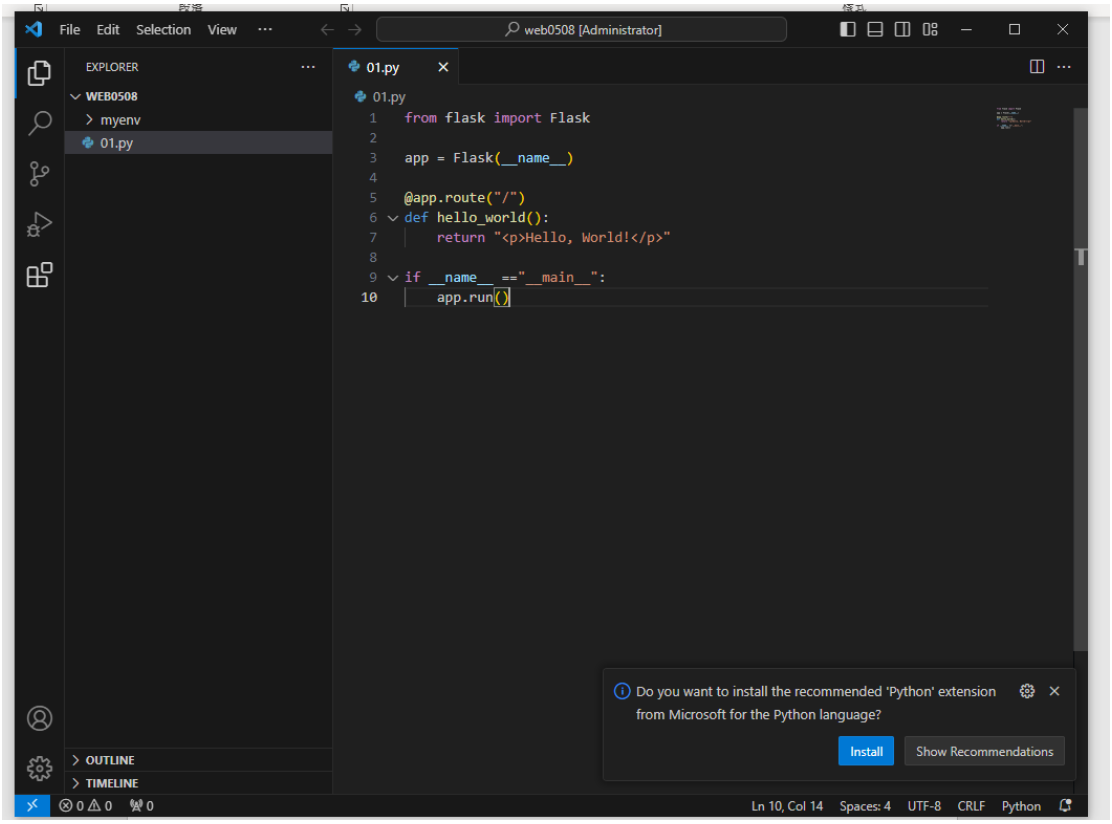
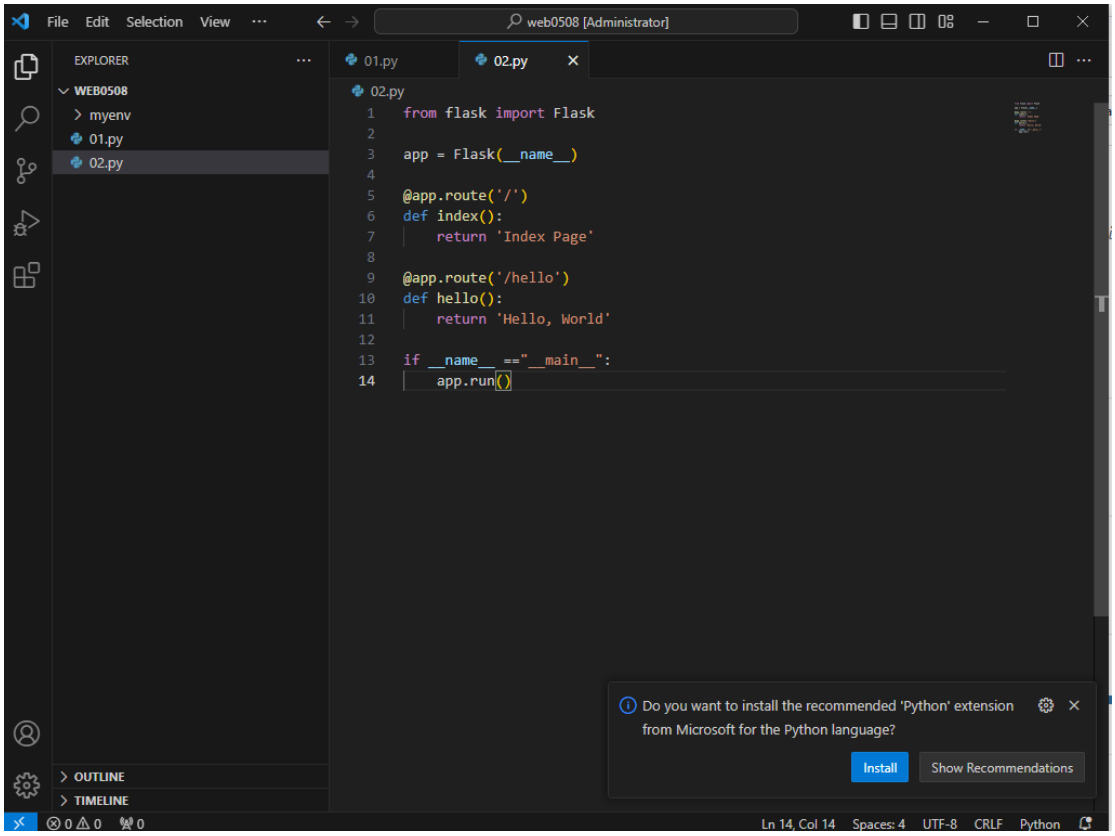
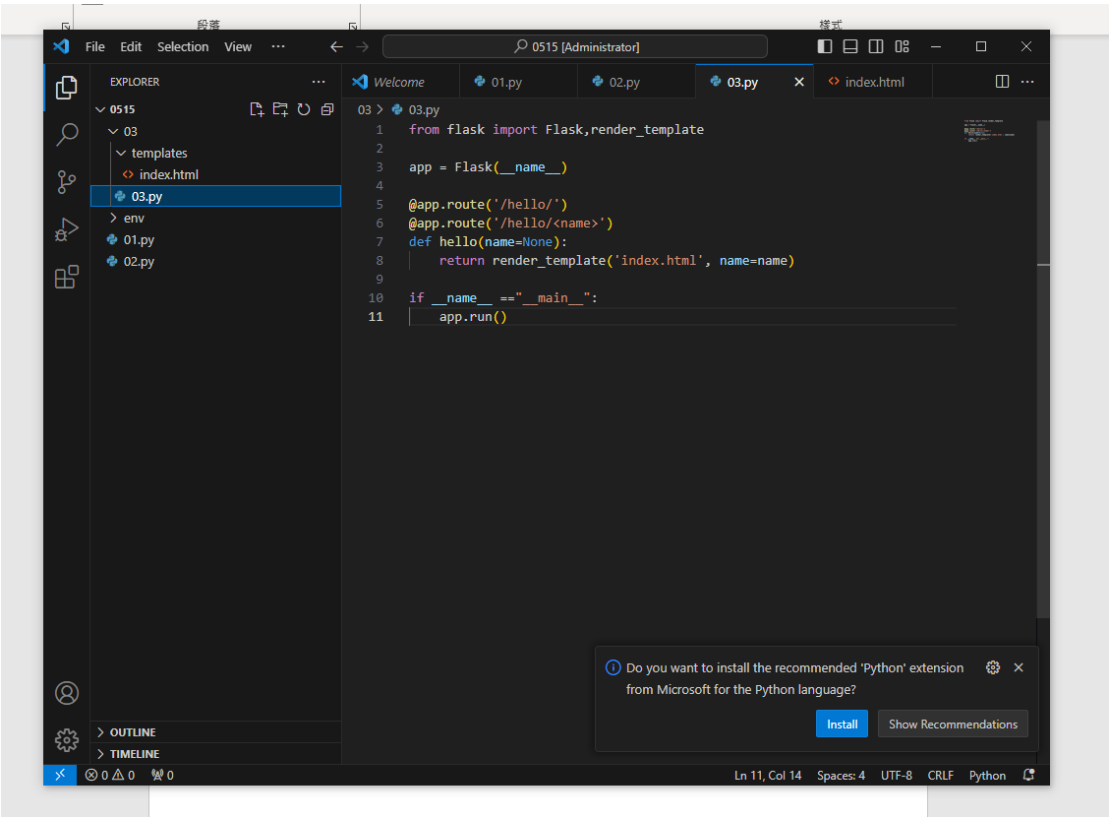
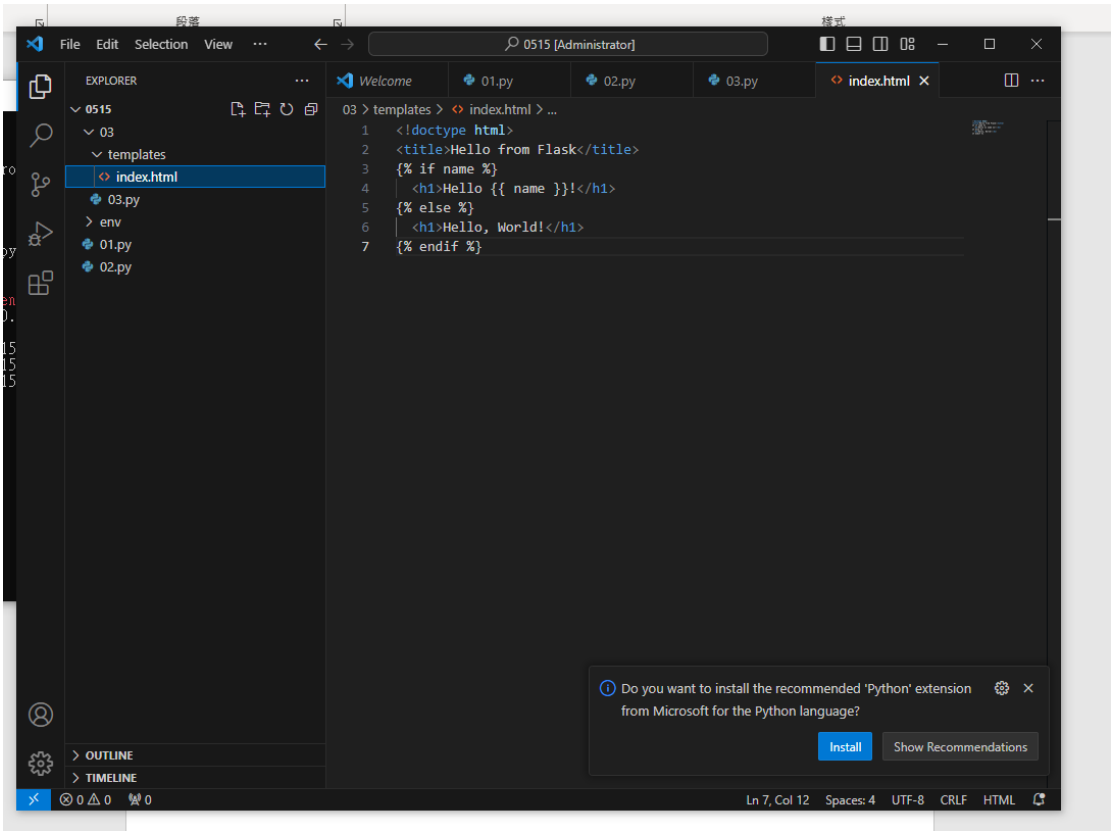
0508

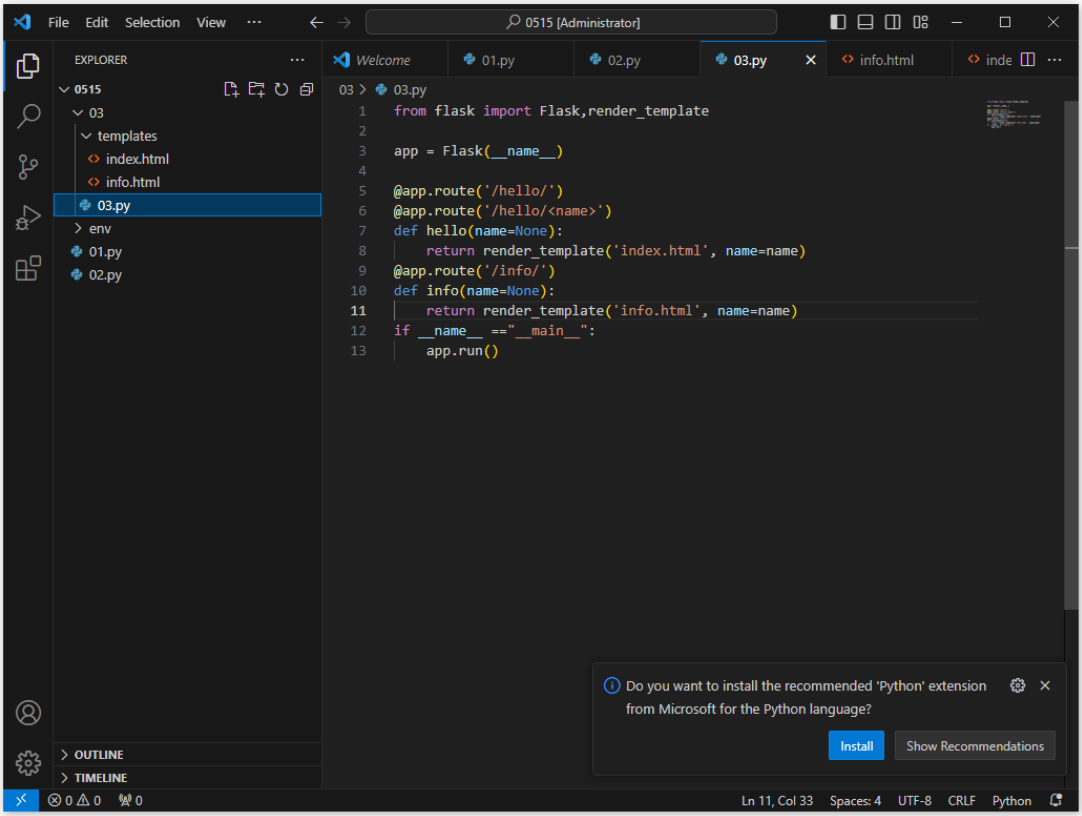

File Edit Selection View
web0508 [Administrator]
EXPLORER
WEB0508
myenv
01.py
01.py
from flask import Flask
app = Flask(__name__)
@app.route("/")
def hello_world():
return " Hello, World! "
if __name__ =="__main__":
app.run()
Do you want to install the recommended 'Python' extension from Microsoft for the Python language?
Install
Show Recommendations
OUTLINE
TIMELINE
Ln 10, Col 14
Spaces: 4
UTF-8
CRLF
Python

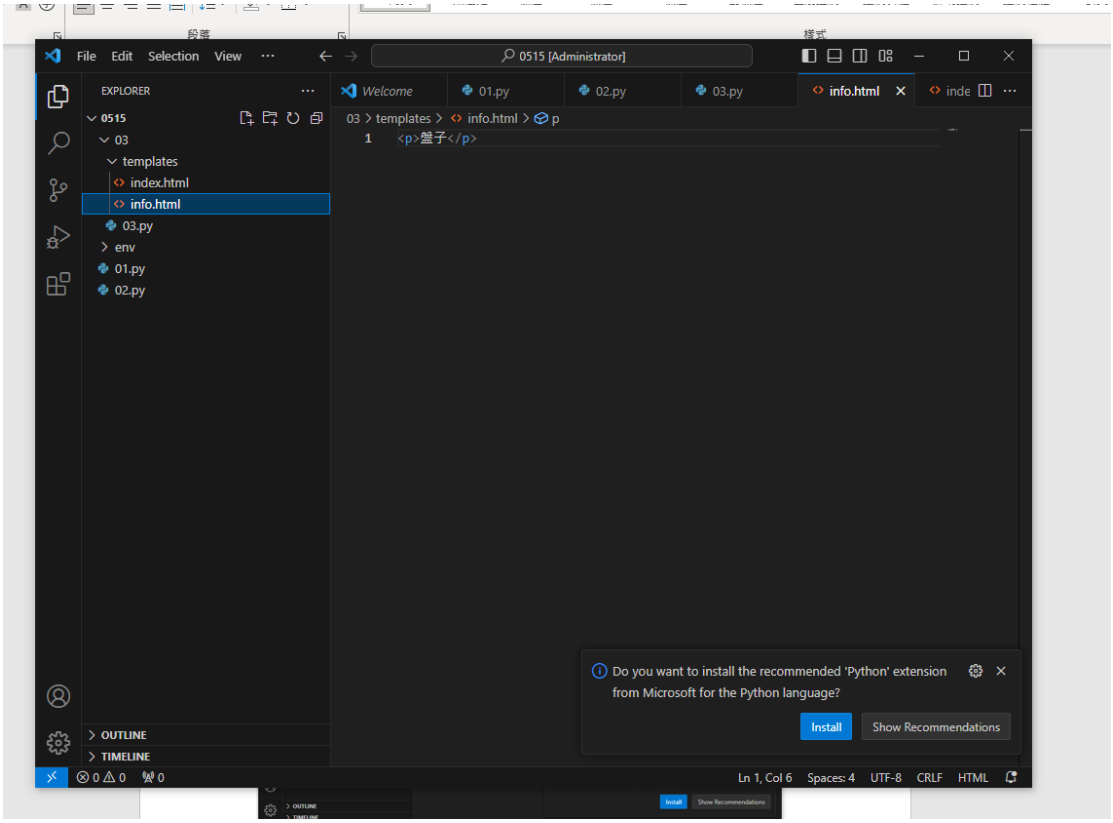

File Edit Selection View
web0508 [Administrator]
EXPLORER
WEB0508
myenv
01.py
02.py
01.py
02.py
from flask import Flask
app = Flask(__name__)
@app.route('/')
def index():
return 'Index Page'
@app.route('/hello')
def hello():
return 'Hello, World'
if __name__ =="__main__":
app.run()
Do you want to install the recommended 'Python' extension from Microsoft for the Python language?
Install
Show Recommendations
OUTLINE
TIMELINE
Ln 14, Col 14
Spaces: 4
UTF-8
CRLF
Python

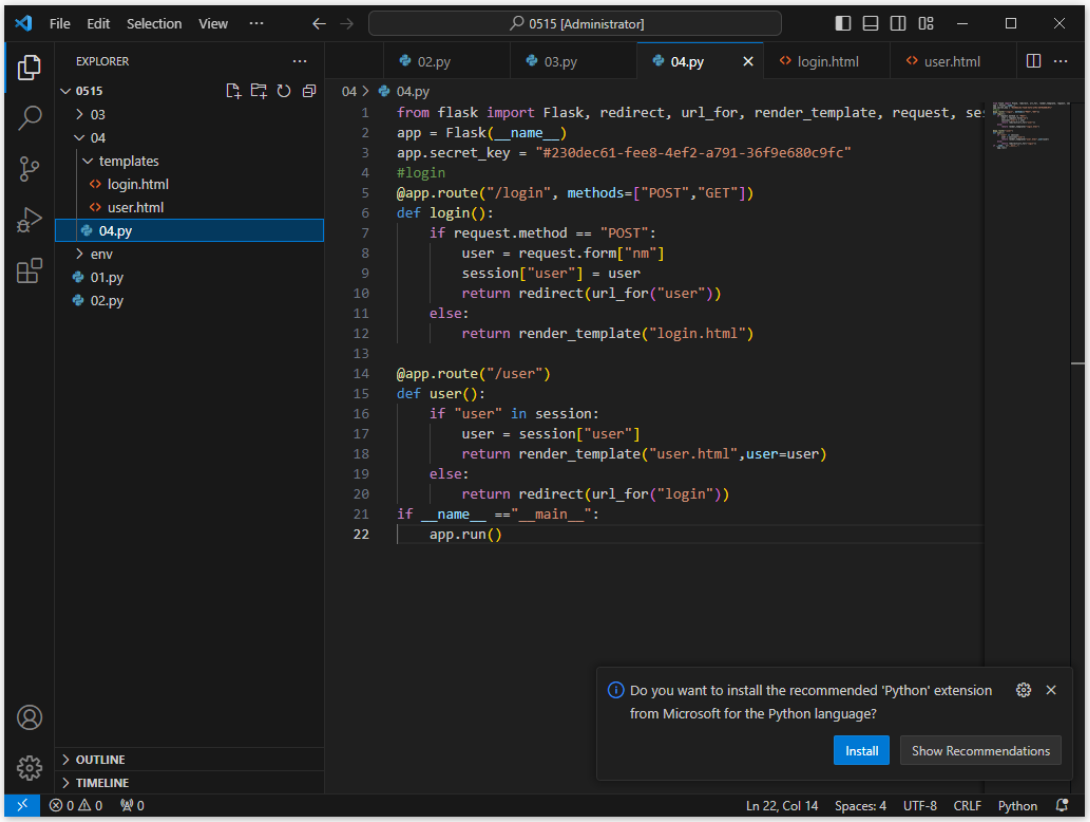
0515

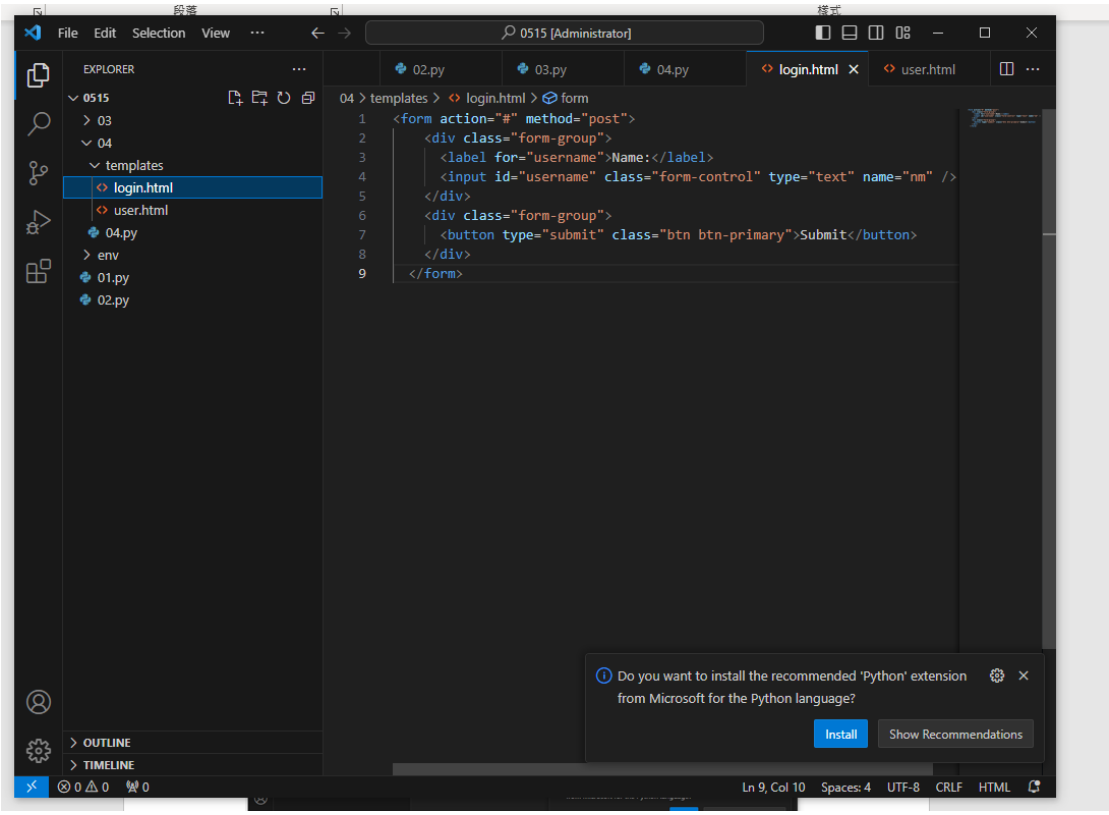

File Edit Selection View
0515 [Administrator]
EXPLORER
0515
03
templates
index.html
03.py
env
01.py
02.py
Welcome
01.py
02.py
03.py
index.html
from flask import Flask,render_template
app = Flask(__name__)
@app.route('/hello/')
@app.route('/hello/<name>')
def hello(name=None):
return render_template('index.html', name=name)
if __name__ =="__main__":
app.run()
Do you want to install the recommended 'Python' extension from Microsoft for the Python language?
Install
Show Recommendations
OUTLINE
TIMELINE
Ln 11, Col 14
Spaces: 4
UTF-8
CRLF
Python


