

웹 프로그래밍

Update 2023.02



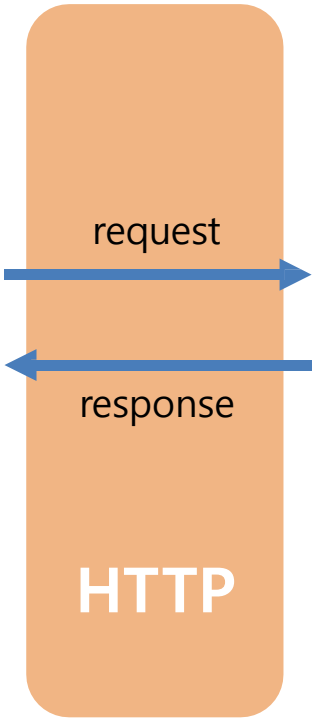
device



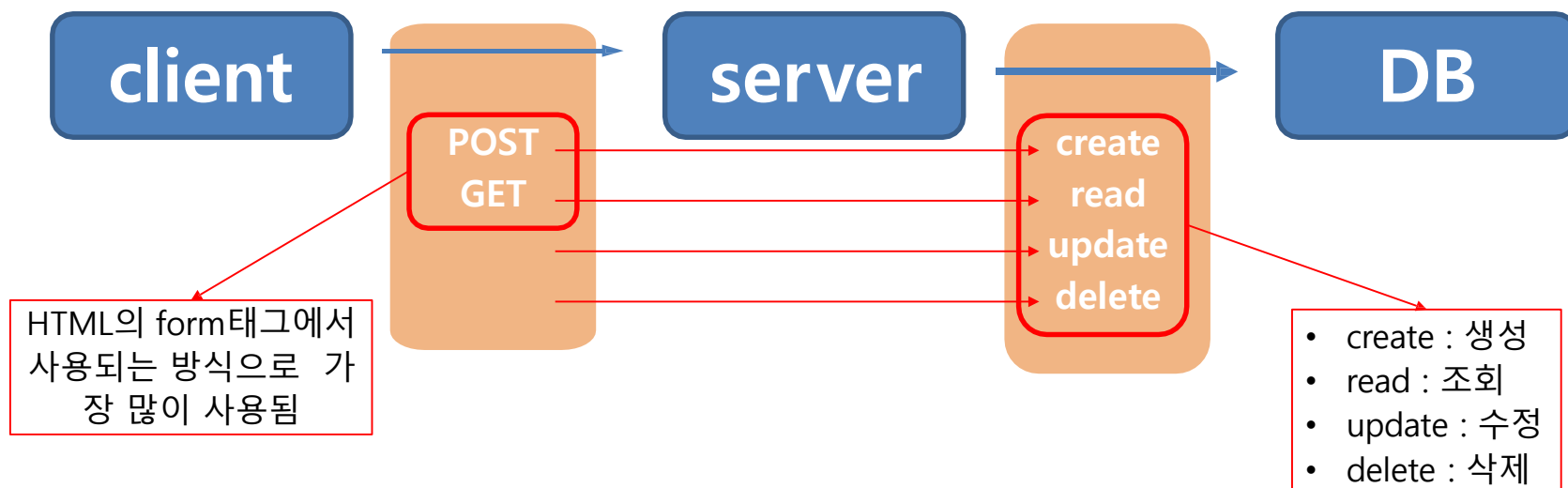
software

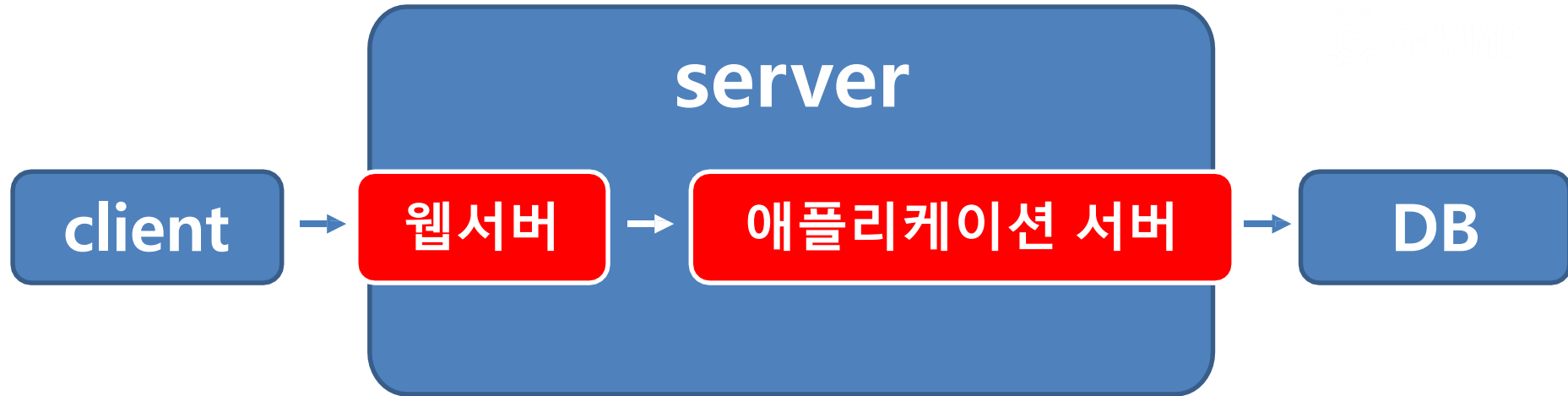


client



server





[웹서버]

- 주로 정적인 데이터 요청 처리
- 동적인 데이터 요청 시 애플리케이션 서버에 전달

[애플리케이션서버]

- 주로 동적인 데이터 요청 처리
- Database 연동

Django설치 및 프로젝트 생성



장고(Django)란

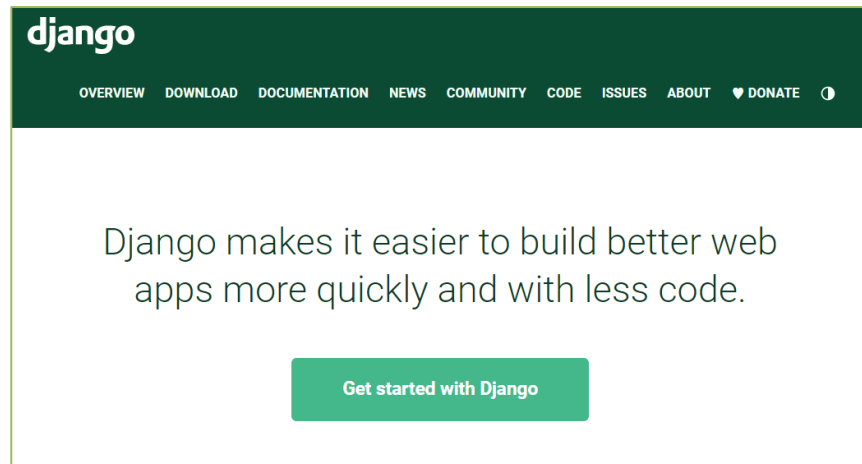
파이썬으로 프로그래밍된 오픈 소스 웹 프레임워크

<https://www.djangoproject.com>

<https://tutorial.djangogirls.org/>

<https://developer.mozilla.org/ko/docs/Learn/Server-side/Django>

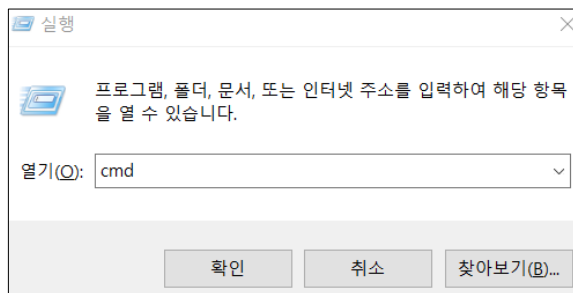
<https://docs.djangoproject.com/en/4.1/>





