
接收 POST 数据

一、接收 POST 数据

使用：

`v=request.POST.get(参数名称[,默认值])`

接收参数数据，在提交时没有指定的参数，值就是默认值（如果设置了默认值），或者出现异常（如果没有设置了默认值）。

(*) view.py

```
from django.http import HttpResponse

def respForm(request):
    response = HttpResponse()
    try:
        name=request.POST.get("name")
        sex=request.POST.get("sex","男")
        response.write("name="+name+"<br>")
        response.write("sex="+sex+"<br>")
    except Exception as e:
        response.write(e)
    return response

def showForm(request):
    s=''
    <html>
<body>
<h3>Hello 静态网页</h3>
<form method="post" action="/resp">
<input type="text" name="name" value="测试" />
<input type="text" name="sex" value="女" />
<input type="submit" value="提交" />
</form>
</body>
</html>
    ''
    return HttpResponse(s)
```

(*) urls.py

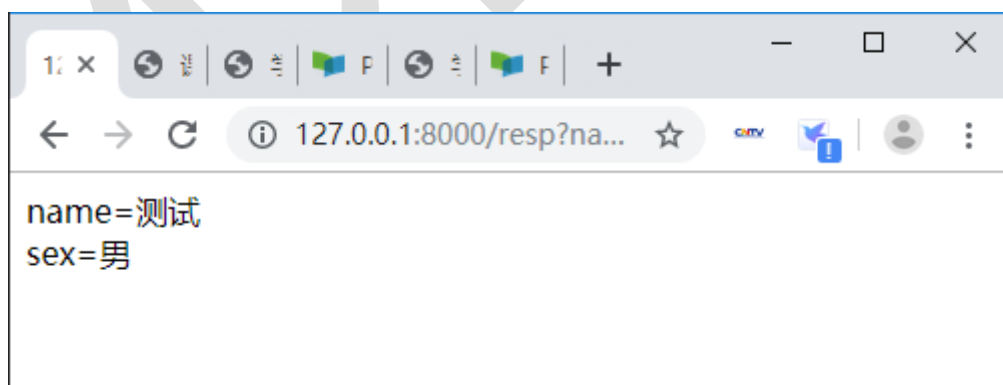
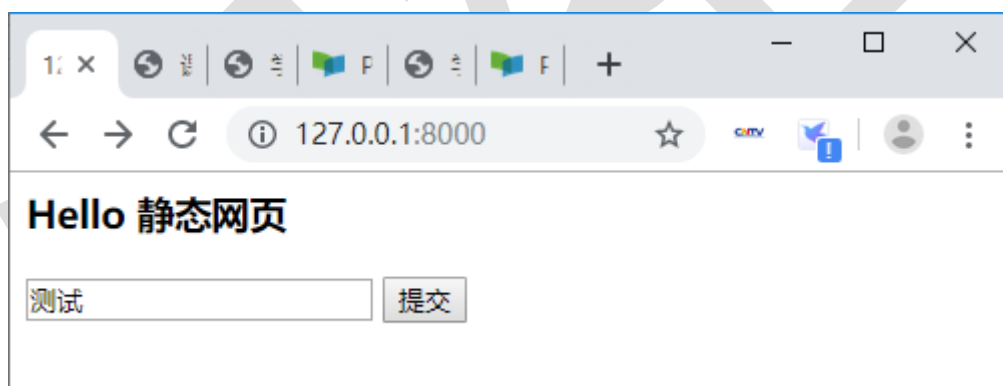
```
from django.conf.urls import url
from . import view
```

```
urlpatterns = [
    url(r'^$', view.showForm),
    url(r'^resp/$', view.respForm)
]
```

(*) settings.py

```
MIDDLEWARE = [
    'django.middleware.security.SecurityMiddleware',
    'django.contrib.sessions.middleware.SessionMiddleware',
    'django.middleware.common.CommonMiddleware',
    # 'django.middleware.csrf.CsrfViewMiddleware',
    'django.contrib.auth.middleware.AuthenticationMiddleware',
    'django.contrib.messages.middleware.MessageMiddleware',
    'django.middleware.clickjacking.XFrameOptionsMiddleware',
]
```

接受 POST 数据时必须设置#`django.middleware.csrf.CsrfViewMiddleware`',无效!



二、GET 提交

注意：使用：

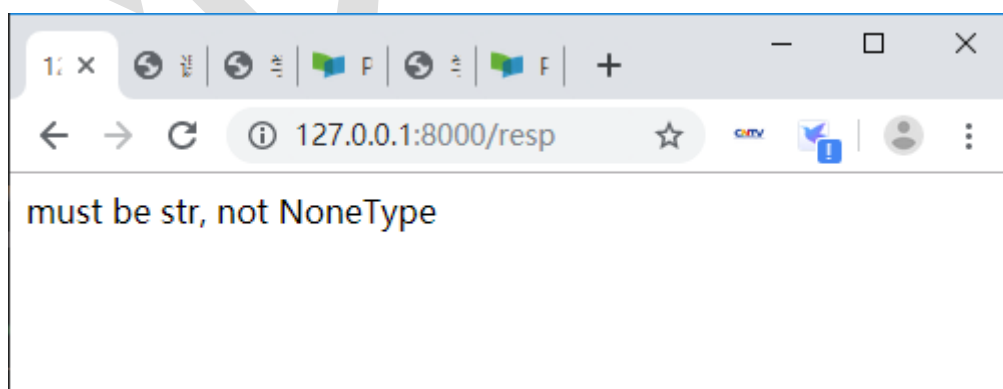
`v=request.GET.get(参数名称[,默认值])`
不能接收 `get` 提交数据。

