

問題 1. 如何不更改網址?

在做查詢、寫入等各種動作時，如果使用"GET"，將會更改網址

“GET”的設定出現在哪些地方？

index_html

```
{% extends 'base.html'%}

{% block title %}{{pageTitle}}{% endblock %}

{% block main %}
<form action='' method="GET">{% csrf_token %}

    搜尋：<input type="search" name="cName" placeholder="請輸入人名"><br>

    <input type="submit" value="送出">
    <!--<h1>{{mainTitle}}</h1-->
    <!--<p>{{mainContent}}</p-->
</form>
<p>{{urls}}</p>
    {% if unit %}
        <tr>
            <td>編號：{{unit.id}}</td><br>
            <td>姓名：{{unit.cName}}</td><br>
            <td>性別：{{unit.cSex}}</td><br>
            <td>生日：{{unit.cBirthday}}</td><br>
            <td>Email：{{unit.cEmail}}</td><br>
            <td>手機：{{unit.cPhone}}</td><br>
            <td>地址：{{unit.cAddr}}</td><br>
        </tr>
    {% else %}
    {% endif %}
    {% if cc %}
        <p>{{username}}，沒這個人</p>
    {% endif %}
{% endblock %}
```

views.py

```
def main(request):
    pageTitle="子網頁繼承"
    # mainTitle="段落標題"
    # mainContent="段落內文"
    # article1={"aTitle":"人名","aContent":"資料"}
    # article2={"aTitle":"文章標題","aContent":"文章2內文"}
    # articles=[article1]

    cc = False
    if request.method == "GET":
        search = request.GET
        username = search.get('cName')

        check = student.objects.filter(cName__exact=username)
        if check:

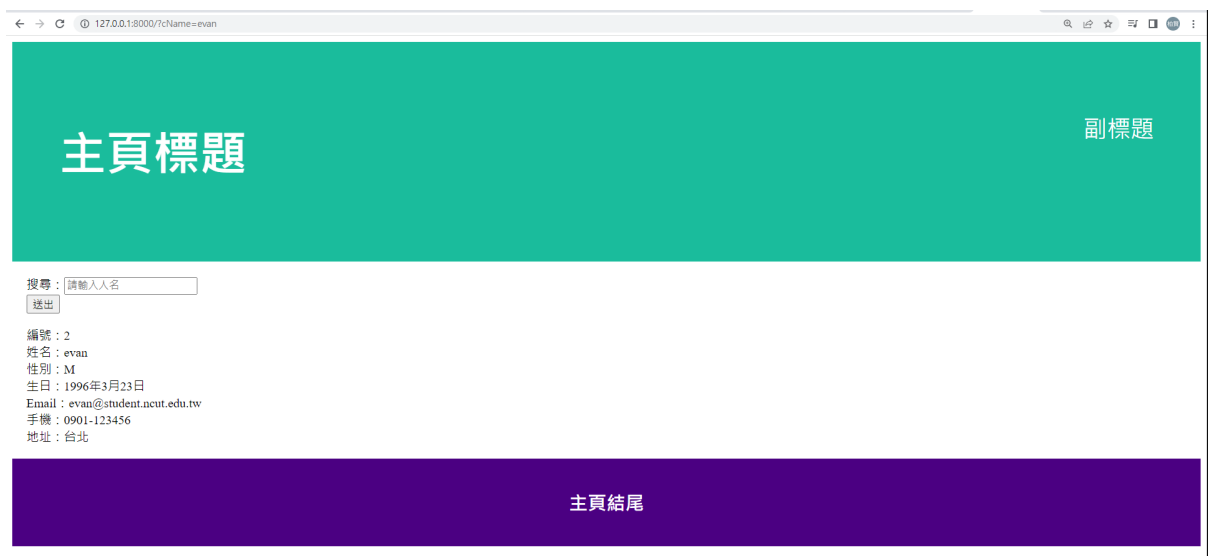
            unit = student.objects.get(cName=username)

        else:
            cc = True

    return render(request, 'index.html', locals())
```

如果使用"GET"，查詢資料後，網址會發生什麼改變？

搜尋框輸入evan，並且按下"送出"



注意看上方網址，原本網址是<http://127.0.0.1:8000/>

原網址後方出現搜尋的內容

網址變成<http://127.0.0.1:8000/?cName=evan>

網址的更動會暴露我們的操作內容，但是我們不希望讓他人看到操作內容，所以網址是不可以出現更動的

=====

操作之後，如何不更改網址？

A：把"GET"改成"POST"

index.html(之前建立過，繼承base.html)

```
{% extends 'base.html'%}

{% block title %}{{pageTitle}}{% endblock %}

{% block main %}
<form action='' method="POST">{% csrf_token %}

    搜尋：<input type="search" name="cName" placeholder="請輸入人名"><br>

    <input type="submit" value="送出">
    <!--<h1>{{mainTitle}}</h1-->
    <!--<p>{{mainContent}}</p-->
</form>

    {% if unit %}
        <tr>
            <td>編號：{{unit.id}}</td><br>
            <td>姓名：{{unit.cName}}</td><br>
            <td>性別：{{unit.cSex}}</td><br>
            <td>生日：{{unit.cBirthDay}}</td><br>
            <td>Email：{{unit.cEmail}}</td><br>
            <td>手機：{{unit.cPhone}}</td><br>
            <td>地址：{{unit.cAddr}}</td><br>
        </tr>
    {% else %}
    {% endif %}
    {% if cc %}
        <p>{{username}}，沒這個人</p>
    {% endif %}

{% endblock %}
```

views.py

```
from django.shortcuts import render, redirect
from django.http import HttpResponse
from datetime import datetime
from myapp.models import student
from django.http import Http404
from django.contrib import auth
from django.http import HttpResponseRedirect

def main(request):
    pageTitle="子網頁繼承"
    # mainTitle="段落標題"
    # mainContent="段落內文"
    # article1={"aTitle":"人名","aContent":"資料"}
    # article2={"aTitle":"文章標題","aContent":"文章2內文"}
    # articles=[article1]

    cc = False
    if request.method == "POST":

        username = request.POST['cName']

        check = student.objects.filter(cName__exact=username)
        if check:

            unit = student.objects.get(cName=username)

        else:
            cc = True

    return render(request, 'index.html', locals())
```

=====

參考：

Django學習紀錄 10.使用者互動與表單 - iT 邦幫忙::一起幫忙解決難題，拯救 IT 人的一天



之前有在用Django寫一些小網站，現在暑假想說再來複習一下之前買的這本書 於是我就把它寫成一系列的文章，也方便查語法 而且因為這本書大概是2014年出的，如今Django也已經出到2.多版 有些內容...

 <https://ithelp.ithome.com.tw/articles/10212441>

10.1.3 數據提交訊息-GET與POST

GET和POST是Http裡面最重要的兩個方法，這兩個方法負責提交使用者表單的數據，簡單來說，兩者在表現上的差別在:GET方法會透過在請求路徑後面添增查詢字符來提交數據，而POST方法是隱性地傳送數據的鍵值對，比如說我們在請求的時候，需要一個參數id

方法	範例URL
GET	<code>www.restaurants.com/restaurant/?id=1</code>
POST	<code>www.restaurants.com/restaurant</code>

這使得POST方法更適合拿來做資訊需隱密的請求，而GET方法則能漂亮的提供一個查詢相同頁面的URL(每次使用這個URL總是能獲得一個id=1的頁面，POST方法可能就不行了)，另外我們也會覺得GET方法適合用來取得數據和對應的頁面，POST方法適合利用提交數據去更動資料庫