



Network Programming

- 網路程式設計 -

資料新增、刪除和修改

授課教師：張珀銀 老師



05.

資料新增、刪除和修改

1. 表單資料的傳送與接收資料新增
2. 表單模型化
3. 資料刪除
4. 資料修改
5. 課堂練習

6

前一單元內容回顧

1

表單資料的傳送與接收資料新增

5-1 表單資料的傳送與接收

5.1.1 表單傳送資料的方法

屬性	說明
GET	表單資料將以字串的方式附加在網址的後面傳送，在網址後會以「？」符號開啟跟著表單中的資料，每個欄位間的值以「&」連接起來。
POST	表單資料將放置在 HTTP 標頭的方式傳送。

5.1.2 表單接收資料的方法

request 提供下列重要方法：

方法	用途	例如
GET	取得以 GET 方法傳送的表單欄位內容。	<code>request.GET["boardsubject"]</code>
POST	取得以 POST 方法傳送的表單欄位內容。	<code>request.POST["boardsubject"]</code>
method	取得表單傳送方式。	<code>if request.method=="POST":</code>

HTML Form 表單設計

HTML Form 語法

HTML 表單是由 `<form>` 標籤開始，結尾使用 `</form>` 標籤
兩個標籤之間就是放置表單元素的地方，

- `action` : form 會將網友填完的資料，傳送至 action 所設定的
資料傳送目的地，例如傳給一個頁面，進行資料的接收
- `method` : 是資料傳遞的方式，有 get 與 post 兩種方式

```
<form action="資料傳送目的地" method="資料傳遞方式">  
表單內容與表單元素  
</form>
```

HTML Form 表單設計

HTML Form 表單範例

```
<form action="get.php" method="get">  
姓名 : <input type='text'><br>  
電話 : <input type='text'><br>  
<p><input type='submit' value='送出表單'></p>  
</form>
```

姓名 :

電話 :

送出表單

HTML Form 表單設計

範例在 <form> 與 </form> 標籤之中使用了兩個表單元素，分別為文字輸入欄位（<input type='text'>）與送出表單的按鈕（submit），這樣的表單設計完成後，當填寫完資料並送出表單，系統就會自動以 get 的方式將資料傳送到 get.php 這支後端程式。

```
<form action="get.php" method="get">  
    姓名 : <input type='text'><br>  
    電話 : <input type='text'><br>  
    <p><input type='submit' value='送出表單'></p>  
</form>
```

HTML Form 表單設計

方法POST：傳送的資訊不呈現於網址

A screenshot of a web browser window. The address bar shows the URL `127.0.0.1:8000/post/`. The page content includes a form with a text input field labeled "請輸入姓名:" and a button labeled "送出資料". Below the form, the text "接收到的資料 : 321" is displayed.

方法GET：傳送的資訊直接呈現於網址

A screenshot of a web browser window. The address bar shows the URL `127.0.0.1:8000/post/?csrfmiddlewaretoken=COAchXBQNGRiWh5cjyorYRec3tWm3m2FjJOCVljKh1duBZAdOof4tKUoff09Dst0...`. The page content includes a form with a text input field labeled "請輸入姓名:" and a button labeled "送出資料". Below the form, the text "接收到的資料 : 111" is displayed.

HTML Form 表單設計

常見表單元素

表單元素	語法
文字輸入欄位 input type= text	<input type="text" name="" value="">
密碼輸入欄位 input type= password	<input type="password" name="" value="">
下拉選單 select option	<select name=""><option value=""></option></select>
選項按鈕 radio buttons	<input type="radio" name="" value="">
核取方塊 checkbox	<input type="checkbox" name="" value="">
文字輸入欄位 textarea	<textarea name=""></textarea>
隱藏欄位 input hidden	<input type="hidden" name="" value="">
顯示表單按鈕 button	<input type="button" name="">
送出表單按鈕 submit	<input type="submit" name="" value="送出表單">

表單元素可自由搭配成為一個完整的表單，每個元素都有他獨特的屬性以及語法規則，建議每個元素都弄清楚，以後使用上才能比較順手。

HTML Form 表單設計

文字輸入欄位 `input type= text` 範例

文字輸入欄位 `input type= text` 範例

```
姓名 : <input type="text" name="UserName">
```

呈現結果

姓名 :

HTML Form 表單設計

密碼輸入欄位 `input type= password`

密碼輸入欄位 `input type= password` 範例

```
請輸入您的密碼 : <input type="password" name="YourPassword">
```

呈現結果

請輸入您的密碼 :

HTML Form 表單設計

表單下拉選單 select option

下拉選單 select option 範例

```
<select name="select_one">  
<option value="1">選擇第一個選項</option>  
<option value="2">選擇第二個選項</option>  
<option value="3">選擇第三個選項</option>  
</select>
```

範例呈現結果

選擇第一個選項 ▾

HTML Form 表單設計

表單選項按鈕 radio buttons

表單選項按鈕 radio buttons 範例

```
<form>  
請選擇喜歡吃的水果<br><br>  
<input type="radio" name="Fruit" value="Apple"> 蘋果<br>  
<input type="radio" name="Fruit" value="Bananas"> 香蕉<br>  
</form>
```

範例呈現結果

請選擇喜歡吃的水果

- 蘋果
- 香蕉

HTML Form 表單設計

HTML參考資料：

- WEBTECH 網頁設計教學站 - HTML Form 表單設計
- [W3C SCHOOL](#)

範例：使用 POST 方式送傳送資料

在這個範例中，佈置了一個簡單的表單，並使用POST 方式傳送資料到目標網頁接收顯示。

The image consists of two side-by-side screenshots of a web browser window titled "表單傳送範例". Both screenshots show the URL `127.0.0.1:8000/post/`.
Left Screenshot: The input field contains the text "林三義". The "送出資料" button is visible and has a mouse cursor hovering over it. Below the input field, the text "接收到的資料：表單資料尚未送出！" is displayed.
Right Screenshot: The input field is empty. The "送出資料" button is visible. Below the input field, the text "接收到的資料：林三義" is displayed, indicating the data was successfully sent.

post.html

```
<html>
<head>
    <title>表單傳送範例</title>
</head>
<body>
<form action="/post/" method="POST" name="form1">
    {% csrf_token %} DJANGO框架提供防止CSRF攻擊機制
    <div>請輸入姓名 : <input type="text" name="username" />
    <input type="submit" value="送出資料" /></div>
    <div>接收到的資料 : {{mess}}</div>
</form>
</body>
</html>
```

views.py 運用view 控制流程

```
def post(request):
    if request.method == "POST": # 如果是以POST方式才處理
        mess = request.POST['username'] # 取得表單輸入資料
    else:
        mess = "表單資料尚未送出!"
    return render(request, "post.html", locals())
```

```
<form action="/post/" method="GET" name="form1">
    {% csrf_token %}
    <div>請輸入姓名：<input type="text" name="username" />
    <input type="submit" value="送出資料" /></div>
    <div>接收到的資料：{{mess}}</div>
</form>
```

