
Python - Django 사용법 (기초)

2024-08-13(화)

정도윤(rabbitsun2@gmail.com)

목차

- 1. 장고 - 설치(pip)
- 2. 장고 - 프로젝트 생성
- 3. 장고 - 개발하기
- 4. 장고 - 실행하기
- 5. 웹 브라우저 -> GET 방식
- 6. 웹 브라우저 -> POST 방식
- 7. 웹 브라우저 -> GET(입력), GET(출력) 방식(예제)
- 8. 웹 브라우저 -> GET(입력), POST(출력) 방식(예제)

1. 장고 - 설치(pip)

```
pip install django
```

2. 장고 - 프로젝트 생성

원하는 위치에 작업공간 폴더를 생성한 후 django프로젝트를 생성하기 위해 다음의 명령어를 입력한다.

```
python -m django startproject [프로젝트명]
```

Visual Studio Code에서 폴더 열기를 누른 후 프로젝트 폴더를 열고 명령 프롬프트를 실행한 후 다음의 명령어로 데이터베이스이 기본 테이블을 생성한다.

```
python manage.py migrate
```

```
Anaconda Prompt - conda in: x + v
(llama3) C:\Users\pc\Documents>python -m django startproject django_web

(llama3) C:\Users\pc\Documents>dir
C 드라이브의 볼륨: os
볼륨 일련 번호: 30E1-DA1B

C:\Users\pc\Documents 디렉터리

2024-08-13 오전 09:09 <DIR> .
2024-08-13 오전 08:43 <DIR> ..
2024-08-13 오전 08:01 <DIR> 210,939
2024-08-12 오후 06:43 <DIR>
2024-08-10 오후 05:56 <DIR> Adobe
2024-08-13 오전 09:09 <DIR> django_web
2024-08-10 오후 05:56 <DIR> ezPDFReader3.0
2024-05-24 오후 02:15 <DIR>
2024-05-14 오후 09:07 <DIR>
2024-08-10 오후 05:56 <DIR>
2024-08-13 오전 08:51 <DIR>
2024-08-10 오후 12:59 <DIR>
2024-08-10 오후 05:56 <DIR>
2024-08-12 오후 09:46 <DIR> 160
2024-08-10 오후 05:56 <DIR> Visual Studio 2022
2021-09-15 오전 06:25 26,257 Welcome to Cell.cell
2021-09-15 오전 06:25 23,552 Welcome to Hwp.hwp
2021-09-15 오전 06:25 219,962 Welcome to Show.show
2021-09-15 오전 06:25 15,185 Welcome to Word.hwdt
2024-05-05 오후 08:11 <DIR> 사용자 지정 Office 서식 파일
2024-08-11 오후 08:50 <DIR> 카카오톡 받은 파일
6개 파일 496,055 바이트
15개 디렉터리 321,418,108,928 바이트 남음

(llama3) C:\Users\pc\Documents>cd django_web
```

```
Anaconda Prompt - conda in: x + v
(llama3) C:\Users\pc\Documents>cd django_web

(llama3) C:\Users\pc\Documents\django_web>dir
C 드라이브의 볼륨: os
볼륨 일련 번호: 30E1-DA1B

C:\Users\pc\Documents\django_web 디렉터리

2024-08-13 오전 09:09 <DIR> .
2024-08-13 오전 09:09 <DIR> ..
2024-08-13 오전 09:09 <DIR> django_web
2024-08-13 오전 09:09 688 manage.py
1개 파일 688 바이트
3개 디렉터리 321,409,335,296 바이트 남음

(llama3) C:\Users\pc\Documents\django_web>python manage.py migrate
Operations to perform:
  Apply all migrations: admin, auth, contenttypes, sessions
Running migrations:
  Applying contenttypes.0001_initial... OK
  Applying auth.0001_initial... OK
  Applying admin.0001_initial... OK
  Applying admin.0002_logentry_remove_auto_add... OK
  Applying admin.0003_logentry_add_action_flag_choices... OK
  Applying contenttypes.0002_remove_content_type_name... OK
  Applying auth.0002_alter_permission_name_max_length... OK
  Applying auth.0003_alter_user_email_max_length... OK
  Applying auth.0004_alter_user_username_opts... OK
  Applying auth.0005_alter_user_last_login_null... OK
  Applying auth.0006_require_contenttypes_0002... OK
  Applying auth.0007_alter_validators_add_error_messages... OK
  Applying auth.0008_alter_user_username_max_length... OK
  Applying auth.0009_alter_user_last_name_max_length... OK
  Applying auth.0010_alter_group_name_max_length... OK
```

3. 장고 - 개발하기

1. 앱 설치하기

프로젝트 폴더에서 명령 프롬프트를 실행하여 다음의 명령어를 입력하여 프로젝트의 django app을 설치한다.

```
python manage.py startapp [앱이름]
```

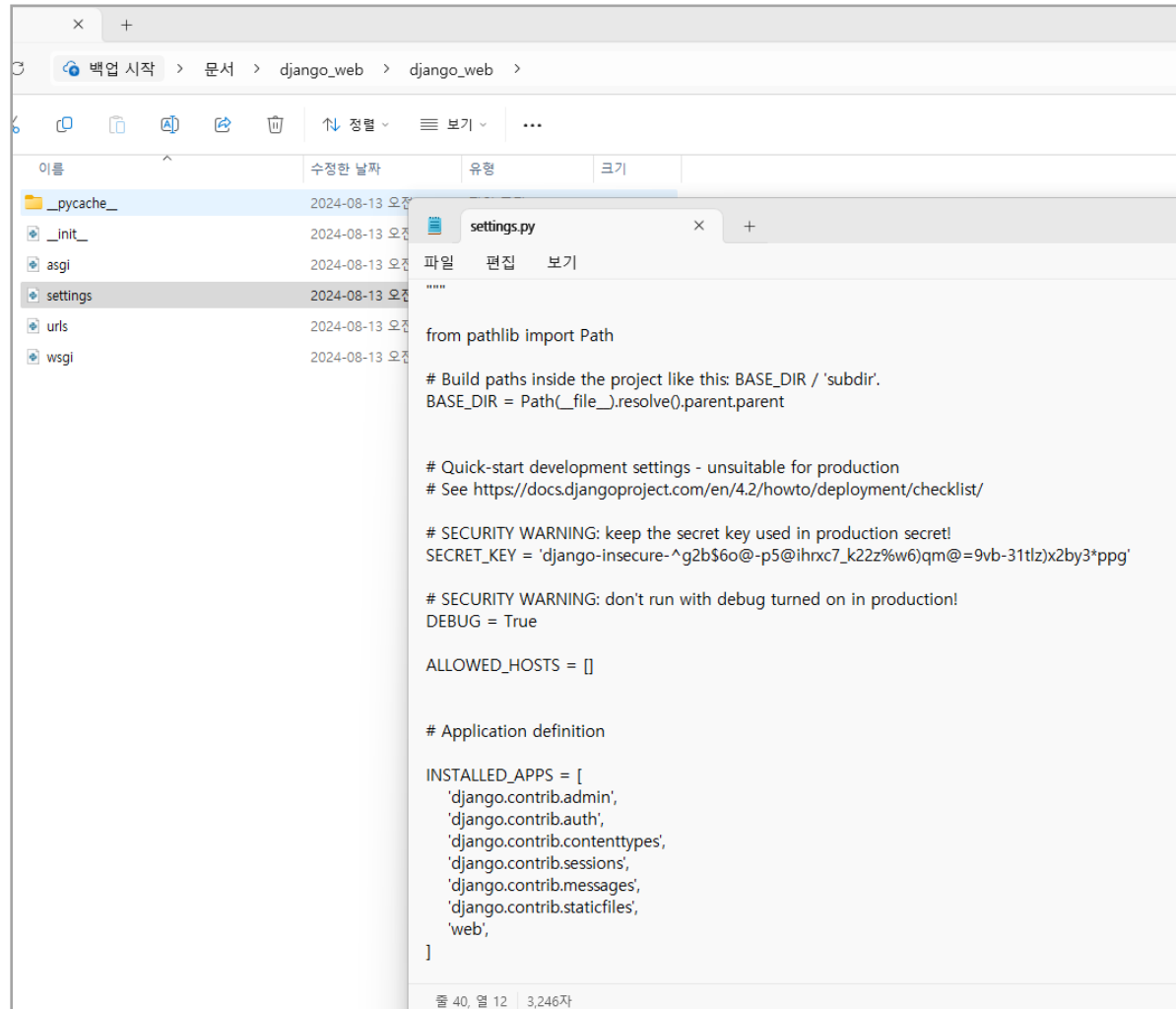
앱 설치가 완료되면 [프로젝트명]/settings.py파일을 열어 다음과 같이 설치한 앱을 설정한다.

```
INSTALLED_APPS = [  
    'django.contrib.admin',  
    'django.contrib.auth',  
    'django.contrib.contenttypes',  
    'django.contrib.sessions',  
    'django.contrib.messages',  
    'django.contrib.staticfiles',  
    '[앱이름]' # 설치한 앱의 이름을 추가  
]
```

```
(llama3) C:\Users\pc\Documents\django_web>python manage.py startapp django_web  
CommandError: 'django_web' conflicts with the name of an existing Python module and cannot be used as an app name. Please try another name.  
  
(llama3) C:\Users\pc\Documents\django_web>python manage.py startapp web  
  
(llama3) C:\Users\pc\Documents\django_web>dir  
C 드라이브의 볼륨: os  
볼륨 일련 번호: 30E1-DA1B  
  
C:\Users\pc\Documents\django_web 디렉터리  
  
2024-08-13 오전 09:15 <DIR> .  
2024-08-13 오전 09:09 <DIR> ..  
2024-08-13 오전 09:10 131,072 db.sqlite3  
2024-08-13 오전 09:10 <DIR> django_web  
2024-08-13 오전 09:09 688 manage.py  
2024-08-13 오전 09:15 <DIR> web  
2개 파일 131,760 바이트  
4개 디렉터리 321,736,093,696 바이트 남음  
  
(llama3) C:\Users\pc\Documents\django_web>
```

