安装Django:pip install django

创建项目：`django-admin startproject HelloWorld`

启动服务器：`python manage.py runserver 0.0.0.0:8000`

测试访问：http://localhost:8000

配置url

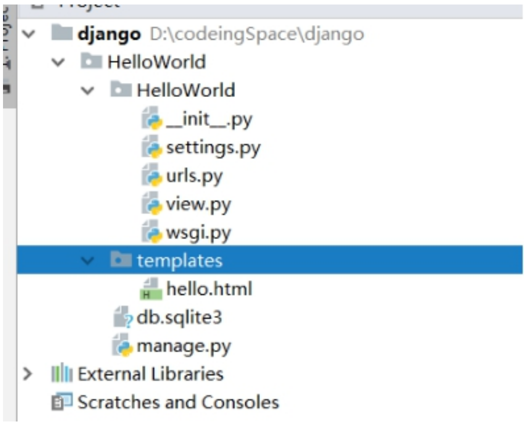
```
from django.conf.urls import url
from . import view
urlpatterns = [
    url(r'^hello$', view.hello),
]
```

url变成如下：

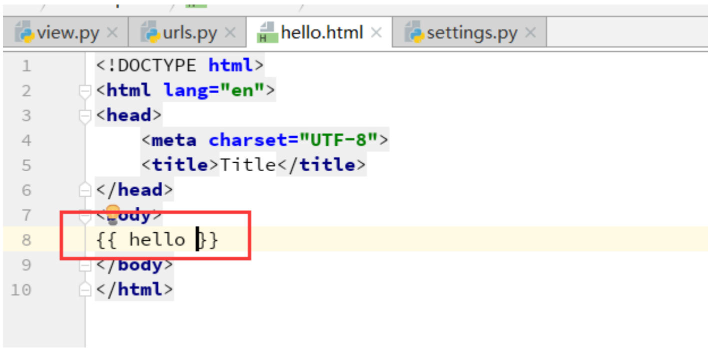
http://localhost:8000/hello

按照mvc模式开发：

建立**视图**文件（视图文件中的一些标签，if else,for之类的看看手册就好了）

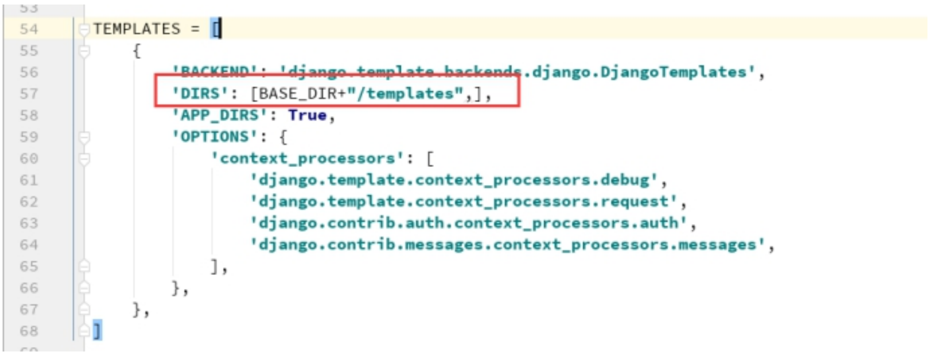


```
django D:\codeingSpace\django
  HelloWorld
    HelloWorld
      __init__.py
      settings.py
      urls.py
      view.py
      wsgi.py
    templates
      hello.html
    db.sqlite3
    manage.py
External Libraries
Scratches and Consoles
```



```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Title</title>
</head>
<body>
{{ hello }}
</body>
</html>
```

配置视图文件路径setting.py



```
TEMPLATES = [
    {
        'BACKEND': 'django.template.backends.django.DjangoTemplates',
        'DIRS': [BASE_DIR+"/templates",],
        'APP_DIRS': True,
        'OPTIONS': {
            'context_processors': [
                'django.template.context_processors.debug',
                'django.template.context_processors.request',
                'django.contrib.auth.context_processors.auth',
                'django.contrib.messages.context_processors.messages',
            ],
        },
    },
]
```