CSRF 攻擊原理

- CSRF是一種常見的網路攻擊手法，全名是 Cross Site Request Forgery(跨站請求偽造)
- 大部分網站應用採用cookie或session的方式進行登入驗證，當通過登入驗證之後，網站就會給你一個通行證存在cookie或sesion中，代表之後的動作中都不需要重複驗證身分
- CSRF的攻擊鑽了這機制漏洞達到攻擊目的

CSRF 攻擊原理

Cookies就像是每個人身上的筆記本，當小明在A銀行登入之後，A銀行在小明的筆記本上蓋了個印章，接下來三個小時內小明的所有動作都不需要重複驗證身分。

但小明在這三小時內逛了其他網頁，不巧逛到駭客架設的惡意網頁，這個惡意網頁偷偷複製小明的cookie筆記本的印章，跑去打A銀行的轉帳API，請A銀行把小明的存款匯10萬元到駭客的帳戶，由於A銀行認印章不認人，A銀行就接受了這筆匯款請求。

只是登入了A銀行的帳戶，同時逛逛網拍、找找資料，小明就損失了10萬元！

CSRF 攻擊原理

普遍防禦CSRF的方法有兩種：

■ 1檢查 referer欄位 2加入驗證token

■ 1檢查 referer欄位

確認需求的來源，Http協定中有referer欄位記錄請求是從哪個網站發出來的，只要不是銀行網站發出來的請求，網站這邊一律不接受。

駭客若可偽裝成銀行來源，則小明仍然會被攻擊。

CSRF 攻擊原理

2加入驗證token

- CSRF的核心概念，就是銀行不知道發出請求的人是不是攻擊者。只要小明能提出一個唯一且保密的序號，攻擊者拿不到這個序號，自然就不能偽裝成小明，
- 這序號就是CSRF Token，這個Token是由銀行server產生，並且加密存在Session變數中，其他人無法仿造，只有透過Server給小明，並在一定時間內刷新。當小明想做任何交易的時候，銀行Server就會請小明提供CSRF Token，
- 如果不能提供，就代表這次的請求就是攻擊者，那銀行Server就不予理會。

5-2 資料新增

利用objects.create 方法可以新增一筆資料，語法：

```
記錄 = 資料表.objects.create( 欄位 1= 內容 1,...)
```

以 save 方法可以將該筆記錄寫入資料庫。例如：將unit 資料寫入資料庫。

```
unit.save()
```

```
def mypost1(request): # 新增資料，資料不作驗證
    if request.method == 'POST':
        cName = request.POST['cName']
        cSex = request.POST['cSex']
        cBirthday = request.POST['cBirthday']
        cEmail = request.POST['cEmail']
        cPhone = request.POST['cPhone']
        cAddr = request.POST['cAddr']

        unit = student.objects.create(
            cName=cName,
            cSex=cSex,
            cBirthday=cBirthday,
            cEmail=cEmail,
            cPhone=cPhone,
            cAddr=cAddr,
        )
```

範例POST1：資料新增

在這個範例中將表單資料和資料庫結合，在表單中輸入資料後，以 POST 方式傳送資料到資料庫建立一筆記錄。

開啟「127.0.0.1:8000/post1/」新增資料，資料新增完成後會切換到 <index.html> 網頁，圖中編號9 的資料為新增的資料。

顯示所有資料

顯示 student 資料表所有資料 新增一 新增二

編號	姓名	性別	生日	郵件帳號	電話	地址	編輯
1	李采茜	F	1987年4月4日	elvan@superstar.com	0976123456	台北市信義路643號	編輯一 編輯二 刪除
2	林木田	M	2001年8月9日	david@superstar.com	0978125643	台北市仁愛路43號	編輯一 編輯二 刪除
3	田出火	M	1998年12月30日	lily@superstar.com	0945678912	台北市忠孝東路12號	編輯一 編輯二 刪除
4	張美麗	F	2008年6月24日	beauty@superstar.com	0912563478	台北市信義路98號	編輯一 編輯二 刪除
5	鄭麗珠	F	1980年8月1日	joe@superstar.com	0956784321	台北市和平東路56號	編輯一 編輯二 刪除
6	黃竹林	M	1994年1月29日	john@superstar.com	0966665555	台北市南昌路65號	編輯一 編輯二 刪除
7	邱諒金	M	2005年4月23日	goan@superstar.com	0913427777	台北市信義路1008號	編輯一 編輯二 刪除
8	黃珠文	F	1992年4月25日	mary@superstar.com	0933338888	台北市南海路13號	編輯一 編輯二 刪除
9	高先進	M	2001年1月12日	test@test.com	02-23456789	台北市信義路123號	編輯一 編輯二 刪除

資料表新增(一)

student 資料表新增(一)

姓名	高先進
性別	<input checked="" type="radio"/> 男生 <input type="radio"/> 女生
生日	2001-01-12
郵件帳號	test@test.com
電話	02-23456789
地址	台北市信義路123號

請輸入資料(資料不作驗證)

2

表單模型化

5-3 表單模型化

建立表單模型

在from.py中建立類別 PostForm

studentsapp\form.py

```
1 from django import forms  
2  
3 class PostForm(forms.Form):  
4     cName = forms.CharField(max_length=20,initial='')  
5     cSex = forms.CharField(max_length=2,initial='M')  
6     cBirthday = forms.DateField()  
7     cEmail = forms.EmailField(max_length=100,initial='',required=False)  
8     cPhone = forms.CharField(max_length=50,initial='',required=False)  
9     cAddr = forms.CharField(max_length=255,initial='',required=False)
```

建立表單模型物件

```
postform = PostForm()
```

輸出表單

首先建立一個空白物件，並將它傳送給顯示樣版<postform.html>。

studentsapp\views.py

```
略.....  
def postform(request):  
    略.....  
    postform = PostForm()  
    return render(request, "postform.html", locals())
```

在 <postform.html> 樣版中以 {{postform}} 就可以顯示此空白的 postform 物件。

templates\postform.html

```
<form action"." method="POST" name="form1">  
    {% csrf_token %}  
    {{postform}}  
</form>
```

範例：基本表單驗證

利用表單模型物件，建立基本的表單驗證。

開啟「127.0.0.1:8000/postform/」，以表格方式顯示新增資料的畫面，輸入的資料會被驗證。

The screenshot shows a web browser window with the URL `127.0.0.1:8000/postform/` in the address bar. The page title is "基本表單驗證". Below the title is a table with six rows, each containing a label and an input field:

Cname :	林三田
Csex :	M
Cbirthday :	2001-03-15
Cemail :	
Cphone :	
Caddr :	

表單樣式客製化

templates\post2.html

```
<form action"." method="POST" name="form1">
    {% csrf_token %}
    <p class="css1">姓名 : {{postform.cName}}</p>
    <p class="css2">郵件帳號 : {{postform.cEmail}}</p>
</form>
```

