

# 實驗26-28

26. Hello Jiann-Ming 網站設計

27. 使用SymPy 與Flask建構微分網站

28. 使用SymPy 與 Flask建構多項式相加網站

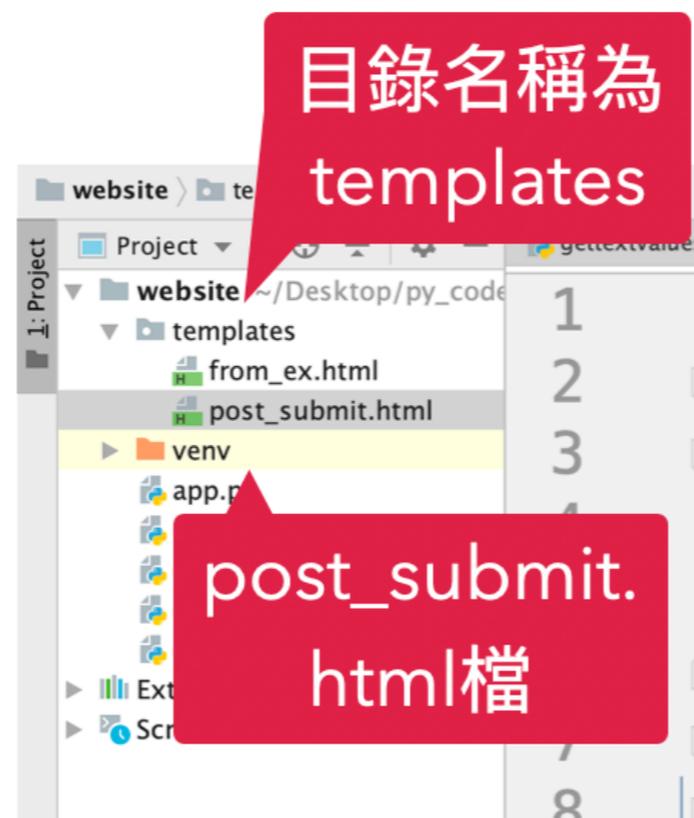
# **EX26. 建立網站：Hello Jiann-Ming**

# 步驟一、匯入模組 **flask**

```
from flask import Flask, request, render_template
```

**步驟二、建立html檔案**  
**post\_submit.html**

步驟二 A、建立目錄  
**templates**，並且在該目錄  
下新增**post\_submit.html**檔



步驟二B、設計網頁：標題、文字輸入欄位、按鍵  
傳送

目錄名稱為  
templates

html檔

網頁標題

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <title>Hello Page</title>
6 </head>
7 <body>
8   <form method='post' action="/">
9     <input type='text' name='username'>
10    <button type='submit'>Submit</button>
11 </form>
12 </body>
13 </html>
```