Django 模板标签

if/else 标签

基本语法格式如下：

```
{% if condition %}
     ... display
{% endif %}
```

或者：

```
{% if condition1 %}
   ... display 1
{% elif condition2 %}
   ... display 2
{% else %}
   ... display 3
{% endif %}
```

根据条件判断是否输出。if/else 支持嵌套。

{% if %} 标签接受 and ， or 或者 not 关键字来对多个变量做判断 ，或者对变量取反（ not)，例如：

```
{% if athlete_list and coach_list %}
     athletes 和 coaches 变量都是可用的。
{% endif %}
```

for 标签

{% for %} 允许我们在一个序列上迭代。

与Python的 for 语句的情形类似，循环语法是 for X in Y ，Y是要迭代的序列而X是在每一个特定的循环中使用的变量名称。

每一次循环中，模板系统会渲染在 {% for %} 和 {% endfor %} 之间的所有内容。

例如，给定一个运动员列表 athlete_list 变量，我们可以使用下面的代码来显示这个列表：

```
<ul>
{% for athlete in athlete_list %}
    <li>{{ athlete.name }}</li>
{% endfor %}
</ul>
```

给标签增加一个 reversed 使得该列表被反向迭代：

```
{% for athlete in athlete_list reversed %}
...
{% endfor %}
```

可以嵌套使用 {% for %} 标签：

```
{% for athlete in athlete_list %}
    <h1>{{ athlete.name }}</h1>
    <ul>
    {% for sport in athlete.sports_played %}
        <li>{{ sport }}</li>
    {% endfor %}
    </ul>
{% endfor %}
```

ifequal/ifnotequal 标签

{% ifequal %} 标签比较两个值，当他们相等时，显示在 {% ifequal %} 和 {% endifequal %} 之中所有的值。

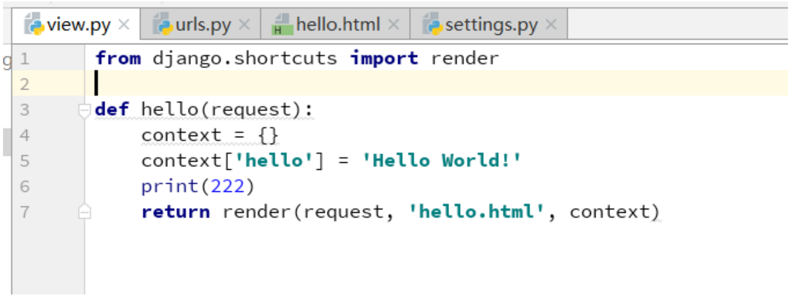
下面的例子比较两个模板变量 user 和 currentuser：

```
{% ifequal user currentuser %}
    <h1>Welcome!</h1>
{% endifequal %}
```

和 {% if %} 类似， {% ifequal %} 支持可选的 {% else%} 标签：8

```
{% ifequal section 'sitenews' %}
    <h1>Site News</h1>
{% else %}
    <h1>No News Here</h1>
{% endifequal %}
```

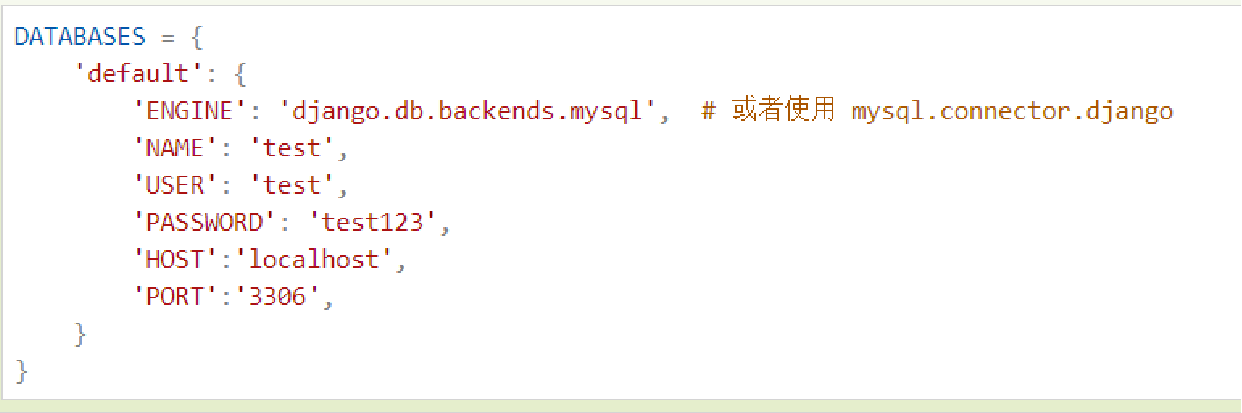
控制器中把参数分配到视图中



```
from django.shortcuts import render

def hello(request):
    context = {}
    context['hello'] = 'Hello World!'
    print(222)
    return render(request, 'hello.html', context)
```