3. 장고 - 개발하기

1. 앱 설치하기



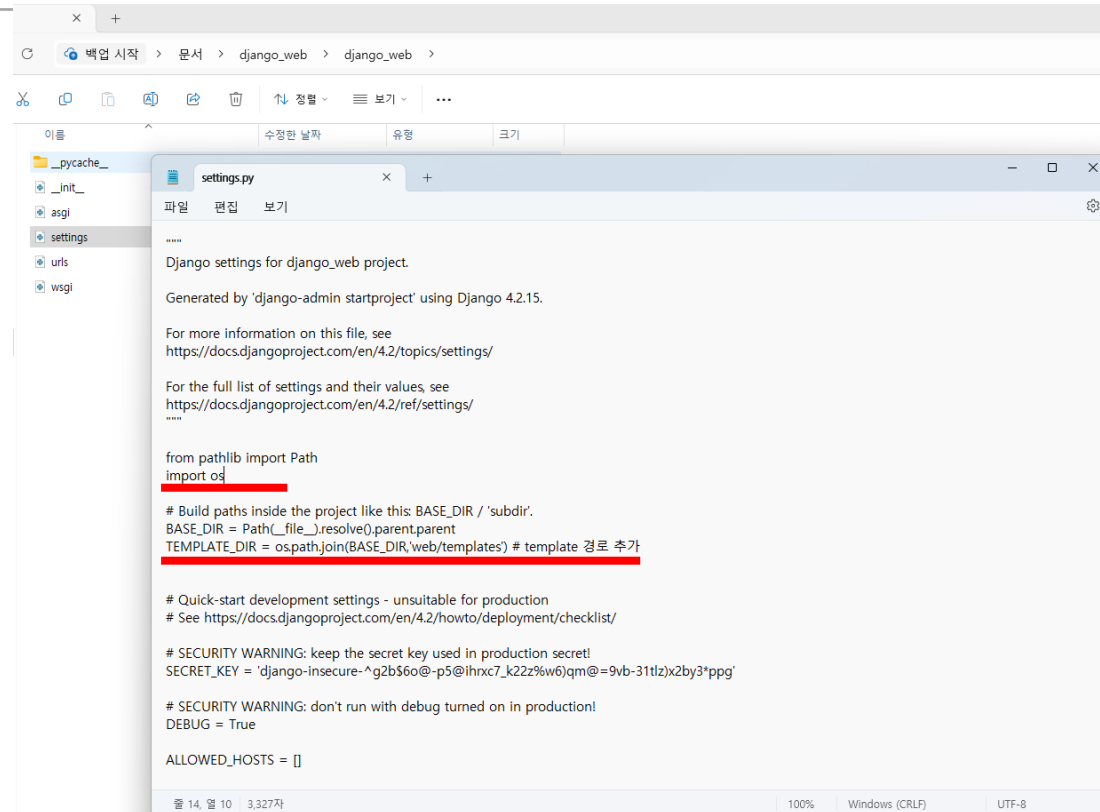
3. 장고 - 개발하기

2. template 경로 설정

Spring프레임워크는 MVC패턴을 따르는 것 처럼 django프레임워크는 MTV패턴을 따른다. 여기서 M이 의미하는 것은 template이고 MVC패턴에서의 View처럼 사용자에게 보여지는 페이지를 의미한다. [프로젝트명]/settings.py파일을 열어 다음과 같이 코드를 작성하여 templates폴더의 경로를 설정한다.

```
import os
```

```
BASE_DIR = Path(__file__).resolve().parent.parent # 해당 코드 밑에  
TEMPLATE_DIR = os.path.join(BASE_DIR, '[앱이름]/templates') # template 경로 추가
```

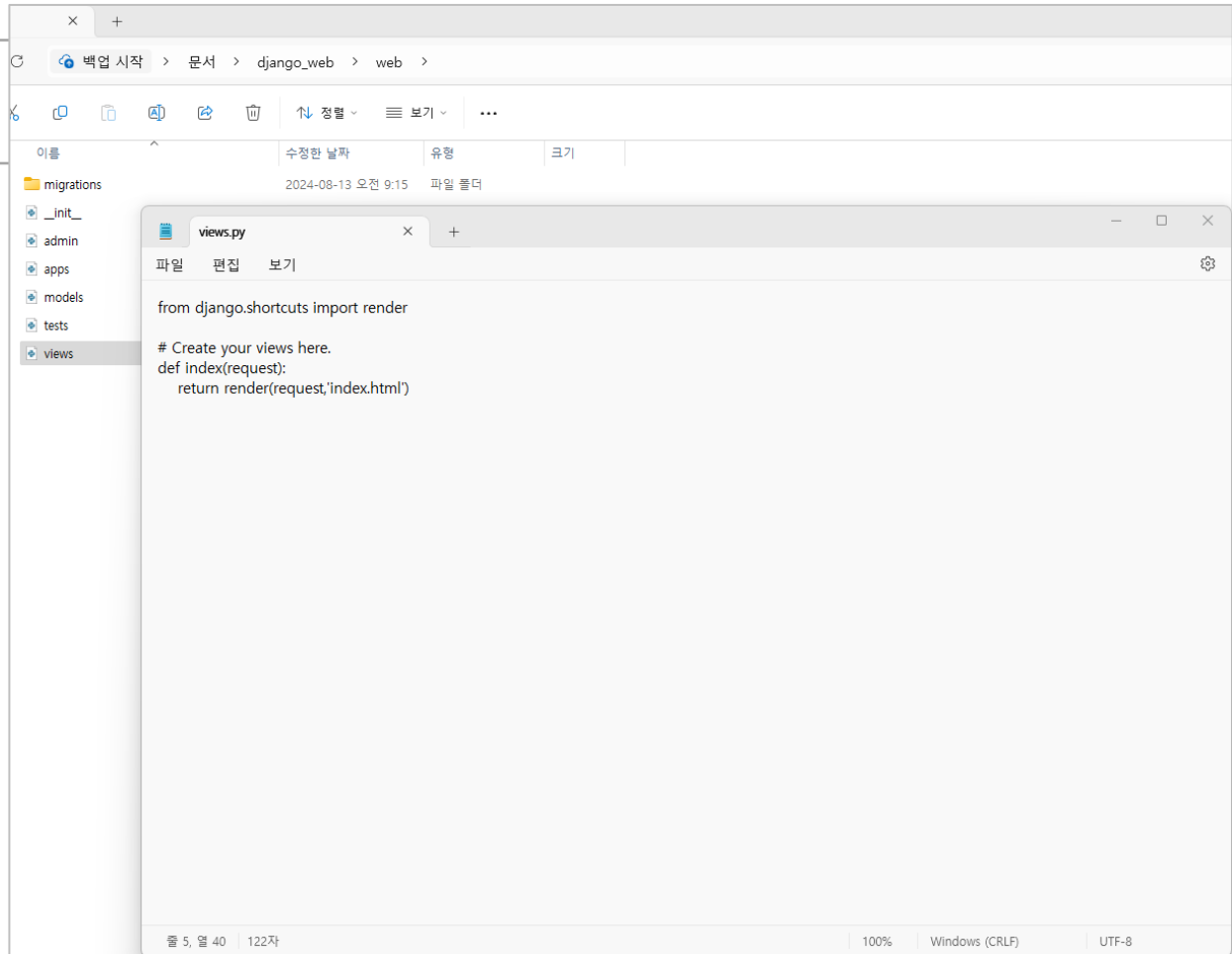


3. 장고 - 개발하기

3. hello Django! 출력하기

설치된 앱 폴더 안에 view.py파일에서 MVC패턴에서 Controller와 같은 역할을 담당하는 View를 설정할 수 있다. 해당 파일에서 다음의 코드를 추가한다.

```
def index(request):  
    return render(request,'index.html')
```



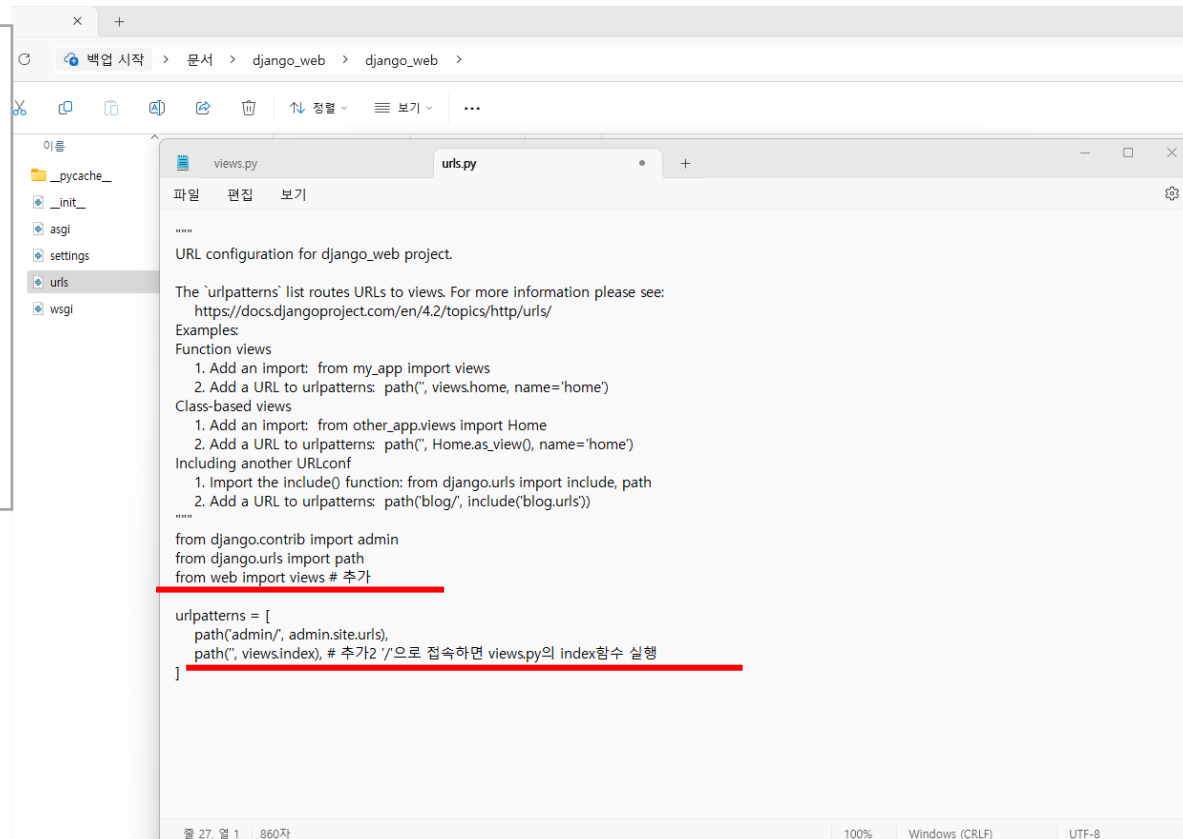
3. 장고 - 개발하기

3. hello Django! 출력하기

[프로젝트명]/urls.py파일에서 다음과 같이 작성하여 해당 경로로 접속하면 view.py파일에 정의한 함수가 동작하도록 한다.

```
from django.contrib import admin
from django.urls import path
from [앱이름] import views # 추가

urlpatterns = [ path('admin/', admin.site.urls),
                path('', views.index) # 추가 '/'으로 접속하면 views.py의 index함수 실행
]
```



3. 장고 - 개발하기

3. hello Django! 출력하기

설치된 앱 폴더 안에 templates폴더를 생성한 후 해당 폴더 안에 index.html파일을 생성하여 다음과 같이 작성한다.