File Edit Selection View
0515 [Administrator]
EXPLORER
0515
03
templates
index.html
03.py
env
01.py
02.py
Welcome
01.py
02.py
03.py
index.html
<!doctype html>
<title>Hello from Flask</title>
{% if name %}
<h1>Hello {{ name }}!</h1>
{% else %}
<h1>Hello, World!</h1>
{% endif %}
Do you want to install the recommended 'Python' extension from Microsoft for the Python language?
Install
Show Recommendations
OUTLINE
TIMELINE
Ln 7, Col 12
Spaces: 4
UTF-8
CRLF
HTML


File Edit Selection View
0515 [Administrator]
EXPLORER
0515
03
templates
index.html
info.html
03.py
env
01.py
02.py
Welcome
01.py
02.py
03.py
info.html
03 > 03.py
1 from flask import Flask,render_template
2
3 app = Flask(__name__)
4
5 @app.route('/hello/')
6 @app.route('/hello/<name>')
7 def hello(name=None):
8 return render_template('index.html', name=name)
9 @app.route('/info/')
10 def info(name=None):
11 return render_template('info.html', name=name)
12 if __name__ =="__main__":
13 app.run()
Do you want to install the recommended 'Python' extension from Microsoft for the Python language?
Install
Show Recommendations
OUTLINE
TIMELINE
Ln 11, Col 33 Spaces: 4 UTF-8 CRLF Python


File Edit Selection View
0515 [Administrator]
EXPLORER
0515
03
templates
index.html
info.html
03.py
env
01.py
02.py
Welcome
01.py
02.py
03.py
info.html
03 > templates > info.html > p
1 盤子
Do you want to install the recommended 'Python' extension from Microsoft for the Python language?
Install
Show Recommendations
OUTLINE
TIMELINE
Ln 1, Col 6 Spaces: 4 UTF-8 CRLF HTML

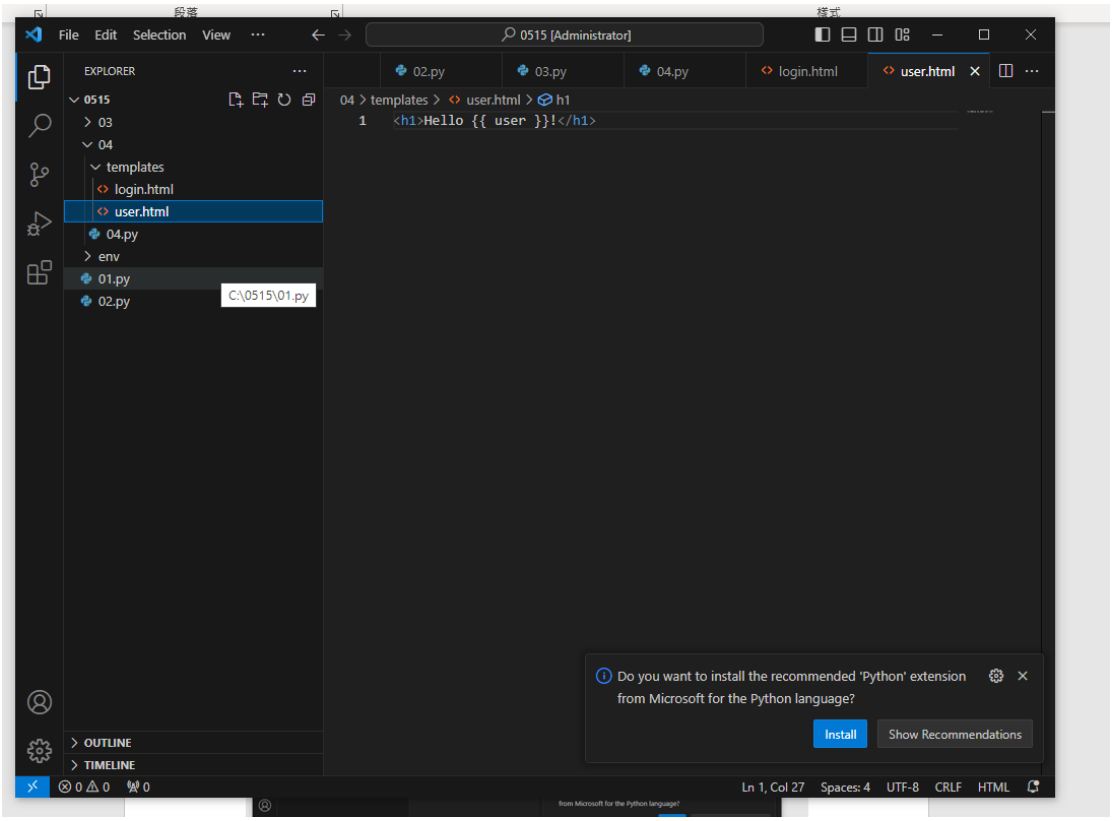

File Edit Selection View
0515 [Administrator]
EXPLORER
0515
03
04
templates
login.html
user.html
04.py
env
01.py
02.py
02.py
03.py
04.py
login.html
user.html
04 > 04.py
from flask import Flask, redirect, url_for, render_template, request, se
app = Flask(__name__)
app.secret_key = "#230dec61-fee8-4ef2-a791-36f9e680c9fc"
#login
@app.route("/login", methods=["POST","GET"])
def login():
if request.method == "POST":
user = request.form["nm"]
session["user"] = user
return redirect(url_for("user"))
else:
return render_template("login.html")
@app.route("/user")
def user():
if "user" in session:
user = session["user"]
return render_template("user.html",user=user)
else:
return redirect(url_for("login"))
if __name__ =="__main__":
app.run()
Do you want to install the recommended 'Python' extension from Microsoft for the Python language?
Install
Show Recommendations
OUTLINE
TIMELINE
Ln 22, Col 14
Spaces: 4
UTF-8
CRLF
Python

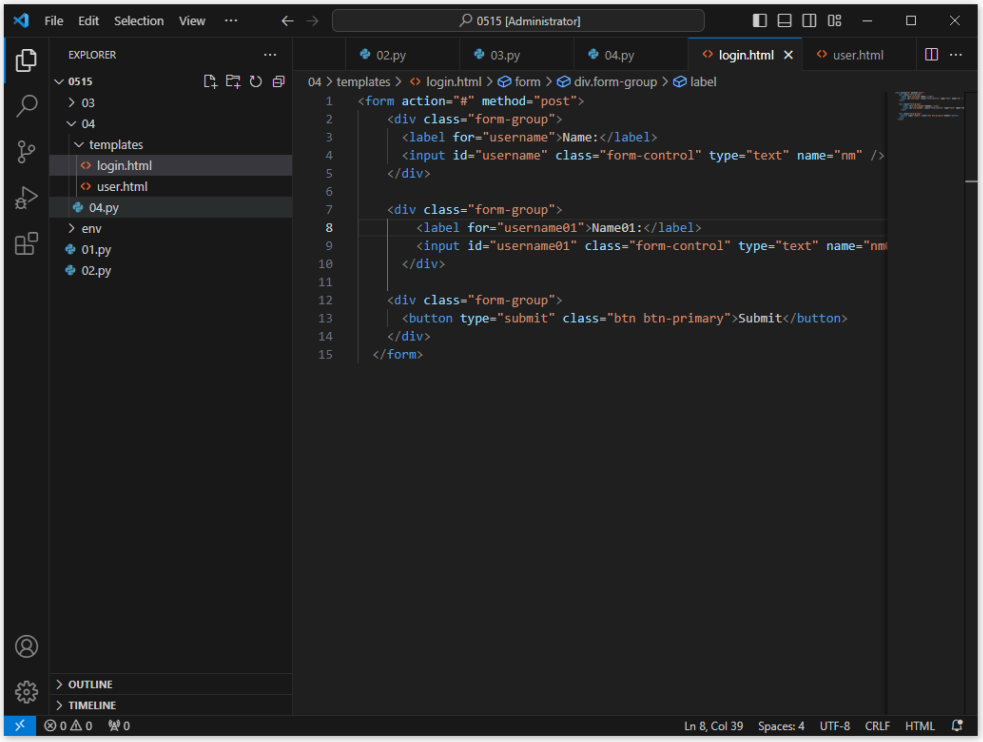

File Edit Selection View
0515 [Administrator]
EXPLORER
0515
03
04
templates
login.html
user.html
04.py
env
01.py
02.py
02.py
03.py
04.py
login.html
user.html
04 > templates > login.html > form
<form action="#" method="post">

<label for="username">Name:</label>
<input id="username" class="form-control" type="text" name="nm" />

<button type="submit" class="btn btn-primary">Submit</button>

</form>
Do you want to install the recommended 'Python' extension from Microsoft for the Python language?
Install
Show Recommendations
OUTLINE
TIMELINE
Ln 9, Col 10
Spaces: 4
UTF-8
CRLF
HTML


File Edit Selection View
0515 [Administrator]
EXPLORER
0515
03
04
templates
login.html
user.html
04.py
env
01.py
02.py
C:\0515\01.py
02.py
03.py
04.py
login.html
user.html
04 > templates > user.html > h1
1 <h1>Hello {{ user }}!</h1>
Do you want to install the recommended 'Python' extension from Microsoft for the Python language?
Install
Show Recommendations
OUTLINE
TIMELINE
Ln 1, Col 27
Spaces: 4
UTF-8
CRLF
HTML

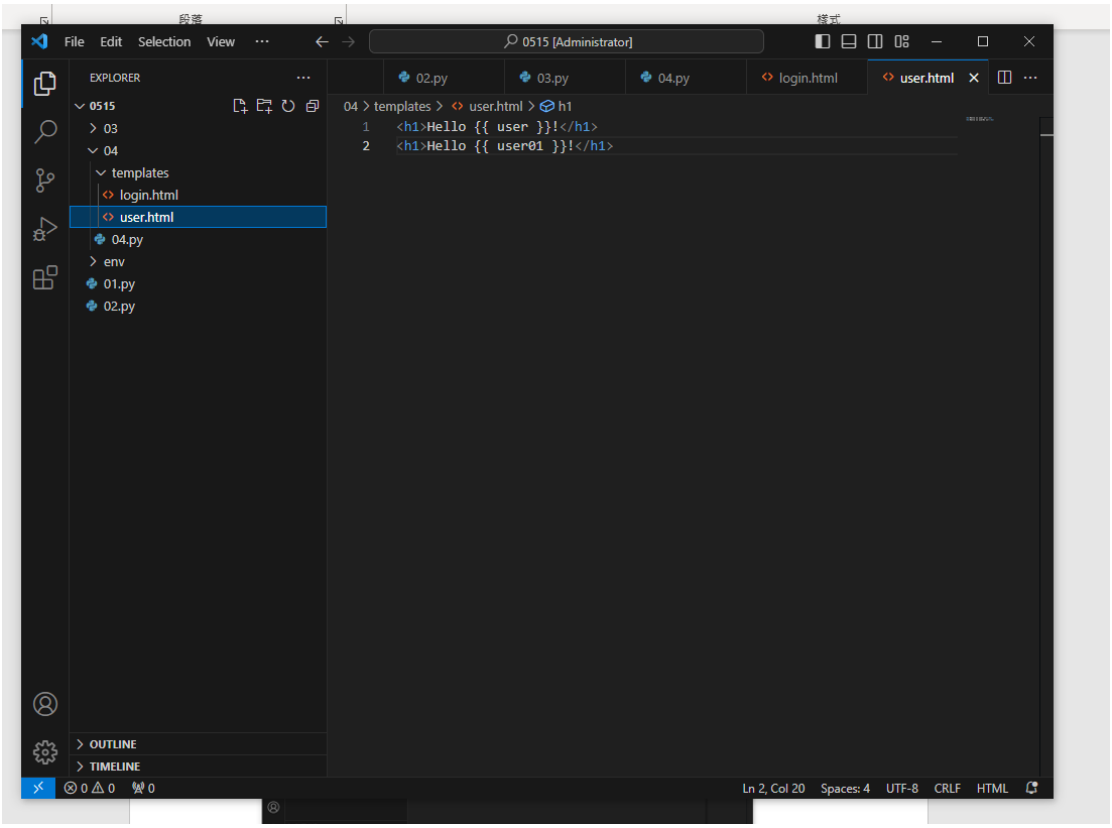

File Edit Selection View
0515 [Administrator]
EXPLORER
0515
03
04
templates
login.html
user.html
04.py
env
01.py
02.py
02.py
03.py
04.py
login.html
user.html
04 > templates > login.html > form > div.form-group > label
<form action="#" method="post">

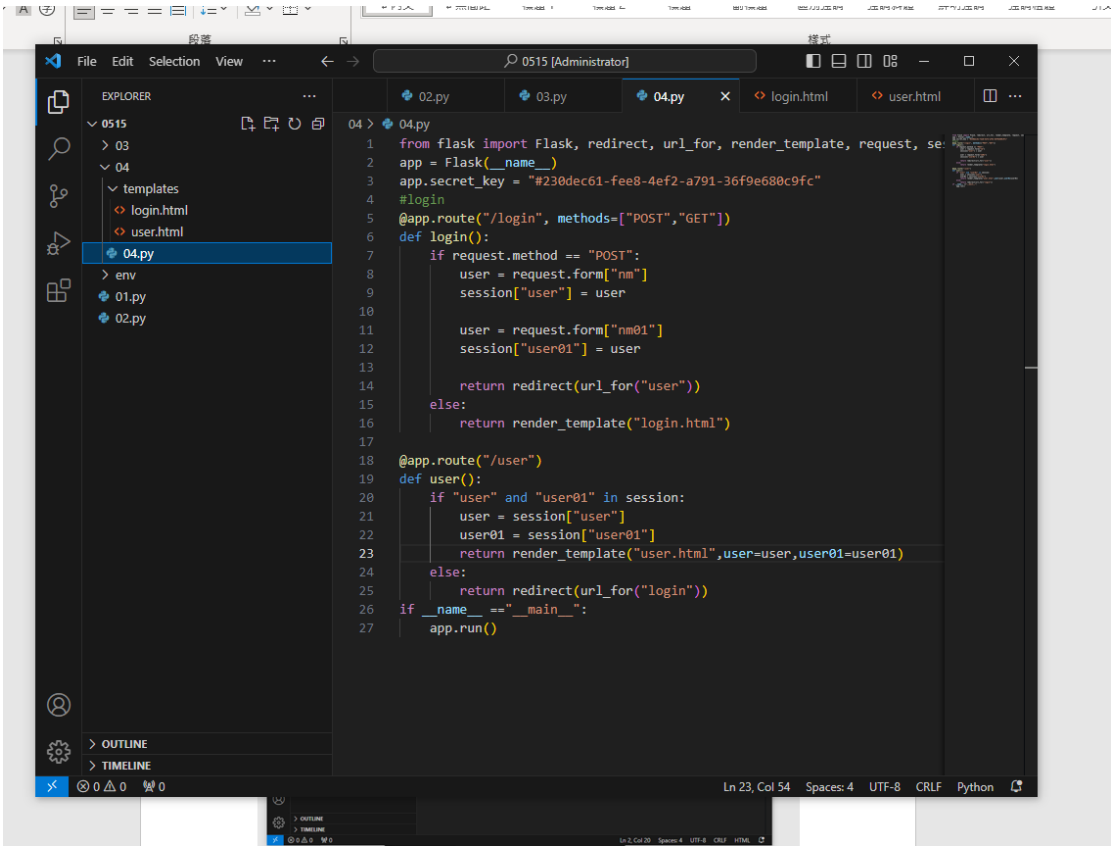
<label for="username">Name:</label>
<input id="username" class="form-control" type="text" name="nm" />

<label for="username01">Name01:</label>
<input id="username01" class="form-control" type="text" name="nm

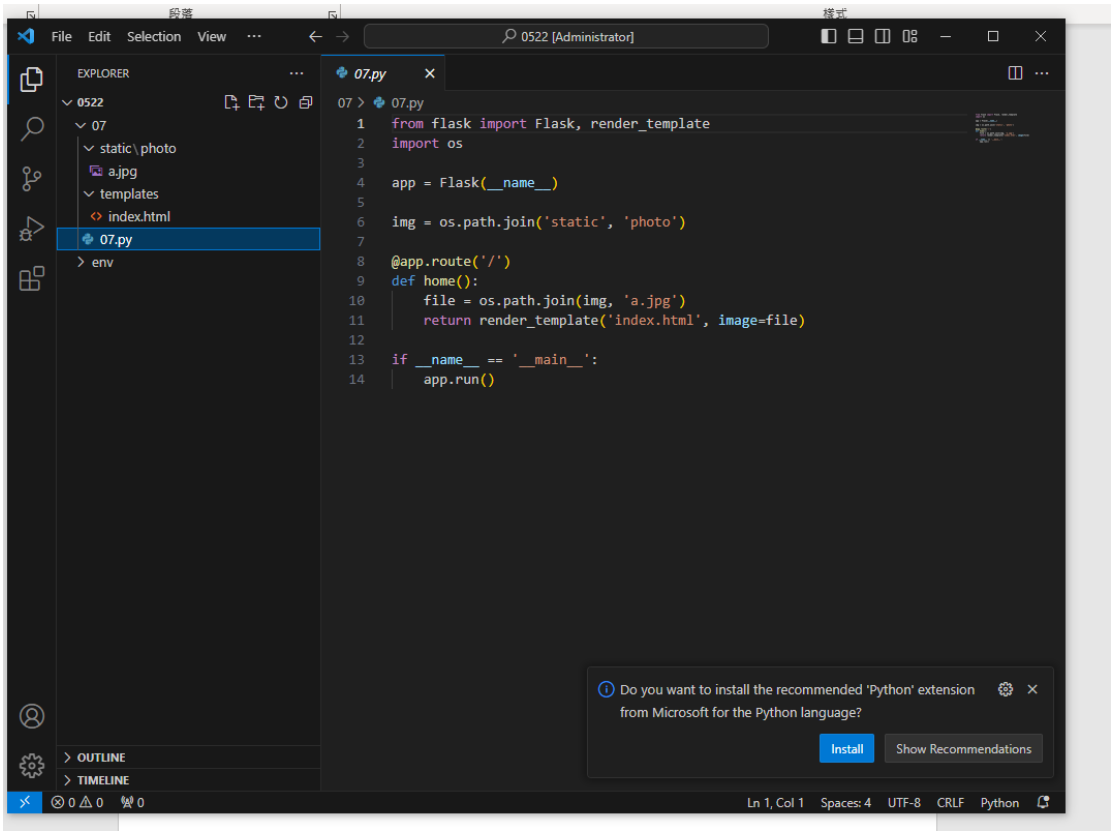
<button type="submit" class="btn btn-primary">Submit</button>

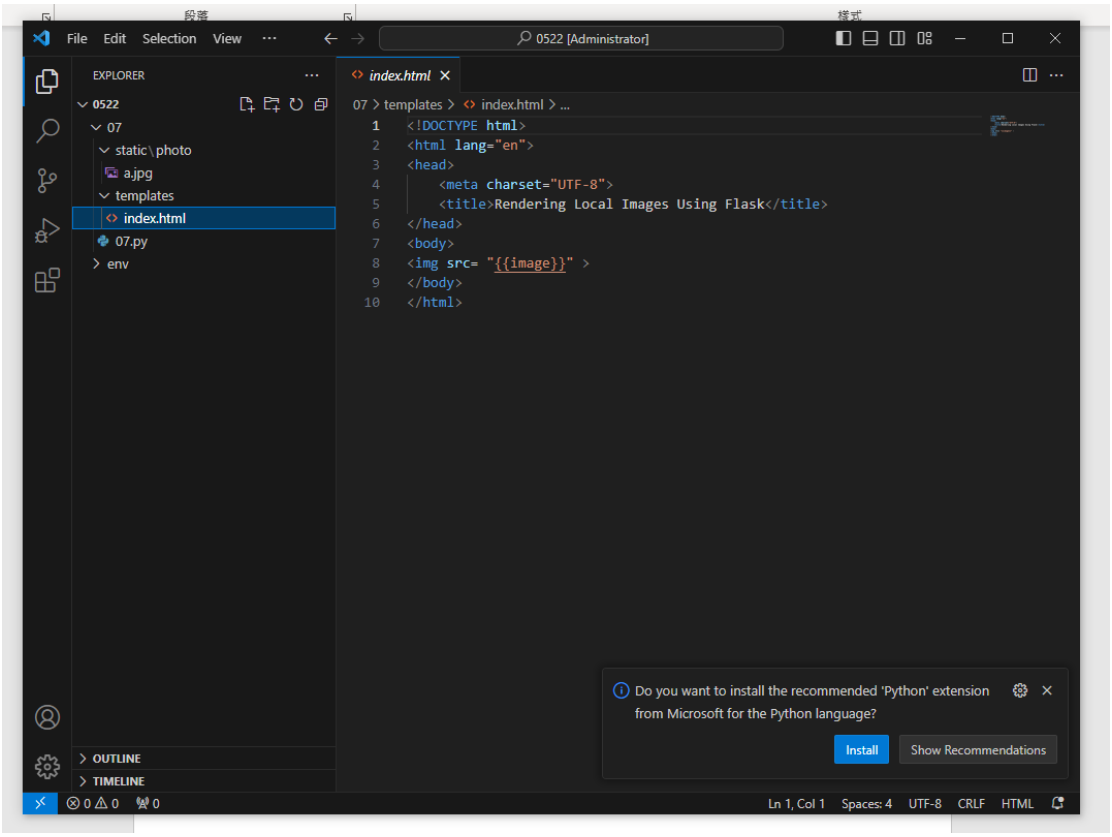
</form>
OUTLINE
TIMELINE
Ln 8, Col 39
Spaces: 4
UTF-8
CRLF
HTML