python 설치 확인 및 : pip 업그레이드

pip : Python Install Packag의 약자로 PyPI(Python Package Index)의 SW패키지를 사용하기 위한 명령어



커맨드(Command) 창 실행 : [Window] + [R]

```
C:\Users\Wlkiop>python -m pip install --upgrade pip --user
Collecting pip
  Using cached pip-21.0.1-py3-none-any.whl (1.5 MB)
Installing collected packages: pip
  WARNING: The scripts pip.exe, pip3.8.exe and pip3.exe are installed in 'C:\Users\Wlkiop\AppData\Roaming\Python\Python38\Scripts' which is not on PATH.
  Consider adding this directory to PATH or, if you prefer to suppress this warning, use --no-warn-script-location.
Successfully installed pip-21.0.1

C:\Users\Wlkiop>pip --version
pip 21.0.1 from C:\Users\Wlkiop\AppData\Roaming\Python\Python38\site-packages
pip (python 3.8)
```



작업 폴더 생성 및 가상환경 생성

작업폴더 생성

```
>> cd c:/
```

```
>> mkdir C:\django
```

가상환경 생성

```
C:\django> conda install virtualenv 또는 pip install virtualenv
```

```
C:\django> virtualenv venv
```

venv 생성확인

```
C:\django> dir
```

C:\django 디렉터리

2023-02-12	오후 11:07	<DIR>	.
2023-02-12	오후 11:07	<DIR>	..
2023-02-12	오후 11:07	<DIR>	venv
	0개 파일		0 바이트
	3개 디렉터리	63,047,069,696 바이트	남음



가상환경 실행

```
cd C:\django\venv\Scripts
```

```
C:\django\venv\Scripts> activate
```

```
C:\django\venv\Scripts>activate  
(venv) C:\django\venv\Scripts>
```

가상환경 해제

```
(venv) C:\django\venv\Scripts>deactivate  
C:\django\venv\Scripts>
```



```
C:\django\venv\Scripts> activate
```

```
(venv) > pip list
```

```
(venv) c:\django\venv\Scripts>pip list
```

Package	Version
asgiref	3.6.0
backports.zoneinfo	0.2.1
Django	4.1.6
pip	23.0
setuptools	65.7.0
sqlparse	0.4.3
tzdata	2022.7
wheel	0.38.4



장고 작업폴더로 이동

```
cd C:\django
```

project는 프로젝트명

```
C:\django>> django-admin startproject project
```

dir 명령으로 project 폴더가 생성되었는지 확인

```
C:\django>> dir
```

c:\django\project 디렉터리

2023-02-12	오후 11:57	<DIR>	.
2023-02-12	오후 11:57	<DIR>	..
2023-02-12	오후 11:57		685 manage.py
2023-02-12	오후 11:57	<DIR>	project
	1개 파일		685 바이트
	3개 디렉터리	64,217,149,440 바이트	남음



Django 앱 생성 및 실행

장고 프로젝트 폴더로 이동

```
(venv) django >> cd project
```

myApp 은 앱이름

```
(venv) django/project >> python manage.py startapp myApp
```

3개 확인 project, myApp, manage.py

```
>> dir
```

```
c:\w\django\project 디렉터리
2023-02-13 오전 12:08 <DIR> .
2023-02-13 오전 12:08 <DIR> ..
2023-02-12 오후 11:57      685 manage.py
2023-02-13 오전 12:08 <DIR> myApp
2023-02-13 오전 12:05 <DIR> project
                        1개 파일      685 바이트
                        4개 디렉터리 63,896,457,216 바이트 남음
```



Django 앱 생성 및 실행

서버 실행

(venv) django/project >> python manage.py runserver

```
(venv) c:\#django#project>python manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).

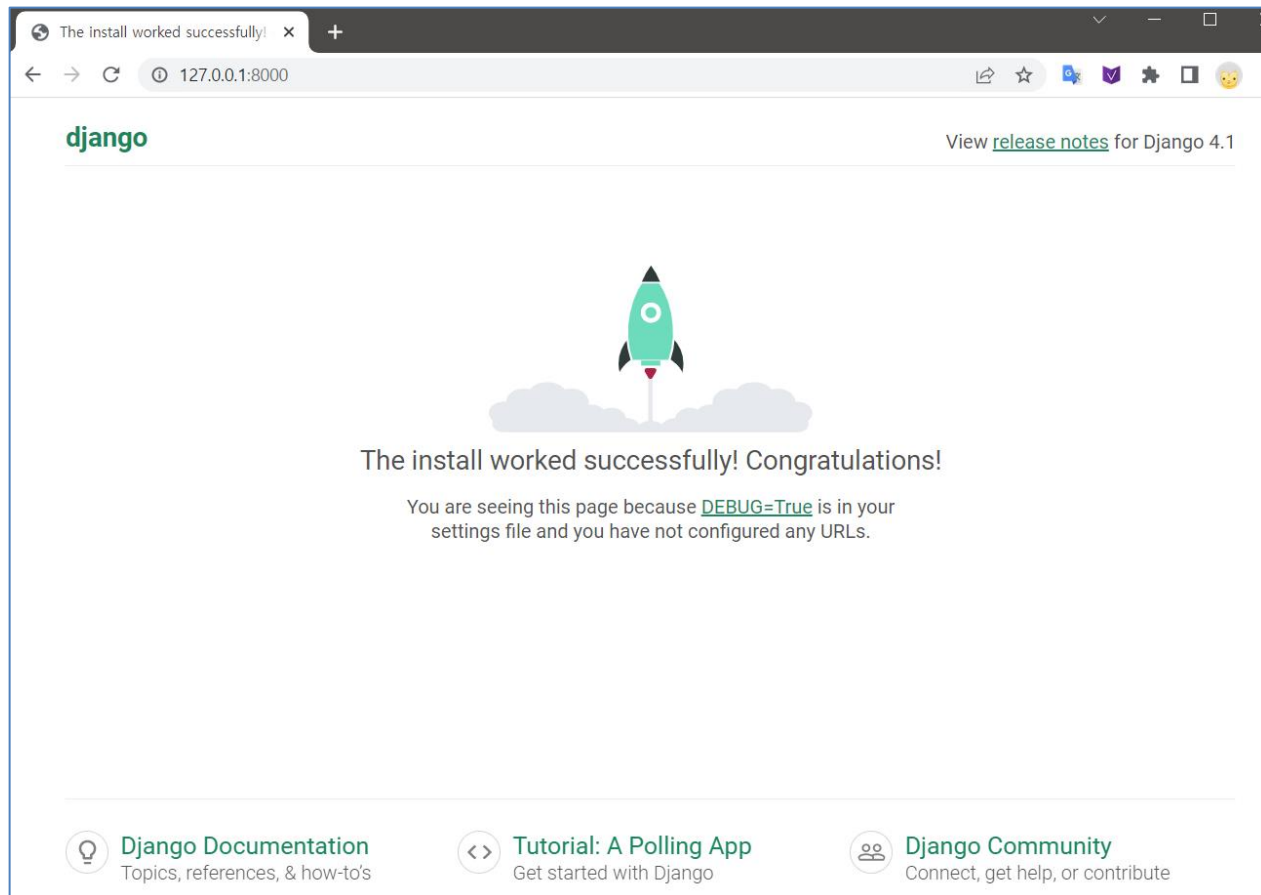
You have 18 unapplied migration(s). Your project may not work properly until you apply the migrations for app(s): admin, auth, contenttypes, sessions.
Run 'python manage.py migrate' to apply them.
February 13, 2023 - 00:12:07
Django version 4.1.6, using settings 'project.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CTRL-BREAK.
```