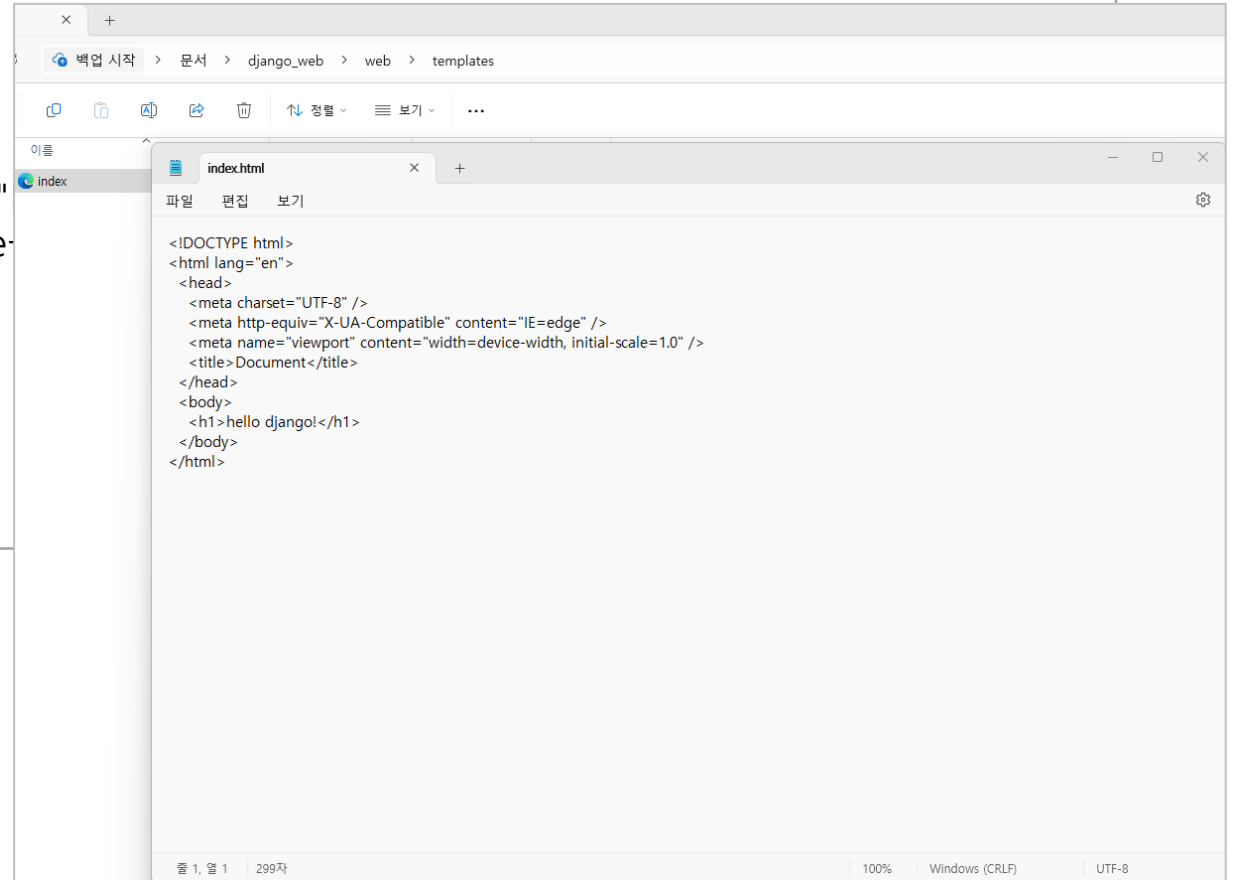
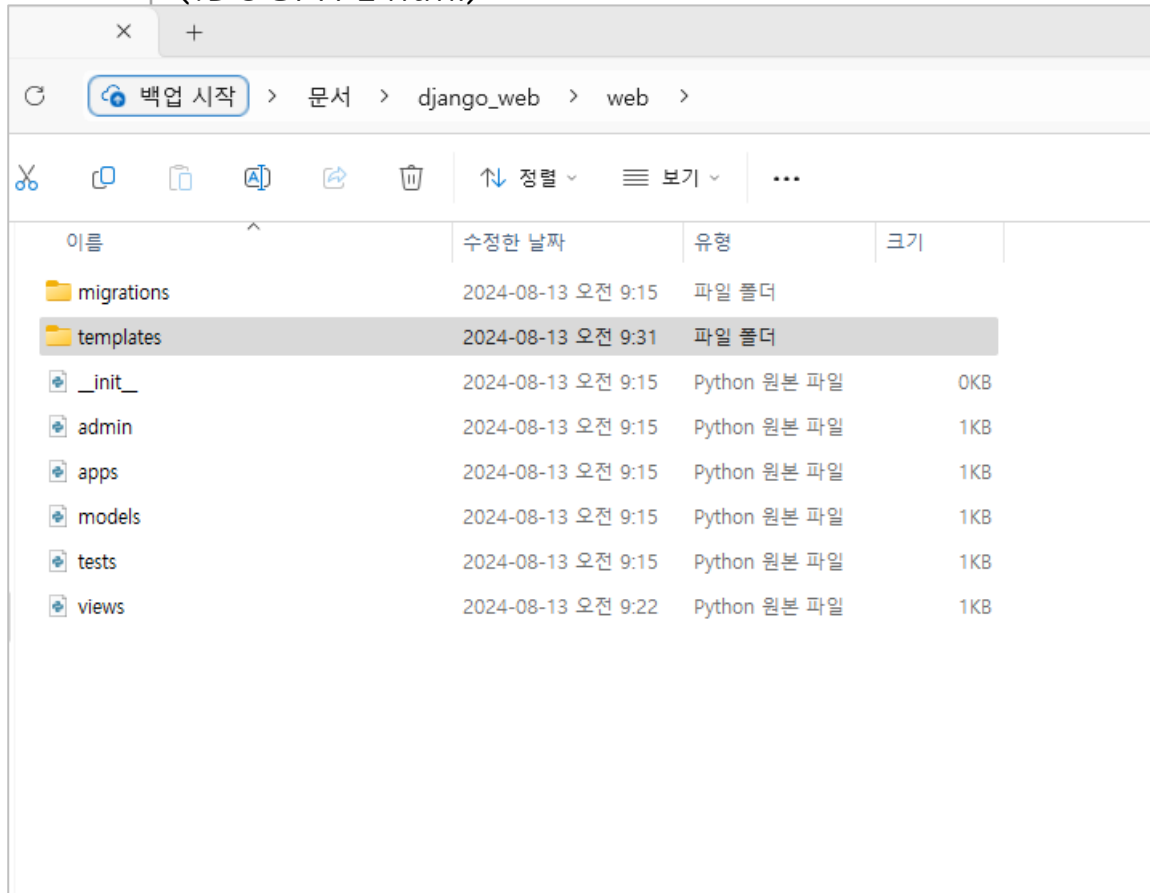
```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Document</title>
</head>
<body>
    <h1>hello django!</h1>
</body>
</html>
```

3. 장고 - 개발하기

3. hello Django! 출력하기

설치된 앱 폴더 안에 templates폴더를 생성한 후 해당 폴더 안에 index.html파일을 생성하여 다음과 같이 작성한다.

```
<!DOCTYPE html>
```



4. 장고 - 실행하기

실행하기

프로젝트 폴더에서 명령 프롬프트를 실행하고 다음의 명령어를 입력하여 프로젝트를 실행한다.

```
python manage.py runserver
```

```
Anaconda Prompt - conda in: x + v
2024-08-13 오전 09:09 <DIR> ..
2024-08-13 오전 09:10 131,072 db.sqlite3
2024-08-13 오전 09:10 <DIR> django_web
2024-08-13 오전 09:09 688 manage.py
2024-08-13 오전 09:31 <DIR> web
2개 파일 131,760 바이트
4개 디렉터리 321,684,262,912 바이트 남음

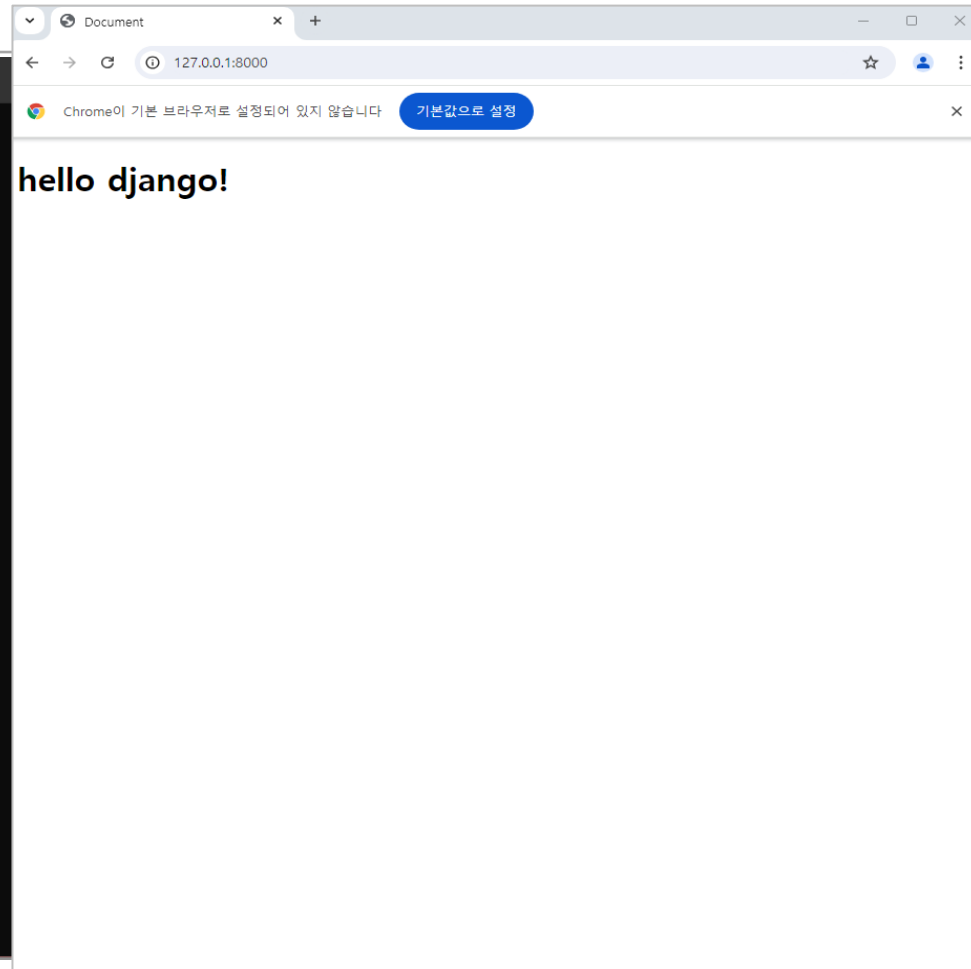
(llama3) C:\Users\pc\Documents\django_web>dir
C 드라이브의 볼륨: os
볼륨 일련 번호: 30E1-DA1B

C:\Users\pc\Documents\django_web 디렉터리

2024-08-13 오전 09:15 <DIR> .
2024-08-13 오전 09:09 <DIR> ..
2024-08-13 오전 09:10 131,072 db.sqlite3
2024-08-13 오전 09:10 <DIR> django_web
2024-08-13 오전 09:09 688 manage.py
2024-08-13 오전 09:31 <DIR> web
2개 파일 131,760 바이트
4개 디렉터리 321,684,262,912 바이트 남음

(llama3) C:\Users\pc\Documents\django_web>python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).
August 13, 2024 - 09:39:24
Django version 4.2.15, using settings 'django_web.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```



5. 웹 브라우저 -> GET 방식

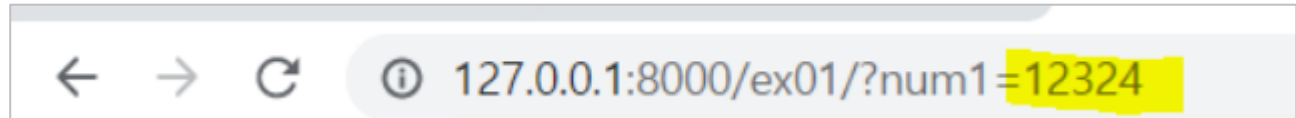
웹 브라우저 -> 서버 (GET)

- URL에 포함시켜서 전송하는 방식
- 서버가 DATA를 받는 것
- 요청하는 URL 뒤에 ? 다음 보내려는 데이터를 키=값 쌍으로 전송

