Mysqldb使用过程中遇到的问题：

1. 读出来的数据只显示数据库数据，并不会显示出来数据表字段名的名称

解决方案：（1）使用MySQLdb，在构造cursor时，指定cursor类型即可。

即传入参数cursorclass=MySQLdb.cursors.DictCursor

（2）另外，普通的cursor中cursor.description有所选列的信息。下面的代码会打印结果中各列的名字：

for field\_desc in cursor.description:  
   print field\_desc[0]

1. fetchall()返回结果集中的全部数据，结果为一个tuple的列表。每个tuple元素是按建表的字段顺序排列。

3、编译运行总是出现AssertionError: Unimplemented method 'GET'错误

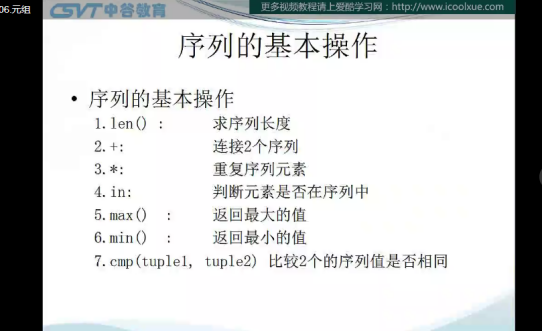
解决方法：flask restful api 里面支持def get(self):

所以将所有的函数名换成get即可成功运行。

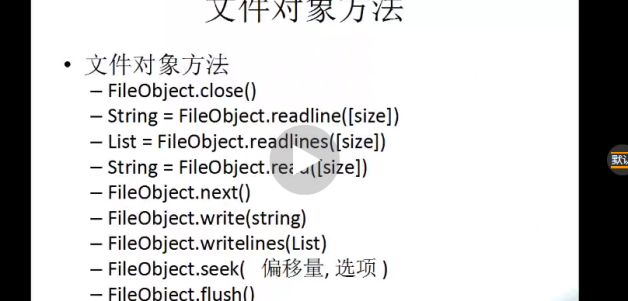
1. 过滤条件data = filter(lambda t: (t['longtitude'] == da\_id and t['latitude'] == de\_id),datas)，、

根据longtitude和latitude来查询MySQL数据表中的结果，切记中间用and，不能用&连字符

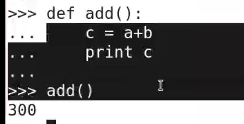
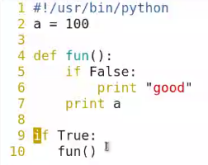
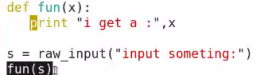
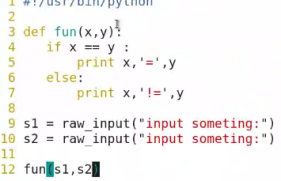
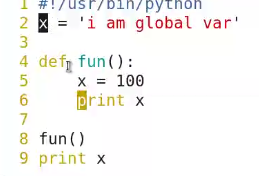
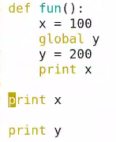
注意：**python语句中并且要用and，或要用or, 非要用not**

1. 打印字符串时，若要输出双引号，则可用（\）转义
2. 换行的转义字符是‘\n’
3. 数据类型：a=‘abcde’ a[0]=a a[1]=b ‘ab’=a[0]+a[1]而不能是啊a[0][1] ‘bcd’=a[1:4(从第二位到第四位数前)] a[:]表示区全部值=‘abcde’
4. 元组（小括号( )表示）：

t=(“milo”,30,”male”)圆括号中用逗号隔开

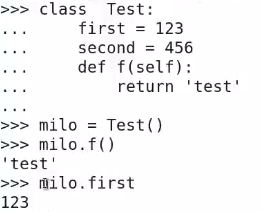
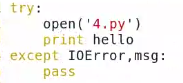
1. 列表（中括号[ ]表示）：
2. { } 大括号，表示定义一个对象，大部分情况下要有成对的属性和值，或是函数
3. 字典：key()和value()返回键列表或值列表，items()返回带键值对的元组，创建字典用 { key：value}，如d={‘name’:’valerie’,’age’:’23’} d[‘name’] =’valerie’通过key 取value值, .item()可以取出字典的key，value，.keys()可以取出字典的所有key，字典是无序的
4. 文件读取方法
5. sublime3光标咋变成下划线了 按insert键复位
6. 循环取数据的时候，一定要注意全局变量和局部变量，稍有不注意就会导致数据不对

也可按照索引将数值遍历出来，s=[‘’,’’]是一个列表，如果要循环的数据为int类型，是不能用fo x in 数据：必须将其按照索引（添加一个buffer）循环

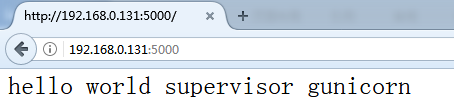
1. 定义函数def fun(形参)并引用，（不传参）
2. 调用函数（实参），
3. 全局变量和局部变量相同时，不影响，但看其是在哪里调用，返回什么值即，仅仅是名字相同，但含义不同。global 将变量全局化
4. 定义一个class类如：class glob(不管有无参数都可):