模型：

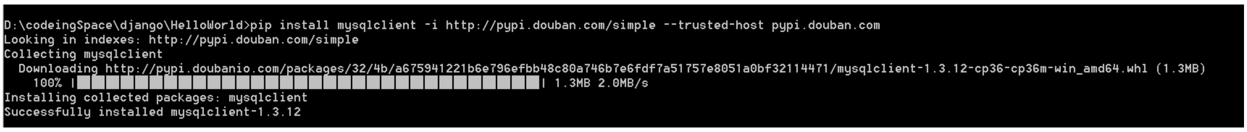
先在setting中配置好数据库



```
DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.mysql',  # 或者使用 mysql.connector.django
        'NAME': 'test',
        'USER': 'test',
        'PASSWORD': 'test123',
        'HOST':'localhost',
        'PORT':'3306',
    }
}
```

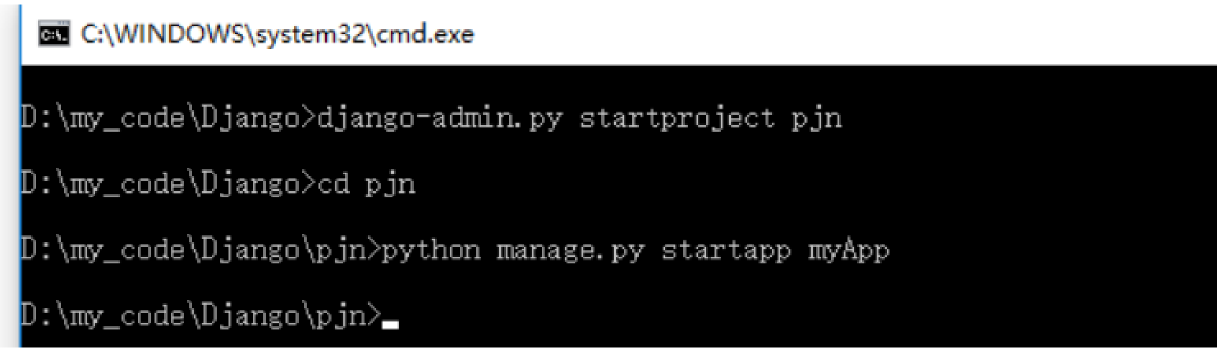
这一步记得安装驱动

pip install mysqlclient -i http://pypi.douban.com/simp;e --trusted-host pypi.douban.com



```
D:\codeingSpace\django\HelloWorld>pip install mysqlclient -i http://pypi.douban.com/simple --trusted-host pypi.douban.com
Looking in indexes: http://pypi.douban.com/simple
Collecting mysqlclient
  Downloading http://pypi.doubanio.com/packages/32/4b/a675941221b6e796efbb48c80a746b7e6fdf7a51757e8051a0bf32114471/mysqlclient-1.3.12-cp36-cp36m-win_amd64.whl (1.3MB)
    100% |████████████████████████████████| 1.3MB 2.0MB/s
Installing collected packages: mysqlclient
Successfully installed mysqlclient-1.3.12
```