```
def func1(request):  
    numvar1 = request.GET.get('num1', None)  
    print(numvar1)  
    return render(request, 'page1.html')
```

- GET 방식이기에 GET num1 : 키, url에서 ? 뒤에 해당 키에 해당하는 값을 따와서 출력

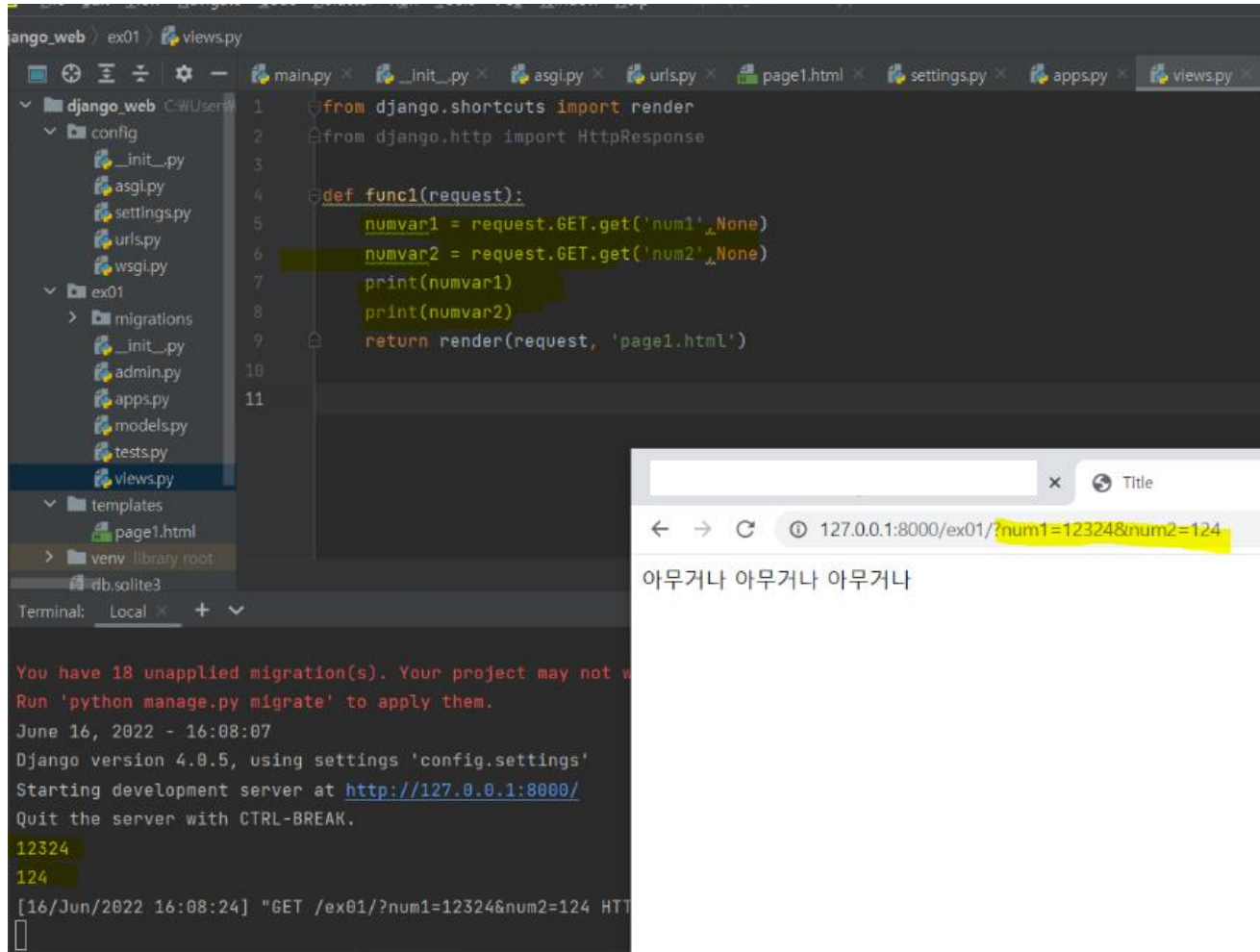
```
[16/Jun/2022  
12324
```



5. 웹 브라우저 -> GET 방식

웹 브라우저 -> 서버 (GET)

- 여러 값을 전송할 때는 &를 사용하여 키=값&키=값으로 전송



The screenshot displays a Django development environment. On the left, a file explorer shows the project structure, including `django_web`, `config`, `ex01`, `migrations`, `templates`, and `venv`. The `views.py` file is open in the editor, showing the `func1` function. The function uses `request.GET.get` to retrieve query parameters `num1` and `num2`, prints their values, and renders `page1.html`. Below the code editor, a terminal window shows the Django development server output, including the message "Starting development server at http://127.0.0.1:8000/" and the received GET request: `[16/Jun/2022 16:08:24] "GET /ex01/?num1=12324&num2=124 HTTP/1.1"`. To the right, a web browser window shows the URL `127.0.0.1:8000/ex01/?num1=12324&num2=124` and the response text "아무거나 아무거나 아무거나".

```
from django.shortcuts import render
from django.http import HttpResponse

def func1(request):
    numvar1 = request.GET.get('num1', None)
    numvar2 = request.GET.get('num2', None)
    print(numvar1)
    print(numvar2)
    return render(request, 'page1.html')
```

12324
124

[16/Jun/2022 16:08:24] "GET /ex01/?num1=12324&num2=124 HTTP/1.1"

아무거나 아무거나 아무거나

6. 웹 브라우저 -> POST 방식

웹 브라우저 -> 서버 (POST)

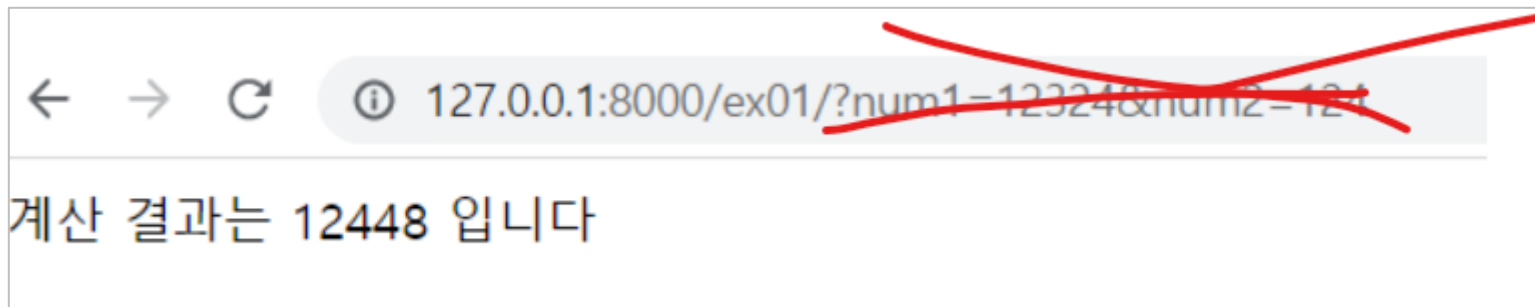
- form 태그로 전송하는 post 방식, 전송할 수 있는 html 페이지 필요
- 이 html 페이지에서 form 태그로 전송 render : 페이지를 렌더링해 줌
- context에 담아 보내야 하며, 이 context는 딕셔너리 형태여야 함. (이름이 context일 필요는 없음)

```
context = {'result': int(numvar1) + int(numvar2)}  
return render(request, 'page1.html', context)
```

- html에서 context 값을 사용하려면 {{ 중괄호 2개 안에 키값을 입력해야 한다. 이는 template 문법이다.

