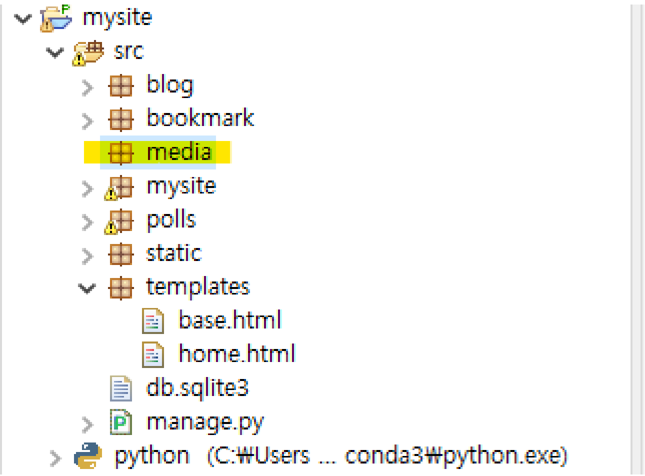
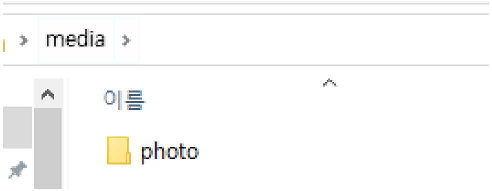
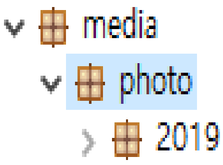
- mysite에 media 폴더 만들기(mysite-settings의 media)



```
mysite
  src
    blog
    bookmark
    media
    mysite
    polls
    static
    templates
      base.html
      home.html
    db.sqlite3
    manage.py
  python  (C:₩Users ... conda3₩python.exe)
```

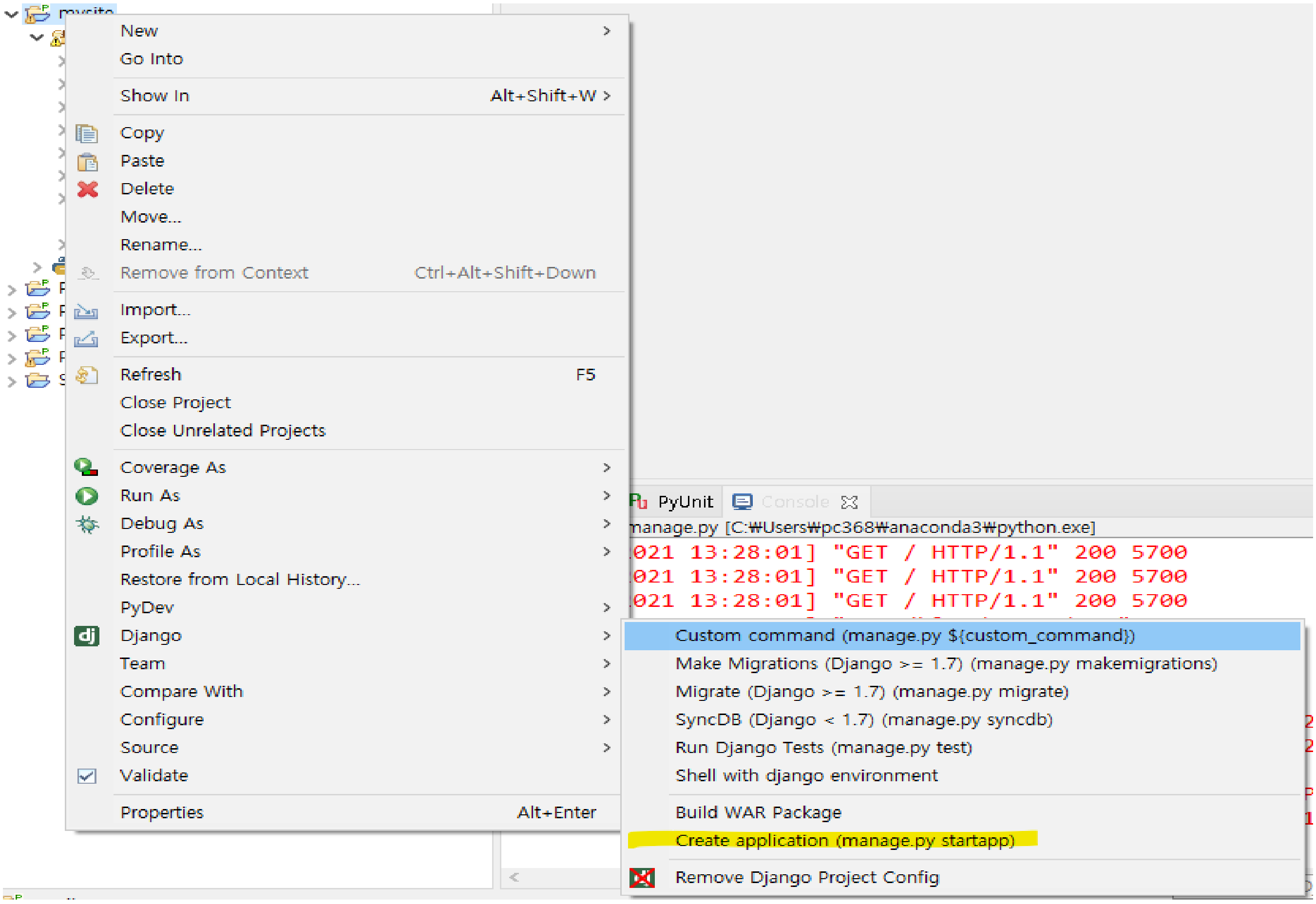
- 다음 카페 media 다운로드해서 압축 풀고 media 안에 있는 photo 폴더 → 이클립스에 붙여 넣기(사진 자료들)





```
media
  photo
    2019
```