创建app



C:\WINDOWS\system32\cmd.exe

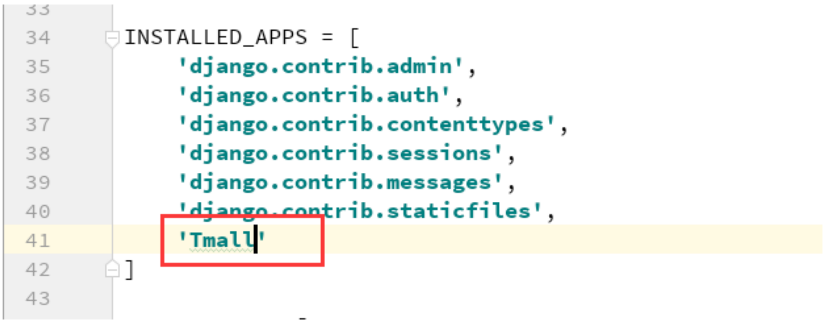
```
D:\my_code\Django>django-admin.py startproject pjn

D:\my_code\Django>cd pjn

D:\my_code\Django\pjn>python manage.py startapp myApp

D:\my_code\Django\pjn>_
```

生成模型：

python manage.py inspectdb

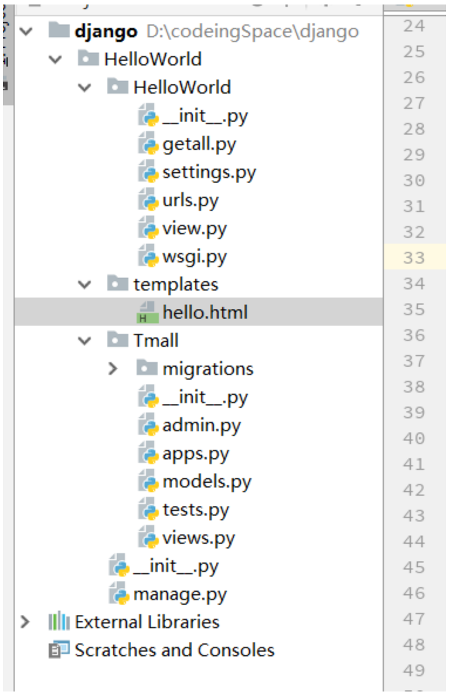
python manage.py inspectdb > models.py

在setting中添加已安装app



```
INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    'Tmall'