```
<body>  
계산 결과는 {{ result }} 입니다  
</body>
```

- 결과



7. 웹 브라우저 -> GET(입력), GET(출력) 방식(예제)

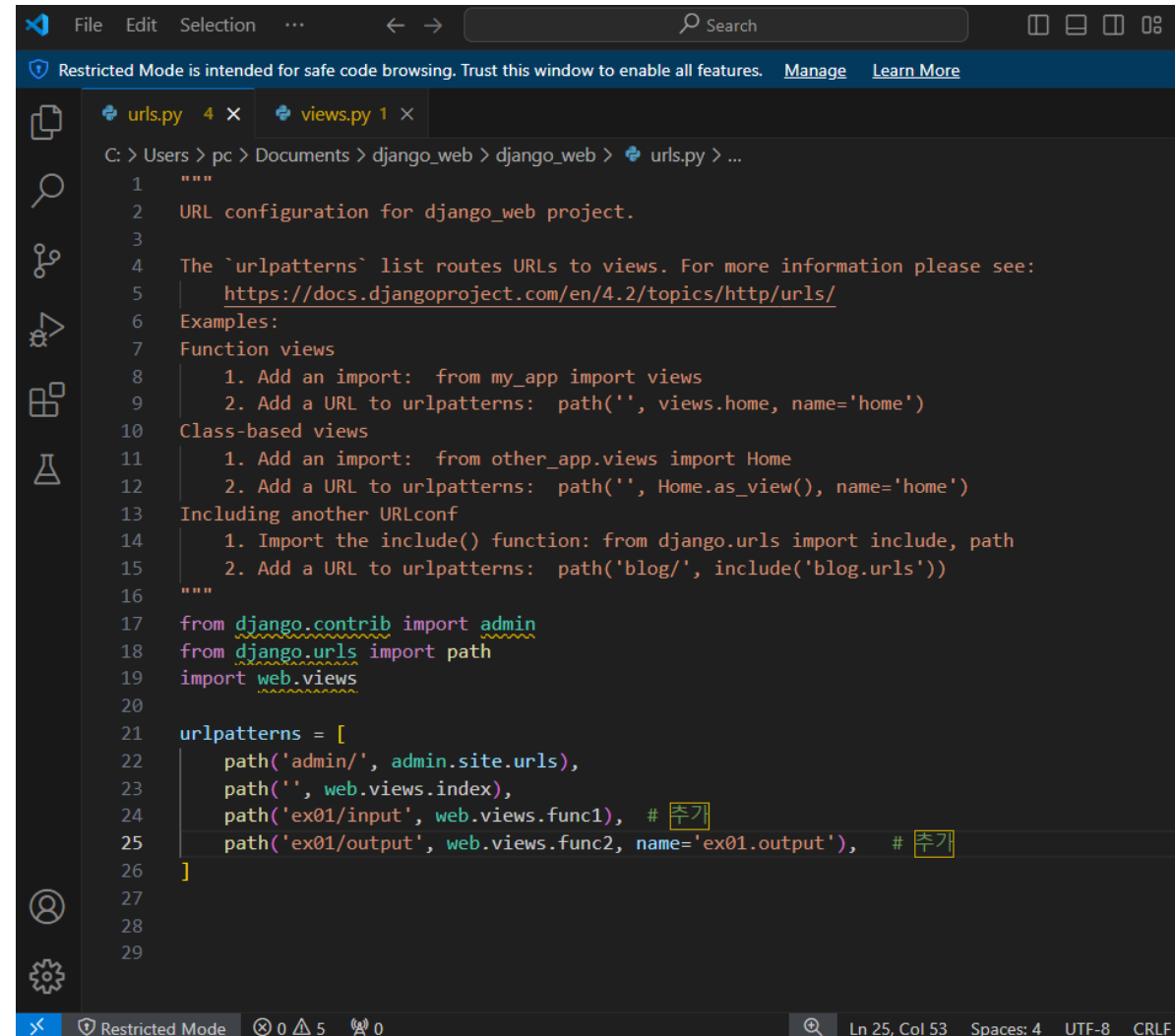
1. 프로젝트 환경(urls.py urlpatterns 변경하기)

- from [앱이름] import views을 지우고,
import [앱이름]으로 변경한다.

import web.views

- from [앱이름] import views을 지우고,
import [앱이름]으로 변경한다.

```
urlpatterns = [  
    path('admin/', admin.site.urls),  
    path(' ', web.views.index),  
    path('ex01/input', web01.views.func1),  
    path('ex01/output', web01.views.func2, name='ex01.output'),  
]
```



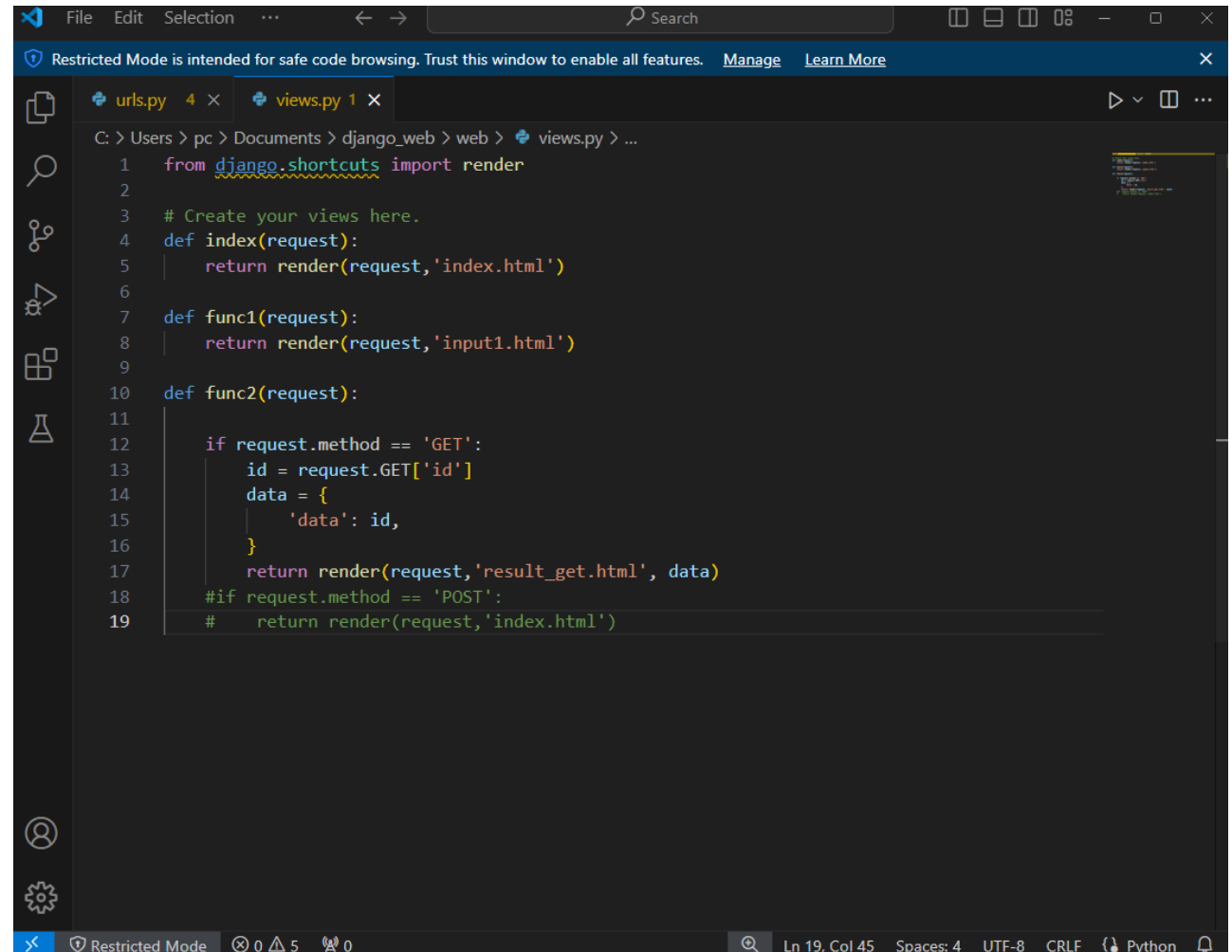
```
File Edit Selection ... Search
Restricted Mode is intended for safe code browsing. Trust this window to enable all features. Manage Learn More

urls.py 4 x views.py 1 x
C:\Users\pc\Documents\django_web\django_web> urls.py ...
1 """
2 URL configuration for django_web project.
3
4 The `urlpatterns` list routes URLs to views. For more information please see:
5     https://docs.djangoproject.com/en/4.2/topics/http/urls/
6 Examples:
7 Function views
8     1. Add an import: from my_app import views
9     2. Add a URL to urlpatterns: path('', views.home, name='home')
10 Class-based views
11     1. Add an import: from other_app.views import Home
12     2. Add a URL to urlpatterns: path('', Home.as_view(), name='home')
13 Including another URLconf
14     1. Import the include() function: from django.urls import include, path
15     2. Add a URL to urlpatterns: path('blog/', include('blog.urls'))
16 """
17 from django.contrib import admin
18 from django.urls import path
19 import web.views
20
21 urlpatterns = [
22     path('admin/', admin.site.urls),
23     path('', web.views.index),
24     path('ex01/input', web.views.func1), # 추가
25     path('ex01/output', web.views.func2, name='ex01.output'), # 추가
26 ]
27
28
29
Ln 25, Col 53 Spaces: 4 UTF-8 CRLF
```


7. 웹 브라우저 -> GET(입력), GET(출력) 방식(예제)

2. 앱 폴더에서 views.py의 내용 수정하기

```
def func1(request):  
    return render(request, 'input1.html')  
  
def func2(request):  
  