表單驗證

可以用is_valid() 方法檢查輸入資料是否通過驗證，通過驗證後就可以cleaned_data取得表單輸入資料。例如：通過驗證後取得cName 欄位資料。

```
if postform.is_valid():          # 通過 forms 驗證
    cName = postform.cleaned_data['cName'] # 取得表單輸入資料
```

範例：資料新增並驗證

和上一範例相同，在表單中輸入資料後，以POST方式傳送資料到資料庫建立一筆記錄，輸入資料必須作驗證。

開啟「127.0.0.1:8000/post2/」新增資料，左下圖為姓名欄位未填，右下圖為郵件帳號格式不符。

資料表新增(二) [Python] Dj... 127.0.0.1:8000/post2/

student 資料表新增(二)

姓名	<input type="text"/>
性別	<input checked="" type="radio"/> 男生 <input type="radio"/> 女生
生日	<input type="text"/>
郵件帳號	<input type="text"/>
電話	<input type="text"/>
地址	<input type="text"/>
<input type="button" value="送出"/> <input type="button" value="重設"/>	

姓名、性別、生日必須輸入！

資料表新增(二) [Python] Dj... 127.0.0.1:8000/post2/

student 資料表新增(二)

姓名	林三田
性別	<input checked="" type="radio"/> 男生 <input type="radio"/> 女生
生日	2001-01-12
郵件帳號	<input type="text"/> user@
! 請輸入「@」後面的部分。「user@」不是完整	
地址	<input type="text"/>
<input type="button" value="送出"/> <input type="button" value="重設"/>	

姓名、性別、生日必須輸入！

3

資料刪除

5-4 資料刪除

例如：刪除「id=1」的記錄

```
id=1  
unit = student.objects.get(id=id)  
unit.delete()
```

範例：資料刪除

輸入「127.0.0.1:8000/delete/id/」後(例如：id=29) , 會依據id 刪除指定的資料，若id 存在會直接刪除該筆資料，並切換到<index.html> 頁面，否則會出現下面畫面等待小明重新輸入編號。



4

資料修改

5-5 資料修改

利用save方法可以將資料寫回資料庫，如此就可以修改資料庫。

例如：將編號為「1」的cName欄位內容更改為「李天龍」並儲存。

```
unit = student.objects.get(id=1)
unit.cName="李天龍"
unit.save() # 寫入資料庫
```

範例：資料修改(一)

輸入「127.0.0.1:8000/edit/id/」後會依據 id 取得指定的資料，更改資料後按 送出 鈕，即會更新資料庫中該筆資料並重導到<index.html> 頁面，若id 不存在會顯示空白表單並出現「此 id 不存在！」訊息。

A screenshot of a web browser window titled "資料表修改(一)". The URL bar shows "127.0.0.1:8000/edit/8/". The page title is "student 資料表修改(一)". The form contains the following fields:

姓名	黃珠文
性別	<input type="radio"/> 男生 <input checked="" type="radio"/> 女生
生日	1992-04-25
郵件帳號	mary@superstar.com
電話	0933338888
地址	台北市南海路13號

At the bottom are two buttons: "送出" and "重設". A cursor arrow points to the "送出" button.

A screenshot of a web browser window titled "資料表修改(一)". The URL bar shows "127.0.0.1:8000/edit/18/". The page title is "student 資料表修改(一)". The form is identical to the one on the left, but it displays an error message at the bottom: "此 id 不存在！".

範例：整合範例

輸入「127.0.0.1:8000」或「127.0.0.1:8000/index/」都會執行<index.html> 檔，以表格方式顯示所有的資料，按下 **新增一**、**新增二** 鈕分別執行「/post1/」、「/post2/」新增資料，按下 **編輯一**、**編輯二** 鈕分別執行「/edit/」、「/edit2/」編輯該筆資料，按下 **刪除** 鈕則刪除該筆資料。



The screenshot shows a web browser window with the URL '127.0.0.1:8000'. The page title is '顯示 student 資料表所有資料' with underlined links for '新增一' and '新增二'. Below the title is a table with 9 rows of student data. Each row contains fields: 编号 (ID), 姓名 (Name), 性别 (Gender), 生日 (Birthday), 邮件帳號 (Email), 電話 (Phone), 地址 (Address), and 编辑 (Edit) buttons. The edit buttons are labeled '編輯一' and '編輯二'.

編號	姓名	性別	生日	郵件帳號	電話	地址	編輯
1	李采茜	F	1987年4月4日	elvan@superstar.com	0976123456	台北市信義路643號	編輯一 編輯二 刪除
2	林木田	M	2001年8月9日	david@superstar.com	0978125643	台北市仁愛路43號	編輯一 編輯二 刪除
3	田出火	M	1998年12月30日	lily@superstar.com	0945678912	台北市忠孝東路12號	編輯一 編輯二 刪除
4	張美麗	F	2008年6月24日	beauty@superstar.com	0912563478	台北市信義路98號	編輯一 編輯二 刪除
5	鄭麗珠	F	1980年8月1日	joe@superstar.com	0956784321	台北市和平東路56號	編輯一 編輯二 刪除
6	黃竹林	M	1994年1月29日	john@superstar.com	0966665555	台北市南昌路65號	編輯一 編輯二 刪除
7	邱諒金	M	2005年4月23日	goan@superstar.com	0913427777	台北市信義路1008號	編輯一 編輯二 刪除
8	黃珠文	F	1992年4月25日	mary@superstar.com	0933338888	台北市南海路13號	編輯一 編輯二 刪除
9	高先進	M	2001年1月12日	test@test.com	02-23456789	台北市信義路123號	編輯一 編輯二 刪除

範例：資料修改(二)

在<index.html> 頁面按 **編輯二** 連結，依據id 取得該筆資料，更改資料後按 **送出** 鈕，即會更新資料庫該筆資料並切換到<index.html> 頁面。

The screenshot displays two browser windows. The left window, titled '顯示所有資料' (Display All Data), shows a table of student information with 9 rows. The right window, titled '資料表修改(二)' (Data Table Modification (2)), shows a form for editing student ID 1. The '編輯二' (Edit 2) link in the left window has been clicked, which triggered the right window to appear.