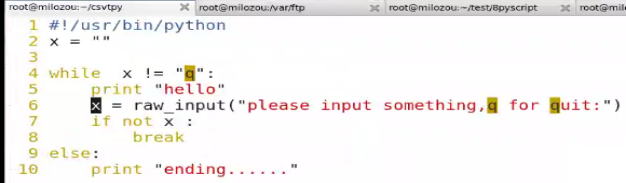
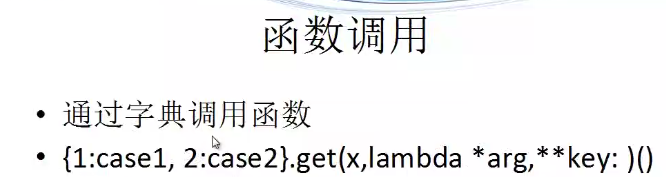
def function(self必须加，类里至少有一个形参self):

后面引用类时只需定义一个对象如a=glob()

1. Flask restful api定义函数时小括号里需传入参数self；单纯的flask则不需要
2. 调用类别
3. 可以查看每个元素，比如列表都有哪些方法，如append
4. 异常
5. from flask import abort，abort函数用于处理错误，返回状态码
6. 在Linux环境下运行程序，Web 服务器可使用 http://a.b.c.d:5000/ 网络中的任一台电脑进行访问，其中“ a.b.c.d”是服务器所在计算机的外网 IP 地址 ，如：



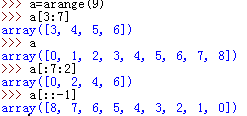


1. Flask初始化
2. 启动服务器
3. 'dict' object is not callable错误：（1）代码里重新定义了dict，比如 dict= {...}，这时调用的是代码里定义的dict而不是python内置类型，（2）取字典内容时用了（）而不是［］。比如sdict("content\_id"),应该是sdict["content\_id"]
4. IndentationError: unindent does not match any outer indentation level错误：运行环境是win7 x64 sublime text2，百度发现是对齐问题。具体来说是由于有的地方使用了4个空格，有的地方使用了tab键。代码区直接全选就会看到有的地方是四个点有个地方是一个横线，改一致了就好了
5. UnboundLocalError: local variable 'r2' referenced before assignment错误：这是因为在函数内部对变量赋值进行修改后，该变量就会被Python解释器认为是局部变量而非全局变量，而因为这个局部变量还没有定义，自然就会抛出这样的错误。（引用的局部变量没有定义）
6. Zip()函数可以友好的对两个列表进行一一对应组合
7. r1=float(r[1].replace('.','.'))，replace()函数的使用，当’.’转换不了小数点的时候可将它替换成英文’.’
8. 
9. Switch函数调用 如31图1，可以很方便的使用
10. 可以将自己的xx.py文件作为模块单独引入（当前目录）（import，import as，from import），引入时方法为import xx；直接调用即单独的.py文件（可用 if \_name\_==’\_main\_’：后面为直接执行语句，即name的值就为main，间接调用as作用为更方便理解的词，（从cal文件中导入jia函数使用）（cal模块为cal.py文件），，导入模块先从当前目录下导入，然后才会去python模块包里找，最后找不到就会报错
11. 模块打包，通常会建立一个\_init\_.py（可为空，只要有就代表为包的结构）的文件，将其放在要引入的.py文件同级目录里，然后引入的时候，从上层文件.引包文件 renzheng.py 引入模块使import cat.textcat目录下
12. %and.format() '%s %s' % ('one', 'two') '{} {}'.format('one', 'two') output:one two
13. '{1} {0}'.format('one', 'two') output:two one
14. '{:>10}'.format('test') 将test按10个占位符右对齐（ test）； '{:10}'.format('test') 将test按10个占位符左对齐（test ） '{:^10}'.format('test') 将test按10个占位符居中（ test ）
15. 截断字符串'{:.5}'.format('xylophone') output （xylop）
16. 截断填充'{:10.5}'.format('xylophone') output （xylop ）
17. 数字形式。'{:d}'.format(42) output （42）；'{:4d}'.format(42) output（ 42）； '{:06.2f}'.format(3.141592653589793) output（003.14）
18. 也可用于字典value值的获取
19. Flaot 固定8位，double固定格式为11位
20. Pro.jsonlint.com可查看正确的json格式，网页版
21. Restful设计理念：（1）统一接口。服务端和客户端；（2）无状态。每次请求和响应 不依赖前一次请求；（3）可缓存。浏览器可缓存；（4）服务端和客户端分离；（5） 分层系统；（6）按需编码。可用不同的编程语言进行编码，选择合适的语言进行程序 的编译。

46、

**Python 数据分析**

**Numpy模块：**

1. 切片模式
2. 引入 from numpy import \*；加载数据np.loadtxt(“a.csv”)
3. 处理数组的形状。拆解、拉直、用元组指定数组形状、转置、调整大小
4. 堆叠数组，水平叠加hstack()，将后组数据加到前一个列表的后方；垂直叠加vstack()，将后一组数据加到前一个列表数据的下方;深度叠加dstack()，将两个列表纵向排列放入一个列表中；列式堆叠column\_stack()二维数组时相当于hstack()函数；行式堆叠row\_stack()对二维

数组进行堆叠时相当于vstack()函数；；concatente((a,b),axis=0/1)