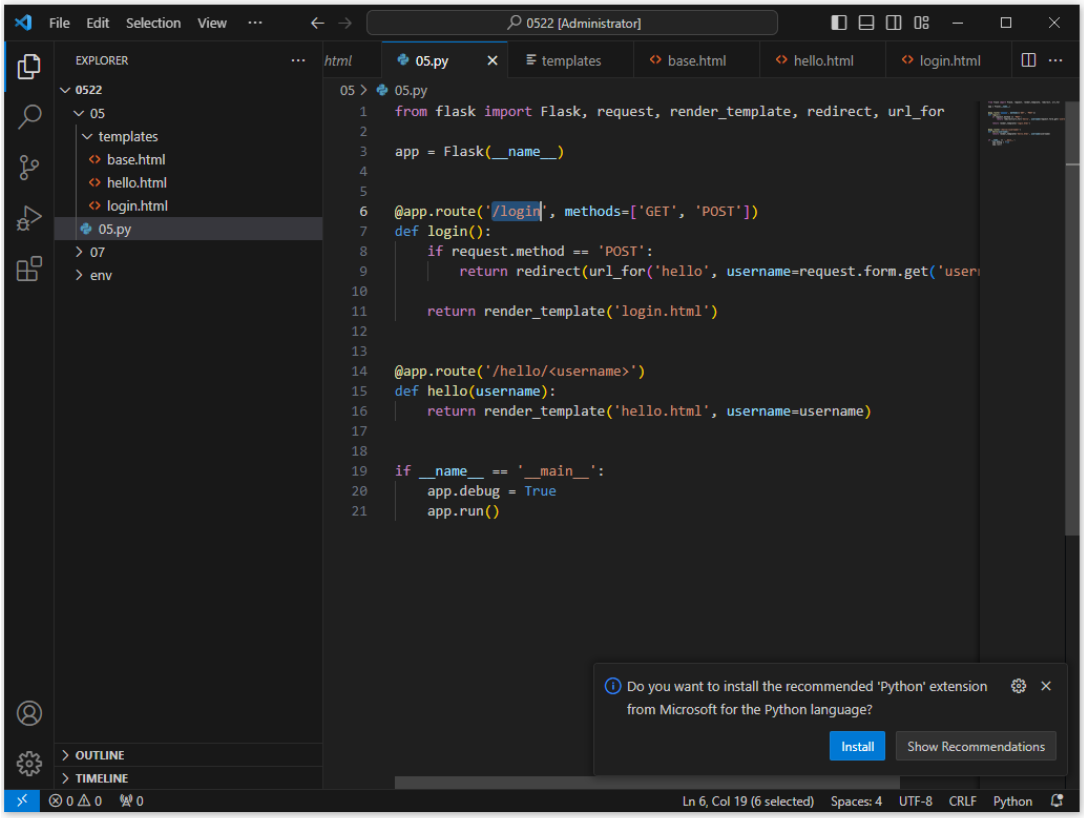

File Edit Selection View
0515 [Administrator]
EXPLORER
0515
03
04
templates
login.html
user.html
04.py
env
01.py
02.py
02.py
03.py
04.py
login.html
user.html
04 > templates > user.html > h1
1 <h1>Hello {{ user }}!</h1>
2 <h1>Hello {{ user01 }}!</h1>
OUTLINE
TIMELINE
Ln 2, Col 20 Spaces: 4 UTF-8 CRLF HTML

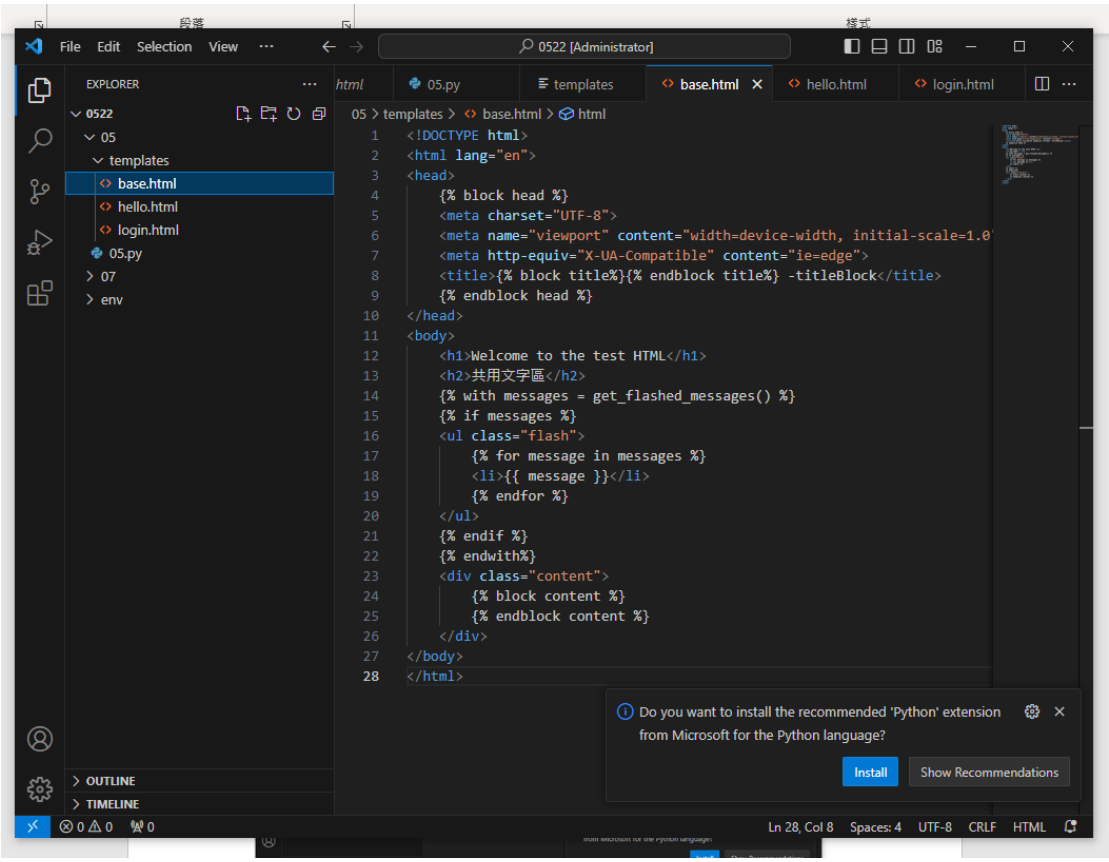

File Edit Selection View
0515 [Administrator]
EXPLORER
0515
03
04
templates
login.html
user.html
04.py
env
01.py
02.py
02.py
03.py
04.py
login.html
user.html
04 > 04.py
1 from flask import Flask, redirect, url_for, render_template, request, se
2 app = Flask(__name__)
3 app.secret_key = "#230dec61-fee8-4ef2-a791-36f9e680c9fc"
4 #login
5 @app.route("/login", methods=["POST","GET"])
6 def login():
7 if request.method == "POST":
8 user = request.form["nm"]
9 session["user"] = user
10
11 user = request.form["nm01"]
12 session["user01"] = user
13
14 return redirect(url_for("user"))
15 else:
16 return render_template("login.html")
17
18 @app.route("/user")
19 def user():
20 if "user" and "user01" in session:
21 user = session["user"]
22 user01 = session["user01"]
23 return render_template("user.html",user=user,user01=user01)
24 else:
25 return redirect(url_for("login"))
26 if __name__ =="__main__":
27 app.run()
OUTLINE
TIMELINE
Ln 23, Col 54 Spaces: 4 UTF-8 CRLF Python

0522


File Edit Selection View
0522 [Administrator]
EXPLORER
0522
07
static\photo
a.jpg
templates
index.html
07.py
env
07.py
07 > 07.py
1 from flask import Flask, render_template
2 import os
3
4 app = Flask(__name__)
5
6 img = os.path.join('static', 'photo')
7
8 @app.route('/')
9 def home():
10 file = os.path.join(img, 'a.jpg')
11 return render_template('index.html', image=file)
12
13 if __name__ == '__main__':
14 app.run()
Do you want to install the recommended 'Python' extension from Microsoft for the Python language?
Install
Show Recommendations
OUTLINE
TIMELINE
Ln 1, Col 1 Spaces: 4 UTF-8 CRLF Python


File Edit Selection View
0522 [Administrator]
EXPLORER
0522
07
static\photo
a.jpg
templates
index.html
07.py
env
index.html
07 > templates > index.html > ...
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4 <meta charset="UTF-8">
5 <title>Rendering Local Images Using Flask</title>
6 </head>
7 <body>
8 <img src= "{{image}}" >
9 </body>
10 </html>
Do you want to install the recommended 'Python' extension from Microsoft for the Python language?
Install
Show Recommendations
OUTLINE
TIMELINE
Ln 1, Col 1 Spaces: 4 UTF-8 CRLF HTML

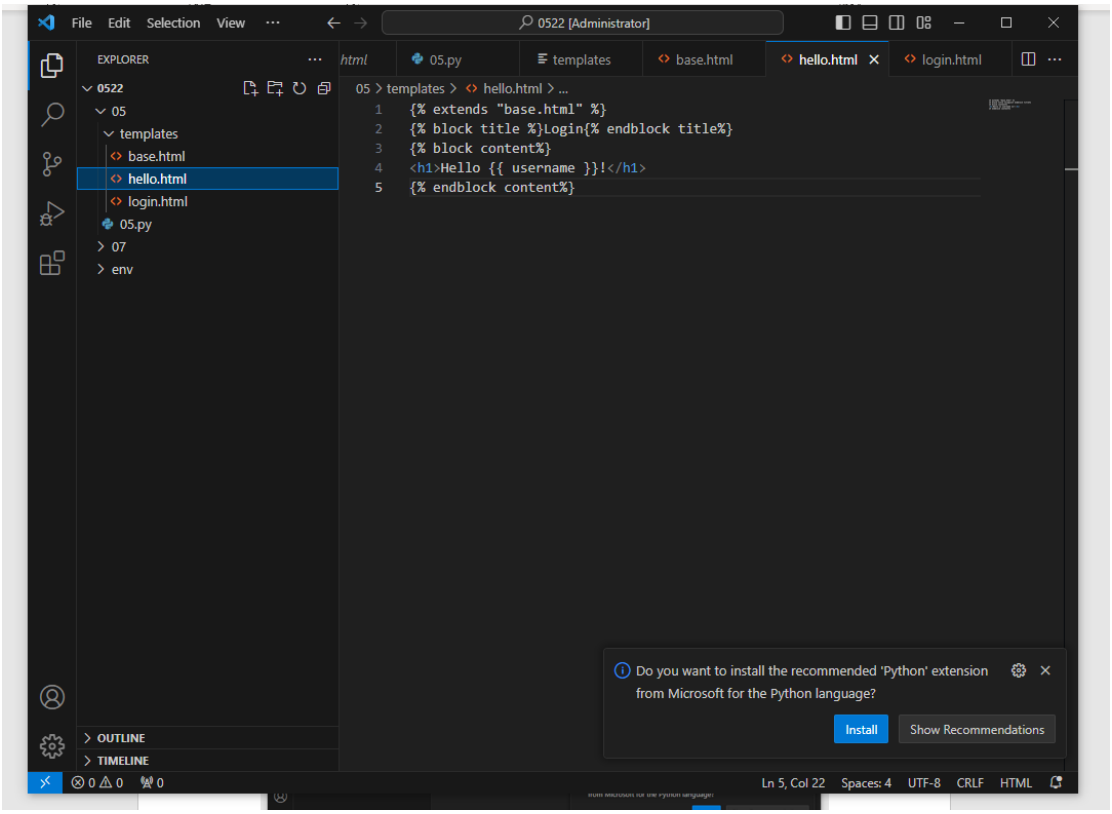

File Edit Selection View
0522 [Administrator]
EXPLORER
0522
05
templates
base.html
hello.html
login.html
05.py
07
env
OUTLINE
TIMELINE
05.py
templates
base.html
hello.html
login.html
from flask import Flask, request, render_template, redirect, url_for
app = Flask(__name__)
@app.route('/login', methods=['GET', 'POST'])
def login():
if request.method == 'POST':
return redirect(url_for('hello', username=request.form.get('user
return render_template('login.html')
@app.route('/hello/<username>')
def hello(username):
return render_template('hello.html', username=username)
if __name__ == '__main__':
app.debug = True
app.run()
Do you want to install the recommended 'Python' extension from Microsoft for the Python language?
Install
Show Recommendations
Ln 6, Col 19 (6 selected)
Spaces: 4
UTF-8
CRLF
Python

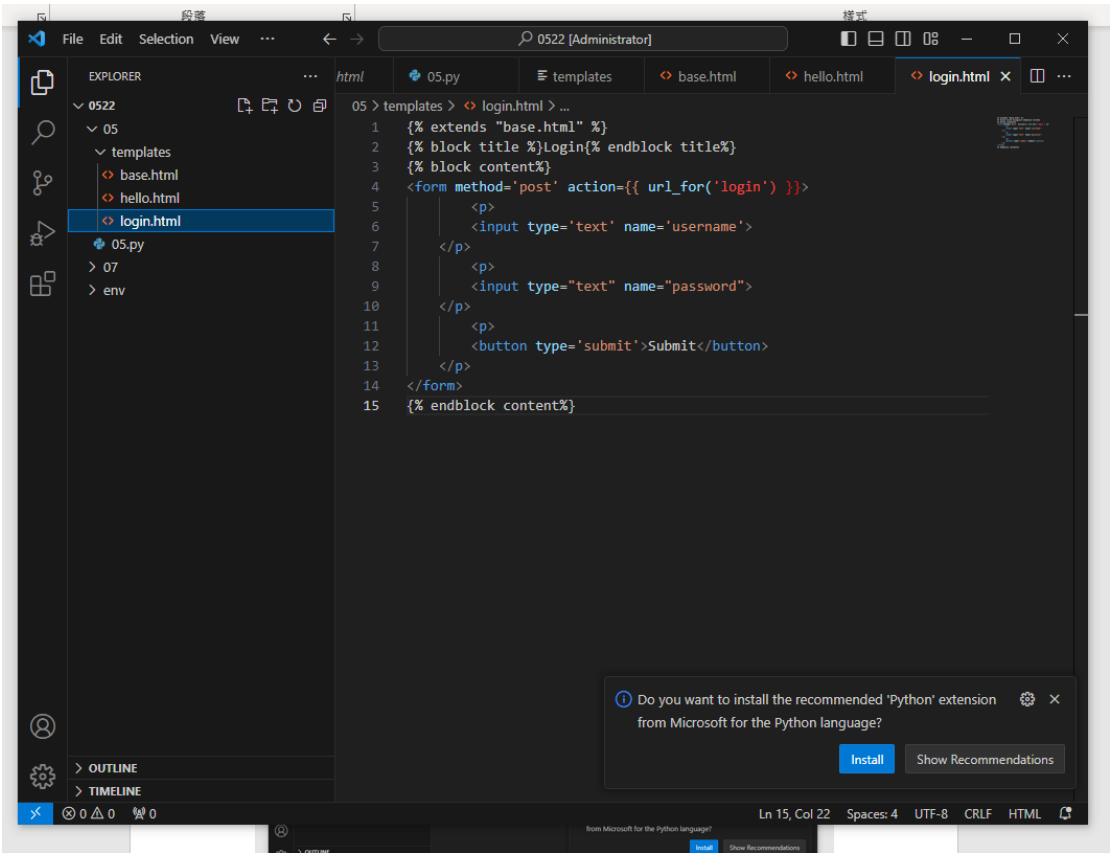

File Edit Selection View
0522 [Administrator]
EXPLORER
0522
05
templates
base.html
hello.html
login.html
05.py
07
env
OUTLINE
TIMELINE
05.py
templates
base.html
hello.html
login.html
05 > templates > base.html > html
<!DOCTYPE html>
<html lang="en">
<head>
{% block head %}
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0
<meta http-equiv="X-UA-Compatible" content="ie=edge">
<title>{% block title%}{% endblock title%} -titleBlock</title>
{% endblock head %}
</head>
<body>
<h1>Welcome to the test HTML</h1>
<h2>共用文字區</h2>
{% with messages = get_flashed_messages() %}
{% if messages %}
<ul class="flash">
{% for message in messages %}
{{ message }}
{% endfor %}

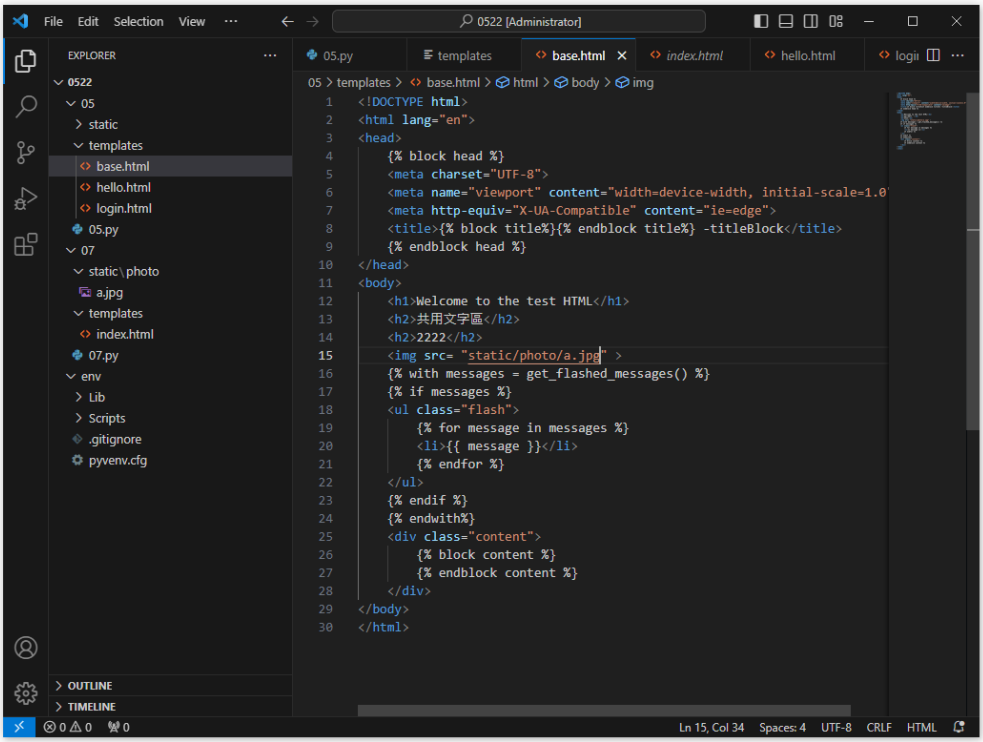
{% endif %}
{% endwith%}

{% block content %}
{% endblock content %}

</body>
</html>
Do you want to install the recommended 'Python' extension from Microsoft for the Python language?
Install
Show Recommendations
Ln 28, Col 8
Spaces: 4
UTF-8
CRLF
HTML


File Edit Selection View
0522 [Administrator]
EXPLORER
0522
05
templates
base.html
hello.html
login.html
05.py
07
env
OUTLINE
TIMELINE
html
05.py
templates
base.html
hello.html
login.html
05 > templates > hello.html > ...
1 {% extends "base.html" %}
2 {% block title %}Login{% endblock title%}
3 {% block content%}
4 <h1>Hello {{ username }}!</h1>
5 {% endblock content%}
Do you want to install the recommended 'Python' extension from Microsoft for the Python language?
Install
Show Recommendations
Ln 5, Col 22
Spaces: 4
UTF-8
CRLF
HTML