編號	姓名	性別	生日	郵件帳號	電話	地址
1	李采茜	F	1987年4月4日	elvan@superstar.com	0976123456	台北市
2	林木田	M	2001年8月9日	david@superstar.com	0978125643	台北市
3	田出火	M	1998年12月30日	lily@superstar.com	0945678912	台北市
4	張美麗	F	2008年6月24日	beauty@superstar.com	0912563478	台北市
5	鄭麗珠	F	1980年8月1日	joe@superstar.com	0956784321	台北市
6	黃竹林	M	1994年1月29日	john@superstar.com	0966665555	台北市
7	邱諒金	M	2005年4月23日	goan@superstar.com	0913427777	台北市
8	黃珠文	F	1992年4月25日	mary@superstar.com	0933338888	台北市
9	高先進	M	2001年1月12日	test@test.com	02-23456789	台北市信義路123號

student 資料表修改(二)

姓名	高先進
性別M	<input checked="" type="radio"/> 男生 <input type="radio"/> 女生
生日	2001-01-12
郵件帳號	test@test.com
電話	02-23456789
地址	台北市信義路123號

送出 重設

編輯一 編輯二 刪除



E

Exercise 1: post

完成post1

- 建立views.py檔中的 post函數。
- 設定 urls.py
- 建立post.html

urls.py

```
from django.contrib import admin
from django.urls import path
from studentsapp.views import index, postform, post, post1, post2, mypost1

urlpatterns = [
    path('admin/', admin.site.urls),
    path('index/', index),
    path('post/', post), path('post/')
    path('post1/', post1),
    path('post2/', post2),
    path('postform/', postform),
    path('mypost1/', mypost1),
    path('mypf/', mypost1),
]
```

post.html

```
1 <html>
2   <head>
3     <title>表單傳送範例</title>
4   </head>
5   <body>
6     <form action="/post/" method="POST" name="form1">
7       {% csrf_token %}
8       <div>請輸入姓名：<input type="text" name="username" />
9       <input type="submit" value="送出資料" /></div>
10      <div>接收到的資料：{{mess}}</div>
11    </form>
12  </body>
13 </html>
```

views.py

```
24     def post(request):
25         if request.method == "POST": # 如果是以POST方式才處理
26             print('POST')
27             mess = request.POST['username'] # 取得表單輸入資料
28         else:
29             print('GET')
30             mess = "表單資料尚未送出!"
31             # mess = request.GET['username']
32
33     return render(request, "post.html", locals())
```

Results



請輸入姓名 : 送出資料

接收到的資料 : 表單資料尚未送出!



請輸入姓名 : 送出資料

接收到的資料 : myname



請輸入姓名 : 送出資料

接收到的資料 : S01234



E

Exercise 2: post1

動手做看看

完成post1

- 建立views.py檔中的 post1 函數。
- 設定 urls.py
- 建立post1.html

views.py

```
59 def post1(request): # 新增資料，資料不作驗證
60     if request.method == "POST": # 如果是以POST方式才處理
61         cName = request.POST['cName'] # 取得表單輸入資料
62         cSex = request.POST['cSex']
63         cBirthday = request.POST['cBirthday']
64         cEmail = request.POST['cEmail']
65         cPhone = request.POST['cPhone']
66         cAddr = request.POST['cAddr']
67         # 新增一筆記錄
68         unit = student.objects.create(cName=cName,
69                                         cSex=cSex,
70                                         cBirthday=cBirthday,
71                                         cEmail=cEmail,
72                                         cPhone=cPhone,
73                                         cAddr=cAddr)
74         unit.save() # 寫入資料庫
75         return redirect('/index/')
76     else:
77         message = '請輸入資料(資料不作驗證)'
78         return render(request, "post1.html", locals())
```

post1.html

```
1  {% extends "base.html" %}  
2  {% block title %}<title>資料表新增(一)</title>{% endblock %}  
3  {% block content %}  
4      <h2>student 資料表新增(一)</h2>  
5      <form action="." method="POST" name="form1">  
6          {% csrf_token %}  
7          <table border="1" cellspacing="1" cellpadding="4">  
8              <tr>  
9                  <th>姓名</th>  
10                 <td><input type="text" id="cName" name="cName"/></td>  
11             </tr>  
12             <tr>  
13                 <th>性別</th>  
14                 <td>  
15                     <input name="cSex" type="radio" id="cSex1" value="M" checked="checked"/>  
16                     男生  
17                     <input name="cSex" type="radio" id="cSex2" value="F"/>  
18                     女生  
19                 </td>  
20             </tr>  
21             <tr>  
22                 <th>生日</th>  
23                 <td><input type="text" id="cBirthday" name="cBirthday"/></td>  
24             </tr>  
25             <tr>  
26                 <th>郵件帳號</th>  
27                 <td><input type="text" id="cEmail" name="cEmail"/></td>  
28             </tr>
```

post1.html

```
28
29
30      </tr>
31      <tr>
32          <th>電話</th>
33          <td><input type="text" id="cPhone" name="cPhone"/></td>
34      </tr>
35      <tr>
36          <th>地址</th>
37          <td><input type="text" id="cAddr" name="cAddr"/></td>
38      </tr>
39      <tr>
40          <th colspan="2" align="center">
41              <input type="submit" name="button" id="button" value="送出">
42              <input type="reset" name="button2" id="button2" value="重設"></th>
43      </tr>
44  </table>
45  <span style="color:red">{{ message }}</span>
46
47  </form>
48
49  {% endblock %}
```

Result

student 資料表新增(一)

姓名	Jack
性別	<input checked="" type="radio"/> 男生 <input type="radio"/> 女生
生日	2000-01-01
郵件帳號	email
電話	phone
地址	address
<input type="button" value="送出"/> <input type="button" value="重設"/>	

試看看直接從127.0.0.1/post1/
進入網頁



student 資料表新增(一)

姓名	
性別	<input checked="" type="radio"/> 男生 <input type="radio"/> 女生
生日	
郵件帳號	
電話	
地址	
<input type="button" value="送出"/> <input type="button" value="重設"/>	

顯示 student 資料表所有資料 [新增一](#) [新增二](#)

Method = POST
, forward to index.html

編號	姓名	性別	生日	郵件帳號	電話	地址	編輯
1	李采茜	F	April 4, 1987	elvan@superstar.com	0976123456	台北市信義路643號	編輯一 編輯二 刪除
2	林木田	M	Aug. 9, 2001	david@superstar.com	0978125643	台北市仁愛路43號	編輯一 編輯二 刪除
3	田出火	M	Dec. 30, 1998	lily@superstar.com	0945678912	台北市忠孝東路12號	編輯一 編輯二 刪除
4	張美麗	F	June 24, 2008	beauty@superstar.com	0912563478	台北市信義路98號	編輯一 編輯二 刪除
5	鄭麗珠	F	Aug. 1, 1980	joe@superstar.com	0956784321	台北市和平東路56號	編輯一 編輯二 刪除
6	黃竹林	M	Jan. 29, 1994	john@superstar.com	0966665555	台北市南昌路65號	編輯一 編輯二 刪除
7	邱諒金	M	April 23, 2005	goan@superstar.com	0913427777	台北市信義路1008號	編輯一 編輯二 刪除
8	黃珠文	F	April 25, 1992	mary@superstar.com	0933338888	台北市南海路13號	編輯一 編輯二 刪除
9	高先進	M	Jan. 12, 2001	test@test.com	02-23456789	台北市信義路123號	編輯一 編輯二 刪除
13	jack	M	Jan. 1, 2000	gm@gmail.com	123	this is address	編輯一 編輯二 刪除
14	123	F	Jan. 1, 1990	ez@gmail.com	0932	this is address	編輯一 編輯二 刪除
15	Jess	F	Jan. 1, 1990				編輯一 編輯二 刪除
16	Jess	F	Jan. 1, 1990				編輯一 編輯二 刪除
17	Jess	F	Jan. 1, 1990				編輯一 編輯二 刪除
18	test	M	Sept. 9, 1990				編輯一 編輯二 刪除
19	jack	M	Jan. 1, 2000	EMAIL	09123	ADDRESS	編輯一 編輯二 刪除