Django 앱 생성 및 실행

웹브라우저에 주소 입력

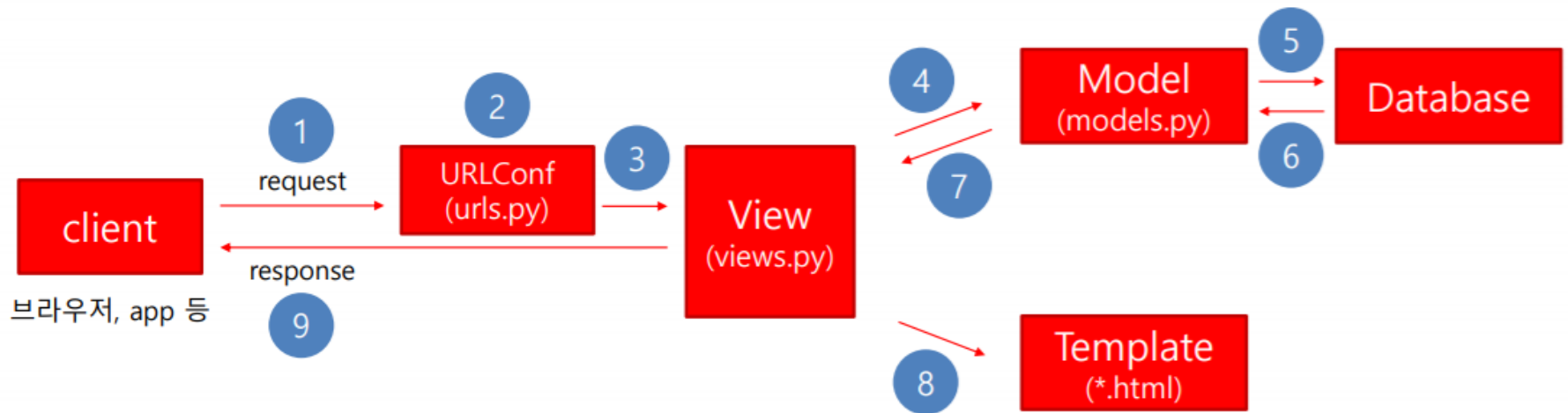
http://127.0.0.1:8000/



주소 생성과 웹페이지 연동

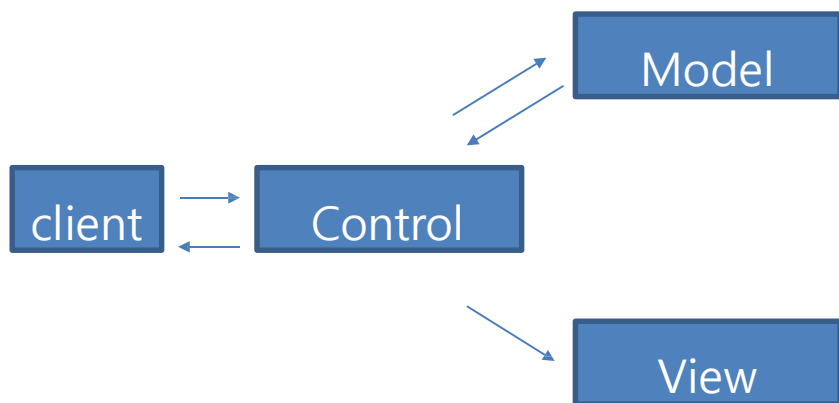


Django framework flow

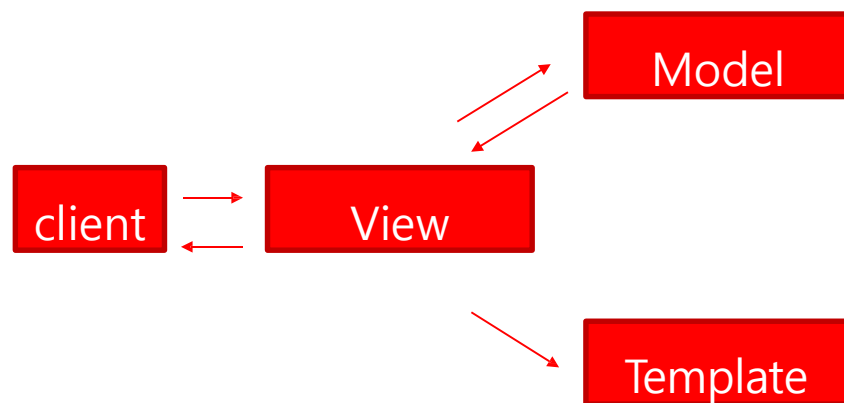




MVC 패턴



MVT 패턴



작업과정

- 1) 프로젝트에서 앱등록과 타임존 변경: `project/settings.py`
- 2) 프로젝트에 앱 주소 등록 : `project/urls.py`
- 3) 앱에 주소 등록 : `myApp/urls.py`
- 4) 뷰함수 등록 : `myApp/views.py`
- 5) 웹사이트에서 결과 확인

프로젝트에서 앱등록과 타임존 변경

1) project/settings.py

```
33  INSTALLED_APPS = [  
34      'django.contrib.admin',  
35      'django.contrib.auth',  
36      'django.contrib.contenttypes',  
37      'django.contrib.sessions',  
38      'django.contrib.messages',  
39      'django.contrib.staticfiles',  
40      'myApp', #추가 : 앱 등록  
41  ]  
  
109  # TIME_ZONE = 'UTC'  
110  TIME_ZONE = 'Asia/Seoul'
```