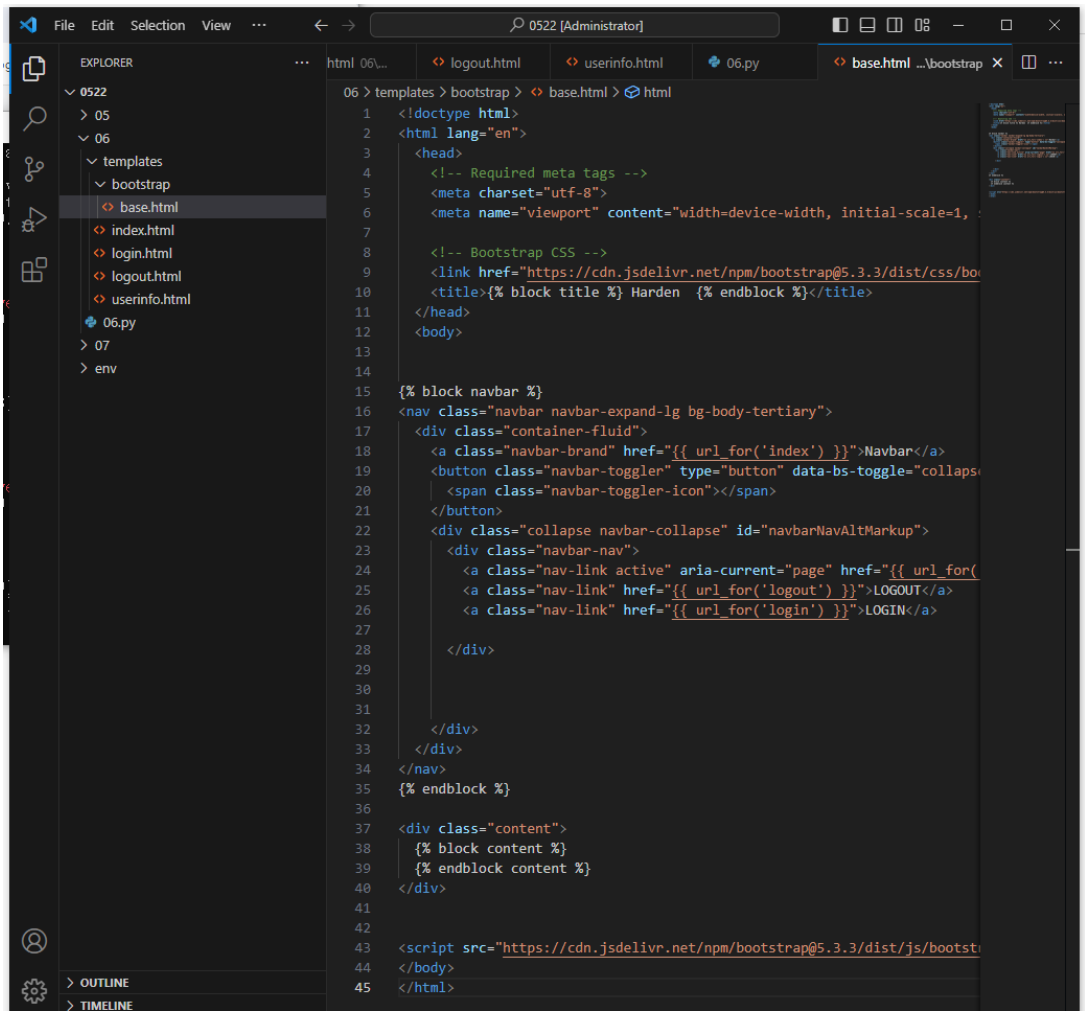

File Edit Selection View
0522 [Administrator]
EXPLORER
0522
05
templates
base.html
hello.html
login.html
05.py
07
env
OUTLINE
TIMELINE
html
05.py
templates
base.html
hello.html
login.html
05 > templates > login.html > ...
1 {% extends "base.html" %}
2 {% block title %}Login{% endblock title%}
3 {% block content%}
4 <form method='post' action={{ url_for('login') }}>
5
6 <input type='text' name='username'>
7
8
9 <input type="text" name="password">
10
11
12 <button type='submit'>Submit</button>
13
14 </form>
15 {% endblock content%}
Do you want to install the recommended 'Python' extension from Microsoft for the Python language?
Install
Show Recommendations
Ln 15, Col 22
Spaces: 4
UTF-8
CRLF
HTML


File Edit Selection View
0522 [Administrator]
EXPLORER
0522
05
static
templates
base.html
hello.html
login.html
05.py
07
static\photo
a.jpg
templates
index.html
07.py
env
Lib
Scripts
.gitignore
pyvenv.cfg
OUTLINE
TIMELINE
05.py
templates
base.html
index.html
hello.html
05 > templates > base.html > html > body > img
<!DOCTYPE html>
<html lang="en">
<head>
{% block head %}
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0
<meta http-equiv="X-UA-Compatible" content="ie=edge">
<title>{% block title%}{% endblock title%} -titleBlock</title>
{% endblock head %}
</head>
<body>
<h1>Welcome to the test HTML</h1>
<h2>共用文字區</h2>
<h2>2222</h2>
<img src= "static/photo/a.jpg" >
{% with messages = get_flashed_messages() %}
{% if messages %}
<ul class="flash">
{% for message in messages %}
{{ message }}
{% endfor %}

{% endif %}
{% endwith%}

{% block content %}
{% endblock content %}

</body>
</html>
Ln 15, Col 34
Spaces: 4
UTF-8
CRLF
HTML


File Edit Selection View
0522 [Administrator]
EXPLORER
0522
05
06
templates
bootstrap
base.html
index.html
login.html
logout.html
userinfo.html
06.py
07
env
logout.html
userinfo.html
06.py
base.html ...\bootstrap
06 > templates > bootstrap > base.html > html
<!doctype html>
<html lang="en">
<head>
<!-- Required meta tags -->
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1,
<!-- Bootstrap CSS -->
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/css/bo
<title>{% block title %} Harden {% endblock %}</title>
</head>
<body>
{% block navbar %}
<nav class="navbar navbar-expand-lg bg-body-tertiary">

Navbar
<button class="navbar-toggler" type="button" data-bs-toggle="collaps

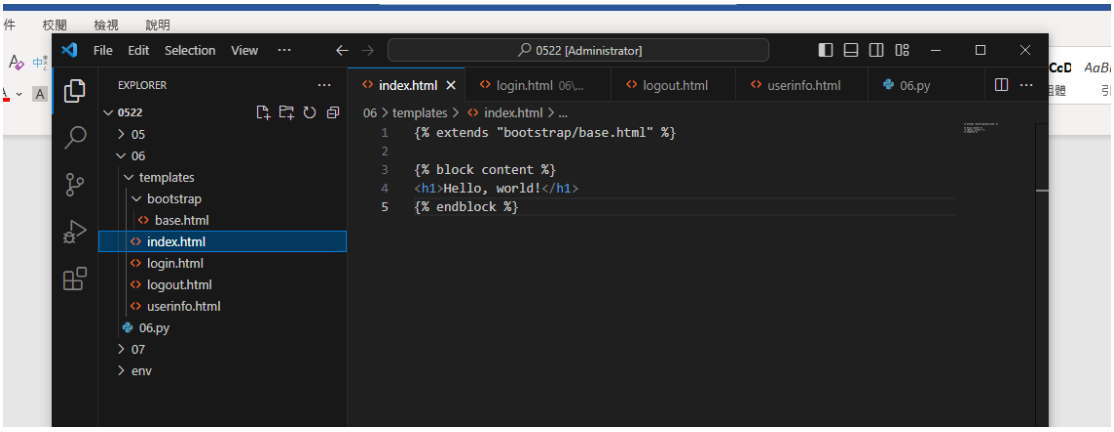
</button>

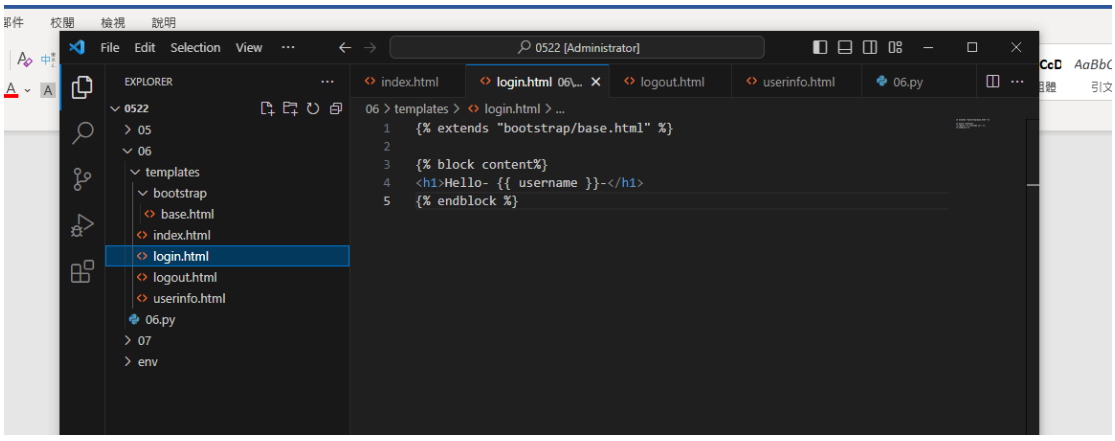
<a class="nav-link active" aria-current="page" href="{{ url_for(
LOGOUT
LOGIN

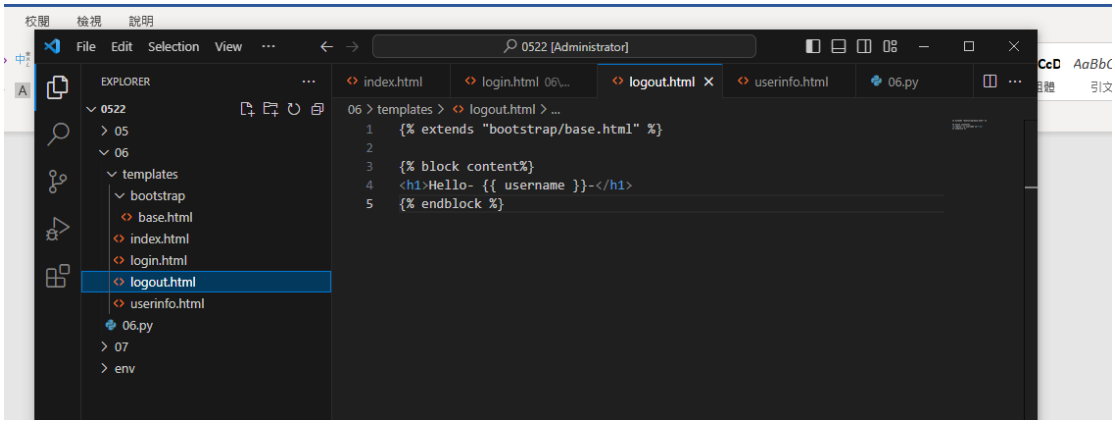
</nav>
{% endblock %}

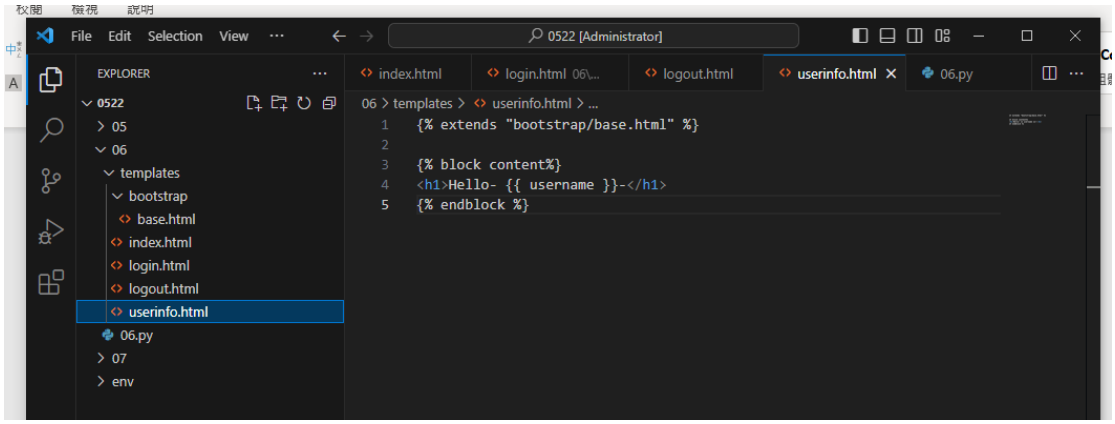
{% block content %}
{% endblock content %}

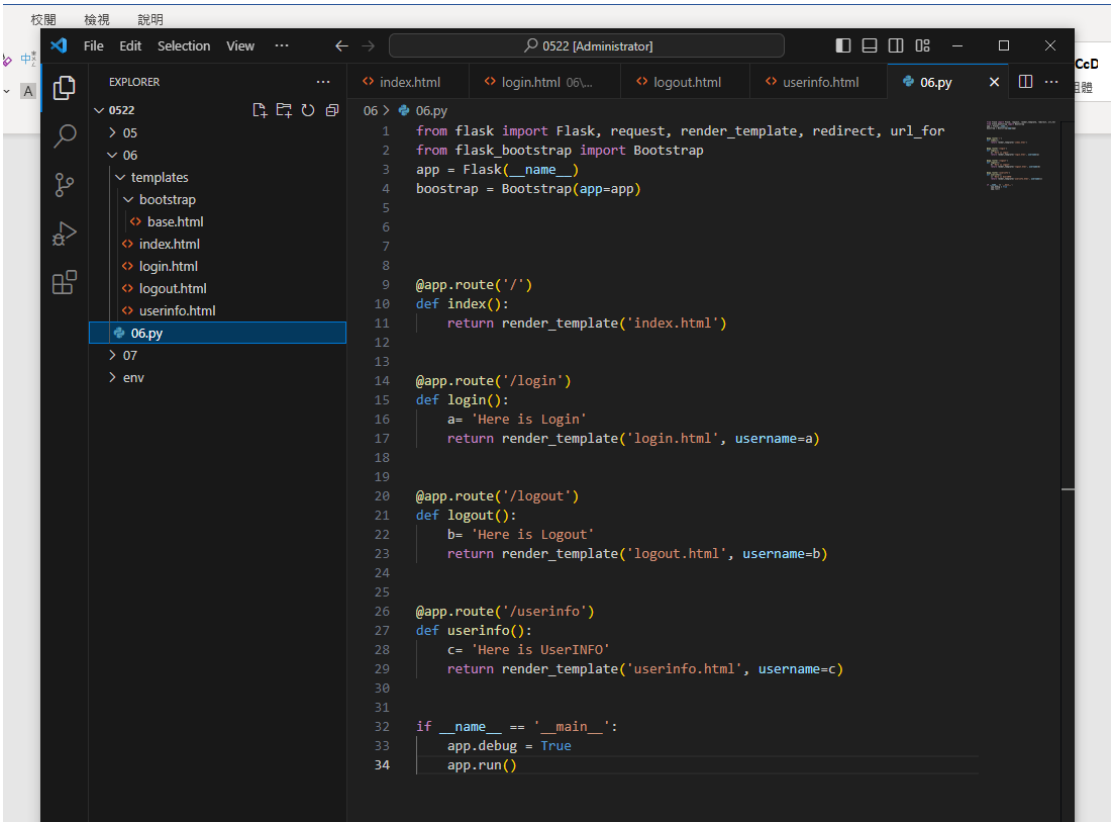
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootst
</body>
</html>
OUTLINE
TIMELINE


File Edit Selection View
0522 [Administrator]
EXPLORER
0522
05
06
templates
bootstrap
base.html
index.html
login.html
logout.html
userinfo.html
06.py
07
env
index.html
login.html 06\...
logout.html
userinfo.html
06.py
06 > templates > index.html > ...
{% extends "bootstrap/base.html" %}
{% block content %}
<h1>Hello, world!</h1>
{% endblock %}

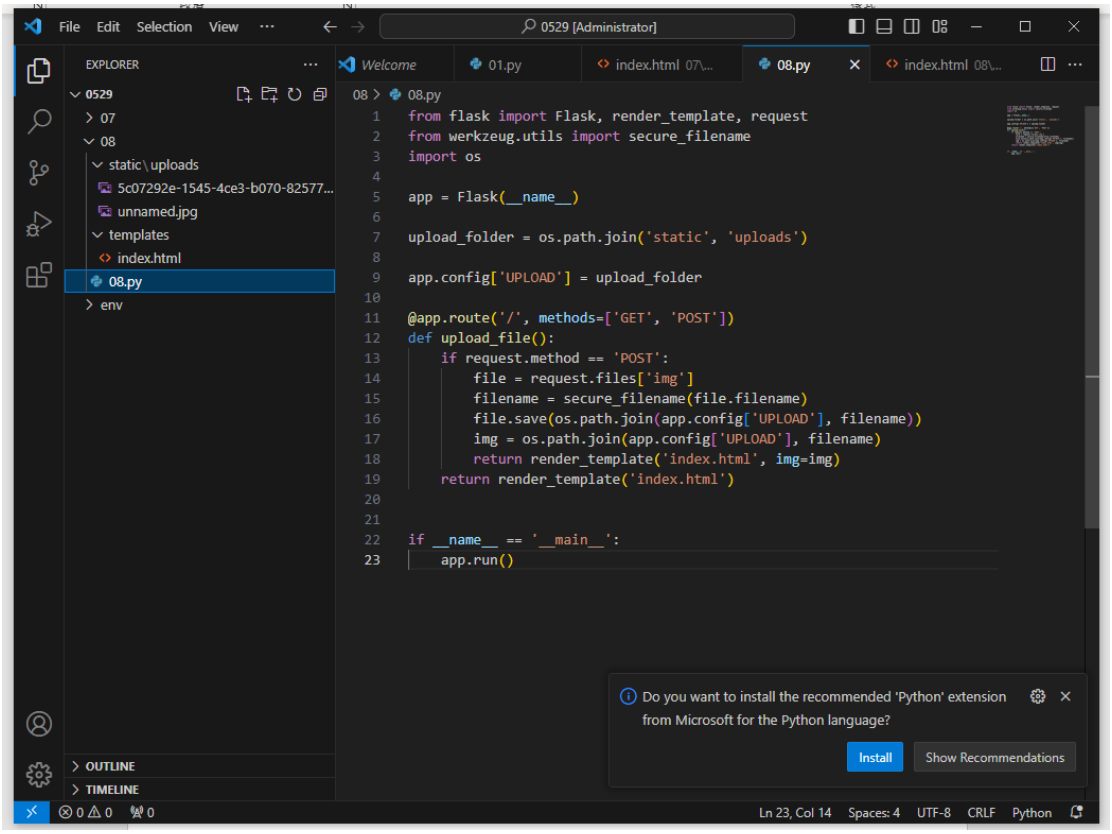

File Edit Selection View
0522 [Administrator]
EXPLORER
0522
05
06
templates
bootstrap
base.html
index.html
login.html
logout.html
userinfo.html
06.py
07
env
index.html
login.html 06\...
logout.html
userinfo.html
06.py
06 > templates > login.html > ...
1 {% extends "bootstrap/base.html" %}
2
3 {% block content%}
4 <h1>Hello- {{ username }}-</h1>
5 {% endblock %}

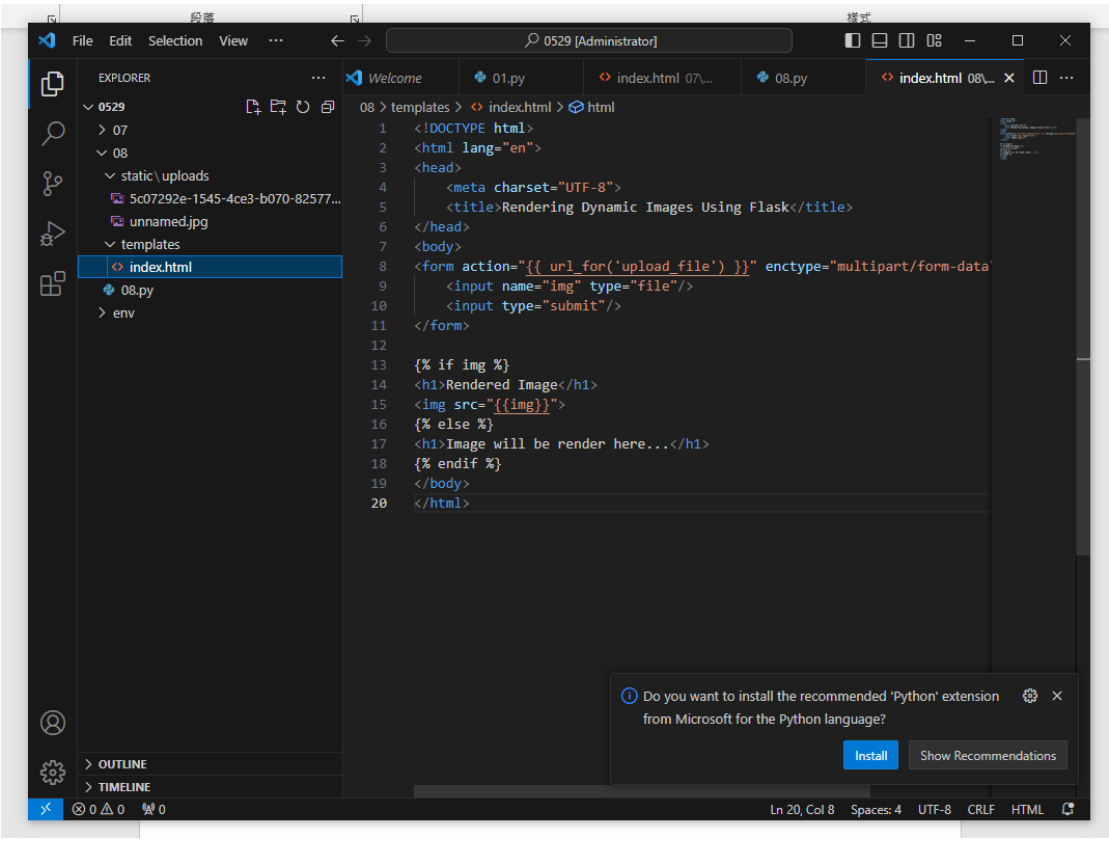

File Edit Selection View
0522 [Administrator]
EXPLORER
0522
05
06
templates
bootstrap
base.html
index.html
login.html
logout.html
userinfo.html
06.py
07
env
index.html
login.html 06\...
logout.html
userinfo.html
06.py
06 > templates > logout.html > ...
1 {% extends "bootstrap/base.html" %}
2
3 {% block content%}
4 <h1>Hello- {{ username }}-</h1>
5 {% endblock %}

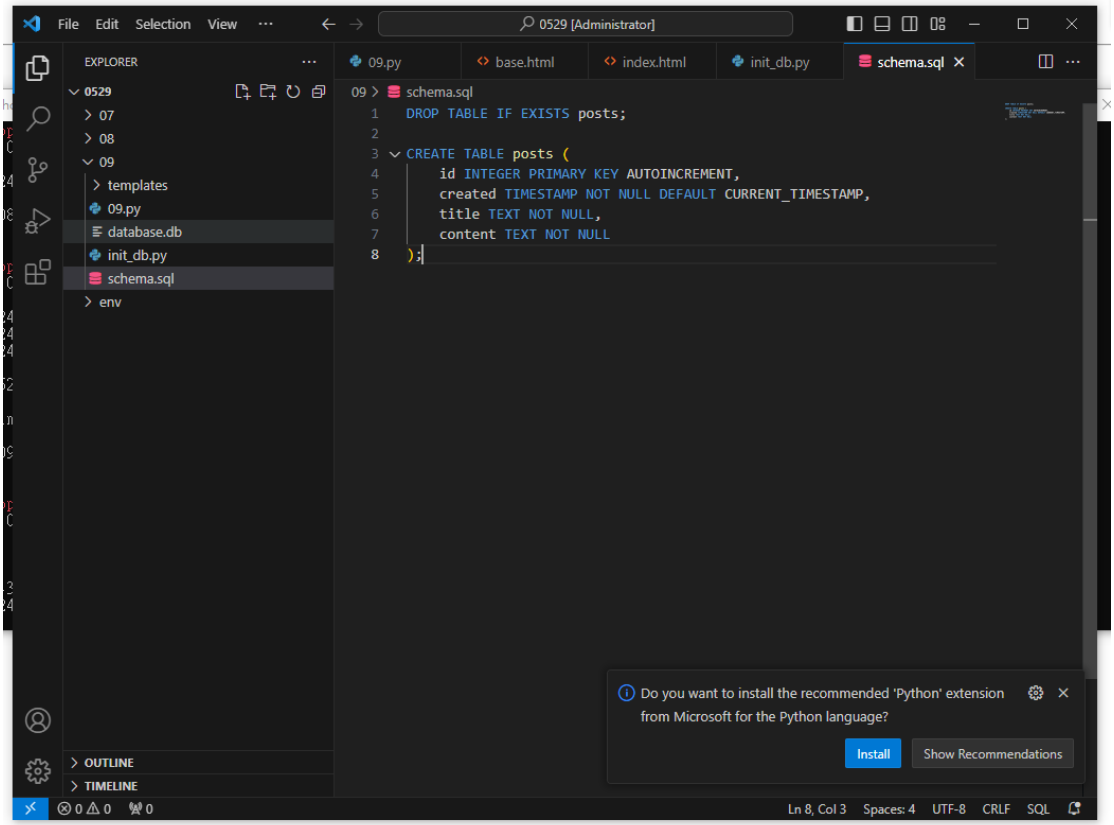

File Edit Selection View
0522 [Administrator]
EXPLORER
0522
05
06
templates
bootstrap
base.html
index.html
login.html
logout.html
userinfo.html
06.py
07
env
index.html
login.html 06\...
logout.html
userinfo.html
06.py
06 > templates > userinfo.html > ...
1 {% extends "bootstrap/base.html" %}
2
3 {% block content%}
4 <h1>Hello- {{ username }}-</h1>
5 {% endblock %}

