

python部分

1. 请列举python2与python3的区别，请将下面的python2代码转换成python3

```
class Point:
    def __init__(self, x, y):
        self.x = x
        self.y = y

    def __str__(self):
        return '({}, {})'.format(self.x, self.y)

points = [Point(9, 2), Point(1,5), Point(2, 7), Point(3, 8), Point(2, 5)]

sorted_points = sorted(
    points,
    lambda (x0, y0), (x1, y1): x0 - x1 if x0 != x1 else y0 - y1,
    lambda point: (point.x, point.y))

#预期结果为(1, 5), (2, 5), (2, 7), (3, 8), (9, 2)
print ', '.join(map(str, sorted_points))
```

1.python2 与python3 区别

- 1.python2 默认编码为ascii所以python2 经常遇到编码问题，而python3默认编码为UTF-8,不用在文件顶部写#coding=utf-8
- 2.python2向后兼容，而python3不是
- 3.True 和 False 在Python2中作为全局变量来处理支持赋值，在python3中修改为关键字
- 4.print不再是语句而是函数等等

```
class Point(object):

    def __init__(self,x,y):
        self.x = x
        self.y = y

    def __str__(self):
        return '({}, {})'.format(self.x,self.y)

points = [Point(9, 2), Point(1,5), Point(2, 7), Point(3, 8), Point(2, 5)]
sorted_points = sorted(points,key=lambda p: (p.x,p.y))
print(', '.join(map(str,sorted_points)))
```

1. 请用python写一个正则表达式实现电话号码的提取功能，下面是个例子

2. 1. 匹配(123)-456-7890和123-456-7890, 不匹配(123-456-7890和123)-456-7890

```
import re

def get_phone(phone):
    pattern = r'\((\d{3})\)-\d{3}-\d{4}|\d{3}-\d{3}-\d{4}'
    result = re.search(pattern, phone)
    if result:
        return result.group(0)

print(get_phone(input('请输入匹配内容')))
```

3. 请用python写一个正则表达式实现如下内容的匹配, 并实现数据的结构化

```
b6b2af4a-02cb-4a45-819e-c35a74186608 300363 关于公司为全资子公司Porton
USA,L.L.C. 申请融资提供担保的公告 2017-01-25
{
  'uuid': 'b6b2af4a-02cb-4a45-819e-c35a74186608',
  'code': '300363',
  'title': '关于公司为全资子公司Porton USA,L.L.C. 申请融资提供担保的公告',
  'date': '2017-01-25'
}
```

```
python text = 'b6b2af4a-02cb-4a45-819e-c35a74186608 300363 关于公司为全资子公司 Porton USA,L.L.C.
申请融资提供担保的公告 2017-01-25'
```

```
import re ret = re.match(r'(\w+-\w+-\w+-\w+-\w{4})\d{8}).(\d{6}).(D+)(\d+-\d+-\d+)', text) content = {}
content['uuid'] = ret.group(1) content['code'] = ret.group(2) content['title'] = ret.group(3) content['date'] =
ret.group(4) print(content)
```

1. 描述python开发中, 协程, 线程, 进程以及python3.4 asyncio的异同, 请写一个python程序在多核CPU上并行执行。

```
import os import sys
```

```
def dump_file(input_dir): if not os.path.exists(input_dir): print('dir {} not exists'.format(input_dir)) return
```

```
path_list = [os.path.join(input_dir, f).strip('\n') for f in
os.listdir(input_dir)]
```

```

file_path_list = filter(lambda file_path: os.path.isfile(file_path),
path_list)

for file_path in file_path_list:
    print('{}:{}'.format(os.getpid(), file_path,
os.path.getsize(file_path)))

```

if name == 'main': if len(sys.argv) > 1: dump_file(sys.argv[1]) else: dump_file(os.getcwd())