2) # 프로젝트에 앱 주소 등록 : project/urls.py

```
16 from django.contrib import admin
17 from django.urls import path, include # 변경
18
19 urlpatterns = [
20     path('admin/', admin.site.urls),
21     # 추가
22     path('', include('myApp.urls')),
23     path('myApp/', include('myApp.urls')),
24 ]
```

3) 앱에 주소 등록

myApp 폴더안에 urls.py 생성

myApp/urls.py 소스 삽입

```
◎ django > project > myApp >  urls.py > ...  
1  from django.urls import path  
2  from . import views  
3  
4  # 시작페이지 주소와 뷰함수 연결  
5  urlpatterns = [  
6      path('', views.index),  
7  ]
```

뷰함수 등록

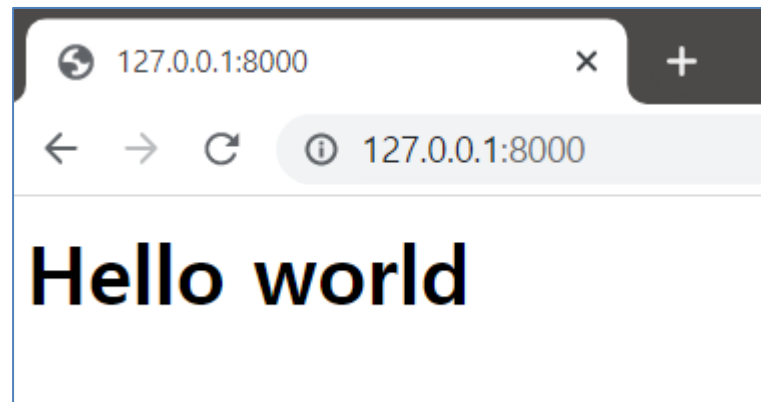
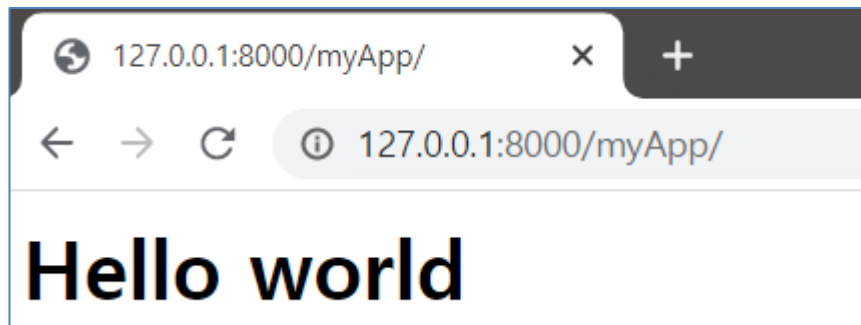
4) myApp/views.py

```
1  from django.shortcuts import render
2  # 추가
3  from django.http import HttpResponse
4
5  # /, /myApp/ 주소와 연결되는 뷰함수
6  def index(request):
7      return HttpResponse("<h1>Hello world</h1>")
```

5) 서버 실행 후 결과 확인

터미널창에서 서버 실행

```
(venv) c:\django\project>python manage.py runserver
```





작업과정

- 1) 앱에 url 등록
- 2) 뷰함수 등록 : url과 html 페이지 연결
- 3) 템플릿 폴더 생성
- 4) HTML 파일 생성
- 5) 웹사이트에서 결과 확인



(1) 앱에 url 등록

myApp/urls.py

```
1  from django.urls import path
2  from . import views
3
4  urlpatterns = [
5      # 시작페이지 주소와 뷰함수 연결
6      path('', views.index),
7      # /myApp/sub1/
8      path('sub1/', views.sub1),
9  ]
```



(2) 뷰함수 등록

myApp/views.py

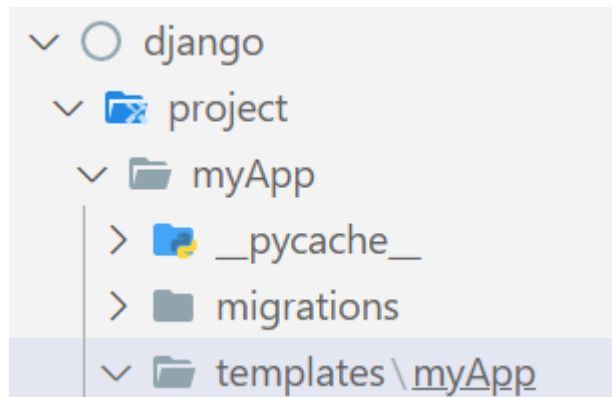
```
9  # /myApp/sub1 주소와 연결되는 뷰함수
10 def sub1(request):
11     return render(request, 'myApp/sub1.html')
```



(3) 템플릿 폴더 생성

앱 폴더(myApp)안에 templates 폴더 생성

templates 폴더안에 myApp 폴더 다시 생성

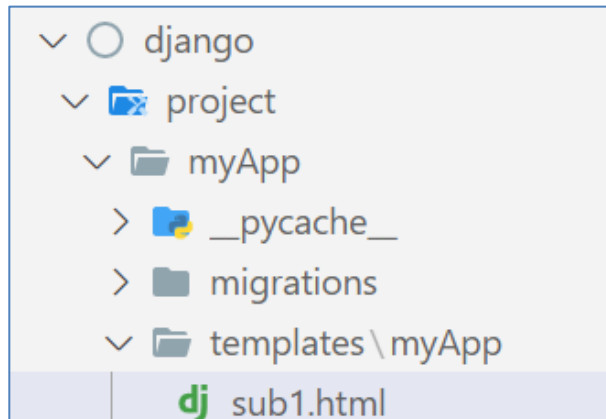




HTML 페이지와 연동하기

(4) HTML 파일 생성 : sub1.html

myApp/templates/myApp/sub1.html 파일 생성



```
◎ django > project > myApp > templates > myApp > 🟢 sub1.html
1  <body>
2    <h1>Sub1</h1>
3    <hr>
4    <h2>welcome to django</h2>
5  </body>
```