]
```

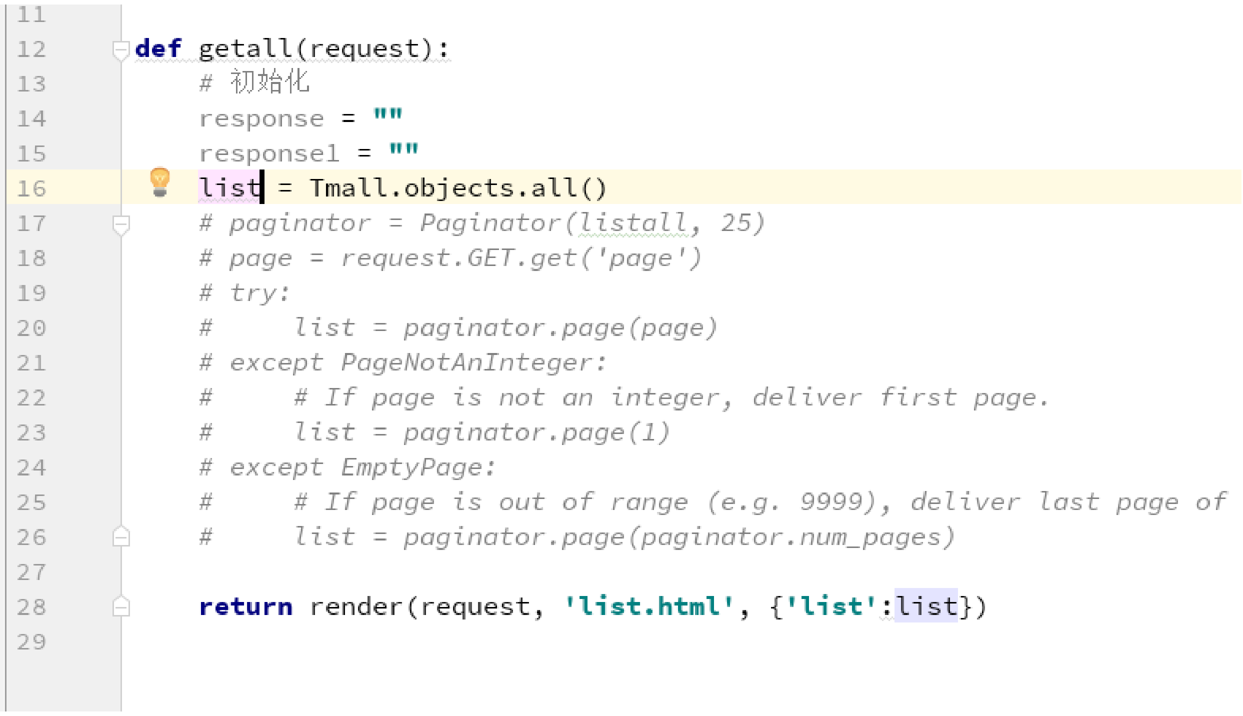
目前结构如下：



```
django D:\codeingSpace\django
  HelloWorld
    HelloWorld
      __init__.py
      getall.py
      settings.py
      urls.py
      view.py
      wsgi.py
    templates
      hello.html
    Tmall
      migrations
      __init__.py
      admin.py
      apps.py
      models.py
      tests.py
      views.py
    __init__.py
    manage.py
External Libraries
Scratches and Consoles
```