```
website [~/Desktop/py_code_2020/website] - .../templates/post_submit.html [website]
website > templates > post_submit.html
Project
website ~/Desktop/py_code_2020/website
  templates
    from_ex.html
    post_submit.html
  venv
  app.py
  gettextvalue.py
  gettextvalues2.py
  External Libraries
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <title>Hello Page</title>
6 </head>
7 <body>
  <form method='post' action="/">
    <input type='text' name='username'>
    <button type='submit'>Submit</button>
  </form>
</body>
</html>
```

設計文字輸入欄位、  
按鍵傳送

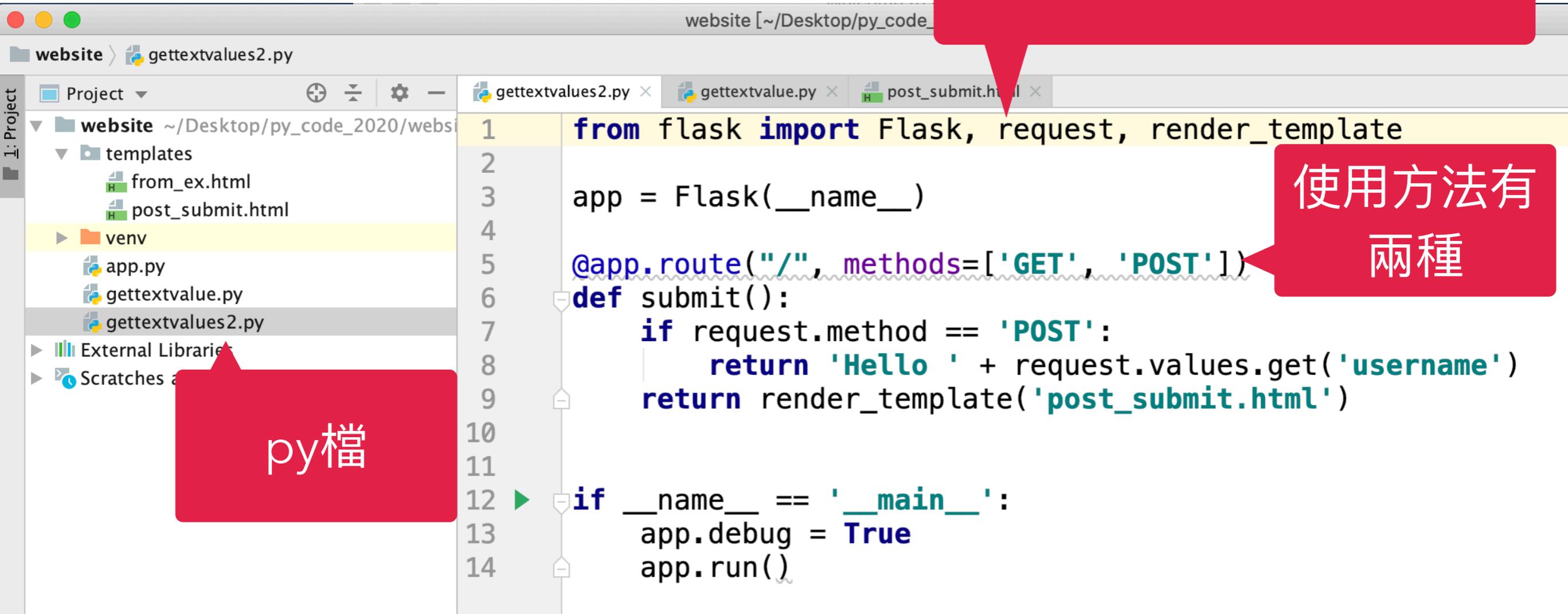
post\_submit.html

```
<!DOCTYPE html>  
<html lang="en">  
<head>  
  <meta charset="UTF-8">  
  <title>Hello Page</title>  
</head>  
<body>  
  <form method='post' action="/">  
    <input type='text' name='username'>  
    <button type='submit'>Submit</button>  
  </form>  
</body>  
</html>
```

步驟三、建立

**gettextvalue2.py**檔案

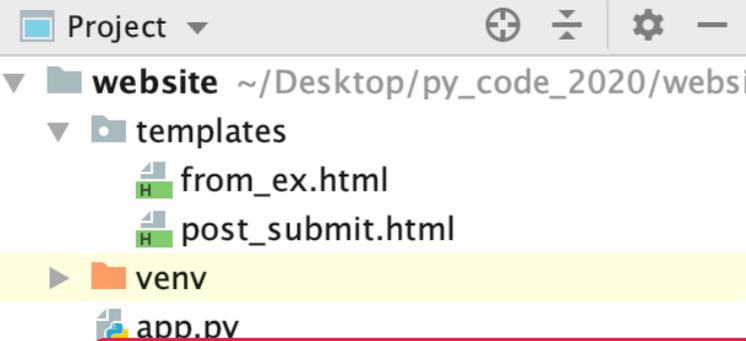
匯入類別Flask, request,  
render\_template