5) 서버 실행 후 결과 확인

터미널창에서 서버 실행하기

```
(venv) c:\django\project>python manage.py runserver
```

웹브라우저 주소 : <http://127.0.0.1:8000/myApp/sub1/>





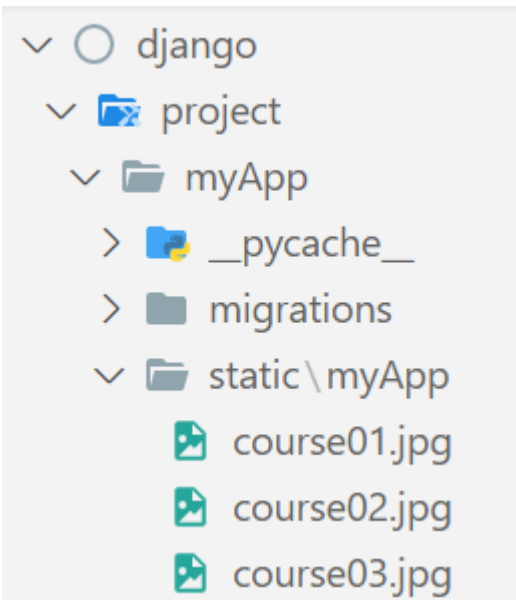
작업과정

- 1) static 폴더 생성 및 이미지 셋팅
- 2) url 등록
- 3) 뷰함수 등록
- 4) html 파일 생성 (이미지 삽입)
- 5) 프로젝트 settings에 static 폴더 등록



static 파일 관리하기

- # 1) static 폴더 생성 및 이미지 셋팅
- # 앱 폴더(myApp)안에 static 폴더 생성
- # static 폴더안에 myApp 폴더 생성
- # project > myApp > static > myApp 구조 확인
- # 위의 폴더(static > myApp)안에 이미지 파일 저장



무료 이미지 사이트

<https://pixabay.com/ko/>

<https://unsplash.com>

<https://picjumbo.com/>



static 파일 관리하기

2) url 등록

myApp/urls.py

🔍 django > project > myApp > 🐍 urls.py > ...

```
1  from django.urls import path
2  from . import views
3
4  urlpatterns = [
5      # 시작페이지 주소와 뷰함수 연결
6      path('', views.index),
7      # /myApp/sub1/
8      path('sub1/', views.sub1),
9      # /myApp/sub2/
10     path('sub2/', views.sub2),
11 ]
```




3) 뷰함수 등록

myApp/views.py

```
13  # /myApp/sub2 주소와 연결되는 뷰함수
14  def sub2(request):
15      return render(request, 'myApp/sub2.html')
```



static 파일 관리하기

4) sub2.html 생성

myApp/templates/myApp/sub2.html 파일 생성

static 폴더안의 이미지 경로는?

/static/앱이름/이미지파일명.확장자

◎ django > project > myApp > templates > myApp > **dj** sub2.html

```
1 <div style="text-align: center;">
2     <h1>Sub2</h1>
3     <hr>
4     <h2>static 폴더를 이용한 이미지 삽입</h2>
5     
6 </div>
```



static 파일 관리하기

5) 프로젝트 settings에 static 폴더 등록

project/settings.py

```
13  from pathlib import Path
14  import os
```

```
121  STATIC_URL = '/static/'
122  # 추가
123  STATICFILES_DIRS = [
124      os.path.join(BASE_DIR, 'static', 'myApp')
125  ]
```



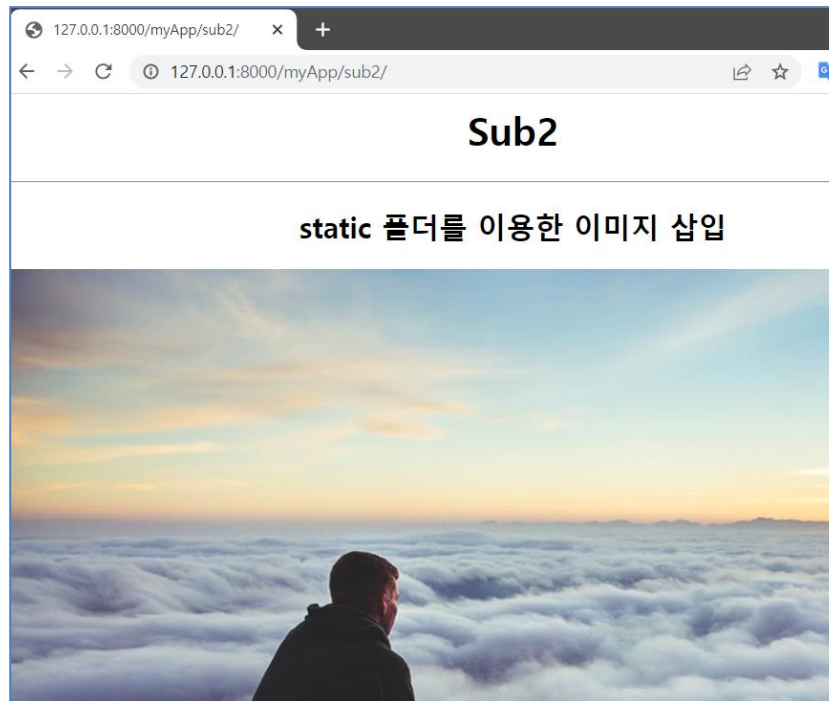
static 파일 관리하기

6) 서버 실행 후 결과 확인

터미널창에서 서버 실행하기

(venv) c:\wdjango\project>python manage.py runserver

웹브라우저 주소 : <http://127.0.0.1:8000/myApp/sub2/>





부트스트랩 적용하기

부트스트랩이란?

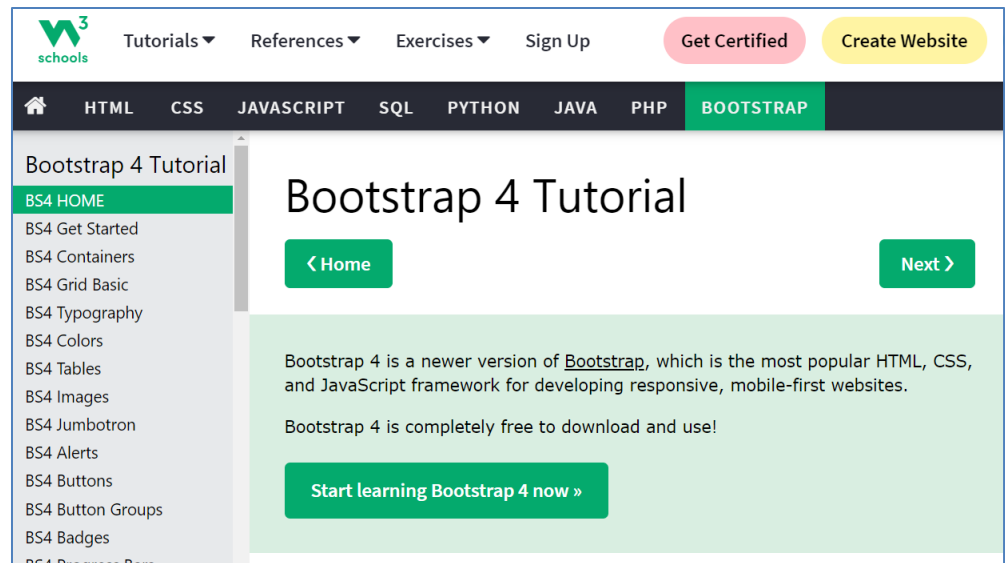
- 반응형 기반 웹사이트 프레임웍스

관련 사이트

<https://www.w3schools.com/bootstrap4/default.asp>

<https://startbootstrap.com/>

<https://getbootstrap.com/>





작업과정

1) 부트스트랩 CDN 소스 복사하기

https://www.w3schools.com/bootstrap4/bootstrap_get_started.asp

jsDelivr:

```
<!-- Latest compiled and minified CSS -->
<link rel="stylesheet"
href="https://cdn.jsdelivr.net/npm/bootstrap@4.6.2/dist/css/bootstrap.min.css">

<!-- jQuery library -->
<script src="https://cdn.jsdelivr.net/npm/jquery@3.6.3/dist/jquery.slim.min.js">
</script>

<!-- Popper JS -->
<script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.1/dist/umd/popper.min.js">
</script>

<!-- Latest compiled JavaScript -->
<script
src="https://cdn.jsdelivr.net/npm/bootstrap@4.6.2/dist/js/bootstrap.bundle.min.js">
</script>
```