Exercise 3:

驗證表單 PostForm

forms.py

A screenshot of a code editor showing the contents of the `forms.py` file. The editor has a tab bar at the top with several tabs: `urls.py`, `views.py`, `forms.py` (which is currently selected), `post2.html`, `post1.html`, and `post.html`. The code itself is a Python class definition for a form:

```
1 from django import forms
2
3
4 class PostForm(forms.Form):
5     cName = forms.CharField(max_length=20, initial='')
6     cSex = forms.CharField(max_length=2, initial='M')
7     cBirthday = forms.DateField()
8     cEmail = forms.EmailField(max_length=100, initial='', required=False)
9     cPhone = forms.CharField(max_length=50, initial='', required=False)
10    cAddr = forms.CharField(max_length=255, initial='', required=False)
```

views.py

```
52     def postform(request):
53         postform = PostForm()
54         return render(request, 'postform.html', locals())
```

postform.html

```
1 1  {% extends 'base.html' %}  
2 2  {% block title %}  
3 3  <title>基本表單驗證</title>  
4 4  {% endblock %}  
5 5  {% block content %}  
6 6  <h2>基本表單驗證</h2>  
7 7  <form action="." method="post">  
8 8  {% csrf_token %}  
9 9  <table border="1" cellpadding="1" cellspacing="4">  
10 10  {{ postform }}  
11 11  </table>  
12 12  </form>  
13 13  {% endblock %}
```

Results

基本表單驗證

Cname:	<input name="name" type="text"/>
Csex:	M
Cbirthday:	2020-01-01
Cemail:	<input name="email" type="text"/>
Cphone:	<input name="phone" type="text"/>
Caddr:	<input name="address" type="text"/>

驗證</title>

```
11 <h2>基本表單驗證</h2>
12 <form action="" method="post">
13     <input type="hidden" name="csrfmiddlewaretoken" value="hr5nBMte8R0o4vbPCd3H3Z0UisvKtp0busWDeqExZ1dbdpTT3iIrj08WsZg8cA7"
14     <table border="1" cellpadding="1" cellspacing="4">
15         <tr>
16             <th><label for="id_cName">Cname:</label></th>
17             <td>
18                 <input type="text" name="cName" maxlength="20" required id="id_cName">
19             </td>
20         </tr>
21         <tr>
22             <th><label for="id_cSex">Csex:</label></th>
23             <td>
24                 <input type="text" name="cSex" value="M" maxlength="2" required id="id_cSex">
25             </td>
26         </tr>
27     </table>
28 </form>
29
30
31
32
33
34
```



E

Exercise 4: Post2 自訂表單及驗證

urls.py

```
20     urlpatterns = [
21         path('admin/', admin.site.urls),
22         path('listone/', views.listone),
23         path('listall/', views.listall),
24         path('', views.index),
25         path('index/', views.index),
26
27         path('post/', views.post), # POST 傳送表單
28         path('post1/', views.post1), # 資料新增，資料不驗證
29         path('post2/', views.post2), # 資料新增，資料作驗證
30
31         path('delete/<int:id>/', views.delete),
32
33         path('edit/<int:id>/', views.edit), # 由 瀏覽器 開啟
34         path('edit/<int:id>/<str:mode>', views.edit), # 由 edit.html 按 送出 鈕
35
36         path('edit2/<int:id>/<str:mode>', views.edit2),
37         path('postform/', views.postform), # 表單驗證
38     ]
```

views.py

```
110     def post2(request): # 新增資料，資料必須驗證
111         if request.method == "POST": # 如果是以POST方式才處理
112             postform = PostForm(request.POST) # 建立forms物件
113             if postform.is_valid(): # 通過forms驗證
114                 cName = postform.cleaned_data['cName'] # 取得表單輸入資料
115                 cSex = postform.cleaned_data['cSex']
116                 cBirthday = postform.cleaned_data['cBirthday']
117                 cEmail = postform.cleaned_data['cEmail']
118                 cPhone = postform.cleaned_data['cPhone']
119                 cAddr = postform.cleaned_data['cAddr']
120                 # 新增一筆記錄
121                 unit = student.objects.create(
122                     cName=cName,
123                     cSex=cSex,
124                     cBirthday=cBirthday,
125                     cEmail=cEmail,
126                     cPhone=cPhone,
127                     cAddr=cAddr
128                 )
129                 unit.save() # 寫入資料庫
130                 message = '已儲存...'
131                 return redirect('/index/')
132             else:
133                 message = '驗證碼錯誤！'
134         else:
135             message = '姓名、性別、生日必須輸入！'
136             postform = PostForm()
137             return render(request, "post2.html", locals())
```

views.py

```
110     def post2(request): # 新增資料，資料必須驗證
111         if request.method == "POST": # 如果是以POST方式才處理
112             postform = PostForm(request.POST) # 建立forms物件
113             if postform.is_valid(): # 通過forms驗證
114                 cName = postform.cleaned_data['cName'] # 取得表單輸入資料
115                 cSex = postform.cleaned_data['cSex']
116                 cBirthday = postform.cleaned_data['cBirthday']
117                 cEmail = postform.cleaned_data['cEmail']
118                 cPhone = postform.cleaned_data['cPhone']
119                 cAddr = postform.cleaned_data['cAddr']
120                 # 新增一筆記錄
121                 unit = student.objects.create(
122                     cName=cName,
123                     cSex=cSex,
124                     cBirthday=cBirthday,
125                     cEmail=cEmail,
126                     cPhone=cPhone,
127                     cAddr=cAddr)
128                 )
129                 unit.save() # 寫入資料庫
130                 message = '已儲存...'
131             return redirect('/index/')
```

