

---

## 模块 操作数据

(\*)urls.py

```
from django.conf.urls import url
import person.views

urlpatterns = [
    url(r'^$', person.views.show),
]
```

(\*) show.html

```
<html>
<body>
<table border="1">
{% for person in persons %}
    <tr><td>{{person.pName}}</td><td>{{person.pSex}}</td><td>{{person.p
Age}}</td><td>{{person.pTel}}</td></tr>
{% endfor %}
</table>
</body>
</html>
```

(\*) person\models.py

```
from django.db import models

# Create your models here.
class PersonModel(models.Model):
    pName=models.CharField(primary_key=True,max_length=30)
    pSex = models.CharField(max_length=8)
    pAge=models.IntegerField()
    pTel=models.CharField(max_length=11)

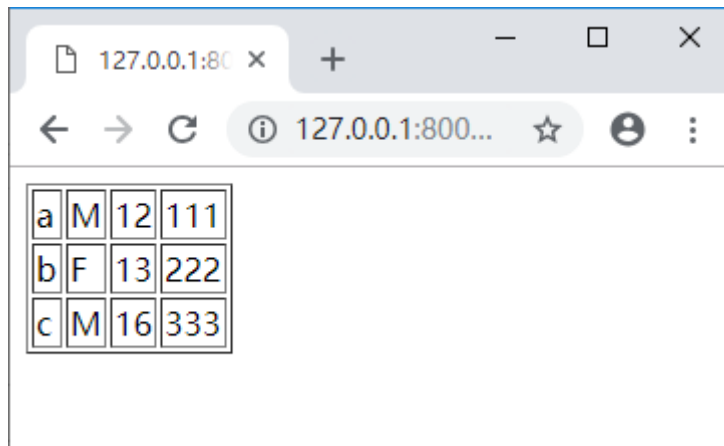
    def __str__(self):
        return self.Name
```

(\*) 排序

(\*)person\views.py 升序排序

```
from django.shortcuts import render
from person.models import PersonModel
```

```
def show(request):
    PersonModel.objects.all().delete()
    PersonModel.objects.create(pName="a",pSex="M",pAge=12,pTel="111")
    PersonModel.objects.create(pName="c",pSex="M",pAge=16,pTel="333")
    PersonModel.objects.create(pName="b",pSex="F",pAge=13,pTel="222")
    persons=PersonModel.objects.all().order_by("pName")
    return render(request,"show.html",{"persons":persons})
```

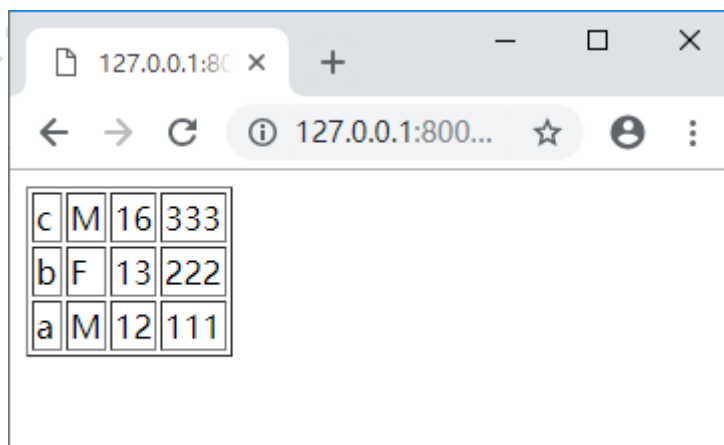


a	M	12	111
b	F	13	222
c	M	16	333

(\*) person\views.py 降序排序

```
from django.shortcuts import render
from person.models import PersonModel

def show(request):
    PersonModel.objects.all().delete()
    PersonModel.objects.create(pName="a",pSex="M",pAge=12,pTel="111")
    PersonModel.objects.create(pName="c",pSex="M",pAge=16,pTel="333")
    PersonModel.objects.create(pName="b",pSex="F",pAge=13,pTel="222")
    persons=PersonModel.objects.all().order_by("-pName")
    return render(request,"show.html",{"persons":persons})
```

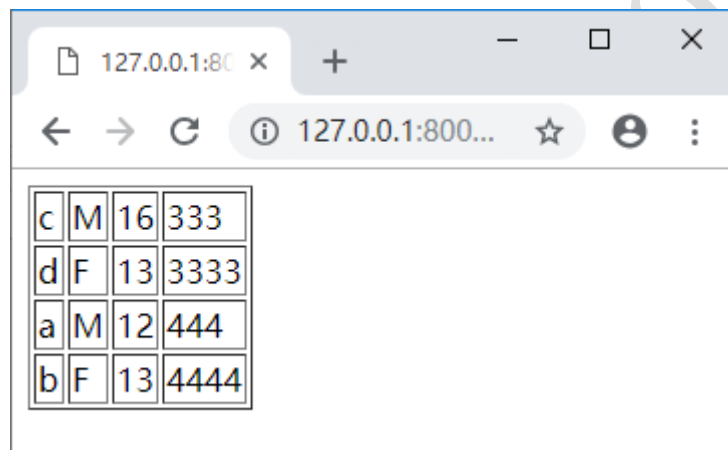


c	M	16	333
b	F	13	222
a	M	12	111

(\*) person\views.py

```
from django.shortcuts import render
from person.models import PersonModel

def show(request):
    PersonModel.objects.all().delete()
    PersonModel.objects.create(pName="a",pSex="M",pAge=12,pTel="444")
    PersonModel.objects.create(pName="c",pSex="M",pAge=16,pTel="333")
    PersonModel.objects.create(pName="b",pSex="F",pAge=13,pTel="4444")
    PersonModel.objects.create(pName="d",pSex="F",pAge=13,pTel="3333")
    persons=PersonModel.objects.all().order_by("pSex").order_by("pTel")
    return render(request,"show.html",{ "persons":persons})
```



c	M	16	333
d	F	13	3333
a	M	12	444
b	F	13	4444

注意:

**persons=PersonModel.objects.all().order\_by("pSex").order\_by("pTel")**

不是按 pAge 排序再按 pTel 排序，它是按 pTel 排序。

使用:

**persons=PersonModel.objects.all().order\_by("pSex","pTel")**

实现按 pAge 排序再按 pTel 排序。

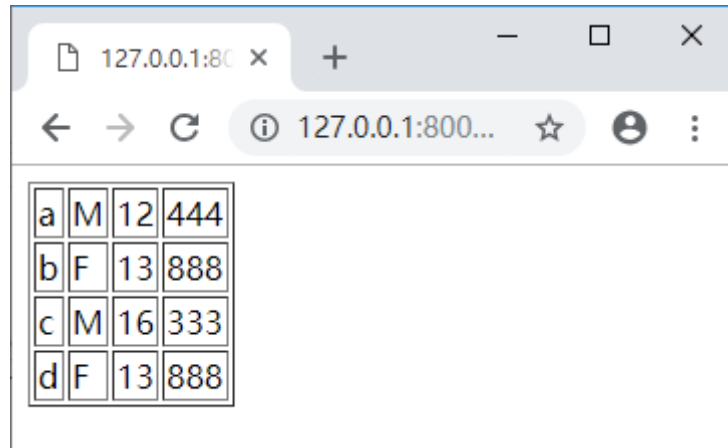
(\*) 更新

(\*) person\views.py

```
from django.shortcuts import render
from person.models import PersonModel

def show(request):
    PersonModel.objects.all().delete()
    PersonModel.objects.create(pName="a",pSex="M",pAge=12,pTel="444")
    PersonModel.objects.create(pName="c",pSex="M",pAge=16,pTel="333")
    PersonModel.objects.create(pName="b",pSex="F",pAge=13,pTel="4444")
    PersonModel.objects.create(pName="d",pSex="F",pAge=13,pTel="3333")
```

```
PersonModel.objects.filter(pSex="F").update(pTel="888")
persons=PersonModel.objects.all().order_by("pName")
return render(request,"show.html",{"persons":persons})
```



a	M	12	444
b	F	13	888
c	M	16	333
d	F	13	888

其中：

`PersonModel.objects.filter(pSex="F").update(pTel="888")`  
对所有的 `pSex="F"`的记录进行修改。

### (\*) 删除

(\*) person\views.py

```
from django.shortcuts import render
from person.models import PersonModel

def show(request):
    PersonModel.objects.all().delete()
    PersonModel.objects.create(pName="a",pSex="M",pAge=12,pTel="444")
    PersonModel.objects.create(pName="c",pSex="M",pAge=16,pTel="333")
    PersonModel.objects.create(pName="b",pSex="F",pAge=13,pTel="4444")
    PersonModel.objects.create(pName="d",pSex="F",pAge=13,pTel="3333")
    PersonModel.objects.filter(pSex="F").delete()
    persons=PersonModel.objects.all().order_by("pName")
    return render(request,"show.html",{"persons":persons})
```

a	M	12	444
c	M	16	333

其中：

```
PersonModel.objects.all().delete()
```

删除全部记录。

```
PersonModel.objects.filter(pSex="F").delete()
```

删除全部 pSex="F"的记录。

总结：

```
class PersonModel(models.Model):
```

```
    pName=models.CharField(primary_key=True,max_length=30)
```

```
    pSex = models.CharField(max_length=8)
```

```
    pAge=models.IntegerField()
```

```
    pTel=models.CharField(max_length=11)
```

```
    def __str__(self):
```

```
        return self.Name
```

(1) 获取记录

获取所有数据行：

```
rows=PersonModel.objects.all()
```

获取所有满足条件的数据行：

```
rows=PersonModel.objects.filter(pSex="F",pAge>=20)
```

(2) 更新记录

```
PersonModel.objects.filter(pSex="F").update(pTel="888",pAge=20)
```

(3) 删除记录

删除全部：

```
PersonModel.objects.all().delete()
```

删除满足条件的记录：

```
PersonModel.objects.filter(pSex="F").delete()
```

(4) 增加记录

create 增加记录，如果关键字重复会抛出异常：

```
PersonModel.objects.create(pName="a",pSex="M",pAge=12,pTel="444")
```

---

(5) 增加与更新记录

**save** 增加或者更新记录。如果关键字重复则就不增加，也不会抛出异常。如果该记录已经存在，就更新记录。

```
p=PersonModel(pName="a",pSex="M",pAge=12,pTel="444")
p.save()
```

(6) 排序记录

符合条件的记录按 **pAge** 从小到大排序:

```
PersonModel.objects.filter(pSex="F").order_by("pAge")
```

符合条件的记录按 **pAge** 从大到小排序:

```
PersonModel.objects.filter(pSex="F").order_by("-pAge")
```

按(**pSex**,**pAge**)组合排序 (从小到大):

```
PersonModel.objects.all().order_by("pSex","pAge")
```

按(**pSex**,**-pAge**)组合排序 (**pSex** 升序、**pAge** 降序):

```
PersonModel.objects.all().order_by("pSex","-pAge")
```