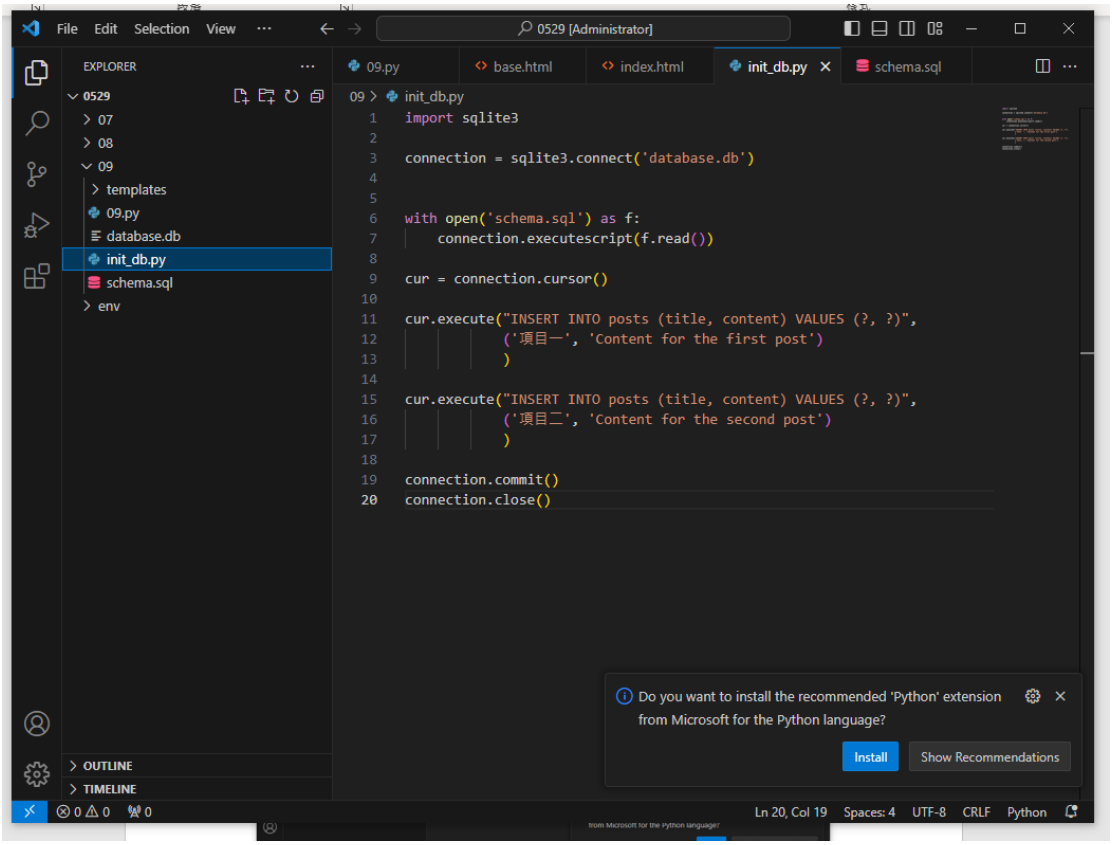

File Edit Selection View
0522 [Administrator]
EXPLORER
0522
05
06
templates
bootstrap
base.html
index.html
login.html
logout.html
userinfo.html
06.py
07
env
index.html
login.html 06\...
logout.html
userinfo.html
06.py
06 > 06.py
from flask import Flask, request, render_template, redirect, url_for
from flask_bootstrap import Bootstrap
app = Flask(__name__)
boostrap = Bootstrap(app=app)
@app.route('/')
def index():
return render_template('index.html')
@app.route('/login')
def login():
a= 'Here is Login'
return render_template('login.html', username=a)
@app.route('/logout')
def logout():
b= 'Here is Logout'
return render_template('logout.html', username=b)
@app.route('/userinfo')
def userinfo():
c= 'Here is UserINFO'
return render_template('userinfo.html', username=c)
if __name__ == '__main__':
app.debug = True
app.run()

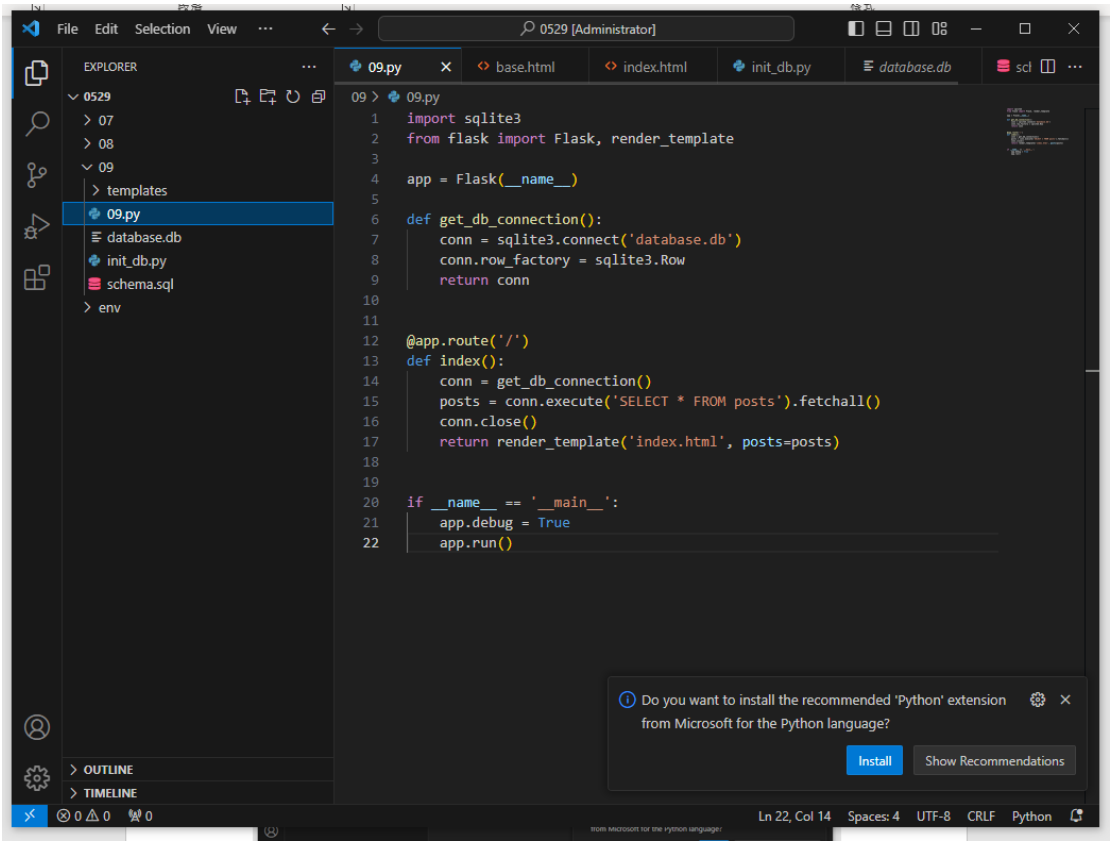
0529

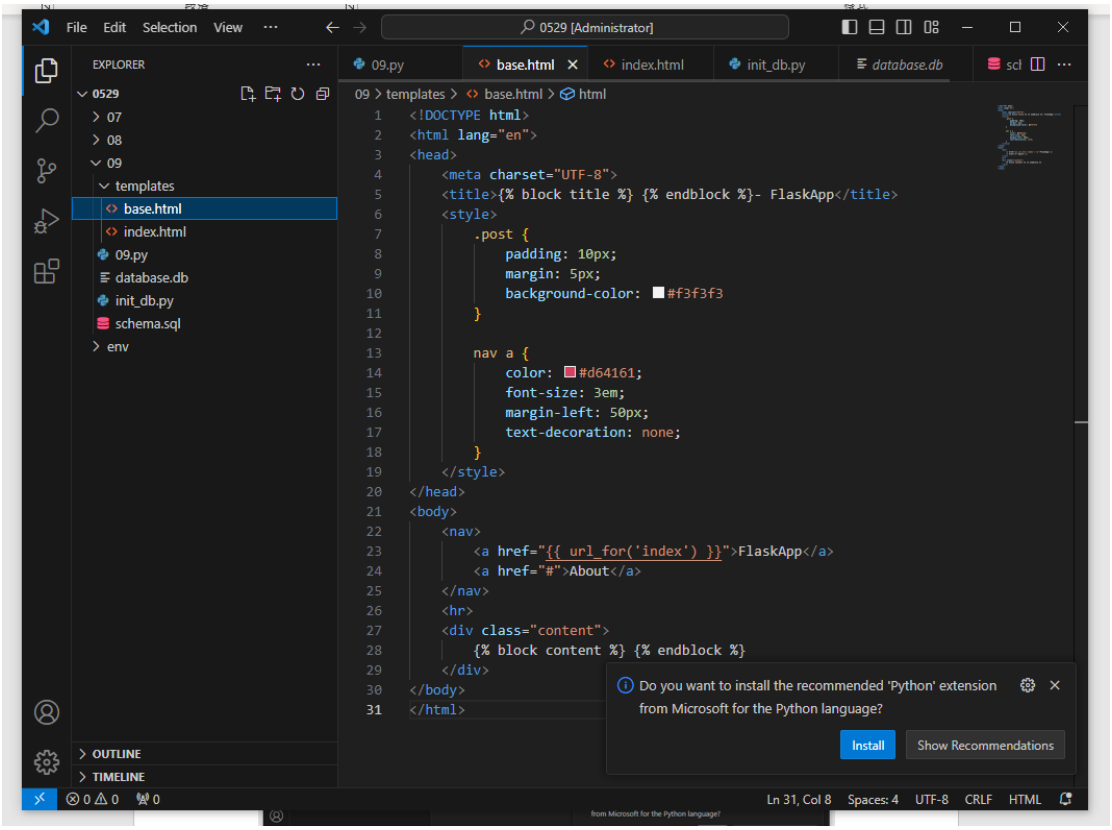

File Edit Selection View
0529 [Administrator]
EXPLORER
0529
07
08
static\uploads
5c07292e-1545-4ce3-b070-82577...
unnamed.jpg
templates
index.html
08.py
env
Welcome
01.py
index.html 07\...
08.py
index.html 08\...
08 > 08.py
from flask import Flask, render_template, request
from werkzeug.utils import secure_filename
import os
app = Flask(__name__)
upload_folder = os.path.join('static', 'uploads')
app.config['UPLOAD'] = upload_folder
@app.route('/', methods=['GET', 'POST'])
def upload_file():
if request.method == 'POST':
file = request.files['img']
filename = secure_filename(file.filename)
file.save(os.path.join(app.config['UPLOAD'], filename))
img = os.path.join(app.config['UPLOAD'], filename)
return render_template('index.html', img=img)
return render_template('index.html')
if __name__ == '__main__':
app.run()
Do you want to install the recommended 'Python' extension from Microsoft for the Python language?
Install
Show Recommendations
OUTLINE
TIMELINE
Ln 23, Col 14
Spaces: 4
UTF-8
CRLF
Python


File Edit Selection View
0529 [Administrator]
EXPLORER
0529
07
08
static\uploads
5c07292e-1545-4ce3-b070-82577...
unnamed.jpg
templates
index.html
08.py
env
Welcome
01.py
index.html 07\...
08.py
index.html 08\...
08 > templates > index.html > html
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<title>Rendering Dynamic Images Using Flask</title>
</head>
<body>
<form action="{{ url_for('upload_file') }}" enctype="multipart/form-data
<input name="img" type="file"/>
<input type="submit"/>
</form>
{% if img %}
<h1>Rendered Image</h1>
<img src="{{img}}">
{% else %}
<h1>Image will be render here...</h1>
{% endif %}
</body>
</html>
Do you want to install the recommended 'Python' extension from Microsoft for the Python language?
Install
Show Recommendations
OUTLINE
TIMELINE
Ln 20, Col 8
Spaces: 4
UTF-8
CRLF
HTML


File Edit Selection View
0529 [Administrator]
EXPLORER
0529
07
08
09
templates
09.py
database.db
init_db.py
schema.sql
env
09.py
base.html
index.html
init_db.py
schema.sql
09 > schema.sql
DROP TABLE IF EXISTS posts;
CREATE TABLE posts (
id INTEGER PRIMARY KEY AUTOINCREMENT,
created TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP,
title TEXT NOT NULL,
content TEXT NOT NULL
);
Do you want to install the recommended 'Python' extension from Microsoft for the Python language?
Install
Show Recommendations
OUTLINE
TIMELINE
Ln 8, Col 3
Spaces: 4
UTF-8
CRLF
SQL

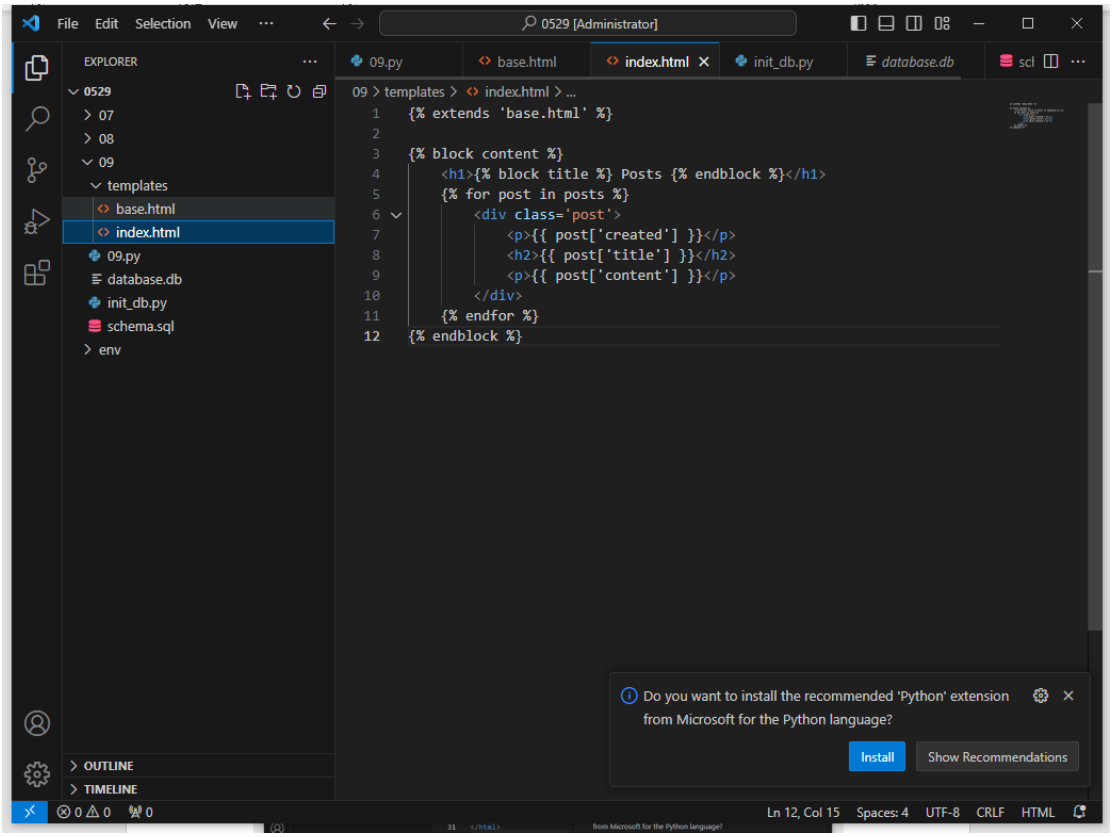

File Edit Selection View
0529 [Administrator]
EXPLORER
0529
07
08
09
templates
09.py
database.db
init_db.py
schema.sql
env
09.py
base.html
index.html
init_db.py
schema.sql
09 > init_db.py
import sqlite3
connection = sqlite3.connect('database.db')
with open('schema.sql') as f:
connection.executescript(f.read())
cur = connection.cursor()
cur.execute("INSERT INTO posts (title, content) VALUES (?, ?)",
('項目一', 'Content for the first post')
)
cur.execute("INSERT INTO posts (title, content) VALUES (?, ?)",
('項目二', 'Content for the second post')
)
connection.commit()
connection.close()
Do you want to install the recommended 'Python' extension from Microsoft for the Python language?
Install
Show Recommendations
OUTLINE
TIMELINE
Ln 20, Col 19
Spaces: 4
UTF-8
CRLF
Python


File Edit Selection View
0529 [Administrator]
EXPLORER
0529
07
08
09
templates
09.py
database.db
init_db.py
schema.sql
env
09.py
base.html
index.html
init_db.py
database.db
09 > 09.py
import sqlite3
from flask import Flask, render_template
app = Flask(__name__)
def get_db_connection():
conn = sqlite3.connect('database.db')
conn.row_factory = sqlite3.Row
return conn
@app.route('/')
def index():
conn = get_db_connection()
posts = conn.execute('SELECT * FROM posts').fetchall()
conn.close()
return render_template('index.html', posts=posts)
if __name__ == '__main__':
app.debug = True
app.run()
Do you want to install the recommended 'Python' extension from Microsoft for the Python language?
Install
Show Recommendations
OUTLINE
TIMELINE
Ln 22, Col 14
Spaces: 4
UTF-8
CRLF
Python


File Edit Selection View
0529 [Administrator]
EXPLORER
0529
07
08
09
templates
base.html
index.html
09.py
database.db
init_db.py
schema.sql
env
09.py
base.html
index.html
init_db.py
database.db
scl
09 > templates > base.html > html
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<title>{% block title %} {% endblock %}- FlaskApp</title>
<style>
.post {
padding: 10px;
margin: 5px;
background-color: #f3f3f3
}
nav a {
color: #d64161;
font-size: 3em;
margin-left: 50px;
text-decoration: none;
}
</style>
</head>
<body>
<nav>
FlaskApp
About
</nav>
<hr>

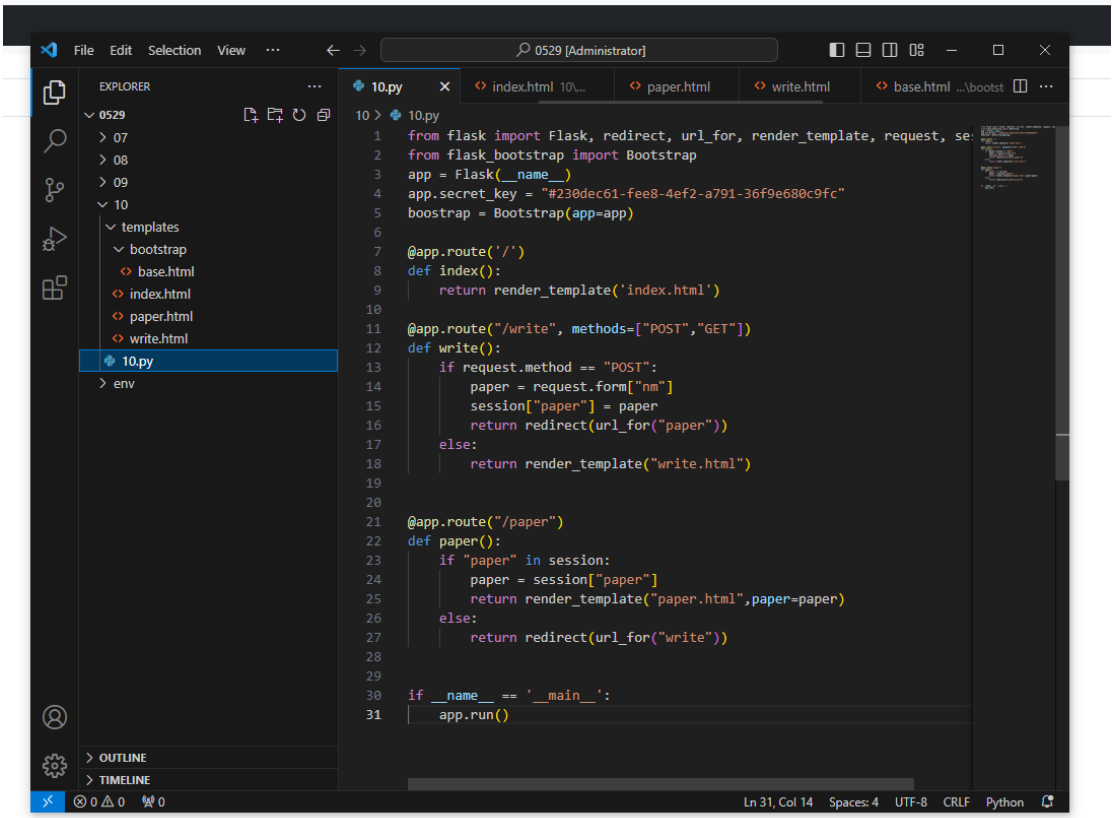
{% block content %} {% endblock %}

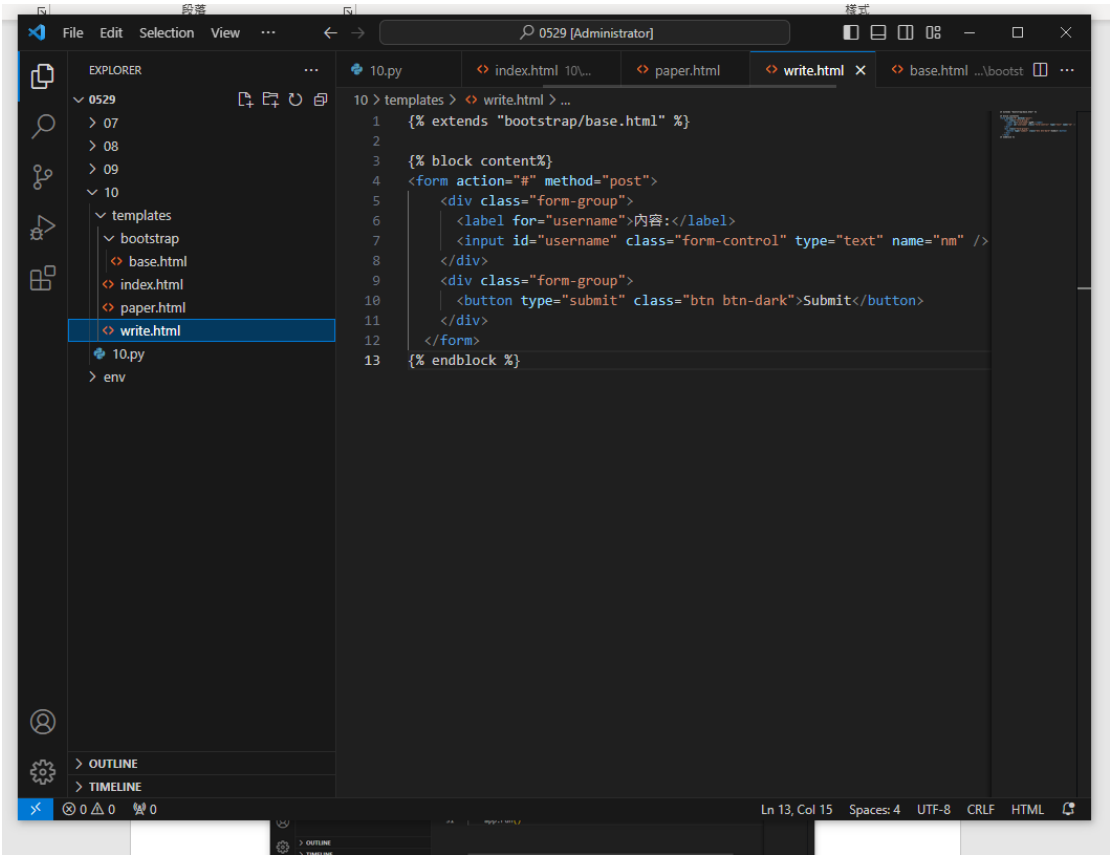
</body>
</html>
Do you want to install the recommended 'Python' extension from Microsoft for the Python language?
Install
Show Recommendations
OUTLINE
TIMELINE
Ln 31, Col 8
Spaces: 4
UTF-8
CRLF
HTML