views.py

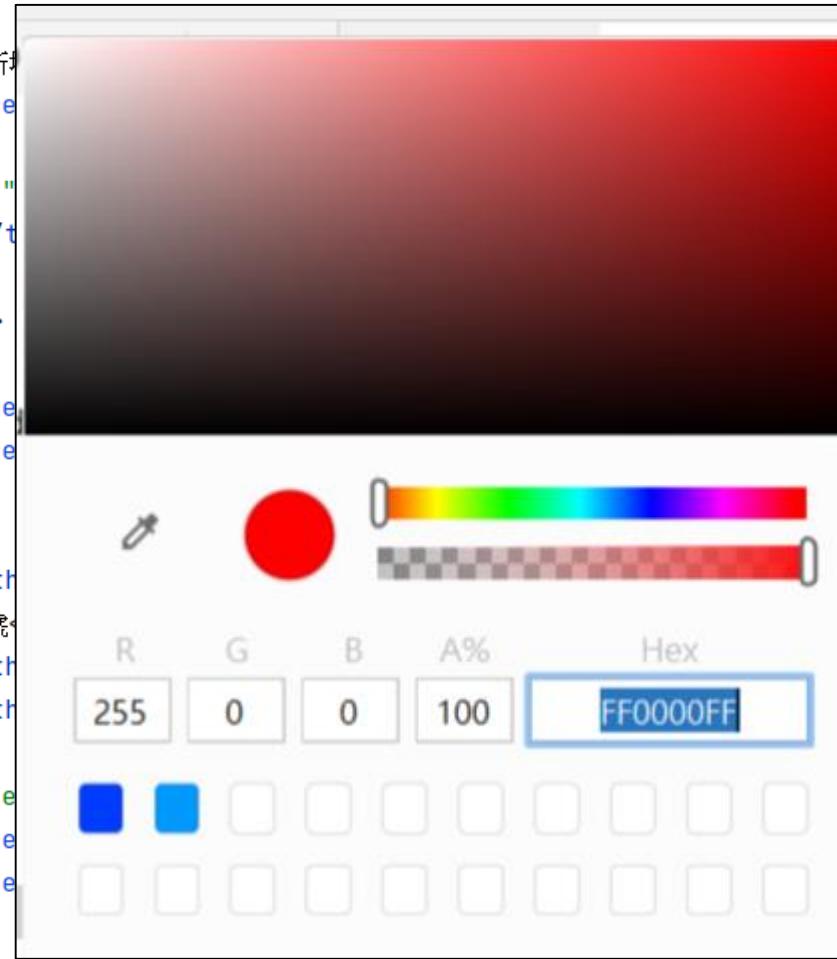
```
131     return redirect('/index/')
132 else:
133     message = '驗證碼錯誤！'
134 else:
135     message = '姓名、性別、生日必須輸入！'
136     postform = PostForm()
137     return render(request, "post2.html", locals())
```

post2.html

```
1  {% extends "base.html" %}|  
2  {% block title %}<title>資料表新增(二)</title>{% endblock %}  
3  {% block content %}  
4      <h2>student 資料表新增(二)</h2>  
5      <form action=". " method="POST" name="form1">  
6          {% csrf_token %}  
7          <table border="1" cellspacing="1" cellpadding="4">  
8              <tr><th>姓名</th><td>{{postform.cName}}</td></tr>  
9              <tr>  
10                 <th>性別</th>  
11                 <td>  
12                     <input name="cSex" type="radio" id="cSex1" value="M" checked="checked" />男生  
13                     <input name="cSex" type="radio" id="cSex2" value="F" />女生  
14                 </td>  
15             </tr>  
16             <tr><th>生日</th><td>{{postform.cBirthday}}</td></tr>  
17             <tr><th>郵件帳號</th><td>{{postform.cEmail}}</td></tr>  
18             <tr><th>電話</th><td>{{postform.cPhone}}</td></tr>  
19             <tr><th>地址</th><td>{{postform.cAddr}}</td></tr>  
20             <tr>  
21                 <th align="center" >  
22                     <input type="submit" name="button" id="button" value="送出">  
23                     <input type="reset" name="button2" id="button2" value="重設"></th>  
24             </tr>  
25         </table>  
26         <span style="color:red">{{message}}</span>  
27     </form>  
28  {% endblock %}
```

post2.html

```
1  {% extends "base.html" %}  
2  {% block title %}<title>資料表新增(二)</title>{% endblock %}  
3  {% block content %}  
4      <h2>student 資料表新增</h2>  
5      <form action=". " method="post">  
6          {% csrf_token %}  
7          <table border="1" style="width:100%; border-collapse: collapse; text-align: center;">  
8              <tr><th>姓名</th><th>性別</th><th>生日</th><th>郵件帳號</th><th>電話</th><th>地址</th><th align="center" style="width: 10%;">刪除9              <tr>  
10                 <th>性別</th>  
11                 <td>  
12                     <input name="gender" type="radio" value="男" checked="" />男  
13                     <input name="gender" type="radio" value="女" />女  
14                 </td>  
15             </tr>  
16             <tr><th>生日</th><td>  
17                 <input type="text" value="1990-01-01" style="width: 100%; height: 25px; border: 1px solid #ccc; padding: 5px; font-size: 14px; margin-bottom: 5px;"/>  
18             </tr>  
19             <tr><th>郵件帳號</th><td>  
20                 <input type="text" value="1234567890@gmail.com" style="width: 100%; height: 25px; border: 1px solid #ccc; padding: 5px; font-size: 14px; margin-bottom: 5px;"/>  
21             </tr>  
22             <tr><th>電話</th><td>  
23                 <input type="text" value="0912345678" style="width: 100%; height: 25px; border: 1px solid #ccc; padding: 5px; font-size: 14px; margin-bottom: 5px;"/>  
24             </tr>  
25             <tr><th>地址</th><td>  
26                 <input type="text" value="新竹市竹北縣竹北市中正路123號" style="width: 100%; height: 25px; border: 1px solid #ccc; padding: 5px; font-size: 14px; margin-bottom: 5px;"/>  
27             </tr>  
28         </table>  
29         <span style="color:red">{{message}}</span>  
30     </form>  
31  {% endblock %}
```



使用PyCharm可以挑選顏色

```
15      </table>  
16      <span style="color:red">{{message}}</span>  
17  </form>  
18  {% endblock %}
```

Result

student 資料表新增(二)

姓名	<input type="text"/>
性別	<input checked="" type="radio"/> 男生 <input type="radio"/> 女生
生日	<input type="text"/>
郵件帳號	<input type="text"/>
電話	<input type="text"/>
地址	<input type="text"/>
<input type="button" value="送出"/> <input type="button" value="重設"/>	

姓名、性別、生日必須輸入！

student 資料表新增(二)

姓名	<input type="text"/>
性別	<input checked="" type="radio"/> 男生 <input type="radio"/> 女生
生日	<input type="text"/> ! 請填寫這個欄位。
郵件帳號	<input type="text"/>
電話	<input type="text"/>
地址	<input type="text"/>
<input type="button" value="送出"/> <input type="button" value="重設"/>	

姓名、性別、生日必須輸入！

student 資料表新增(二)

姓名	Jack <input type="text"/>
性別	<input checked="" type="radio"/> 男生 <input type="radio"/> 女生
生日	2002-01-01 <input type="text"/>
郵件帳號	email <input type="text"/> ! 請在電子郵件地址中包含「@」。「email」未包含「@」。
地址	<input type="text"/>
<input type="button" value="送出"/> <input type="button" value="重設"/>	