```
1 from flask import Flask, request, render_template
2
3 app = Flask(__name__)
4
5 @app.route("/", methods=['GET', 'POST'])
6 def submit():
7     if request.method == 'POST':
8         return 'Hello ' + request.values.get('username')
9     return render_template('post_submit.html')
10
11
12 if __name__ == '__main__':
13     app.debug = True
14     app.run()
```

使用方法有  
兩種

py檔

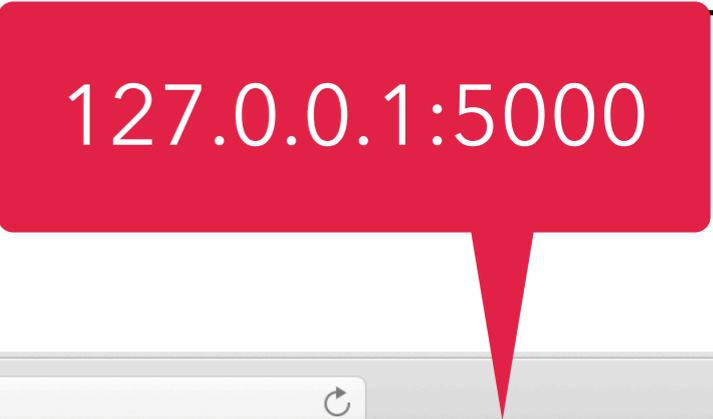
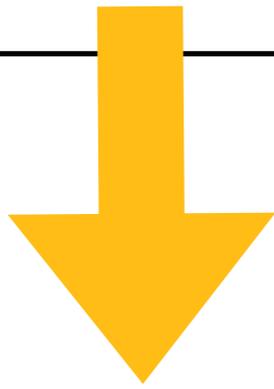
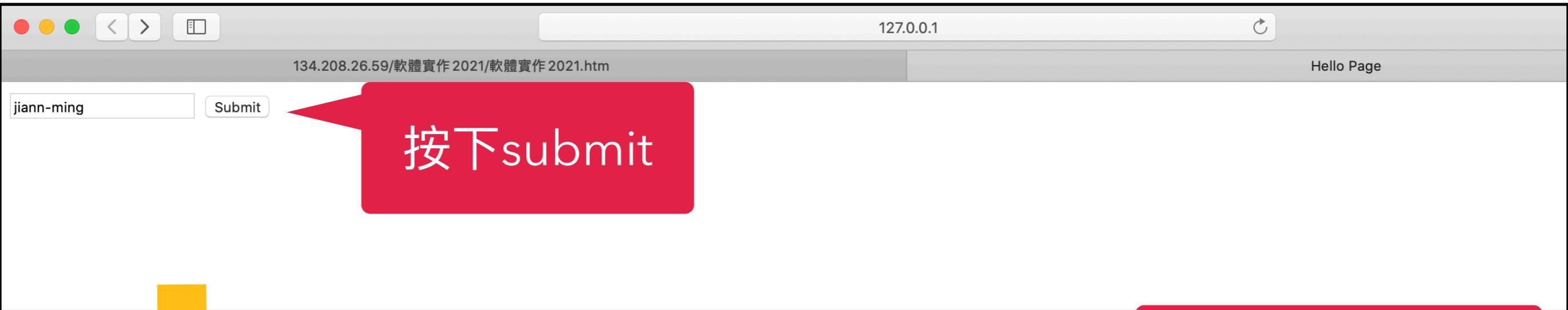


```
1 from flask import Flask, request, render_template
2
3 app = Flask(__name__)
4
5 @app.route("/", methods=['GET', 'POST'])
   def submit():
       if request.method == 'POST':
           return 'Hello ' + request.values.get('username')
       return render_template('post_submit.html')
6
7 if __name__ == '__main__':
   app.debug = True
   app.run()
```

使用方法為'POST'  
將抓取username  
附加在'Hello'之  
後，貼在標題  
為'127.0.0.1:5000'  
的網頁上

使用方法為'GET'  
透過標題為'hello  
page'的網頁，抓  
出username

步驟四、執行  
**gettextvalue2.py**



Hello jiann-ming

# Ex27. 使用SymPy 與 Flask建構微分網站

Lecture 20

步驟一、匯入套件

**flask**與**sympy**

```
from flask import Flask, request, render_template  
import sympy as sp  
from sympy import *
```

# 步驟二、建立html檔案

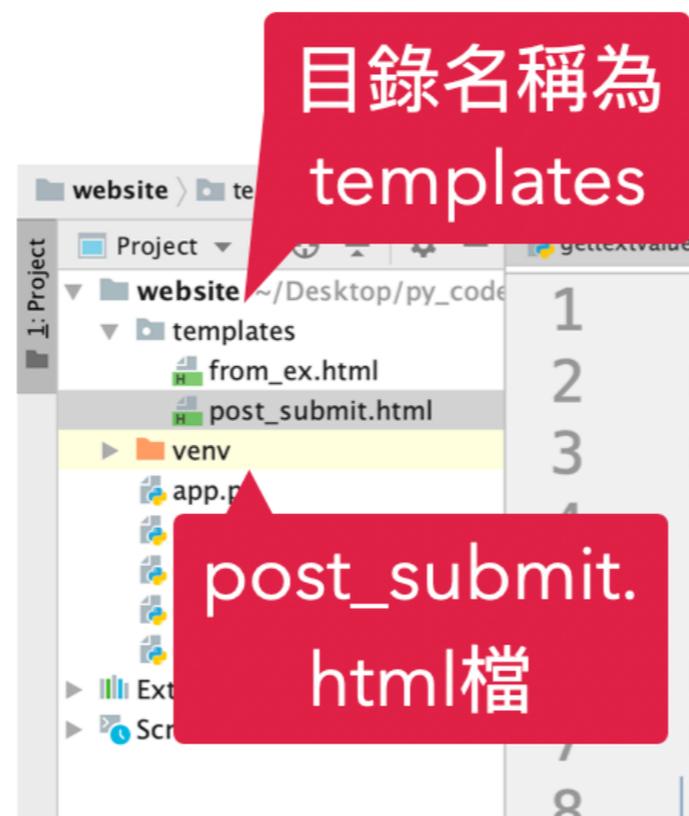
目錄名稱為  
templates

post\_submit.  
html檔

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <title>微分網頁</title>
6 </head>
7 <body>
8 <form method="post" action="/">
9   <p>本網站使用符號運算sympy進行微分</p>
10  <p>微分運算，請輸入函數f(x)</p>
11  <p>範例，輸入函數： $x^2+2x-1$ ，答案： $2x+2$ </p>
12  <p>範例，輸入函數： $\cos(x)$ ，答案： $-\sin(x)$ </p><br>
13  <input type="text" name="username">
14  <button type="submit">按鍵傳送</button>
15 </form>
16 </body>
17 </html>
```

