]  
 styles = re.findall('流派:.\*?>(.\*?)<br />', html.text, re.S)  
 if len(styles) == 0:  
 style = '未知'  
 else:  
 style = styles[0].replace('&nbsp;', '').strip()  
  
 time = re.findall('发行时间:</span>(.\*?)<br />', html.text, re.S)[0].strip().replace('&nbsp;', '')  
 publishers = re.findall('出版者:</span>(.\*?)<br />', html.text, re.S)  
 if len(publishers) == 0:  
 publisher = '未知'  
 else:  
 publisher = publishers[0].strip().replace('&nbsp;', '')  
  
 score = selector.xpath('//\*[@id="interest\_sectl"]/div/div[2]/strong/text()')[0]  
  
 info = {'name': name, 'author': author, 'style': style, 'time': time, 'publisher': publisher, 'score': score}  
 music\_top.insert\_one(info)  
  
  
if \_\_name\_\_ == '\_\_main\_\_':  
 urls = ['https://music.douban.com/top250?start={0}'.format(str(i)) for i in range(0, 50, 25)]  
 for url in urls:  
 get\_url\_music(url)  
 time.sleep(1)