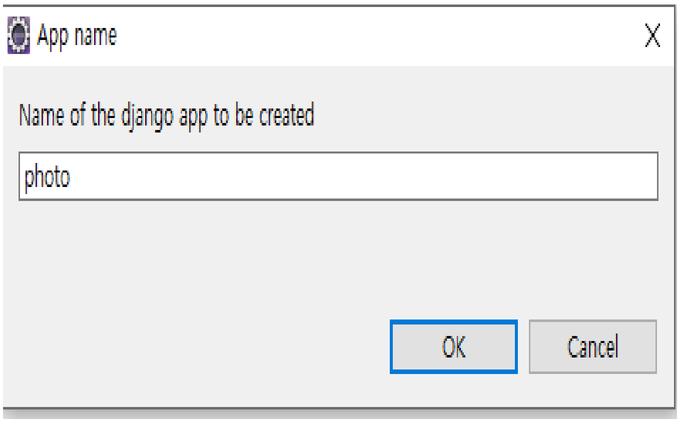
① photo 앱 만들기(Create application ~)



mysite
New
Go Into
Show In Alt+Shift+W
Copy
Paste
Delete
Move...
Rename...
Remove from Context Ctrl+Alt+Shift+Down
Import...
Export...
Refresh F5
Close Project
Close Unrelated Projects
Coverage As
Run As
Debug As
Profile As
Restore from Local History...
PyDev
Django
Team
Compare With
Configure
Source
Validate
Properties Alt+Enter
PyUnit
Console
Custom command (manage.py ${custom_command})
Make Migrations (Django >= 1.7) (manage.py makemigrations)
Migrate (Django >= 1.7) (manage.py migrate)
SyncDB (Django < 1.7) (manage.py syncdb)
Run Django Tests (manage.py test)
Shell with django environment
Build WAR Package
Create application (manage.py startapp)
Remove Django Project Config

```
manage.py [C:₩Users₩pc368₩anaconda3₩python.exe]
021 13:28:01] "GET / HTTP/1.1" 200 5700
021 13:28:01] "GET / HTTP/1.1" 200 5700
021 13:28:01] "GET / HTTP/1.1" 200 5700
```

