马哥教育

IT人的高薪职业学院

Python高阶函数、柯里化

讲师：Wayne

从业十余载，漫漫求知路

高阶函数

- First Class Object
  - 函数在Python中是一等公民
  - 函数也是对象，可调用的对象
  - 函数可以作为普通变量、参数、返回值等等
- 高阶函数
  - 数学概念 y=g(f(x))
  - 在数学和计算机科学中，高阶函数应当是至少满足下面一个条件的函数
    - 接受一个或多个函数作为参数
    - 输出一个函数

高阶函数

- 计数器

```
def counter(base):
    def inc(step=1):
        base += step
        return base
    return inc
```

- 分析：
  - 函数couter是不是一个高阶函数
  - 上面代码有没有什么问题？怎样改进
  - 如何调用完成计数功能
  - f1 = counter(5)和f2=counter(5)，请问f1和f2相等吗？

自定义sort函数

- 排序问题
  - 仿照内建函数sorted，请自行实现一个sort函数（不使用内建函数），能够为列表元素排序
- 思路
  - 内建函数sorted函数是返回一个新的列表，可以设置升序或降序，可以设置一个排序的函数。自定义的sort函数也要实现这个功能
  - 新建一个列表，遍历原列表，和新列表的值依次比较决定如何插入到新列表中
- 思考
  - sorted函数的实现原理，扩展到map、filter函数的实现原理

自定义sort函数

- sort函数实现。下面实现的什么排序？还能怎么改变

```
def sort(iterable):
    ret = [ ]
    for x in iterable:
        for i, y in enumerate(ret):
            if x > y: # 找到大的就地插入。如果换成x < y呢，函数什么意思呢？
                ret.insert(i,x) # 降序
                break
        else: # 不大于，说明是最小的，尾部追加
            ret.append(x)
    return ret
print(sort([1,2,5,4,2,3,5,6]))
```

自定义sort函数

- sort函数实现。用一个参数控制顺序

```
def sort(iterable, reverse=False):
    ret = []
    for x in iterable:
        for i, y in enumerate(ret):
            flag = x>y if reverse else x<y
            if flag: # 是否能进一步改进
                ret.insert(i,x)
                break
        else:
            ret.append(x)
    return ret
print(sort([1,2,5,4,2,3,5,6]))
```


马哥教育 IT人的高薪职业学院

自定义sort函数

- sort函数实现。下面实现的什么排序？还能怎么改变

```
def sort(iterable, key=lambda a,b : a>b):
    ret = [ ]
    for x in iterable:
        for i, y in enumerate(ret):
            if key(x, y): # 函数的返回值是bool
                ret.insert(i,x)
                break
        else:
            ret.append(x)
    return ret
print(sort([1,2,5,4,2,3,5,6]))  # 升序还是降序？如何反序？
```

自定义sort函数

- sort函数实现

```
def sort(iterable, reverse=False, key=lambda x,y:x<y):
    ret = []
    for x in iterable:
        for i,y in enumerate(ret):
            flag = key(x,y) if not reverse else not key(x,y)
            if flag:
                ret.insert(i, x)
                break
        else:
            ret.append(x)
    return ret
print(sort([1,2,5,4,2,3,5,6]))
```

内建函数-高阶函数

- sorted(iterable[, key][, reverse])
  - 排序
- filter(function, iterable) --> filter object
  - 过滤数据
- map(func, *iterables) --> map object
  - 映射

马哥教育
IT人的高薪职业学院

内建函数-高阶函数

- sorted(iterable[, key][, reverse]) 排序
  - 返回一个新的列表，对一个可迭代对象的所有元素排序，排序规则为key定义的函数，reverse表示是否排序翻转
  - sorted(lst,key=lambda x:6-x) # 返回新列表
  - list.sort(key=lambda x: 6-x) # 就地修改

马哥教育
IT人的高薪职业学院

内建函数-高阶函数

- filter(function, iterable)
  - 过滤可迭代对象的元素，返回一个迭代器
  - function一个具有一个参数的函数，返回bool
  - 例如，过滤出数列中能被3整除的数字

    list(filter(lambda x: x%3==0, [1,9,55,150,-3,78,28,123]))
- map(function, *iterables) --> map object
  - 对多个可迭代对象的元素按照指定的函数进行映射，返回一个迭代器
    - list(map(lambda x:2*x+1, range(5)))
    - dict(map(lambda x: (x%5,x) , range(500)))

柯里化Currying

- 柯里化
  - 指的是将原来接受两个参数的函数变成新的接受一个参数的函数的过程。新的函数返回一个以原有第二个参数为参数的函数
  - z = f(x, y) 转换成 z = f(x)(y)的形式

- 举例
  - 将加法函数柯里化

```
def add(x, y):
    return x + y
```

柯里化

- 举例
  - 将加法函数柯里化

```
def add(x, y):
    return x + y
```

  - 转换如下

```
def add(x):
    def _add(y):
        return x+y
    return _add
add(5)(6)
```

  - 通过嵌套函数就可以把函数转换成柯里化函数

马哥教育 IT人的高薪职业学院

谢谢

咨询热线 400-080-6560