姓名、性別、生日必須輸入！



E

Exercise 5: delete

urls.py

```
20     urlpatterns = [
21         path('admin/', admin.site.urls),
22         path('listone/', views.listone),
23         path('listall/', views.listall),
24         path('', views.index),
25         path('index/', views.index),
26
27         path('post/', views.post), # POST 傳送表單
28         path('post1/', views.post1), # 資料新增，資料不驗證
29         path('post2/', views.post2), # 資料新增，資料作驗證
30
31         path('delete/<int:id>/' , views.delete),
32
33         path('edit/<int:id>/' , views.edit), # 由 瀏覽器 開啟
34         path('edit/<int:id>/<str:mode>' , views.edit), # 由 edit.html 按 送出 鈕
35
36         path('edit2/<int:id>/<str:mode>' , views.edit2),
37         path('postform/' , views.postform), # 表單驗證
38     ]
```

delete.html

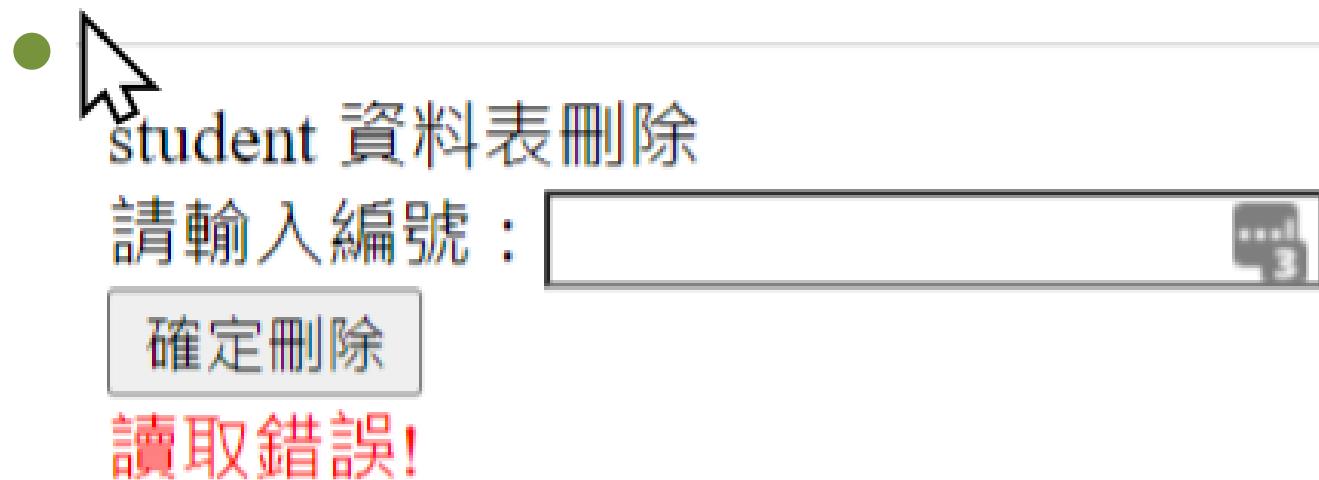
```
1  {% extends "base.html" %}  
2  {% load static %}  
3  {% block title %}<title>資料表刪除</title>{% endblock %}  
4  {% block content %}  
5      <h1>student 資料表刪除</h1>  
6      <form action="." method="POST" name="form1">  
7          {% csrf_token %}  
8          <div>請輸入編號 : <input type="text" name="cId"/>  
9          </div>  
10         <div>  
11             <input type="submit" name="button" id="button" value="確定刪除"/>  
12         </div>  
13         <span style="color:red">{{ message }}</span>  
14     </form>  
15  {% endblock %}
```

views.py

```
85 def delete(request,id=None):
86     if id != None:
87         if request.method == "POST": # 如果是以POST方式才處理
88             id = request.POST['cId'] # 取得表單輸入的編號
89         try:
90             unit = student.objects.get(id=id)
91             unit.delete()
92             return redirect('/index/')
93         except:
94             message = "讀取錯誤！"
95     return render(request, "delete.html", locals())
```

Result

- 於網址輸入網址，刪除記錄。正確會導向index.html的錯誤，
(待後續練習完成index.html)
- 利用資料庫工具查看資料庫變更的內容。





E

Exercise 6: edit1

urls.py

```
20     urlpatterns = [
21         path('admin/', admin.site.urls),
22         path('listone/', views.listone),
23         path('listall/', views.listall),
24         path('', views.index),
25         path('index/', views.index),
26
27         path('post/', views.post), # POST 傳送表單
28         path('post1/', views.post1), # 資料新增，資料不驗證
29         path('post2/', views.post2), # 資料新增，資料作驗證
30
31         path('delete/<int:id>/', views.delete),
32
33         path('edit/<int:id>/', views.edit), # 由 瀏覽器 開啟
34         path('edit/<int:id>/<str:mode>', views.edit), # 由 edit.html 按 送出 鈕
35
36         path('edit2/<int:id>/<str:mode>', views.edit2),
37         path('postform/', views.postform), # 表單驗證
38     ]
```

views.py

```
87     def edit(request, id=None, mode=None):
88         if mode == "edit": # 由 edit.html 按 submit
89             unit = student.objects.get(id=id) # 取得要修改的資料記錄
90             unit.cName = request.GET['cName']
91             unit.cSex = request.GET['cSex']
92             unit.cBirthday = request.GET['cBirthday']
93             unit.cEmail = request.GET['cEmail']
94             unit.cPhone = request.GET['cPhone']
95             unit.cAddr = request.GET['cAddr']
96             unit.save() # 寫入資料庫
97             message = '已修改...'
98             return redirect('/index/')
99         else: # 由網址列
100             try:
101                 unit = student.objects.get(id=id) # 取得要修改的資料記錄
102                 strdate = str(unit.cBirthday)
103                 strdate2 = strdate.replace("年", "-")
104                 strdate2 = strdate.replace("月", "-")
105                 strdate2 = strdate.replace("日", "-")
106                 unit.cBirthday = strdate2
107             except:
108                 message = "此 id不存在!"
109             return render(request, "edit.html", locals())
```

edit.html

```
1  {% extends "base.html" %}  
2  {% block title %}<title>資料表修改(一)</title>{% endblock %}  
3  {% block content %}  
4      <h2>student 資料表修改(一)</h2>  
5      <form action="/edit/{{ unit.id }}/edit" method="GET" name="form1">  
6          <table border="1" cellspacing="1" cellpadding="4">  
7              <tr>  
8                  <th>姓名</th>  
9                  <td><input type="text" id="cName" name="cName" value="{{ unit.cName }}></td>  
10             </tr>  
11             <th>性別{{ unit.cSex }}</th>  
12             {% if unit.cSex == "M" %}  
13                 <td>  
14                     <input name="cSex" type="radio" id="cSex1" value="M" checked="checked"/>男生  
15                     <input name="cSex" type="radio" id="cSex2" value="F"/>女生  
16                 </td>  
17             {% else %}  
18                 <td>  
19                     <input name="cSex" type="radio" id="cSex1" value="M"/>男生  
20                     <input name="cSex" type="radio" id="cSex2" value="F" checked="checked"/>女生  
21                 </td>  
22             {% endif %}  
23             </tr>  
24             <tr>  
25                 <th>生日</th>  
26                 <td><input type="text" id="cBirthday" name="cBirthday" value="{{ unit.cBirthday }}></td>
```

edit.html

```
26      <td><input type="text" id="cBirthday" name="cBirthday" value="{{ unit.cBirthday }}></td>
27  </tr>
28  <tr>
29    <th>郵件帳號</th>
30    <td><input type="text" id="cEmail" name="cEmail" value="{{ unit.cEmail }}></td>
31  </tr>
32  <tr>
33    <th>電話</th>
34    <td><input type="text" id="cPhone" name="cPhone" value="{{ unit.cPhone }}></td>
35  </tr>
36  <tr>
37    <th>地址</th>
38    <td><input type="text" id="cAddr" name="cAddr" value="{{ unit.cAddr }}></td>
39  </tr>
40  <tr>
41    <th colspan="2" align="center">
42      <input type="submit" name="button" id="button" value="送出">
43      <input type="reset" name="button2" id="button2" value="重設"></th>
44  </tr>
45  </table>
46  <span style="color:red">{{ message }}</span>
47  ■
48  </form>
49  {% endblock %}
```