2) html 파일에 부트스트랩 소스 추가 – head 부분

```
8      <!-- Latest compiled and minified CSS -->
9      <link
10         rel="stylesheet"
11         href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css"
12     />
13     <!-- jQuery library -->
14     <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
15
16     <!-- Popper JS -->
17     <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.16.0/umd/popper.min.js"></script>
18
19     <!-- Latest compiled JavaScript -->
20     <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>
21
22     <style>
23         .container {
24             border: 1px solid lightgray;
25             margin-top: 50px;
26             padding: 10px;
27         }
28         .container div {
29             padding: 20px;
30         }
31     </style>
```



```
33 <body>
34   <div class="container text-center">
35     <h1>Sub2</h1>
36     <hr />
37     <h2>static 폴더를 이용한 이미지 삽입</h2>
38     
43   </div>
44 </body>
45 </html>
46
```




3) 서버 실행 후 결과 확인

터미널창에서 서버 실행하기

```
(venv) c:\Wdjango\project>python manage.py runserver
```

웹브라우저 주소 : <http://127.0.0.1:8000/myApp/sub2/>

Sub2

static 폴더를 이용한 이미지 삽입



Django와 파이썬 데이터



템플릿 변수

- html 페이지에서 뷰함수를 이용하여 전달받은 데이터(문자열, 리스트...)를 표시하는 기능
- {{ 템플릿 변수 }}

작업과정

- 1) url 등록
- 2) 뷰함수 등록: url과 html 페이지 연결, context를 이용하여 딕셔너리 형태로 데이터 전송
- 3) html 파일 생성. 템플릿변수 {{}} 로 context의 데이터 출력
- 4) 결과 확인



(1) url 등록

myApp/urls.py

```
1  from django.urls import path
2  from . import views
3
4  urlpatterns = [
5      # 시작페이지 주소와 뷰함수 연결
6      path('', views.index),
7      # /myApp/sub1/
8      path('sub1/', views.sub1),
9      # /myApp/sub2/
10     path('sub2/', views.sub2),
11     # /myApp/sub3/
12     path('sub3/', views.sub3),
13 ]
```



(2) 뷰함수 등록

myApp/views.py

```
18 # /myApp/sub3 주소와 연결되는 뷰함수
19 def sub3(request):
20     # 딕셔너리 형태로 전송 데이터 정의
21     context = {
22         'name': '임영웅',
23         'id': 'YYY',
24         'age': 34,
25         'grade': [ 90, 85, 77]
26     }
27     return render(request, 'myApp/sub3.html', context)
```



(3) sub3.html 생성

myApp/templates/myApp/sub3.html

```
33 <body>
34   <div class="container text-center">
35     <h1>Sub3</h1>
36     <hr />
37     <h2>템플릿 변수</h2>
38     <hr />
39     <!-- 파이썬에서 전달받은 데이터 -->
40     <!-- context 딕셔너리에서 정의된 키로 호출 -->
41     <ul class="list-group text-left" style="width: 50%; margin: 0 auto">
42       <li class="list-group-item active">Name : {{name}}</li>
43       <li class="list-group-item">Id : {{id}}</li>
44       <li class="list-group-item">Age : {{age}}</li>
45       <li class="list-group-item">Grade : {{grade}}</li>
46     </ul>
47   </div>
48 </body>
```

부트스트랩 리스트 그룹 디자인 참조

https://www.w3schools.com/bootstrap4/bootstrap_list_groups.asp



4) 서버 실행 후 결과 확인

터미널창에서 서버 실행하기

```
(venv) c:\Wdjango\project>python manage.py runserver
```

웹브라우저 주소 : <http://127.0.0.1:8000/myApp/sub3/>

Sub3	
템플릿 변수	
Name : 임영웅	
Id : YYY	
Age : 34	
Grade : [90, 85, 77]	



```
{% for 아이템변수 in 리스트명/튜플명/range() %}
```

```
...
```

```
{% endfor %}
```

작업과정

1) url 등록

2) 뷰함수 등록: url과 html 페이지 연결. 리스트안에 튜플로 정의한 후 데이터 전달

3) html 파일 생성 : sub4.html. 템플릿 문법 for 이용

4) 결과 확인 <http://127.0.0.1:8000/myApp/sub4/>



(1) url 등록

myApp/urls.py

```
1  from django.urls import path
2  from . import views
3
4  urlpatterns = [
5      # 시작페이지 주소와 뷰함수 연결
6      path('', views.index),
7      # /myApp/sub1/
8      path('sub1/', views.sub1),
9      # /myApp/sub2/
10     path('sub2/', views.sub2),
11     # /myApp/sub3/
12     path('sub3/', views.sub3),
13     # /myApp/sub4/
14     path('sub4/', views.sub4),
15 ]
```



(2) 뷰함수 등록

myApp/views.py

```
32 def sub4(request):
33     book_info = [
34         ('title', '어린왕자'),
35         ('writer', '생텍쥐페리'),
36         ('price', 15000),
37         ('ISBN', '483731-849-90'),
38     ]
39     student_info = [
40         ('홍길동', '남', '부산', 23),
41         ('고길동', '남', '전주', 28),
42         ('윤길동', '남', '서울', 23),
43         ('박길동', '남', '마산', 23),
44         ('정길순', '여', '울산', 22),
45     ]
46     context = {
47         'book_info': book_info,
48         'student_info': student_info,
49         'numList': range(1,101,2),
50     }
51     return render(request, 'myApp/sub4.html', context)
```



(3) sub4.html 파일 생성

myApp/templates/myApp/sub4.html

```
33     <body>
34         <div class="container text-center">
35             <h1>Sub4</h1>
36             <hr>
37             <h2>템플릿 for 반복문 </h2>
38             <!-- context에 정의된 키로 호출 -->
39             <h3>book_info</h3>
40             {{ book_info }}
41             <ul class="list-group text-left" style="width:50%; margin:25px auto">
42                 {% for i,j in book_info %}
43                 <li class="list-group-item"> {{i}} => {{j}} </li>
44                 {% endfor %}
45             </ul>
46             <hr>
```