    if request.method == 'GET':  
        id = request.GET['id']  
        data = {  
            'data': id,  
        }  
        return render(request, 'result_get.html', data)
```



The screenshot shows a code editor window with the following content:

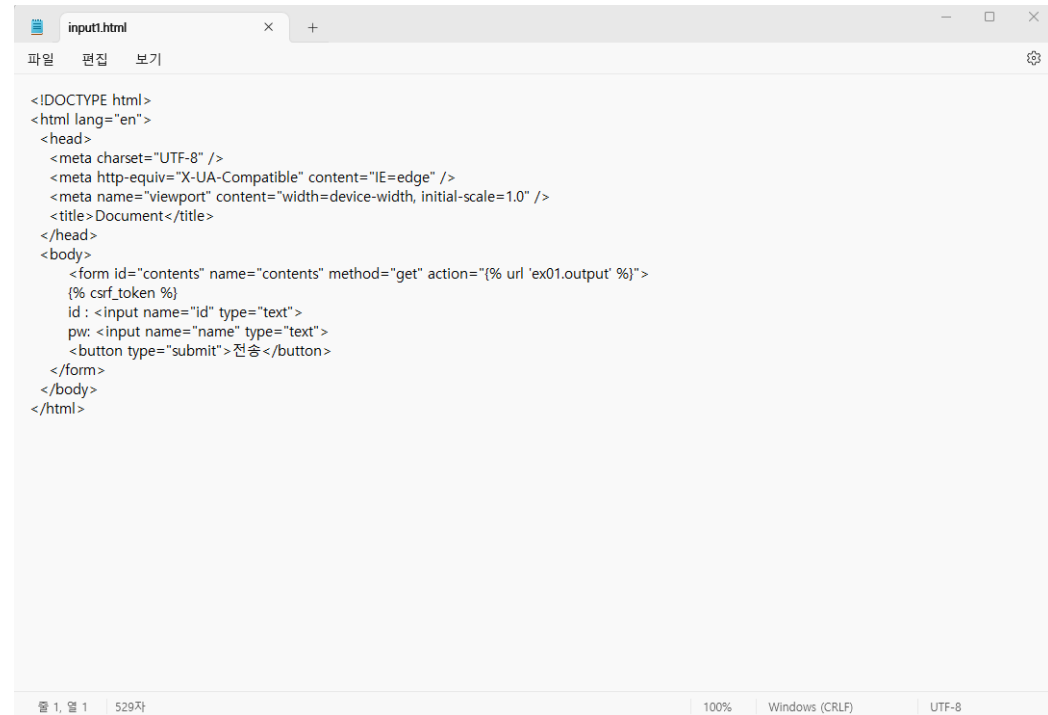
```
File Edit Selection ... Search  
Restricted Mode is intended for safe code browsing. Trust this window to enable all features. Manage Learn More  
urls.py 4 x views.py 1 x  
C:\Users\pc\Documents> django_web > web > views.py > ...  
1 from django.shortcuts import render  
2  
3 # Create your views here.  
4 def index(request):  
5     return render(request, 'index.html')  
6  
7 def func1(request):  
8     return render(request, 'input1.html')  
9  
10 def func2(request):  
11  
12     if request.method == 'GET':  
13         id = request.GET['id']  
14         data = {  
15             'data': id,  
16         }  
17         return render(request, 'result_get.html', data)  
18     #if request.method == 'POST':  
19     #    return render(request, 'index.html')
```

7. 웹 브라우저 -> GET(입력), GET(출력) 방식(예제)

3. templates에서 html 파일 만들기

- input1.html

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Document</title>
  </head>
  <body>
    <form id="contents" name="contents" method="get" action="{% url 'ex01.output' %}">
      {% csrf_token %}
      id : <input name="id" type="text">
      pw: <input name="name" type="text">
      <button type="submit">전송</button>
    </form>
  </body>
</html>
```

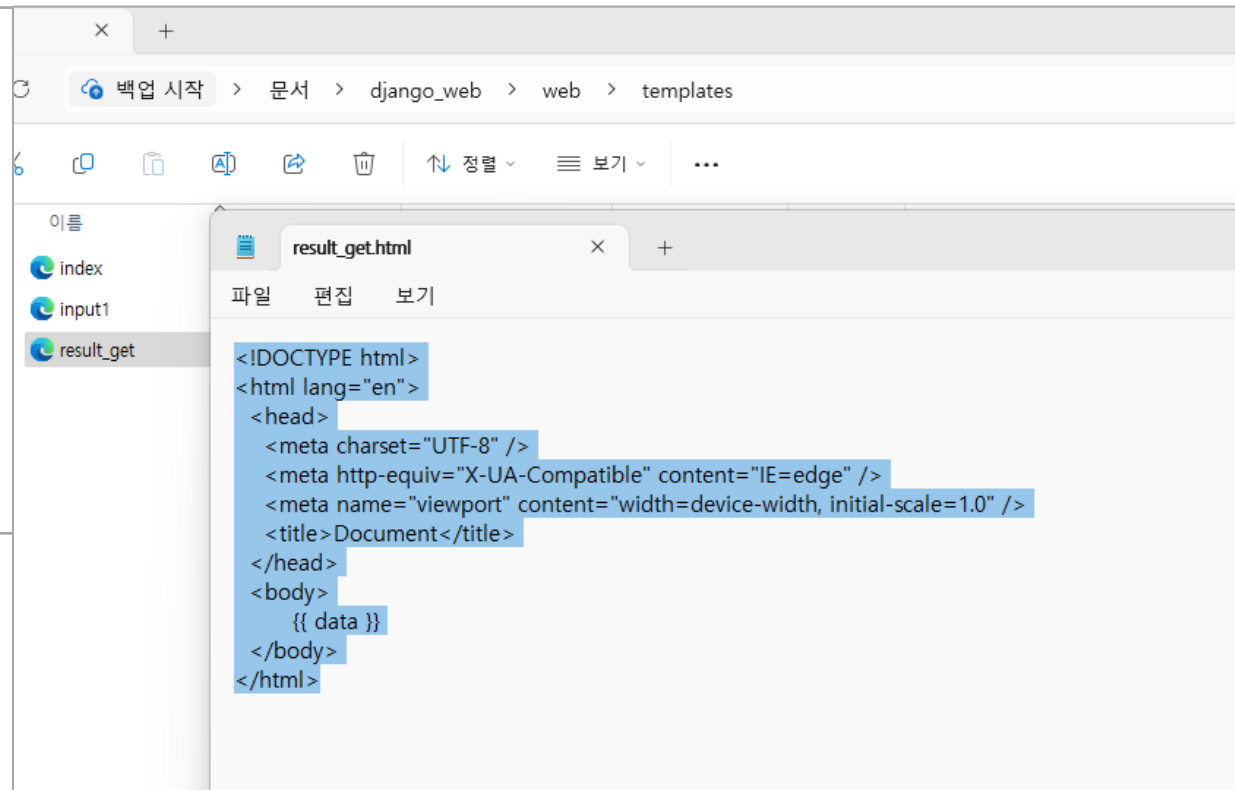


7. 웹 브라우저 -> GET(입력), GET(출력) 방식(예제)

3. templates에서 html 파일 만들기

- result_get.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <meta http-equiv="X-UA-Compatible" content="IE=edge" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <title>Document</title>
</head>
<body>
  {{ data }}
</body>
</html>
```



7. 웹 브라우저 -> GET(입력), GET(출력) 방식(예제)

4. 서버 실행

- python manage.py runserver

```
Anaconda Prompt - python rr x + v
[13/Aug/2024 16:14:55] "GET /ex01/input HTTP/1.1" 200 627
[13/Aug/2024 16:14:55] "GET /ex01/input HTTP/1.1" 200 627
[13/Aug/2024 16:15:00] "GET /ex01/output?csrfmiddlewaretoken=Qg1lbq1ME6UiKgdlBTo1Pqu0jh5etRbH5vX0WR8b7BC06ZMb1cmT38qFQxHkqA2s&id=asdf&name=asdf HTTP/1.1" 200 274
[13/Aug/2024 16:15:49] "GET /ex01/output?csrfmiddlewaretoken=Qg1lbq1ME6UiKgdlBTo1Pqu0jh5etRbH5vX0WR8b7BC06ZMb1cmT38qFQxHkqA2s&id=asdf&name=asdf HTTP/1.1" 200 274
C:\Users\pc\Documents\django_web\web\views.py changed, reloading.
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).
August 13, 2024 - 16:16:12
Django version 4.2.15, using settings 'django_web.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.

[13/Aug/2024 16:16:13] "GET /ex01/output?csrfmiddlewaretoken=Qg1lbq1ME6UiKgdlBTo1Pqu0jh5etRbH5vX0WR8b7BC06ZMb1cmT38qFQxHkqA2s&id=asdf&name=asdf HTTP/1.1" 200 278

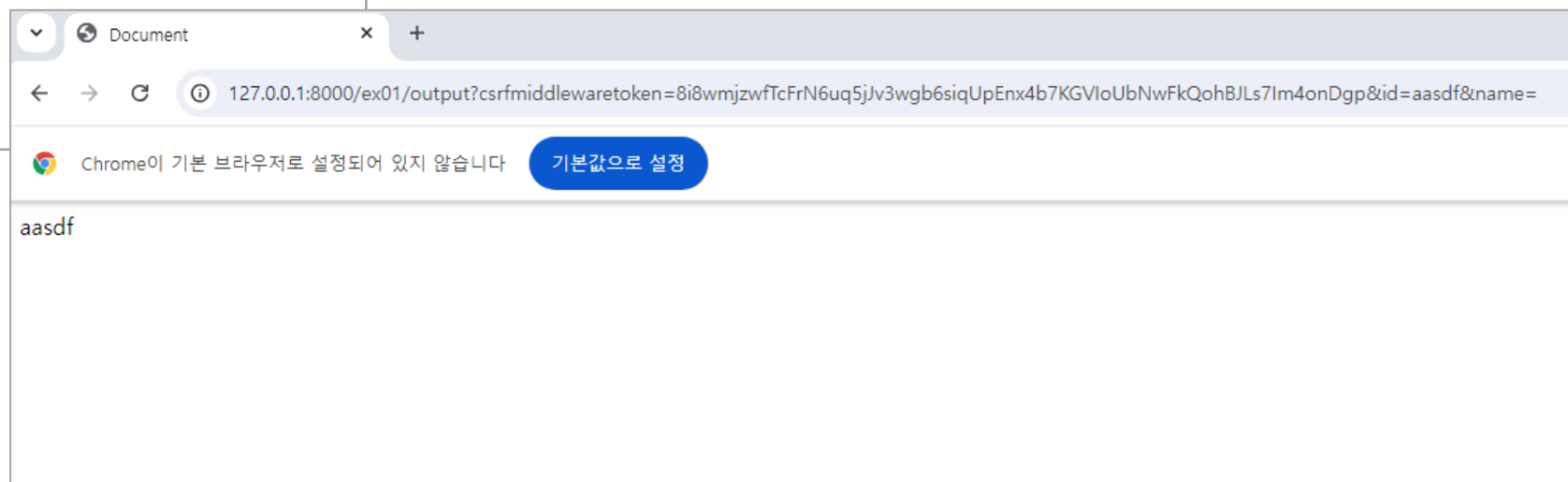
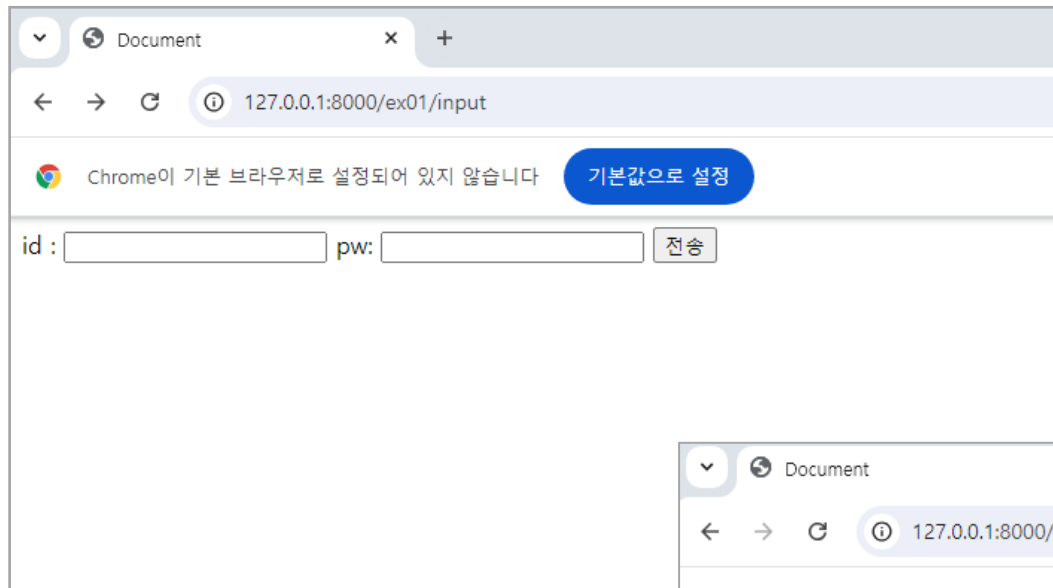
(C:\Users\pc\Documents\django_web>)python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).
August 13, 2024 - 16:22:11
Django version 4.2.15, using settings 'django_web.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