Result

- 輸入網址，帶出資料編輯。
- 可送出儲。



student 資料表修改(一)

姓名	Jack
性別M	<input checked="" type="radio"/> 男生 <input type="radio"/> 女生
生日	2002-01-01
郵件帳號	email@gmail.com
電話	Jack
地址	address
<input type="button" value="送出"/> <input type="button" value="重設"/>	



Exercise 7: edit2及完成index.html

urls.py

```
20     urlpatterns = [
21         path('admin/', admin.site.urls),
22         path('post/', views.post),
23         path('post1/', views.post1),
24         path('postform/', views.postform),
25         path('post2/', views.post2),
26         path('delete/<int:id>/', views.delete),
27         path('edit/<int:id>/', views.edit),
28         path('edit2/<int:id>/<str:mode>', views.edit2),  
         path('index/', views.index),
29     ]
```

views.py

```
112 def edit2(request, id=None, mode=None):
113     if mode == "load": # 由 index.html 按 編輯二 鈕
114         unit = student.objects.get(id=id) # 取得要修改的資料記錄
115         strdate = str(unit.cBirthday)
116         strdate2 = strdate.replace("年", "-")
117         strdate2 = strdate.replace("月", "-")
118         strdate2 = strdate.replace("日", "-")
119         unit.cBirthday = strdate2
120         return render(request, "edit2.html", locals())
121     elif mode == "save": # 由 edit2.html 按 submit
122         unit = student.objects.get(id=id) # 取得要修改的資料記錄
123         unit.cName = request.POST['cName']
124         unit.cSex = request.POST['cSex']
125         unit.cBirthday = request.POST['cBirthday']
126         unit.cEmail = request.POST['cEmail']
127         unit.cPhone = request.POST['cPhone']
128         unit.cAddr = request.POST['cAddr']
129         unit.save() # 寫入資料庫
130         message = '已修改...'
131         return redirect('/index/')
```

views.py

```
19     def index(request):
20         students = student.objects.all().order_by('id') # 讀取資料表，依 id 遞增排序
21         return render(request, "index.html", locals())
```

index.html

```
3  {block content}
4  <h2 align="left">顯示 student 資料表所有資料</h2>
5  <a href="/post1/">新增一</a> <a href="/post2/">新增二</a></h2>
6  <form action"." method="POST" name="form1">
7  {% csrf_token %}
8  <table border="1" cellpadding="0" cellspacing="0">
9  <th>編號</th><th>姓名</th><th>性別</th><th>生日</th>
10 <th>郵件帳號</th><th>電話</th><th>地址</th><th>編輯</th>
11 <% for student in students %>
12 <tr>
13 <td>{{ student.id }} </td>
14 <td>{{ student.cName }} </td>
15 <td>{{ student.cSex}} </td>
16 <td>{{ student.cBirthday }} </td>
17 <td>{{ student.cEmail }} </td>
18 <td>{{ student.cPhone}} </td>
19 <td>{{ student.cAddr }} </td>
20 <td>
21 <a href="/edit/{{student.id}}/" title="編輯一">編輯一</a>
22 <a href="/edit2/{{student.id}}/load" title="編輯二">編輯二</a>
23 <a href="/delete/{{student.id}}/" title="刪除">刪除</a>
24 </td>
25 </tr>
26 <% endfor %>
27 </table>
28 </form>
29 <% endblock %>
```

edit2.html

```
1 1  {% extends "base.html" %}  
2 2  {% block title %}<title>資料表修改(二)</title>{% endblock %}  
3 3  {% block content %}  
4 4    <h2>student 資料表修改(二)</h2>  
5 5    <form action="/edit2/{{ unit.id }}/save" method="POST" name="form1">  
6 6      {% csrf_token %}  
7 7      <table border="1" cellspacing="1" cellpadding="4">  
8 8        <tr>  
9 9          <th>姓名</th>  
10 10         <td><input type="text" id="cName" name="cName" value="{{ unit.cName }}></td>  
11 11       </tr>  
12 12       <th>性別{{ unit.cSex }}</th>  
13 13       {% if unit.cSex == "M" %}  
14 14         <td>  
15 15           <input name="cSex" type="radio" id="cSex1" value="M" checked="checked"/>  
16 16           男生  
17 17           <input name="cSex" type="radio" id="cSex2" value="F"/>  
18 18           女生  
19 19         </td>  
20 20       {% else %}  
21 21         <td>  
22 22           <input name="cSex" type="radio" id="cSex1" value="M"/>  
23 23           男生  
24 24           <input name="cSex" type="radio" id="cSex2" value="F" checked="checked"/>  
25 25           女生  
26 26         </td>  
27 27       {% endif %}  
28 28     </tr>
```

edit2.html

```
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
```

</tr>
<tr>
<th>生日</th>
<td><input type="text" id="cBirthday" name="cBirthday" value="{{ unit.cBirthday }}></td>
</tr>
<tr>
<th>郵件帳號</th>
<td><input type="text" id="cEmail" name="cEmail" value="{{ unit.cEmail }}></td>
</tr>
<tr>
<th>電話</th>
<td><input type="text" id="cPhone" name="cPhone" value="{{ unit.cPhone }}></td>
</tr>
<tr>
<th>地址</th>
<td><input type="text" id="cAddr" name="cAddr" value="{{ unit.cAddr }}></td>
</tr>
<tr>
<th colspan="2" align="center">
<input type="submit" name="button" id="button" value="送出">
<input type="reset" name="button2" id="button2" value="重設"></th>
</tr>
</table>
{{ message }}
</form>
{% endblock %}

Result

- 觀察網址的不同。
- 觀察導向的頁面。

顯示 student 資料表所有資料 [新增一](#) [新增二](#)

編號	姓名	性別	生日	郵件帳號	電話	地址	編輯
6	Jack	M	Jan. 1, 2002	email@gmail.com	0977777777	address	編輯一 編輯二 刪除
7	123	M	Jan. 1, 2000	11	11	11	編輯一 編輯二 刪除
9	tester	M	Jan. 1, 2002	email@gmail.com	phonenumbe	address	編輯一 編輯二 刪除
10	Smith	M	Jan. 1, 1999	email	0999	address	編輯一 編輯二 刪除