템플릿 문법 - for 반복문

```
47 <h3>student_info</h3>
48 {{ student_info }} <br><br>
49 <table class="table table-bordered" style="width:70%;margin:10px auto">
50   <!-- 제목행 -->
51   <tr>
52     <th>학생명</th>
53     <th>성별</th>
54     <th>주소</th>
55     <th>나이</th>
56   </tr>
57   {% for name, gender, addr, age in student_info %}
58   <tr>
59     <td>{{name}}</td>
60     <td>{{gender}}</td>
61     <td>{{addr}}</td>
62     <td>{{age}}</td>
63   </tr>
64   {% endfor %}
65 </table>
```



```
66 <hr>
67 <h3>numList</h3>
68 {{numList}}
69 <p>
70 <!-- &nbsp; 한칸공백 -->
71 {% for i in numList %}
72     {{i}} &nbsp;&nbsp;&nbsp;
73 {% endfor %}
74 </p>
```



4) 서버 실행 후 결과 확인

터미널창에서 서버 실행하기

(venv) c:\Wdjango\project>python manage.py runserver

웹브라우저 주소 : http://127.0.0.1:8000/myApp/sub4/

템플릿 for 반복문 book_info

[('title', '어린왕자'), ('writer', '생텍쥐페리'), ('price', 15000), ('ISBN', '483731-849-90')]

title => 어린왕자

writer => 생텍쥐페리

price => 15000

ISBN => 483731-849-90



student_info

[('홍길동', '남', '부산', 23), ('고길동', '남', '전주', 28), ('윤길동', '남', '서울', 23), ('박길동', '남', '마산', 23), ('정길순', '여', '울산', 22)]

학생명	성별	주소	나이
홍길동	남	부산	23
고길동	남	전주	28
윤길동	남	서울	23
박길동	남	마산	23
정길순	여	울산	22



numList

range(1, 101, 2)

1	3	5	7	9	11	13	15	17	19	21	23	25	27	29	31	33	35	37
39	41	43	45	47	49	51	53	55	57	59	61	63	65	67	69	71	73	
		75	77	79	81	83	85	87	89	91	93	95	97	99				



```
{% for 아이템변수 in 리스트명/튜플명/range() %}
```

```
...
```

```
{% endfor %}
```

작업과정

1) url 등록

2) 뷰함수 등록: url과 html 페이지 연결. 리스트안에 튜플로 정의한 후 데이터 전달

3) html 파일 생성 : sub4.html. 템플릿 문법 for 이용

4) 결과 확인 <http://127.0.0.1:8000/myApp/sub4/>



(1) url 등록

myApp/urls.py

```
from django.urls import path
from . import views

urlpatterns = [
    # 시작페이지 주소와 뷰함수 연결
    path('', views.index),
    # /myApp/sub1/
    path('sub1/', views.sub1),
    # /myApp/sub2/
    path('sub2/', views.sub2),
    # /myApp/sub3/
    path('sub3/', views.sub3),
    # /myApp/sub4/
    path('sub4/', views.sub4),
    # /myApp/sub5/
    path('sub5/', views.sub5),
]
```



(2) 뷰함수 등록

myApp/views.py

```
55 def sub5(request):
56     book_list = [
57         ('파이썬 도장', 33000),
58         ('C언어', 23000),
59         ('JAVA', 43000),
60     ]
61     context = {
62         'book_list':book_list ,
63     }
64     return render(request, 'myApp/sub5.html', context)
```



(3) sub5.html 파일 생성

myApp/templates/myApp/sub5.html/

```
34 <div class="container text-center">
35   <h1>Sub5</h1>
36   <hr />
37   <h2>템플릿 if 제어문</h2>
38   <h3>book_list</h3>
39   {% if book_list %}
40   <p>{{book_list}}</p>
41   {% else %}
42   <p>책이 없습니다.</p>
43   {% endif %}
44   <hr />
45   <h3>for 문 + if 문</h3>
46   {% for title, price in book_list %}
47     {% if title == 'JAVA' %}
48     <p>제목 : {{title}} , 가격 : {{price}}</p>
49     {% else %}
50     <p>...</p>
51     {% endif %}
52   {% endfor %}
53 </div>
```



4) 서버 실행 후 결과 확인

터미널창에서 서버 실행하기

(venv) c:\Wdjangow\project>python manage.py runserver

웹브라우저 주소 : http://127.0.0.1:8000/myApp/sub5/

Sub5

템플릿 if 제어문

book_list

[('파이썬 도장', 33000), ('C언어', 23000), ('JAVA', 43000)]

for 문 + if 문

...

...

제목 : JAVA , 가격 : 43000



```
import csv
저장리스트변수 = []
with open('csv파일경로', 'r') as f:
    csv_data = csv.reader(f)
    for row in csv_data:
        저장리스트변수.append(row)
```

작업과정

- 1) csv 파일 준비
- 2) url 등록
- 3) 뷰함수 등록 : csv => list 구조
- 4) html 파일 생성
- 5) 결과 확인 <http://127.0.0.1:8000/myApp/sub6/>

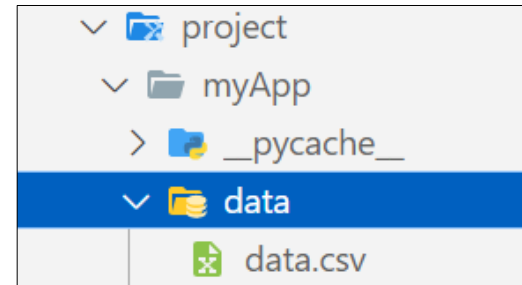


(1) csv 파일 준비

myApp 하단에 data 폴더 생성

data 폴더안에 csv 파일 저장

data/data.csv



(2) url 등록

myApp/urls.py

```
# /myApp/sub6/  
path('sub6/', views.sub6),
```



(3) 뷰함수 등록

myApp/views.py

```
66 ∨ import csv
67     import os
68
69 ∨ def sub6(request):
70     data_list = []
71     print(os.getcwd()) # 현재 경로 출력
72 ∨     with open('myApp/data/data.csv', 'r') as f:
73         csv_data = csv.reader(f)
74 ∨         for row in csv_data:
75             data_list.append(row)
76     context = { 'data_list' : data_list[1:] }
77     return render(request, 'myApp/sub6.html', context)
```




(4) sub6.html 파일 생성

myApp/templates/myApp/sub6.html/

```
<div class="container text-center">
  <h1>Sub6</h1>
  <hr />
  <h2>CSV 데이터 출력</h2>
  <!-- context에 정의된 키로 호출 -->
  {% for row in data_list %}
  <p>{{row}}</p>
  {% endfor %}
```

```
<hr />
<h3>테이블</h3>
<table class="table" style="width: 80%; margin: 50px auto">
  <thead class="thead-dark">
    <th>반</th>
    <th>이름</th>
    <th>국어</th>
    <th>영어</th>
    <th>수학</th>
    <th>과학</th>
  </thead>
  <tbody>
    {% for c, name, kor, eng, mat, bio in data_list %}
    <tr>
      <td>{{c}}</td>
      <td>{{name}}</td>
      <td>{{kor}}</td>
      <td>{{eng}}</td>
      <td>{{mat}}</td>
      <td>{{bio}}</td>
    </tr>
    {% endfor %}
  </tbody>
</table>
```