7. 웹 브라우저 -> GET(입력), GET(출력) 방식(예제)

5. 웹 브라우저에서 실행하기

- `http://127.0.0.1:8000/ex01/input`



8. 웹 브라우저 -> GET(입력), POST(출력) 방식(예제)

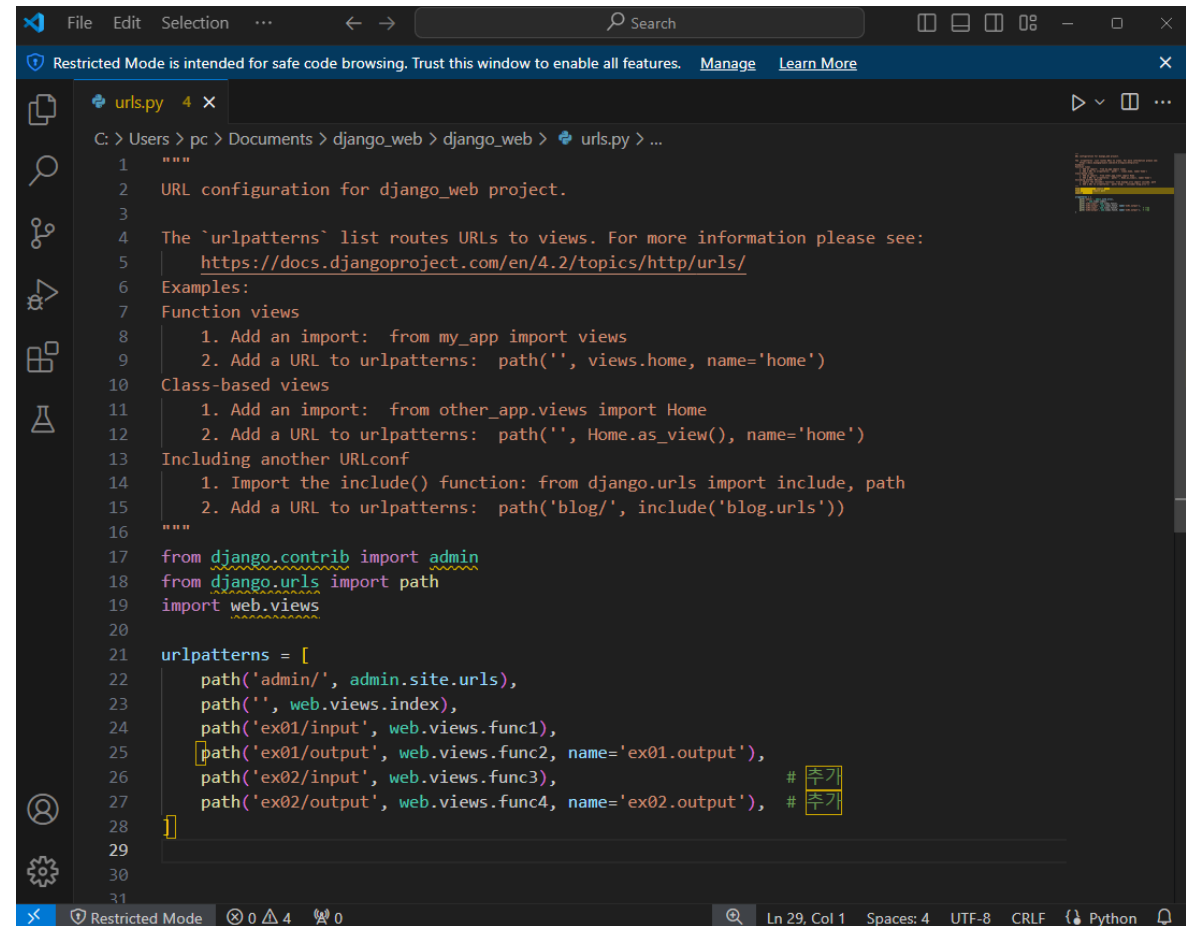
1. 프로젝트 환경(urls.py urlpatterns 변경하기)

- from [앱이름] import views을 지우고,
import [앱이름]으로 변경한다.

import web.views

- from [앱이름] import views을 지우고,
import [앱이름]으로 변경한다.

```
urlpatterns = [  
    path('admin/', admin.site.urls),  
    path(' ', web.views.index),  
    path('ex01/input', web01.views.func1),  
    path('ex01/output', web01.views.func2, name='ex01.output'),  
    path('ex02/input', web01.views.func3),  
    path('ex02/output', web01.views.func4, name='ex02.output'),  
]
```

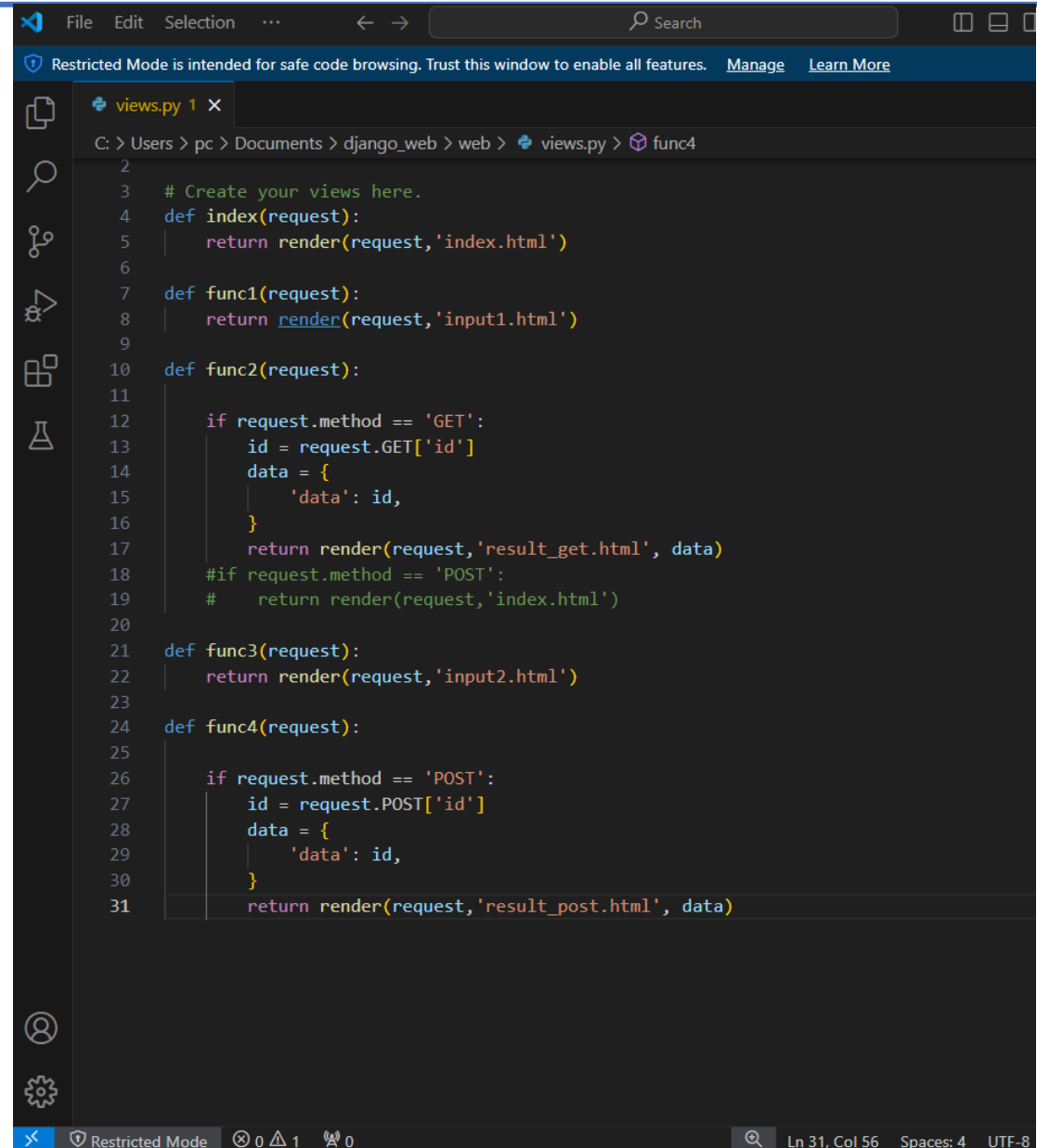


```
File Edit Selection ... Search  
Restricted Mode is intended for safe code browsing. Trust this window to enable all features. Manage Learn More  
urls.py 4 x  
C: > Users > pc > Documents > django_web > django_web > urls.py > ...  
1 """  
2 URL configuration for django_web project.  
3  
4 The `urlpatterns` list routes URLs to views. For more information please see:  
5     https://docs.djangoproject.com/en/4.2/topics/http/urls/  
6 Examples:  
7 Function views  
8     1. Add an import: from my_app import views  
9     2. Add a URL to urlpatterns: path('', views.home, name='home')  
10 Class-based views  
11     1. Add an import: from other_app.views import Home  
12     2. Add a URL to urlpatterns: path('', Home.as_view(), name='home')  
13 Including another URLconf  
14     1. Import the include() function: from django.urls import include, path  
15     2. Add a URL to urlpatterns: path('blog/', include('blog.urls'))  
16 """  
17 from django.contrib import admin  
18 from django.urls import path  
19 import web.views  
20  
21 urlpatterns = [  
22     path('admin/', admin.site.urls),  
23     path('', web.views.index),  
24     path('ex01/input', web.views.func1),  
25     path('ex01/output', web.views.func2, name='ex01.output'),  
26     path('ex02/input', web.views.func3),  
27     path('ex02/output', web.views.func4, name='ex02.output'),  
28 ]  
29  
30  
31  
Ln 29, Col 1 Spaces: 4 UTF-8 CRLF Python
```