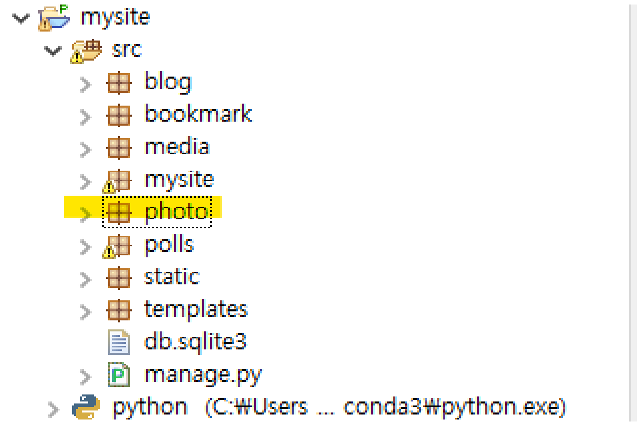
- App name 'photo'로 주고 새로고침하면 photo 생김



App name

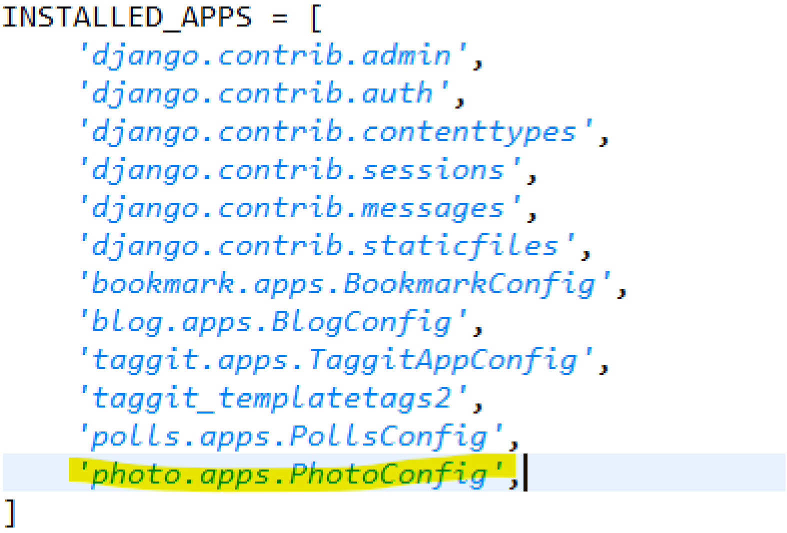
Name of the django app to be created

photo



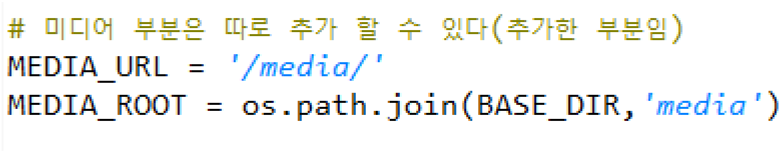
```
mysite
  src
    blog
    bookmark
    media
    mysite
    photo
    polls
    static
    templates
    db.sqlite3
    manage.py
  python  (C:₩Users ... conda3₩python.exe)
```

② mysite – settings.py 추가등록하기(p.196)



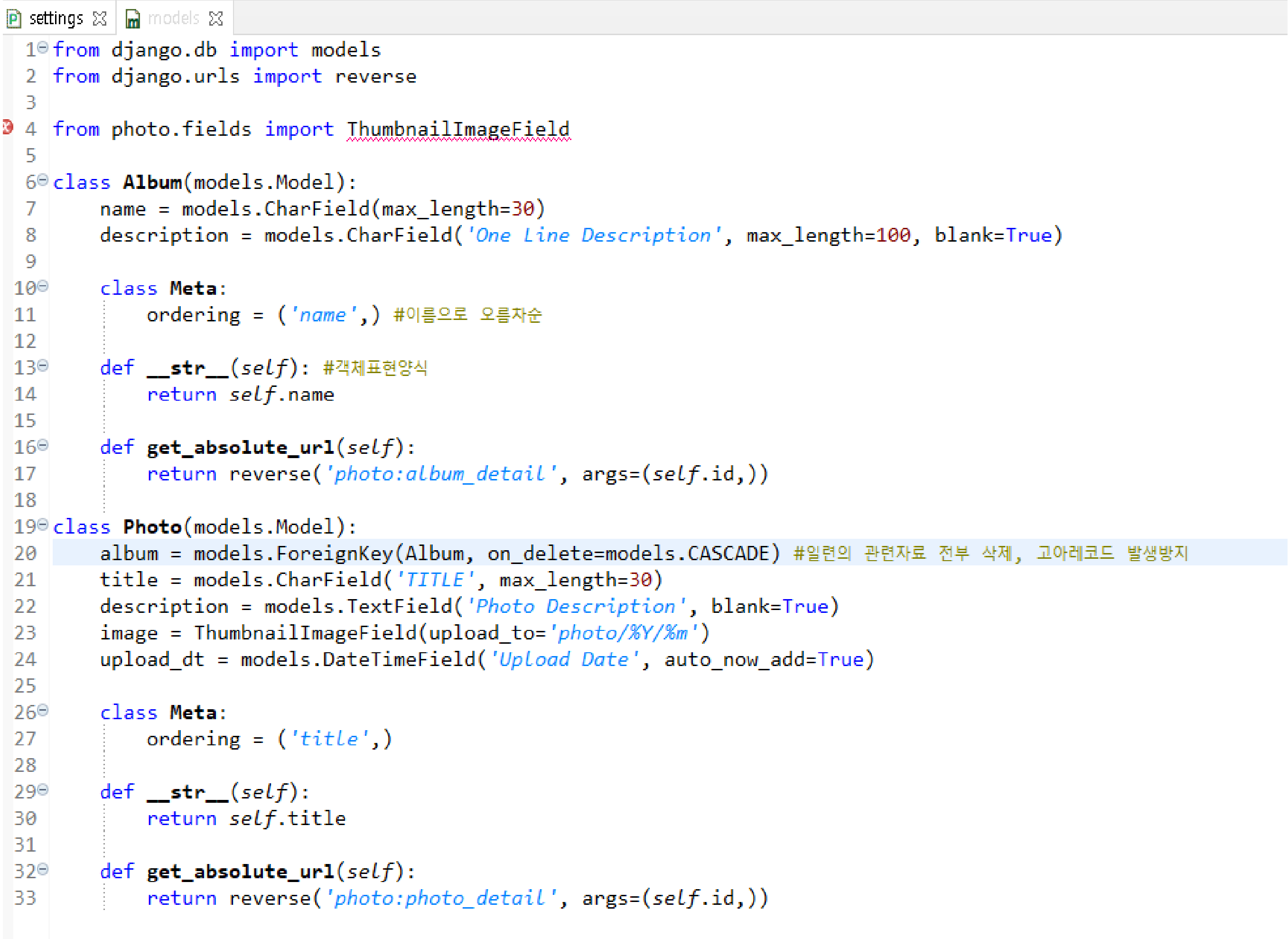
```
INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',
    'bookmark.apps.BookmarkConfig',
    'blog.apps.BlogConfig',
    'taggit.apps.TaggitAppConfig',
    'taggit_templatetags2',
    'polls.apps.PollsConfig',
    'photo.apps.PhotoConfig',
]
```

