启动方式：

Python -m xx.py将当前数据添加到模块查询路径中

Python xx.py 直接启动

区别在于有依赖项的时候查询的结果不同；

字典常用的几个方法：

.items() 返回字典中的所有键值对

Keys() 返回字典中的所有keys

Value() 返回字典中的所有value

Dict.pop() 获取指定key对应的value,删除字典键值对

Popitem() 方法用于随机弹出字典中的一个key-value

Dict.SetDefalut() 设置指定的key的默认值，如果目标dict中有此key，那么设定不生效；

Fromkeys( [keys], value)

只能配置一个value，如果不配置value的话，所有的值默认为none；

查询USB命令：

cat /sys/kernel/debug/usb/devices

Return 和 yield的区别：

return 获取结果后直接返回，并停止函数继续调用

Yield 进行数据的存储，返回一个迭代对象，使用next取出

装饰器函数是一个Python的重要特点

其主要的优势：简短代码，对仗工整，将相关的处理数据的方法封装

存在一个缺点，使得原函数的\_\_name\_\_属性消失了，但是可以使用warp函数对装饰器函数进行处理，之后可以正常返回目标原函数的\_\_name\_\_

Paramiko函数，拥有的方法，做为ssh传输协议的sftp会话，实现远程文件的上传，下载等操作；

Put 方法 和 get方法 其他的方法

分别为

上传put(self,localpath,remotepath,callback=None)

下载(get(self, remotepath, localpath, callback=None))

setup模块进行文件的打包

Str.splite(),返回的值是一个列表，既可以通过遍历获取值，同样也可以通过直接指定获取相应的值

A\_list[5:]如果以超出列表长度的Index进行获取，不会报错，只会返回一个空的列表值

重点在于可变元素和不可变元素；

a. 整型 int、 长整型 long、浮点型 float、 复数 complex

b. 字符串 str、 列表 list、 元祖 tuple

c. 字典 dict 、 集合 set

Shuffle 函数：  
random.shuffle （）将目标函数进行随机处理

Samp(n,x)从目标序列n中取出x个元素，将取到的x个元素以List的形式返回

可变类型和不可变类型：  
进行修改操作的时候，可变类型传递的内存地址

例如list,dict；不可变类型有string,number.tuple

可变类型传递的是内存中的地址，直接修改内存中的值，没有开辟新的内存

不可变类型被改变的时候，没有改变原内存地址的值，开辟了一个新的内存，将源地址的值复制出来，开辟一个新的内存，写入后并进行修改；

洛必达法则写法，lambda：x,y:x+y 目标

前面的是处理的参数，后面的是处理方法和处理目标

不能处理复杂的逻辑状态，并且调用的时候不方便；

Re.sub方法，替换目标字符串的结果:

Re.sub(x,y,z)

将z中的x 用y替换掉；

Str. Zill(x)返回指定长度x的字符串

sql是否能在docker中启动两个

可以启动多个sql服务，类似于一个服务器启动多个sql；扩展命令，docker images查看某个镜像id

Docker run -d

Docker ps -a 查看生成的容器

Docker restart 容器id 重启某个

Docker stop 容器id, 停止此容器

Docker rm 容器id, 删除此容器

Ps -ef|grep mysql 查看当前的sql服务

突出点：在以往项目中自身作用，重点凸显出能力的哪些部分，项目优化的地方，以及自身做了些什么，是否只是将需求转化成代码，做了哪些优化；

深拷贝和浅拷贝的区别：

浅拷贝，只拷贝了不可变元素，建立了一个新的地址但是其中的动态数据并没有进行拷贝，核心数据的指向地址还是同一个地址；

int,float,complex,long,str,unicode,tuple 不可以变元素

Linux 操作部分：

cat和tail的区别：一个展示整个文件，一个展示文件的最后几行；如果文件内容比较多，或者文件比较大的时候建议使用tail;

Head 和 tail对照，head展示文件的前多少行

less和more 进行文件的查询

在vim下如何进行特定字符的查找；进入normal模式点击/进行查找；

选择后，点击n 向下翻页，点击N向上翻页；

对装饰器的理解

常用的抓包工具，如何mock数据

模拟测试数据，拦截请求，伪造返回值；