Lecture 20

步驟二 A、建立目錄  
**templates**，並且在該目錄  
下新增**post\_submit.html**檔



步驟二B、設計網頁：標  
題、顯示文字、文字輸入  
欄位、按鍵傳送



本網站使用符號運算sympy進行微分

微分運算，請輸入函數 $f(x)$

範例，輸入函數： $x**2+2*x-1$ ，答案： $2*x+2$

範例，輸入函數： $\cos(x)$ ，答案： $-\sin(x)$

按鍵傳送

在網頁上顯示文字

在網頁上顯示文字輸入欄位，使用者輸入函數

在網頁上顯示按鍵，使用者可以用滑鼠按下按鍵，完成get 模式

2.py × gettextvalue.py × gettextvalues2\_diff.py × gettextvalues2\_diff2.py × app.py × post\_submit.html × from\_ex.ht

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>微分網頁</title>
</head>
<body>
<form method='post' action="/">
  <p>本網站使用符號運算sympy進行微分</p>
  <p>微分運算，請輸入函數f(x)</p>
  <p>範例，輸入函數： $x^2+2x-1$ ，答案： $2x+2$ </p>
  <p>範例，輸入函數： $\cos(x)$ ，答案： $-\sin(x)$ </p><br>
  <input type='text' name='username'>
  <button type='submit'>按鍵傳送</button>
</form>
</body>
</html>
```

檔案名稱

post\_submit.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>微分網頁</title>
</head>
<body>
  <form method='post' action="/">
    <p>本網站使用符號運算sympy進行微分</p>
    <p>微分運算，請輸入函數f(x)</p>
    <p>範例，輸入函數:x**2+2*x-1，答案：2*x+2</p>
    <p>範例，輸入函數:cos(x)，答案：-sin(x)</p><br>
    <input type='text' name='username'>
    <button type='submit'>按鍵傳送</button>
  </form>
</body>
</html>
```

步驟三、建立

**gettextvalue\_diff.py**檔案

# 步驟三A、匯入套件

```
gettextvalues2_diff2.py x | gettextvalues2_diff2
1 from flask import Flask, request, render_templat
2 import sympy as sp
3 from sympy import *
4
```

匯入套件sympy，以設計微分符號運算功能

request  
GET方法:render\_template，執行html檔，擷取輸入文字，完成submit  
POST方法：在網頁上輸出運算結果

# 步驟三B、設計GET方法

## GET方法與POST方法

```
0
7 @app.route("/", methods=['GET', 'POST'])
8 def submit():
9     if request.method == 'POST':
10         x = sp.symbols('x')
11         ss = request.values.get('username')
12         tt = "diff("+ss+")"
13         ans = eval(tt)
14         return '函數'+ss+'的微分: '+str(ans)
15     return render_template('post_submit.html')
```

request.method為'GET'時（先執行GET方法），使用render\_template，執行post\_template.html檔，擷取輸入文字，完成submit