<hr>

- 进程是系统分配资源的最小单位，进程之间可以有多个线程，一个进程中资源共享
- 线程是进行任务调度的最小单位，在一个进程中进行操作
- 协程执行具有原子性，需要程序员来调度

```

python
import os
import sys
import multiprocessing
def dump_file(input_dir):
    if not os.path.exists(input_dir):
        print('dir {} not exists'.format(input_dir))
        return
    path_list = [os.path.join(input_dir, f).strip('\n') for f in
os.listdir(input_dir)]
    file_path_list = filter(lambda file_path:
os.path.isfile(file_path),path_list)
    for file_path in file_path_list:
        print('{}:{}'.format(os.getpid(),
file_path,os.path.getsize(file_path)))

if __name__ == '__main__':
    pool_size = multiprocessing.cpu_count()
    pool = multiprocessing.Pool(processes=pool_size,)
    for i in range(pool_size):
        if len(sys.argv) > 1:
            pool.apply_async(dump_file, args=(sys.argv[1],))
        else:
            pool.apply_async(dump_file, args=(os.getcwd(),))
    pool.close()
    pool.join()

```

1. 请列常用的python library，并说明其作用基本使用示例
2. 请解释什么是CORS问题，在什么样的情况下，会遇到这个问题，常用的python库里面该怎么解决这个问题。

-
- 1. library 指其它语言中的库，而在python中成为module，常用标准库有，string，re，datetime，os，等等

- 2. 前端和后端分别设置了两个不同的域名，前端与后端分处不同的域名，我们需要为后端添加跨域访问的支持。 解决问题：安装django的扩展包django-cors-headers，把域名添加到白名单中，指明在跨域访问中，后端支持cookie的操作。

Linux 部分

1. 假设这里有一台服务器地址为192.168.0.2，开放的端口为6623，如何连接到远程服务器操作？
2. 找出当前目录下2天内新创建的所有json文件
3. 如何查看(监控)CPU使用情况，硬盘读写情况以及网络读写情况
4. 简述在浏览器中输入网址到网页内容展现出来，经过了一个什么样的过程。
5. 通过python server.py开启了某个服务，现在该服务失去响应了，请问如何找到这个进程，并杀掉。
6. 如何查看系统某个端口是否已经被其他程序占用
7. 如何设置定时任务，如果定时任务执行命令的结果和在命令行直接执行该命令的结果不同，请考虑可能存在什么问题
8. git rebase命令和git merge命令有什么区别，2个命令对系统会造成什么影响。

-
- 1. ssh root@192.168.0.2:6623
 - 2. find ./ -type f -name '*.json' -ctime -2
 - 3. htop
 - 4. 浏览器输入网址-> 浏览器查找域名ip->与服务器建立连接，三次握手->浏览器发送http请求->服务器接收请求，返回响应，和一个html页面-> 浏览器显示html
 - 5. ps -aux 查找服务进程然后kill -s 9 加查找到的进程id
 - 6. lsof -i:端口号
 - 7. 直接在命令行执行 就是在当前的shell环境下执行
比如涉及到一些环境变量的时候 必须在当前shell环境里执行, 在脚本里执行的话 脚本执行的时候会fork一个子进程 所有操作都在子进程进行 如果涉及到一些在脚本里设置环境变量的东西 脚本结束了 环境变量也就消失了
 - 8. git merge # 将branch分支合并到当前分支 git merge origin/master --no-ff # 不要Fast-Foward合并，这样可以生成merge提交 git rebase master # 将master rebase到branch，相当于： git co && git rebase master && git co master && git merge

其他：

1. 请简单描述您对人工智能，知识图谱的了解
2. 请简单描述您未来5年的规划
3. 请介绍一下最近看的5本书

-
- 1. 人工智能是当下的热点话题，也是企业创新发展的动力之源，当一个新兴事物出现时不要急着跟风也不要急着诋毁，了解它学习它，任何事物存在必然有之理性从哲学角度讲，从社会发展来看，ai必将是改变人们生活方式的一个必然工具，从现在日常生活中，人脸识别的检票程序，无人超市，无人餐厅，和无人驾驶智能家居等等，应用领域之广泛，前景之广阔，需要人才，也需要不断创新和研究发展。吴恩达讲，物联网+ai将无所不能，我想这不仅仅是一个口号而已。
 - 2. 在未来1-2年内，在岗位沉淀下来，关注行业最新发展，致力研究钻研技术，提高技术能力，成为技术中坚力量，未来三到五年如果得到公司认可，希望可以负责开发相关项目，成为管理者或算法研究，同时希望能够在企业中实现价值，得到认可。
 - 3. 算法导论、深度学习、linux系统管理实战、平凡的世界、python数据分析与挖掘实战(还没来及看)