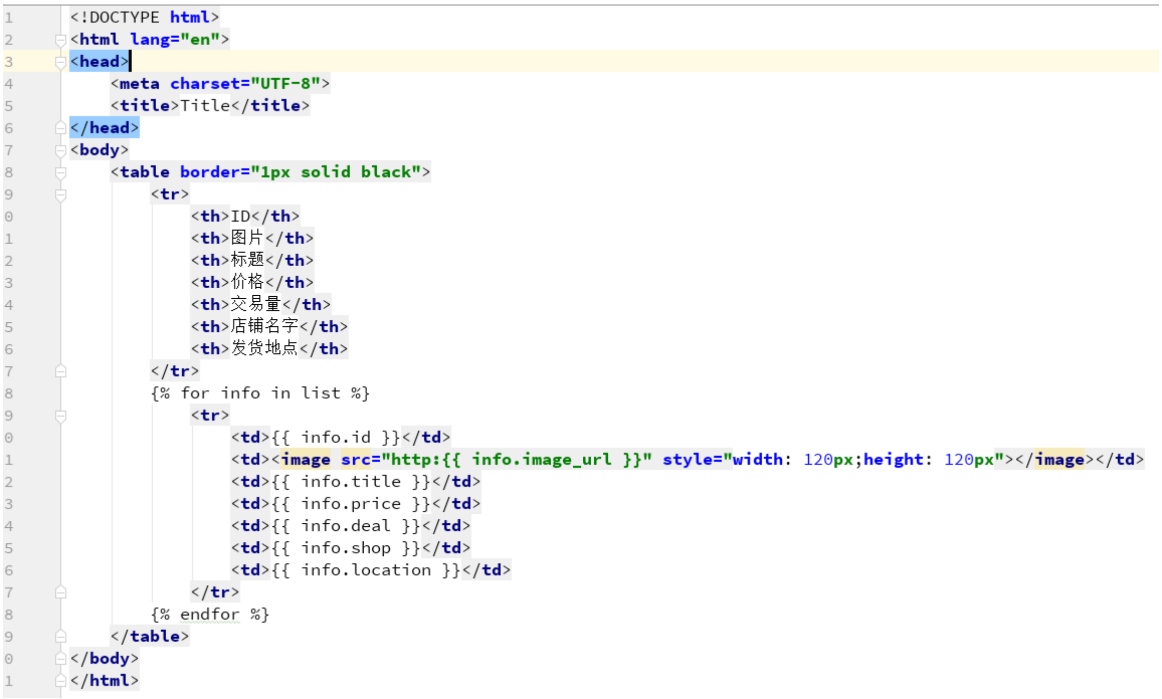
查询数据

控制器：



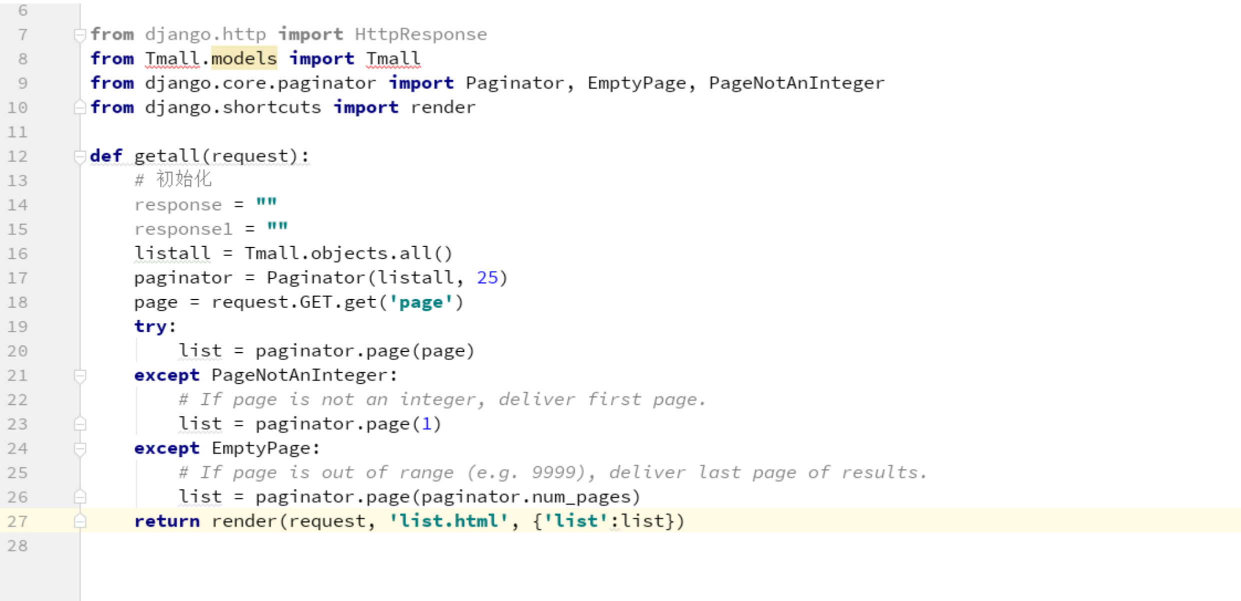
```python
def getall(request):
    # 初始化
    response = ""
    response1 = ""
    list = Tmall.objects.all()
    # paginator = Paginator(listall, 25)
    # page = request.GET.get('page')
    # try:
    #     list = paginator.page(page)
    # except PageNotAnInteger:
    #     # If page is not an integer, deliver first page.
    #     list = paginator.page(1)
    # except EmptyPage:
    #     # If page is out of range (e.g. 9999), deliver last page of
    #     list = paginator.page(paginator.num_pages)

    return render(request, 'list.html', {'list':list})
```