8. 웹 브라우저 -> GET(입력), POST(출력) 방식(예제)

2. 앱 폴더에서 views.py의 내용 수정하기

```
def func3(request):  
    return render(request, 'input2.html')  
  
def func4(request):  
  
    if request.method == 'POST':  
        id = request.POST['id']  
        data = {  
            'data': id,  
        }  
        return render(request, 'result_post.html', data)
```



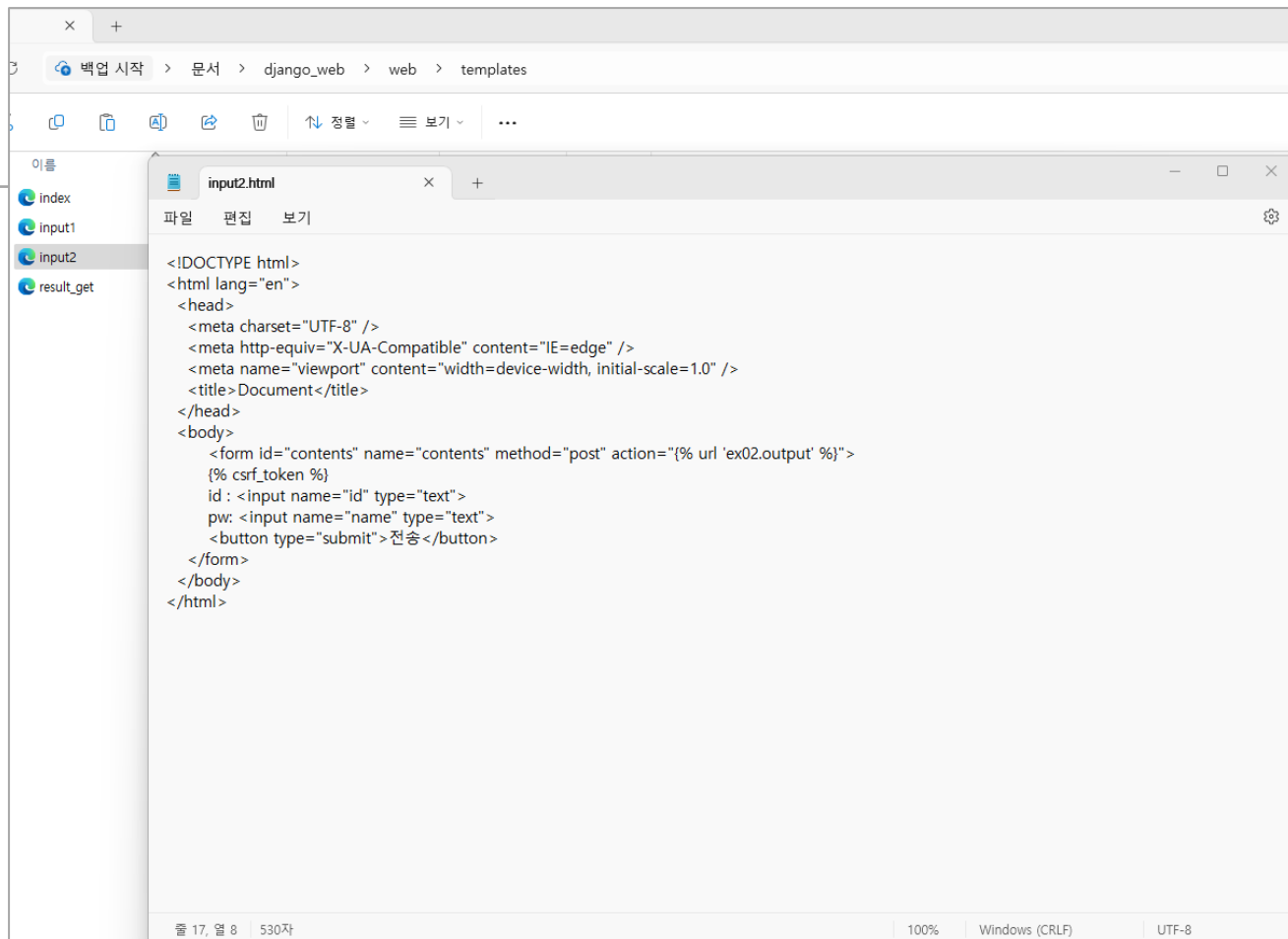
```
views.py 1 x  
C: > Users > pc > Documents > django_web > web > views.py > func4  
2  
3 # Create your views here.  
4 def index(request):  
5     return render(request, 'index.html')  
6  
7 def func1(request):  
8     return render(request, 'input1.html')  
9  
10 def func2(request):  
11  
12     if request.method == 'GET':  
13         id = request.GET['id']  
14         data = {  
15             'data': id,  
16         }  
17         return render(request, 'result_get.html', data)  
18     #if request.method == 'POST':  
19     #    return render(request, 'index.html')  
20  
21 def func3(request):  
22     return render(request, 'input2.html')  
23  
24 def func4(request):  
25  
26     if request.method == 'POST':  
27         id = request.POST['id']  
28         data = {  
29             'data': id,  
30         }  
31         return render(request, 'result_post.html', data)
```

8. 웹 브라우저 -> GET(입력), POST(출력) 방식(예제)

3. templates에서 html 파일 만들기

- input2.html

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Document</title>
  </head>
  <body>
    <form id="contents" name="contents" method="post" action="{% url 'ex02.output' %}">
      {% csrf_token %}
      id : <input name="id" type="text">
      pw: <input name="name" type="text">
      <button type="submit">전송</button>
    </form>
  </body>
</html>
```

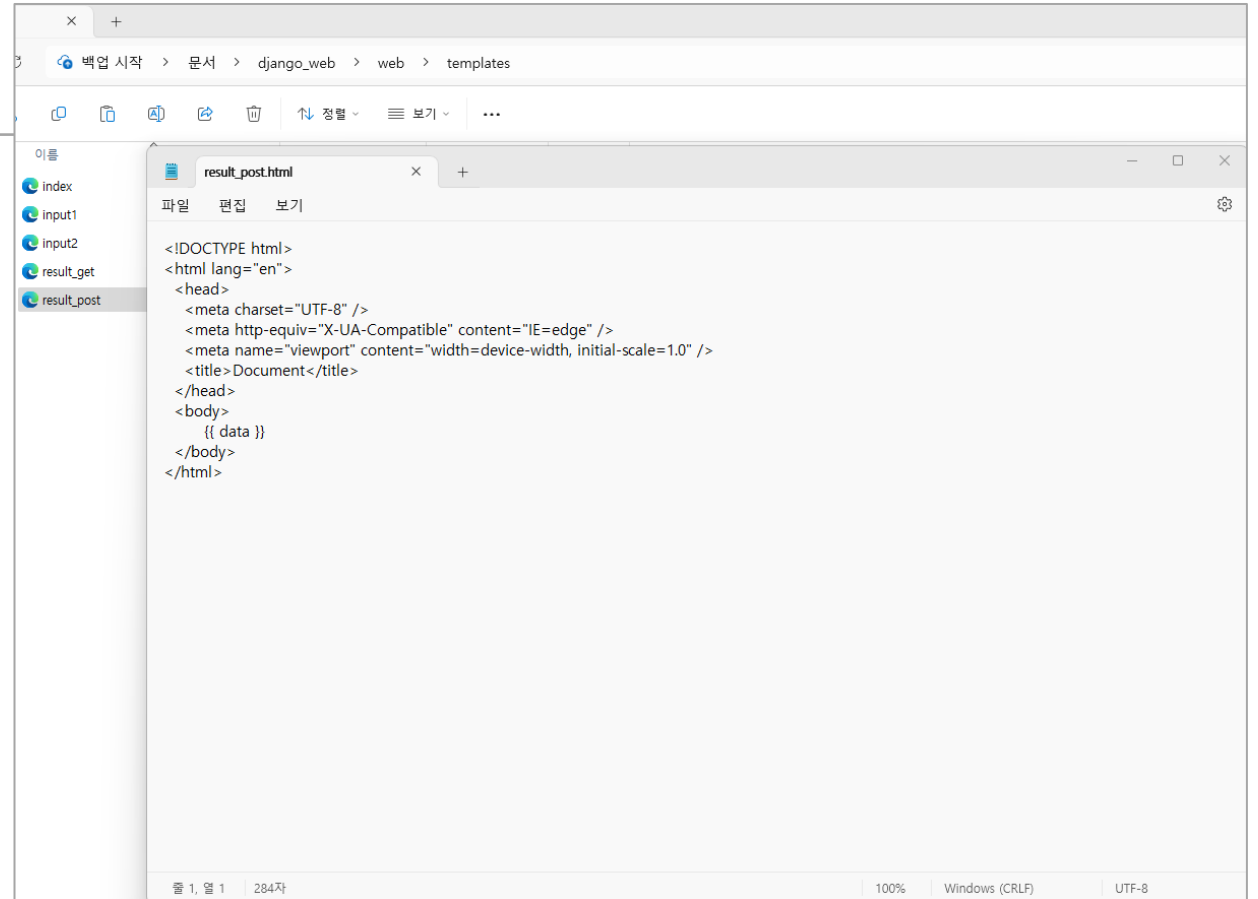


8. 웹 브라우저 -> GET(입력), POST(출력) 방식(예제)

3. templates에서 html 파일 만들기

- result_post.html

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Document</title>
  </head>
  <body>
    {{ data }}
  </body>
</html>
```



8. 웹 브라우저 -> GET(입력), POST(출력) 방식(예제)

4. 서버 실행

- python manage.py runserver

```
Anaconda Prompt - python 11 x + v
2024-08-13 오전 10:09 <DIR> web
                2개 파일 131,760 바이트
                4개 디렉터리 314,940,022,784 바이트 남음

(llama3) C:\Users\pc\Documents\django_web>dir
C 드라이브의 볼륨: os
볼륨 일련 번호: 30E1-DA1B

C:\Users\pc\Documents\django_web 디렉터리

2024-08-13 오전 09:15 <DIR> .
2024-08-13 오후 03:38 <DIR> ..
2024-08-13 오전 09:10 131,072 db.sqlite3
2024-08-13 오전 09:10 <DIR> django_web
2024-08-13 오전 09:09 688 manage.py
2024-08-13 오전 10:09 <DIR> web
                2개 파일 131,760 바이트
                4개 디렉터리 314,940,022,784 바이트 남음

(llama3) C:\Users\pc\Documents\django_web>python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).
August 13, 2024 - 16:30:32
Django version 4.2.15, using settings 'django_web.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```

8. 웹 브라우저 -> GET(입력), POST(출력) 방식(예제)

5. 웹 브라우저에서 실행하기

- `http://127.0.0.1:8000/ex01/input`