File Edit Selection View
0529 [Administrator]
EXPLORER
0529
07
08
09
templates
base.html
index.html
09.py
database.db
init_db.py
schema.sql
env
09.py
base.html
index.html
init_db.py
database.db
scl
09 > templates > index.html > ...
{% extends 'base.html' %}
{% block content %}
<h1>{% block title %} Posts {% endblock %}</h1>
{% for post in posts %}

{{ post['created'] }}
<h2>{{ post['title'] }}</h2>
{{ post['content'] }}

{% endfor %}
{% endblock %}
Do you want to install the recommended 'Python' extension from Microsoft for the Python language?
Install
Show Recommendations
OUTLINE
TIMELINE
Ln 12, Col 15
Spaces: 4
UTF-8
CRLF
HTML

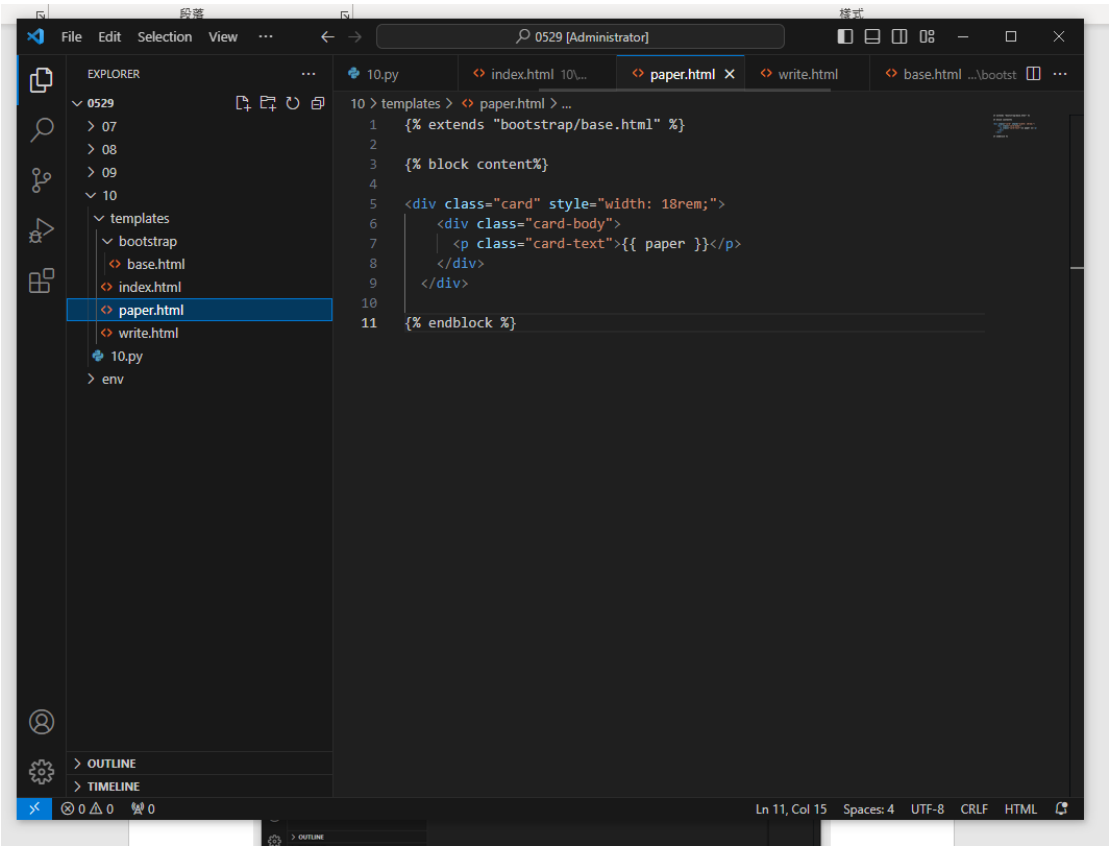

File Edit Selection View
0529 [Administrator]
EXPLORER
0529
07
08
09
10
templates
bootstrap
base.html
index.html
paper.html
write.html
10.py
env
OUTLINE
TIMELINE
10.py
index.html 10\...
paper.html
write.html
base.html ...\bootst
10 > 10.py
from flask import Flask, redirect, url_for, render_template, request, se
from flask_bootstrap import Bootstrap
app = Flask(__name__)
app.secret_key = "#230dec61-fee8-4ef2-a791-36f9e680c9fc"
boostrap = Bootstrap(app=app)
@app.route('/')
def index():
return render_template('index.html')
@app.route("/write", methods=["POST","GET"])
def write():
if request.method == "POST":
paper = request.form["nm"]
session["paper"] = paper
return redirect(url_for("paper"))
else:
return render_template("write.html")
@app.route("/paper")
def paper():
if "paper" in session:
paper = session["paper"]
return render_template("paper.html",paper=paper)
else:
return redirect(url_for("write"))
if __name__ == '__main__':
app.run()
Ln 31, Col 14
Spaces: 4
UTF-8
CRLF
Python

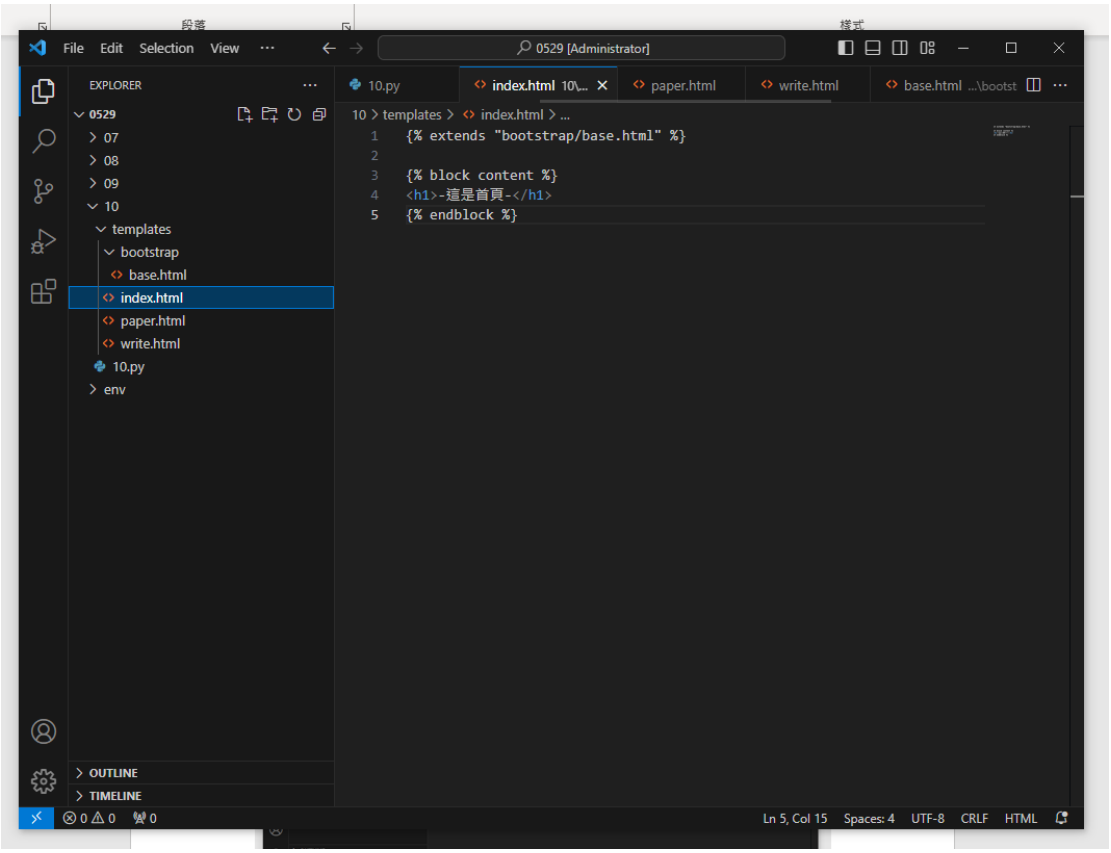

File Edit Selection View
0529 [Administrator]
EXPLORER
0529
07
08
09
10
templates
bootstrap
base.html
index.html
paper.html
write.html
10.py
env
OUTLINE
TIMELINE
10.py
index.html 10\...
paper.html
write.html
base.html ...\bootst
10 > templates > write.html > ...
{% extends "bootstrap/base.html" %}
{% block content%}
<form action="#" method="post">

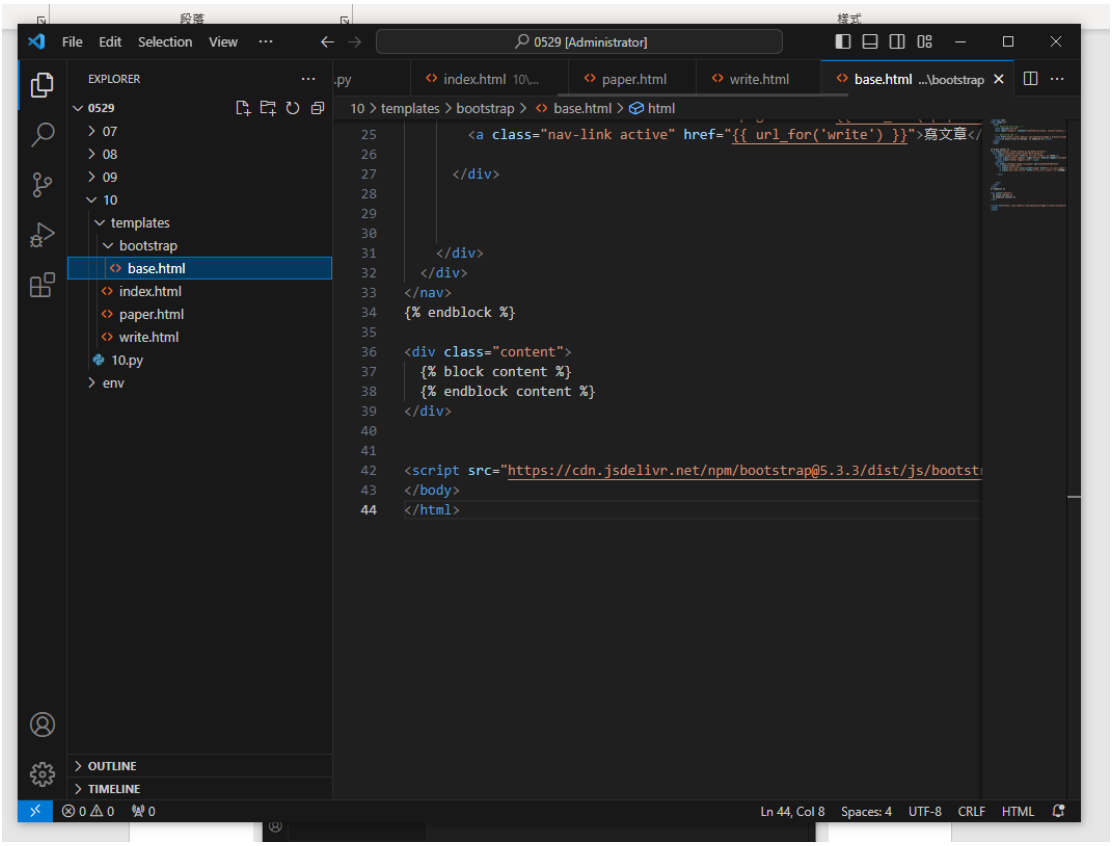
<label for="username">内容:</label>
<input id="username" class="form-control" type="text" name="nm" />

<button type="submit" class="btn btn-dark">Submit</button>

</form>
{% endblock %}
Ln 13, Col 15
Spaces: 4
UTF-8
CRLF
HTML


File Edit Selection View
0529 [Administrator]
EXPLORER
0529
07
08
09
10
templates
bootstrap
base.html
index.html
paper.html
write.html
10.py
env
OUTLINE
TIMELINE
10.py
index.html 10\...
paper.html
write.html
base.html ...\bootst
10 > templates > paper.html > ...
1 {% extends "bootstrap/base.html" %}
2
3 {% block content%}
4
5
6
7 {{ paper }}
8
9
10
11 {% endblock %}
Ln 11, Col 15
Spaces: 4
UTF-8
CRLF
HTML

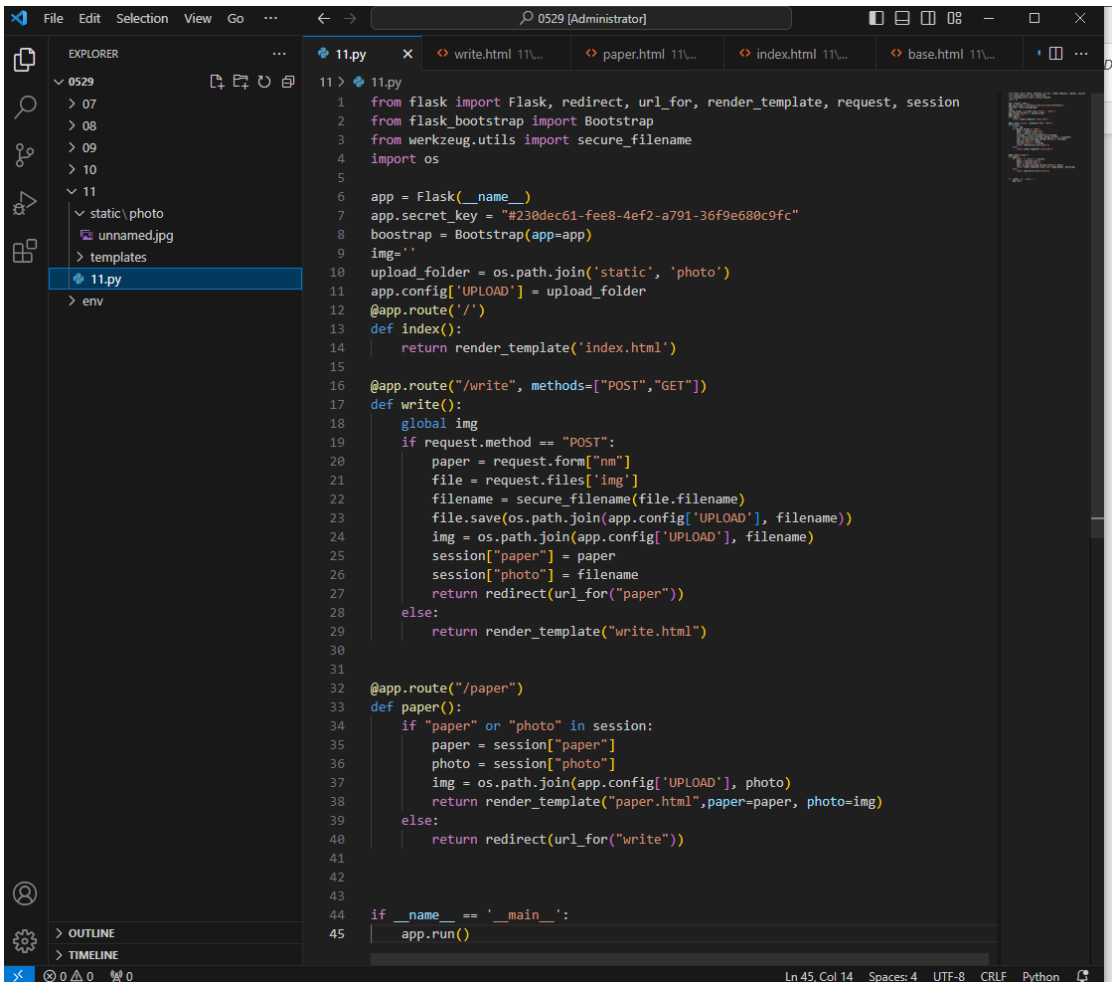

File Edit Selection View
0529 [Administrator]
EXPLORER
0529
07
08
09
10
templates
bootstrap
base.html
index.html
paper.html
write.html
10.py
env
OUTLINE
TIMELINE
10.py
index.html 10\...
paper.html
write.html
base.html ...\bootst
10 > templates > index.html > ...
1 {% extends "bootstrap/base.html" %}
2
3 {% block content %}
4 <h1>-這是首頁-</h1>
5 {% endblock %}
Ln 5, Col 15
Spaces: 4
UTF-8
CRLF
HTML

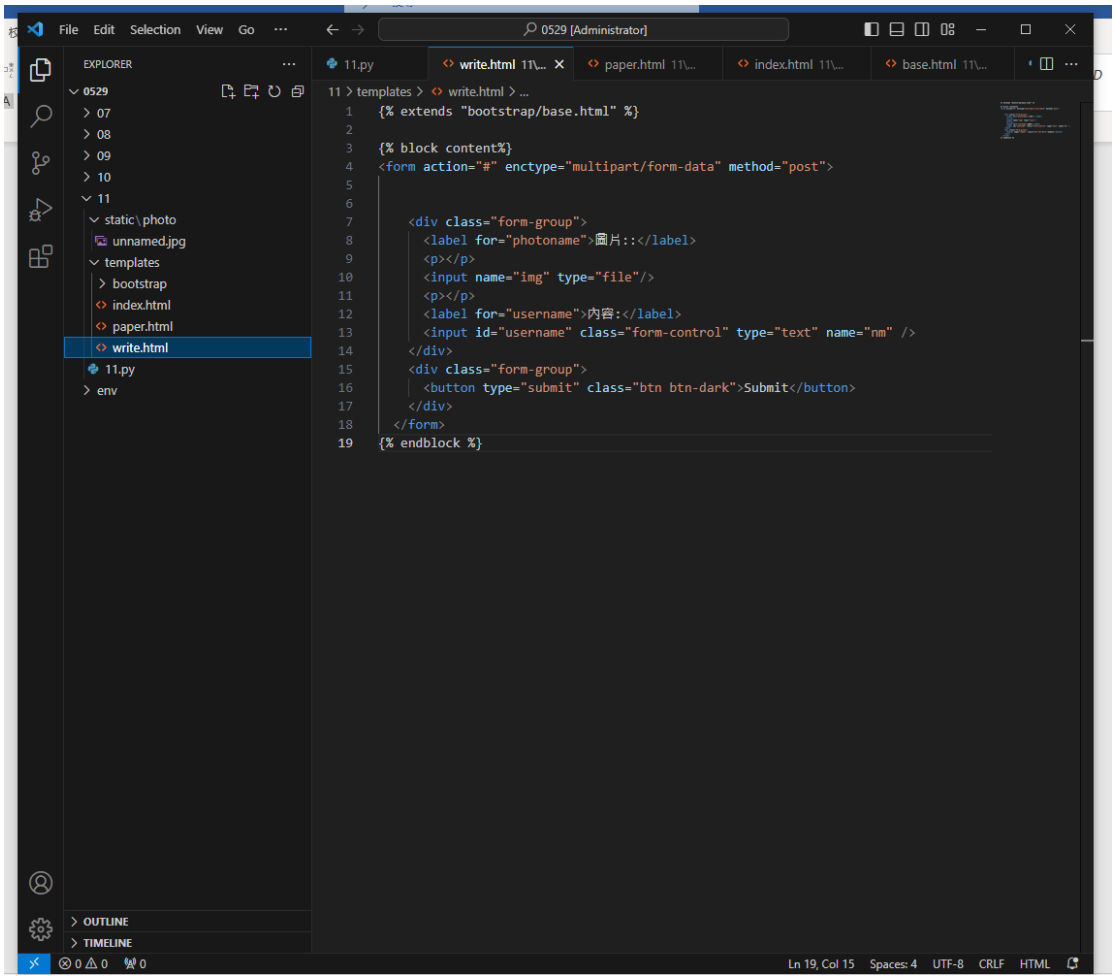

File Edit Selection View
0529 [Administrator]
EXPLORER
0529
07
08
09
10
templates
bootstrap
base.html
index.html
paper.html
write.html
10.py
env
OUTLINE
TIMELINE
index.html 10\...
paper.html
write.html
base.html ...\bootstrap
10 > templates > bootstrap > base.html > html
<a class="nav-link active" href="{{ url_for('write') }}">寫文章</