视图：

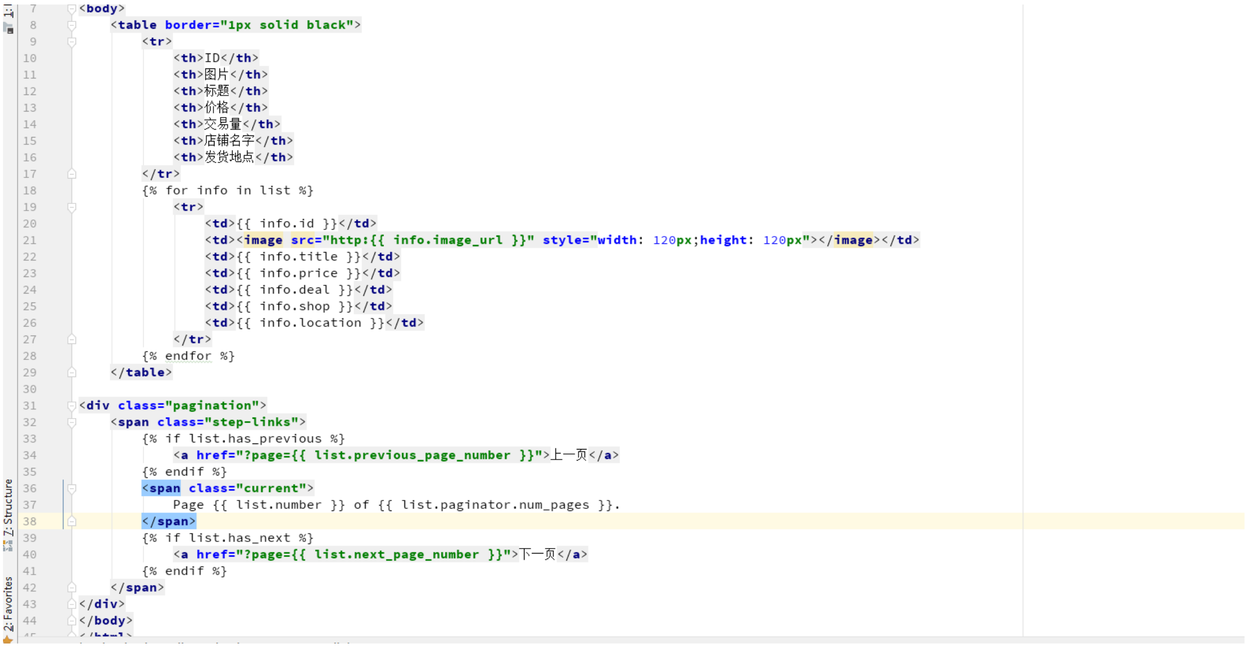


```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Title</title>
</head>
<body>
    <table border="1px solid black">
        <tr>
            <th>ID</th>
            <th>图片</th>
            <th>标题</th>
            <th>价格</th>
            <th>交易量</th>
            <th>店铺名字</th>
            <th>发货地点</th>
        </tr>
        {% for info in list %}
            <tr>
                <td>{{ info.id }}</td>
                <td><image src="http:{{ info.image_url }}" style="width: 120px;height: 120px"></image></td>
                <td>{{ info.title }}</td>
                <td>{{ info.price }}</td>
                <td>{{ info.deal }}</td>
                <td>{{ info.shop }}</td>
                <td>{{ info.location }}</td>
            </tr>
        {% endfor %}
    </table>
</body>
</html>
```

然后分页：



```python
from django.http import HttpResponse
from Tmall.models import Tmall
from django.core.paginator import Paginator, EmptyPage, PageNotAnInteger
from django.shortcuts import render

def getall(request):
    # 初始化
    response = ""
    response1 = ""
    listall = Tmall.objects.all()
    paginator = Paginator(listall, 25)
    page = request.GET.get('page')
    try:
        list = paginator.page(page)
    except PageNotAnInteger:
        # If page is not an integer, deliver first page.
        list = paginator.page(1)
    except EmptyPage:
        # If page is out of range (e.g. 9999), deliver last page of results.
        list = paginator.page(paginator.num_pages)
    return render(request, 'list.html', {'list':list})
```



```
<body>
    <table border="1px solid black">
        <tr>
            <th>ID</th>
            <th>图片</th>
            <th>标题</th>
            <th>价格</th>
            <th>交易量</th>
            <th>店铺名字</th>
            <th>发货地点</th>
        </tr>
        {% for info in list %}
            <tr>
                <td>{{ info.id }}</td>
                <td><image src="http:{{ info.image_url }}" style="width: 120px;height: 120px"></image></td>
                <td>{{ info.title }}</td>
                <td>{{ info.price }}</td>
                <td>{{ info.deal }}</td>
                <td>{{ info.shop }}</td>
                <td>{{ info.location }}</td>
            </tr>
        {% endfor %}
    </table>

<div class="pagination">
    <span class="step-links">
        {% if list.has_previous %}
            <a href="?page={{ list.previous_page_number }}">上一页</a>
        {% endif %}
        <span class="current">
            Page {{ list.number }} of {{ list.paginator.num_pages }}.
        </span>
        {% if list.has_next %}
            <a href="?page={{ list.next_page_number }}">下一页</a>
        {% endif %}
    </span>
</div>
</body>
</html>
```