5) 서버 실행 후 결과 확인

터미널창에서 서버 실행하기

(venv) c:\Wdjango\project>python manage.py runserver

웹브라우저 주소 : <http://127.0.0.1:8000/myApp/sub6/>

CSV 데이터 출력

['1', 'adam', '67', '87', '90', '98']

['1', 'andrew', '45', '45', '56', '98']

['1', 'ben', '95', '59', '96', '88']

['1', 'clark', '65', '94', '89', '98']

['1', 'dan', '45', '65', '78', '98']

['1', 'noel', '78', '76', '98', '89']

['2', 'paul', '87', '67', '65', '56']

['2', 'walter', '89', '98', '78', '78']

['2', 'oscar', '100', '78', '56', '65']

['2', 'martin', '99', '89', '87', '87']

['2', 'hugh', '98', '45', '56', '54']

['2', 'henry', '65', '89', '87', '78']

반	이름	국어	영어	수학	과학
1	adam	67	87	90	98
1	andrew	45	45	56	98
1	ben	95	59	96	88
1	clark	65	94	89	98
1	dan	45	65	78	98
1	noel	78	76	98	89
2	paul	87	67	65	56
2	walter	89	98	78	78
2	oscar	100	78	56	65
2	martin	99	89	87	87
2	hugh	98	45	56	54
2	henry	65	89	87	78

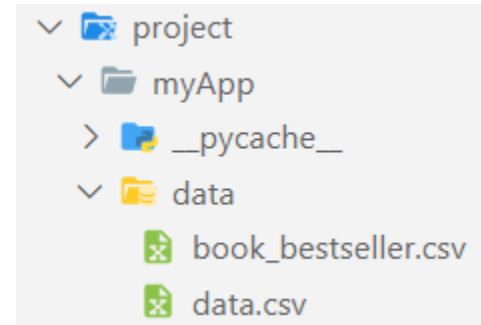
```
import pandas as pd
import numpy as np
df = pd.read_csv('csv파일경로', encoding='cp949')
arr = df.to_numpy()
data_list = arr.tolist()
```

작업과정

- 1) csv 파일 준비
- 2) url 등록
- 3) 뷰함수 등록 : csv => 판다스 데이터프레임 => 넘파이 배열 => list 구조
- 4) html 파일 생성
- 5) 결과 확인 <http://127.0.0.1:8000/myApp/sub7//>

(1) csv 파일 준비

```
# myApp 하단에 data 폴더 생성  
# data 폴더안에 csv 파일 저장  
# data/book_bestseller.csv
```



(2) url 등록

```
# myApp/urls.py
```

```
# /myApp/sub7/  
path('sub7/', views.sub7),
```

(3) 뷰함수 등록

myApp/views.py

```
81 import pandas as pd
82 def sub7(request):
83     df = pd.read_csv('myApp/data/book_bestseller.csv')
84     arr = df.to_numpy()
85     data_list = arr.tolist()
86     context = { 'data_list' : data_list }
87     return render(request, 'myApp/sub7.html', context)
```

(4) sub7.html 파일 생성

myApp/templates/myApp/sub7.html/

```
37 <h2>CSV 데이터 출력</h2>
38 {% for row in data_list %}
39 <p>{{row}}</p>
40 {% endfor %}
41 <hr />
42 <h2>베스트셀러</h2>
43 <table class="table" style="width:98%; margin: 50px auto">
44     <thead class="thead-dark">
45         <th>순위</th>
46         <th>제목</th>
47         <th>저자</th>
48         <th>표지</th>
49         <th>더보기</th>
50     </thead>
```



```
51 <tbody>
52     {% for rank,title,writer,poster,url in data_list %}
53     <tr>
54         <td>{{rank}}</td>
55         <td>{{title}}</td>
56         <td>{{writer}}</td>
57         <td></td>
58         <td>
59             <a href="{{url}}" target="_blank" class="btn btn-dark"
60             >바로가기</a>
61         </td>
62     </tr>
63     {% endfor %}
64 </tbody>
65 </table>
```



5) 서버 실행 후 결과 확인

터미널창에서 서버 실행하기

(venv) c:\Wdjango\project>python manage.py runserver

웹브라우저 주소 : http://127.0.0.1:8000/myApp/sub7/

CSV 데이터 출력

['1위', '달러구트 꿈 백화점 2', '이미에 저', 'http://image.yes24.com/Goods/102789938/XL', 'http://www.yes24.com/Product/Goods/102789938']

['2위', '달러구트 꿈 백화점', '이미에 저', 'http://image.yes24.com/Goods/91065309/XL', 'http://www.yes24.com/Product/Goods/91065309']

['3위', '오늘 밤, 세계에서 이 사랑이 사라진다 해도', '이치조 미사키 저/권영주 역', 'http://image.yes24.com/Goods/102360203/XL',
'http://www.yes24.com/Product/Goods/102360203']

['4위', '작별하지 않는다', '한강 저', 'http://image.yes24.com/Goods/103495056/XL', 'http://www.yes24.com/Product/Goods/103495056']

['5위', '양순이네 떡집', '김리리 글/김이랑 그림', 'http://image.yes24.com/Goods/103303100/XL', 'http://www.yes24.com/Product/Goods/103303100']





['6위', '대한민국 재건축 재개발 지도', '정지영(아임해피) 저', 'http://image.yes24.com/Goods/103401164/XL',
'http://www.yes24.com/Product/Goods/103401164']

['7위', '흔한남매 과학 탐험대 2 지구와 달', '흔한남매 원저/이재국,이현진 글/김덕영 그림/정현철,김희목,권경아,최진수 기획/정현철 감수',
'http://image.yes24.com/Goods/103492634/XL', 'http://www.yes24.com/Product/Goods/103492634']

['8위', '햇빛은 찬란하고 인생은 귀하니까요', '장명숙 저', 'http://image.yes24.com/Goods/103025960/XL', 'http://www.yes24.com/Product/Goods/103025960']

['9위', '피톨로지 피트니스 영양학', '이호욱,정자람 공저', 'http://image.yes24.com/Goods/103555776/XL', 'http://www.yes24.com/Product/Goods/103555776']

['10위', '미드나잇 라이브러리', '매트 헤이그 저/노진선 역', 'http://image.yes24.com/Goods/99534783/XL', 'http://www.yes24.com/Product/Goods/99534783']

순 위	제목	저자	표지	더보기
1위	달러구트 꿈 백화점 2	이미예 저		바로가 기
2위	달러구트 꿈 백화점	이미예 저		바로가 기
3위	오늘 밤, 세계에서 이 사랑이 사라 진다 해도	이치조 미사키 저/권영주 역		바로가 기
4위	작별하지 않는다	한강 저		바로가 기