</nav>
{% endblock %}

{% block content %}
{% endblock content %}

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootst
</body>
</html>
Ln 44, Col 8
Spaces: 4
UTF-8
CRLF
HTML


File Edit Selection View Go
0529 [Administrator]
EXPLORER
0529
07
08
09
10
11
static\photo
unnamed.jpg
templates
11.py
env
OUTLINE
TIMELINE
11.py
write.html 11\...
paper.html 11\...
index.html 11\...
base.html 11\...
11 > 11.py
from flask import Flask, redirect, url_for, render_template, request, session
from flask_bootstrap import Bootstrap
from werkzeug.utils import secure_filename
import os
app = Flask(__name__)
app.secret_key = "#230dec61-fee8-4ef2-a791-36f9e680c9fc"
boostrap = Bootstrap(app=app)
img=''
upload_folder = os.path.join('static', 'photo')
app.config['UPLOAD'] = upload_folder
@app.route('/')
def index():
return render_template('index.html')
@app.route("/write", methods=["POST","GET"])
def write():
global img
if request.method == "POST":
paper = request.form["nm"]
file = request.files['img']
filename = secure_filename(file.filename)
file.save(os.path.join(app.config['UPLOAD'], filename))
img = os.path.join(app.config['UPLOAD'], filename)
session["paper"] = paper
session["photo"] = filename
return redirect(url_for("paper"))
else:
return render_template("write.html")
@app.route("/paper")
def paper():
if "paper" or "photo" in session:
paper = session["paper"]
photo = session["photo"]
img = os.path.join(app.config['UPLOAD'], photo)
return render_template("paper.html",paper=paper, photo=img)
else:
return redirect(url_for("write"))
if __name__ == '__main__':
app.run()
Ln 45, Col 14
Spaces: 4
UTF-8
CRLF
Python


File Edit Selection View Go
0529 [Administrator]
EXPLORER
0529
07
08
09
10
11
static\photo
unnamed.jpg
templates
bootstrap
index.html
paper.html
write.html
11.py
env
OUTLINE
TIMELINE
11.py
write.html 11\...
paper.html 11\...
index.html 11\...
base.html 11\...
11 > templates > write.html > ...
{% extends "bootstrap/base.html" %}
{% block content%}
<form action="#" enctype="multipart/form-data" method="post">

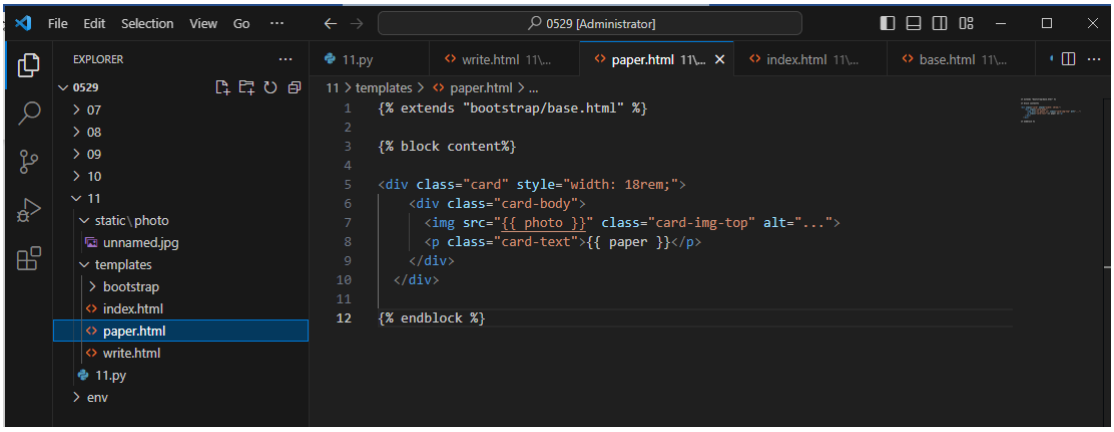
<label for="photoname">圖片::</label>

<input name="img" type="file"/>

<label for="username">內容:</label>
<input id="username" class="form-control" type="text" name="nm" />

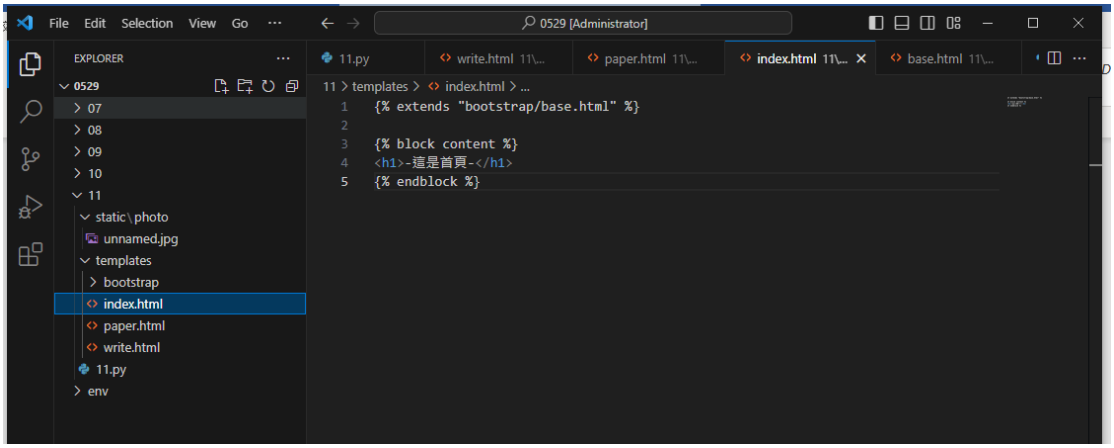
<button type="submit" class="btn btn-dark">Submit</button>

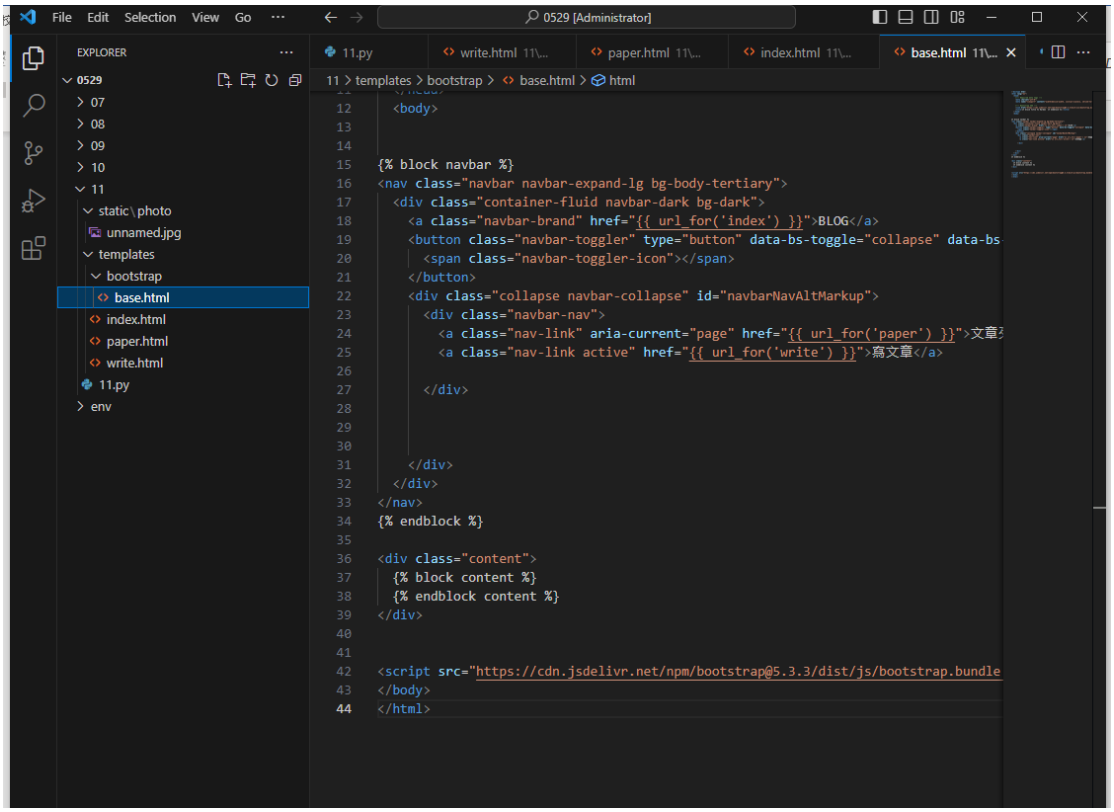
</form>
{% endblock %}
Ln 19, Col 15
Spaces: 4
UTF-8
CRLF
HTML


File Edit Selection View Go
0529 [Administrator]
EXPLORER
0529
07
08
09
10
11
static\photo
unnamed.jpg
templates
bootstrap
index.html
paper.html
write.html
11.py
env
11.py
write.html 11\...
paper.html 11\...
index.html 11\...
base.html 11\...
11 > templates > paper.html > ...
{% extends "bootstrap/base.html" %}
{% block content%}

<img src="{{ photo }}" class="card-img-top" alt="...">
{{ paper }}

{% endblock %}


File Edit Selection View Go
0529 [Administrator]
EXPLORER
0529
07
08
09
10
11
static\photo
unnamed.jpg
templates
bootstrap
index.html
paper.html
write.html
11.py
env
11.py
write.html 11\...
paper.html 11\...
index.html 11\...
base.html 11\...
11 > templates > index.html > ...
{% extends "bootstrap/base.html" %}
{% block content %}
<h1>-這是首頁-</h1>
{% endblock %}


File Edit Selection View Go
0529 [Administrator]
EXPLORER
0529
07
08
09
10
11
static\photo
unnamed.jpg
templates
bootstrap
base.html
index.html
paper.html
write.html
11.py
env
11.py
write.html 11\...
paper.html 11\...
index.html 11\...
base.html 11\...
11 > templates > bootstrap > base.html > html
<body>
{% block navbar %}
<nav class="navbar navbar-expand-lg bg-body-tertiary">

BLOG
<button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs

</button>

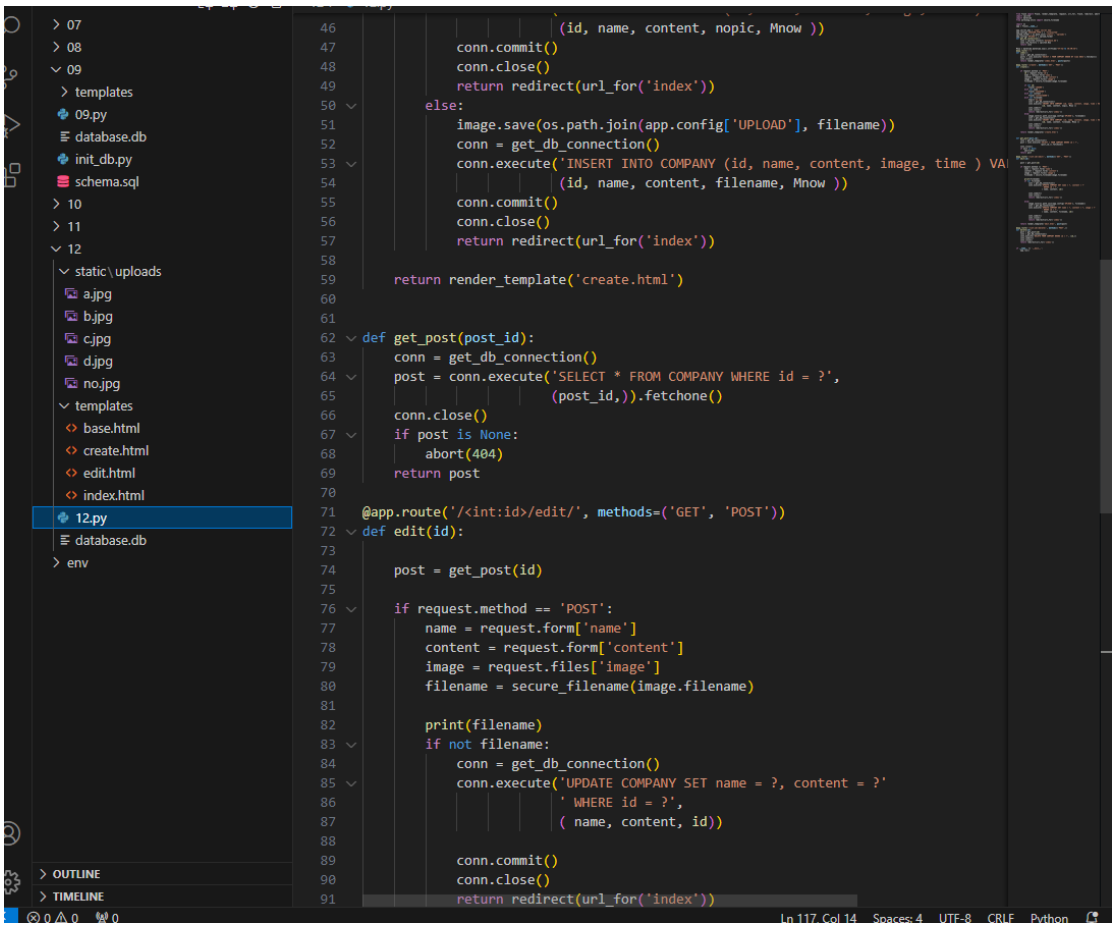
<a class="nav-link" aria-current="page" href="{{ url_for('paper') }}">文章
寫文章

</nav>
{% endblock %}

{% block content %}
{% endblock content %}

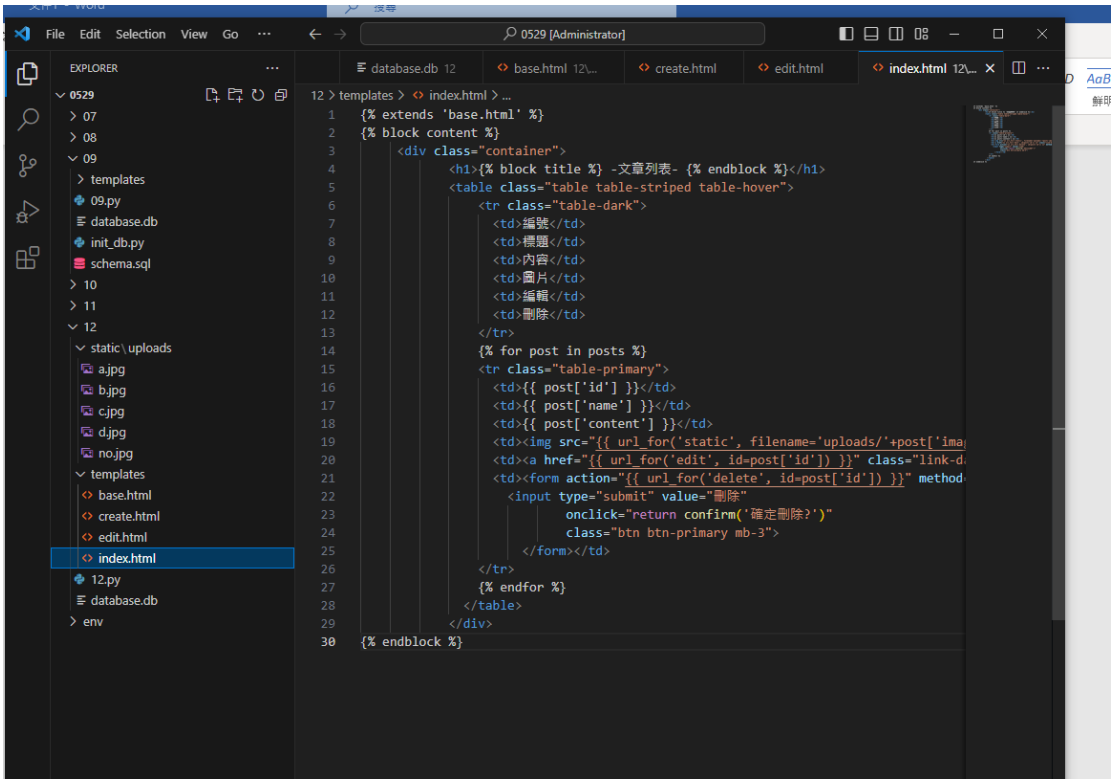
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.bundle
</body>
</html>

```python
from flask import Flask, render_template, request, url_for, flash, redirect, abort
import sqlite3
import datetime
from werkzeug.utils import secure_filename

import os
app = Flask(__name__)

app.secret_key = 'super secret key'
app.config['SESSION_TYPE'] = 'filesystem'
upload_folder = os.path.join('static', 'uploads')
app.config['UPLOAD'] = upload_folder
def get_db_connection():
    conn = sqlite3.connect('database.db')
    conn.row_factory = sqlite3.Row
    return conn

Mnow = datetime.datetime.now().strftime("%Y-%m-%d %H:%M:%S")
@app.route('/')
def index():
    conn = get_db_connection()
    posts = conn.execute('SELECT * FROM COMPANY ORDER BY time DESC').fetchall()
    conn.close()
    return render_template('index.html', posts=posts)

@app.route('/create', methods=('GET', 'POST'))
def create():

    if request.method == 'POST':
        id = request.form['id']
        name = request.form['name']
        content = request.form['content']
        image = request.files['image']
        filename = secure_filename(image.filename)

        if not id:
            flash('id無填')
        elif not name :
            flash('name無填')
        elif not content :
            flash('content無填')
        elif not filename:
            nopic='no.jpg'
            conn = get_db_connection()
            conn.execute('INSERT INTO COMPANY (id, name, content, image, time ) VA
                         (id, name, content, nopic, Mnow ))
```

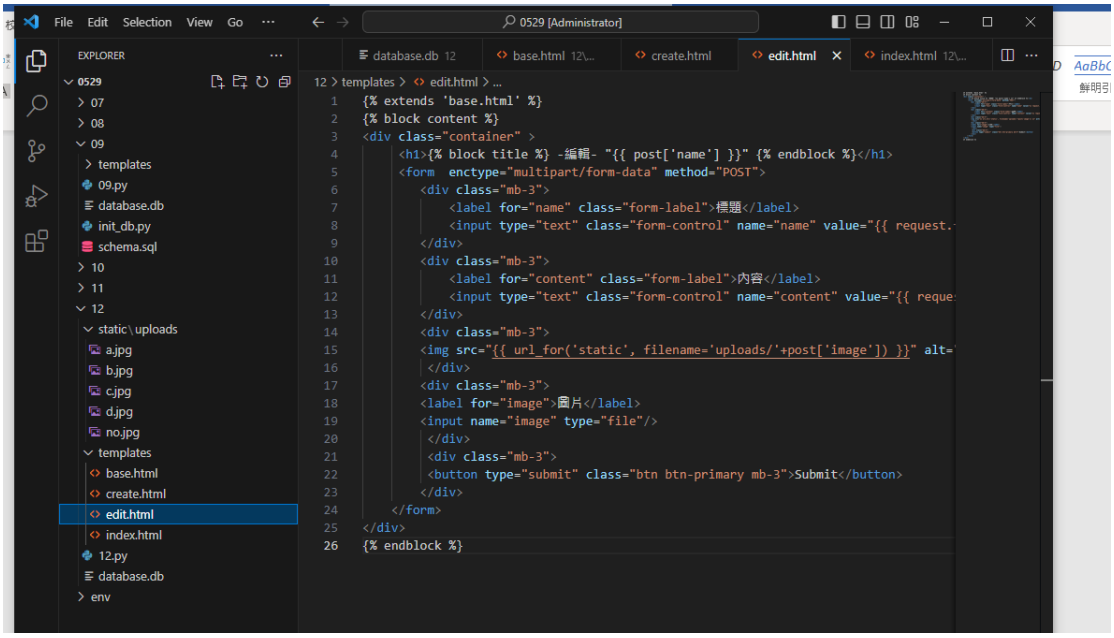

07
08
09
templates
09.py
database.db
init_db.py
schema.sql
10
11
12
static\uploads
a.jpg
b.jpg
c.jpg
d.jpg
no.jpg
templates
base.html
create.html
edit.html
index.html
12.py
database.db
env
OUTLINE
TIMELINE
(id, name, content, nopic, Mnow))
conn.commit()
conn.close()
return redirect(url_for('index'))
else:
image.save(os.path.join(app.config['UPLOAD'], filename))
conn = get_db_connection()
conn.execute('INSERT INTO COMPANY (id, name, content, image, time) VA
(id, name, content, filename, Mnow))
conn.commit()
conn.close()
return redirect(url_for('index'))
return render_template('create.html')
def get_post(post_id):
conn = get_db_connection()
post = conn.execute('SELECT * FROM COMPANY WHERE id = ?',
(post_id,)).fetchone()
conn.close()
if post is None:
abort(404)
return post
@app.route('/<int:id>/edit/', methods=('GET', 'POST'))
def edit(id):
post = get_post(id)
if request.method == 'POST':
name = request.form['name']
content = request.form['content']
image = request.files['image']
filename = secure_filename(image.filename)
print(filename)
if not filename:
conn = get_db_connection()
conn.execute('UPDATE COMPANY SET name = ?, content = ?'
' WHERE id = ?',
(name, content, id))
conn.commit()
conn.close()
return redirect(url_for('index'))
Ln 117, Col 14
Spaces: 4
UTF-8
CRLF
Python

```
12.py
def edit(id):

    post = get_post(id)

    if request.method == 'POST':
        name = request.form['name']
        content = request.form['content']
        image = request.files['image']
        filename = secure_filename(image.filename)

        print(filename)
        if not filename:
            conn = get_db_connection()
            conn.execute('UPDATE COMPANY SET name = ?, content = ?'
                         ' WHERE id = ?',
                         ( name, content, id))

            conn.commit()
            conn.close()
            return redirect(url_for('index'))

        else:
            image.save(os.path.join(app.config['UPLOAD'], filename))
            conn = get_db_connection()
            conn.execute('UPDATE COMPANY SET name = ?, content = ?, image = ?'
                         ' WHERE id = ?',
                         ( name, content, filename, id))

            conn.commit()
            conn.close()
            return redirect(url_for('index'))

    return render_template('edit.html', post=post)

@app.route('/<int:id>/delete/', methods=('POST',))
def delete(id):
    post = get_post(id)
    conn = get_db_connection()
    conn.execute('DELETE FROM COMPANY WHERE id = ?', (id,))
    conn.commit()
    conn.close()
    return redirect(url_for('index'))


if __name__ == '__main__':
    app.run()
```