例如：

```
from django.http import HttpResponse

def respForm(request):
    response = HttpResponse()
    try:
        name=request.POST.get("name")
        sex=request.POST.get("sex","男")
        response.write("name="+name+"<br>")
        response.write("sex="+sex+"<br>")
    except Exception as e:
        response.write(e)
    return response

def showForm(request):
    s=''
    <html>
<body>
<h3>Hello 静态网页</h3>
<form method="get" action="resp">
<input type="text" name="name" value="测试" />
<input type="submit" value="提交" />
</form>
</body>
</html>
    ...
    return HttpResponse(s)
```



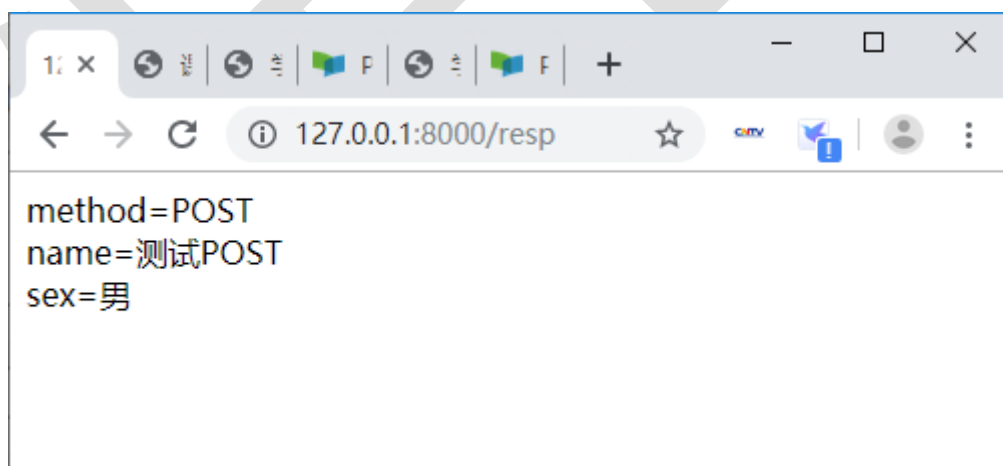
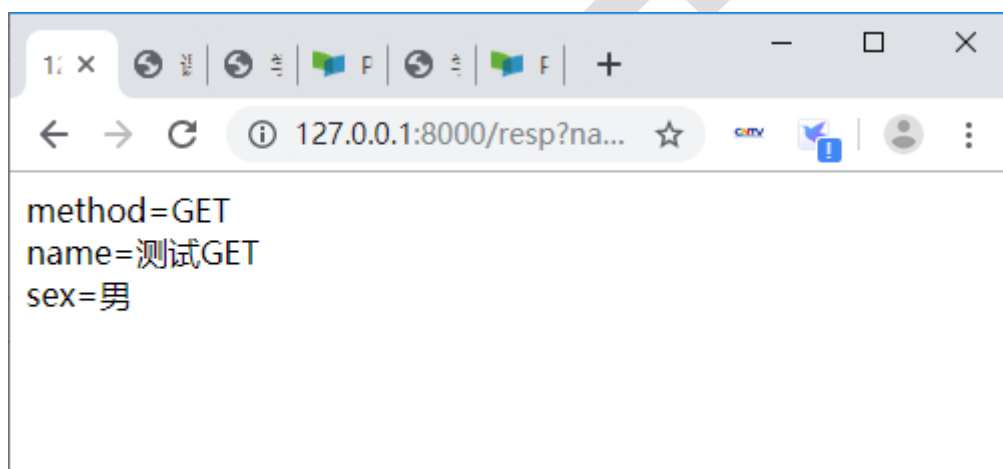
三、接收 GET 与 POST 提交数据

(*) view.py

```
from django.http import HttpResponseRedirect

def respForm(request):
    response = HttpResponseRedirect()
    if request.method=="POST":
        name=request.POST.get("name")
        sex=request.POST.get("sex","男")
    else:
        name=request.GET.get("name")
        sex=request.GET.get("sex","男")
    response.write("method="+request.method+"<br>")
    response.write("name="+name+"<br>")
    response.write("sex="+sex+"<br>")
    return response

def showForm(request):
    s='''
    <html>
<body>
<h3>Hello 静态网页</h3>
<form method="get" action="/resp">
<input type="text" name="name" value="测试 GET" />
<input type="submit" value="提交" />
</form>
<form method="post" action="/resp">
<input type="text" name="name" value="测试 POST" />
<input type="submit" value="提交" />
</form>
</body>
</html>
    '''
    return HttpResponseRedirect(s)
```



使用:

```
if request.method=="POST":  
    name=request.POST.get("name")  
    sex=request.POST.get("sex","男")  
else:  
    name=request.GET.get("name")  
    sex=request.GET.get("sex","男")
```

判断是什么提交，然后使用什么方法获取。

保秘