# 步驟三 C、設計POST方法

## GET方法與POST方法

```
0
7 @app.route("/", methods=['GET', 'POST'])
8 def submit():
9     if request.method == 'POST':
10         x = sp.symbols('x')
11         ss = request.values.get('username')
12         tt = "diff("+ss+")"
13         ans = eval(tt)
14         return '函數'+ss+'的微分: '+str(ans)
15     return render_template('post_submit.html')
```

request.method為'POST'時，執行以下步驟

1. 宣告符號變數x
2. 將ss設定為，GET模式傳回的username內容
3. 將字串tt設定為微分指令
4. 使用eval執行tt所代表的微分指令，儲存在ans中
5. 回傳結果

```
1 from flask import Flask, request, render_template
2 import sympy as sp
3 from sympy import *
4
5 app = Flask(__name__)
6
7 @app.route("/", methods=['GET', 'POST'])
8 def submit():
9     if request.method == 'POST':
10         x = sp.symbols('x')
11         ss = request.values.get('username')
12         tt = "diff("+ss+")"
13         ans = eval(tt)
14         return '函數'+ss+'的微分: '+str(ans)
15     return render_template('post_submit.html')
16 if __name__ == '__main__':
17     app.debug = True
18     app.run()
```

步驟四、執行

**gettextvalue\_diff.py**

```
website > gettextvalues2_diff2.py
Project
templates
  from_ex.html
  post_submit.html
  post_submit_poly_add.html
venv
  app.py
  gettextvalue.py
  gettextvalues2.py
  gettextvalues2_diff2.py
  gettextvalues2_diff2.py
  gettextvalues2_poly_add.py
External Libraries
Scratches and Consoles

6
7 @app.route("/", methods=['GET', 'POST'])
8 def submit():
9     if request.method == 'POST':
10         x = sp.symbols('x')
11         ss = request.values.get('username')
12         tt = "diff("+ss+")"
13         ans = eval(tt)
14         return '函數'+ss+'的微分: '+str(ans)

submit() > if request.method == 'POST'
```

```
Run: gettextvalues2_diff2
production deployment.
Use a production WSGI server instead.
* Debug mode: on
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
* Restarting with stat
* Debugger is active!
```

press

## GET方法

本網站使用符號運算sympy進行微分

微分運算，請輸入函數f(x)

範例，輸入函數： $x^2+2x-1$ ，答案： $2x+2$

範例，輸入函數： $\cos(x)$ ，答案： $-\sin(x)$



submit，將GET  
方法擷取的資料  
傳送給python

## POST方法： 運算與回傳結果

函數 $\tanh(x)$ 的微分： $1 - \tanh(x)^2$

# Ex28. 使用SymPy 與 Flask建構多項式相加網站

Lecture 21

# 步驟一、建立html檔案

目錄名稱為  
templates

post\_submit.  
html檔

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <title>微分網頁</title>
6 </head>
7 <body>
8 <form method="post" action="/">
9   <p>本網站使用符號運算sympy進行微分</p>
10  <p>微分運算，請輸入函數f(x)</p>
11  <p>範例，輸入函數： $x^2+2x-1$ ，答案： $2x+2$ </p>
12  <p>範例，輸入函數： $\cos(x)$ ，答案： $-\sin(x)$ </p><br>
13  <input type="text" name="username">
14  <button type="submit">按鍵傳送</button>
15 </form>
16 </body>
17 </html>
```

Slides 51-62

步驟一 A、建立目錄，在該目錄templates下新增  
**post\_submit\_poly\_add.html**  
檔

步驟一B、設計網頁：標題、顯示文字、文字輸入欄位、**radio**選擇器、按鍵傳送

標題顯示  
多項式相加網頁

多項式相加網頁

本網站使用符號運算sympy進行多項式相加

請分別輸入兩個多項式

f(x):

g(x):

Add  Subtract

在網頁上顯示兩個文字  
輸入欄位，使用者輸入  
函數

在網頁上顯示兩個選項選一項的  
radio選擇器，讓使用者選擇多項  
式相加或相減的功能

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>多項式相加網頁</title>
</head>
<body>
<form method='post' action="/">
  <p>本網站使用符號運算sympy進行多項式相加</p>
  <p>請分別輸入兩個多項式</p><br>
  <p> f(x): </p>
  <input type='text' name='username1'>
  <p> g(x): </p>
  <input type='text' name='username2'><br>
  <INPUT TYPE="Radio" Name="gender" Value="add">Add
  <INPUT TYPE="Radio" Name="gender" Value="subtract">Substract
  <button type='submit'>按鍵傳送</button>
</form>
</body>
</html>
```