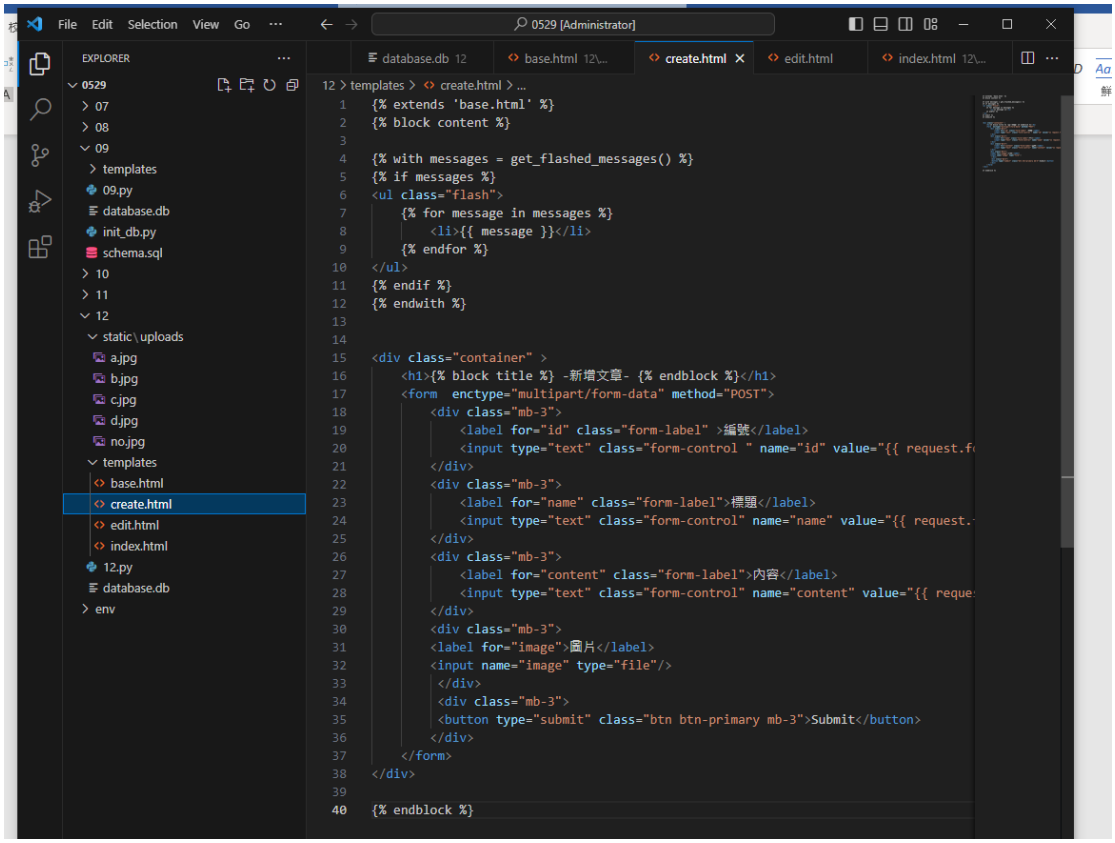
Ln 117, Col 14 Spaces: 4 UTF-8 CRLF Python



```
{% extends 'base.html' %}
{% block content %}
    <div class="container">
        <h1>{% block title %} -文章列表- {% endblock %}</h1>
        <table class="table table-striped table-hover">
            <tr class="table-dark">
              <td>編號</td>
              <td>標題</td>
              <td>內容</td>
              <td>圖片</td>
              <td>編輯</td>
              <td>刪除</td>
            </tr>
            {% for post in posts %}
            <tr class="table-primary">
              <td>{{ post['id'] }}</td>
              <td>{{ post['name'] }}</td>
              <td>{{ post['content'] }}</td>
              <td><img src="{{ url_for('static', filename='uploads/'+post['ima
              <td><a href="{{ url_for('edit', id=post['id']) }}" class="link-d
              <td><form action="{{ url_for('delete', id=post['id']) }}" method
                <input type="submit" value="刪除"
                       onclick="return confirm('確定刪除?')"
                       class="btn btn-primary mb-3">
              </form></td>
            </tr>
            {% endfor %}
          </table>
        </div>
{% endblock %}
```



```
{% extends 'base.html' %}
{% block content %}
<div class="container" >
    <h1>{% block title %} -編輯- "{{ post['name'] }}" {% endblock %}</h1>
    <form  enctype="multipart/form-data" method="POST">
        <div class="mb-3">
            <label for="name" class="form-label">標題</label>
            <input type="text" class="form-control" name="name" value="{{ request.
        </div>
        <div class="mb-3">
            <label for="content" class="form-label">內容</label>
            <input type="text" class="form-control" name="content" value="{{ reque
        </div>
        <div class="mb-3">
        <img src="{{ url_for('static', filename='uploads/'+post['image']) }}" alt=
        </div>
        <div class="mb-3">
        <label for="image">圖片</label>
        <input name="image" type="file"/>
        </div>
        <div class="mb-3">
        <button type="submit" class="btn btn-primary mb-3">Submit</button>
        </div>
    </form>
</div>
{% endblock %}
```


File Edit Selection View Go
0529 [Administrator]
EXPLORER
0529
07
08
09
templates
09.py
database.db
init_db.py
schema.sql
10
11
12
static\uploads
a.jpg
b.jpg
c.jpg
d.jpg
no.jpg
templates
base.html
create.html
edit.html
index.html
12.py
database.db
env
database.db 12
base.html 12\...
create.html
edit.html
index.html 12\...
12 > templates > create.html > ...
{% extends 'base.html' %}
{% block content %}
{% with messages = get_flashed_messages() %}
{% if messages %}
<ul class="flash">
{% for message in messages %}
{{ message }}
{% endfor %}

{% endif %}
{% endwith %}

<h1>{% block title %} -新增文章- {% endblock %}</h1>
<form enctype="multipart/form-data" method="POST">

<label for="id" class="form-label" >編號</label>
<input type="text" class="form-control " name="id" value="{{ request.f

<label for="name" class="form-label">標題</label>
<input type="text" class="form-control" name="name" value="{{ request.

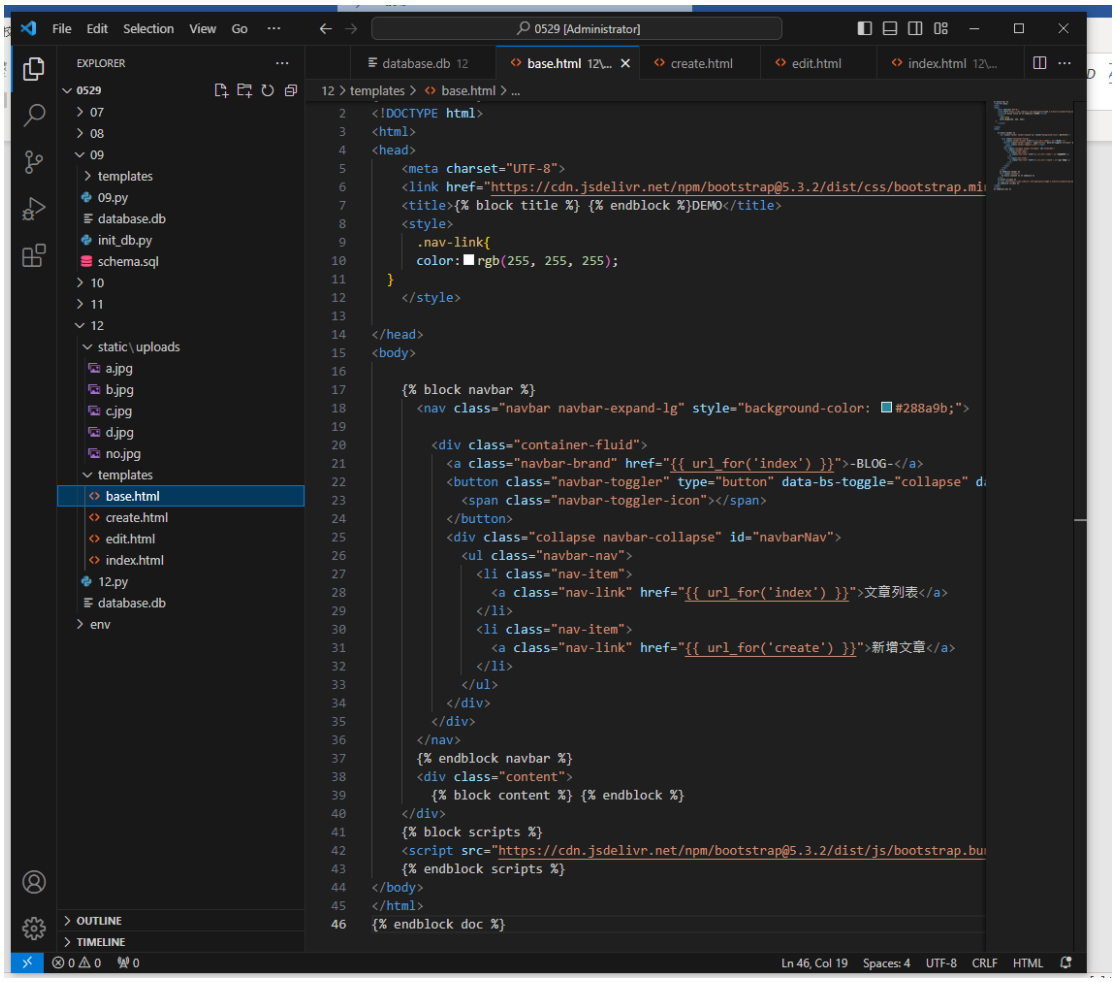
<label for="content" class="form-label">內容</label>
<input type="text" class="form-control" name="content" value="{{ reque

<label for="image">圖片</label>
<input name="image" type="file"/>

<button type="submit" class="btn btn-primary mb-3">Submit</button>

</form>

{% endblock %}


File Edit Selection View Go
0529 [Administrator]
EXPLORER
0529
07
08
09
templates
09.py
database.db
init_db.py
schema.sql
10
11
12
static\uploads
a.jpg
b.jpg
c.jpg
d.jpg
no.jpg
templates
base.html
create.html
edit.html
index.html
12.py
database.db
env
OUTLINE
TIMELINE
database.db 12
base.html 12\...
create.html
edit.html
index.html 12\...
12 > templates > base.html > ...
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.mi
<title>{% block title %} {% endblock %}DEMO</title>
<style>
.nav-link{
color: rgb(255, 255, 255);
}
</style>
</head>
<body>
{% block navbar %}
<nav class="navbar navbar-expand-lg" style="background-color: #288a9b;">

-BLOG-
<button class="navbar-toggler" type="button" data-bs-toggle="collapse" d

</button>

<ul class="navbar-nav">
<li class="nav-item">
文章列表

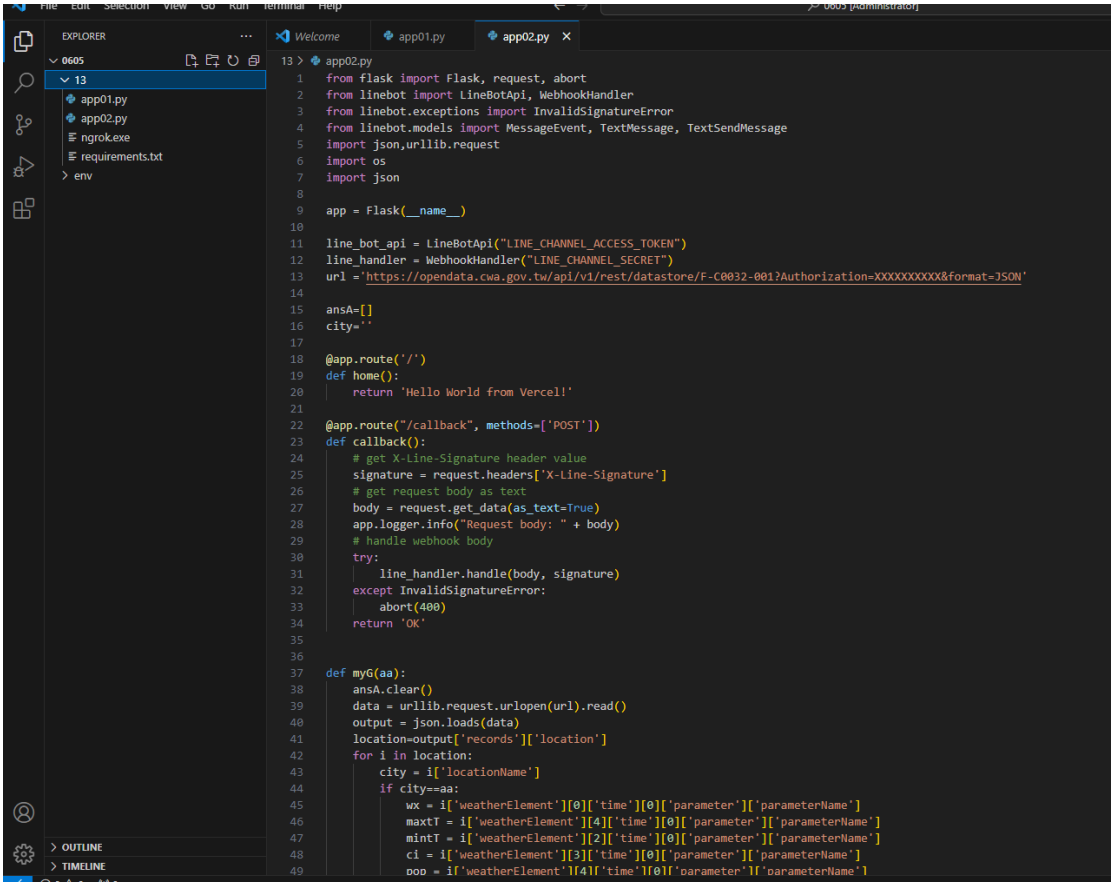
<li class="nav-item">
新增文章

</nav>
{% endblock navbar %}

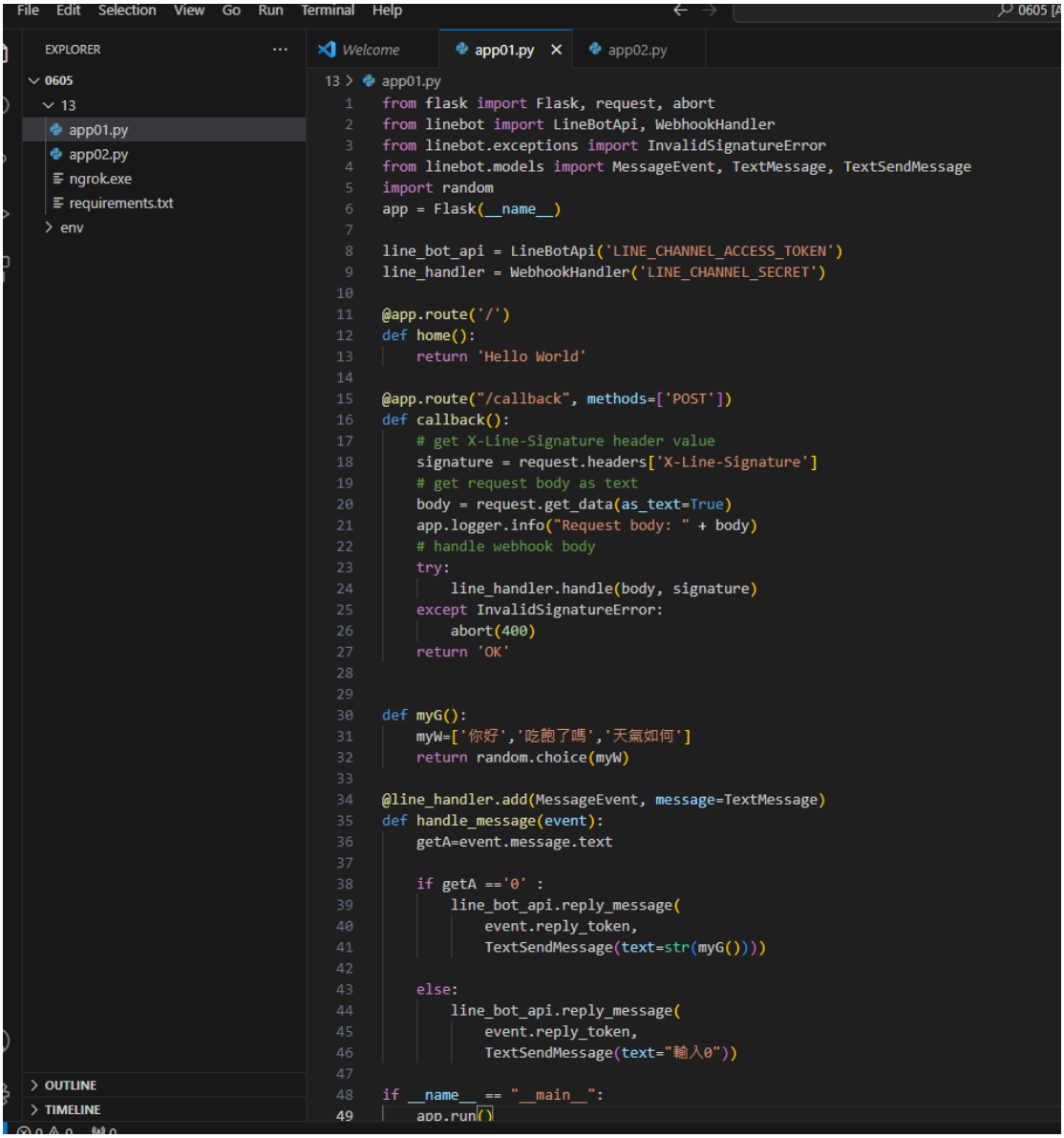
{% block content %} {% endblock %}

{% block scripts %}
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/js/bootstrap.bu
{% endblock scripts %}
</body>
</html>
{% endblock doc %}
Ln 46, Col 19
Spaces: 4
UTF-8
CRLF
HTML

0605


EXPLORER
0605
13
app01.py
app02.py
ngrok.exe
requirements.txt
env
OUTLINE
TIMELINE
Welcome
app01.py
app02.py
13 > app02.py
from flask import Flask, request, abort
from linebot import LineBotApi, WebhookHandler
from linebot.exceptions import InvalidSignatureError
from linebot.models import MessageEvent, TextMessage, TextSendMessage
import json,urllib.request
import os
import json
app = Flask(__name__)
line_bot_api = LineBotApi("LINE_CHANNEL_ACCESS_TOKEN")
line_handler = WebhookHandler("LINE_CHANNEL_SECRET")
url ='https://opendata.cwa.gov.tw/api/v1/rest/datastore/F-C0032-001?Authorization=XXXXXXXXX&format=JSON'
ansA=[]
city=''
@app.route('/')
def home():
return 'Hello World from Vercel!'
@app.route("/callback", methods=['POST'])
def callback():
# get X-Line-Signature header value
signature = request.headers['X-Line-Signature']
# get request body as text
body = request.get_data(as_text=True)
app.logger.info("Request body: " + body)
# handle webhook body
try:
line_handler.handle(body, signature)
except InvalidSignatureError:
abort(400)
return 'OK'
def myG(aa):
ansA.clear()
data = urllib.request.urlopen(url).read()
output = json.loads(data)
location=output['records']['location']
for i in location:
city = i['locationName']
if city==aa:
wx = i['weatherElement'][0]['time'][0]['parameter']['parameterName']
maxtT = i['weatherElement'][4]['time'][0]['parameter']['parameterName']
mintT = i['weatherElement'][2]['time'][0]['parameter']['parameterName']
ci = i['weatherElement'][3]['time'][0]['parameter']['parameterName']

```
51                  ansA.append(city)
52                  ansA.append(wx)
53                  ansA.append(mintT)
54                  ansA.append(maxtT)
55      return ansA
56
57
58
59  @line_handler.add(MessageEvent, message=TextMessage)
60  def handle_message(event):
61
62      if event.message.text == "1":
63          line_bot_api.reply_message(
64              event.reply_token,
65              TextSendMessage(text=str(myG('臺北市'))))
66
67      elif event.message.text == "2":
68          line_bot_api.reply_message(
69              event.reply_token,
70              TextSendMessage(text=str(myG('桃園市'))))
71
72      elif event.message.text=='3':
73          line_bot_api.reply_message(
74              event.reply_token,
75              TextSendMessage(text=str(myG('新竹市'))))
76
77      elif event.message.text=='4':
78          line_bot_api.reply_message(
79              event.reply_token,
80              TextSendMessage(text=str(myG('臺中市'))))
81
82      elif event.message.text=='5':
83          line_bot_api.reply_message(
84              event.reply_token,
85              TextSendMessage(text=str(myG('臺南市'))))
86
87      elif event.message.text=='6':
88          line_bot_api.reply_message(
89              event.reply_token,
90              TextSendMessage(text=str(myG('高雄市'))))
91
92      else:
93          line_bot_api.reply_message(
94              event.reply_token,
95              TextSendMessage(text="只有1-6的數字"))
96
97  if __name__ == "__main__":
98      app.run()
```


File Edit Selection View Go Run Terminal Help
EXPLORER
0605
13
app01.py
app02.py
ngrok.exe
requirements.txt
env
OUTLINE
TIMELINE
Welcome
app01.py
app02.py
13 > app01.py
from flask import Flask, request, abort
from linebot import LineBotApi, WebhookHandler
from linebot.exceptions import InvalidSignatureError
from linebot.models import MessageEvent, TextMessage, TextSendMessage
import random
app = Flask(__name__)
line_bot_api = LineBotApi('LINE_CHANNEL_ACCESS_TOKEN')
line_handler = WebhookHandler('LINE_CHANNEL_SECRET')
@app.route('/')
def home():
return 'Hello World'
@app.route("/callback", methods=['POST'])
def callback():
# get X-Line-Signature header value
signature = request.headers['X-Line-Signature']
# get request body as text
body = request.get_data(as_text=True)
app.logger.info("Request body: " + body)
# handle webhook body
try:
line_handler.handle(body, signature)
except InvalidSignatureError:
abort(400)
return 'OK'
def myG():
myW=['你好','吃飽了嗎','天氣如何']
return random.choice(myW)
@line_handler.add(MessageEvent, message=TextMessage)
def handle_message(event):
getA=event.message.text
if getA =='0' :
line_bot_api.reply_message(
event.reply_token,
TextSendMessage(text=str(myG())))
else:
line_bot_api.reply_message(
event.reply_token,
TextSendMessage(text="輸入0"))
if __name__ == "__main__":
app.run()