text輸入欄位，變數名稱name分別設定為username1與username2

輸入型態為Radio，變數名稱gender，選擇內容分別設定為add與subtract

步驟三、設計POST方法，建立 `gettextvalue_poly_add.py`

## GET方法與POST方法

```
@app.route("/", methods=['GET', 'POST'])
def submit():
    if request.method == 'POST':
        x = sp.symbols('x')
        ss1 = request.values.get('username1')
        ss2 = request.values.get('username2')
        op = request.values.get('gender')
        if op == "add":
            tt = "("+ss1+")"+"+"+"+"+"(" + ss2 + ").simplify()
            op = "+"
        else:
            tt = "("+ss1+")"+"-"+"(" + ss2 + ").simplify()"
            op = "-"
        ans = eval(tt)
        output_str = "("+ss1+")" + op + "(" + ss2 + ")" + " => " + str(ans)
    return render_template('post_submit_poly_add.html')
```

request.method為'GET'時  
(先執行  
GET方法)，使用  
render\_template，執行  
post\_submit\_poly\_add.html  
檔，擷取輸入文字，完成  
submit

request.method為'POST'時，執行以下步驟

1. 宣告符號變數x
2. 將ss1與ss2分別設定為，GET模式傳回的username1與username2內容
3. 將op設定為GET模式傳回的gender內容
4. 將字串tt設定為多項式運算指令
5. 使用eval執行tt所代表的多項式運算指令，儲存在ans中
6. 回傳結果

```
@app.route("/", methods=['GET', 'POST'])
def submit():
    if request.method == 'POST':
        x = sp.symbols('x')
        ss1 = request.values.get('username1')
        ss2 = request.values.get('username2')
        op = request.values.get('gender')
        if op == "add":
            tt = "("+ss1+")"+"+"+"("+ss2+").simplify()"
            op = "+"
        else:
            tt = "("+ss1+")"+"-"+"("+ss2+").simplify()"
            op = "-"
        ans = eval(tt)
        output_str = "("+ss1+")"+ op + "("+ ss2 +")" + ' => '+str(ans)
        return output_str
    return render_template('post_submit_poly_add.html')
```

```
gettextvalues2_diff2.py x gettextvalues2_poly_add.py x post_submit_poly_add.html x post_submit.html x
1 from flask import Flask, request, render_template
2 import sympy as sp
3 from sympy import *
4 app = Flask(__name__)
5 @app.route("/", methods=['GET', 'POST'])
6 def submit():
7     if request.method == 'POST':
8         x = sp.symbols('x')
9         ss1 = request.values.get('username1')
10        ss2 = request.values.get('username2')
11        op = request.values.get('gender')
12        if op == "add":
13            tt = "("+ss1+")"+"+"+"("+ss2+").simplify()"
14            op = "+"
15        else:
16            tt = "("+ss1+")"+"-"+"("+ss2+).simplify()"
17            op = "-"
18        ans = eval(tt)
19        output_str = "("+ss1+")"+ op + "("+ ss2 +")" + ' => '+str(ans)
20        return output_str
21    return render_template('post_submit_poly_add.html')
22 if __name__ == '__main__':
23     app.debug = True
24     app.run()
```

步驟四、執行

**gettextvalue\_poly\_add.py**

```
website > gettextvalues2_diff2.py
Project
templates
  from_ex.html
  post_submit.html
  post_submit_poly_add.html
venv
  app.py
  gettextvalue.py
  gettextvalues2.py
  gettextvalues2_diff2.py
  gettextvalues2_diff2.py
  gettextvalues2_poly_add.py
External Libraries
Scratches and Consoles

6
7 @app.route("/", methods=['GET', 'POST'])
8 def submit():
9     if request.method == 'POST':
10         x = sp.symbols('x')
11         ss = request.values.get('username')
12         tt = "diff("+ss+")"
13         ans = eval(tt)
14         return '函數'+ss+'的微分: '+str(ans)

submit() > if request.method == 'POST'
```

```
Run: gettextvalues2_diff2
production deployment.
Use a production WSGI server instead.
* Debug mode: on
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
* Restarting with stat
* Debugger is active!
```

press

## GET方法

本網站使用符號運算sympy進行微分

微分運算，請輸入函數f(x)

範例，輸入函數： $x^2+2x-1$ ，答案： $2x+2$

範例，輸入函數： $\cos(x)$ ，答案： $-\sin(x)$



submit，將GET  
方法擷取的資料  
傳送給python

## POST方法： 運算與回傳結果

函數 $\tanh(x)$ 的微分： $1 - \tanh(x)^2$