- 미디어 관련부분 있는 지 확인



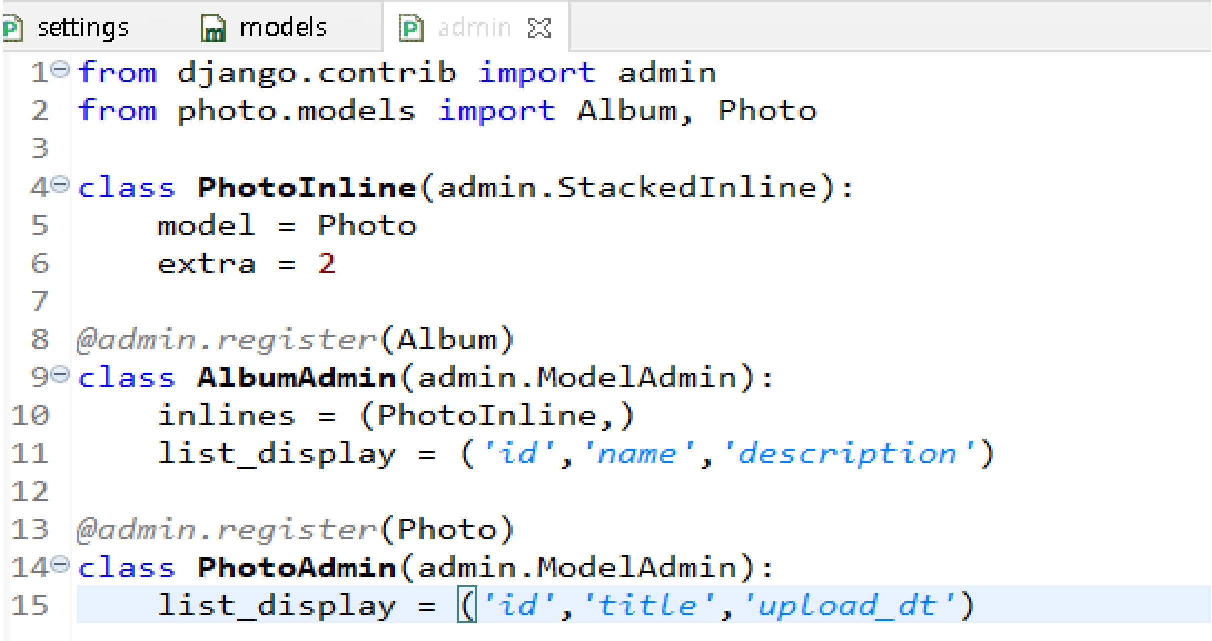
```
# 미디어 부분은 따로 추가 할 수 있다(추가한 부분임)
MEDIA_URL = '/media/'
MEDIA_ROOT = os.path.join(BASE_DIR,'media')
```

③ photo – models.py 코딩(p.197~198)



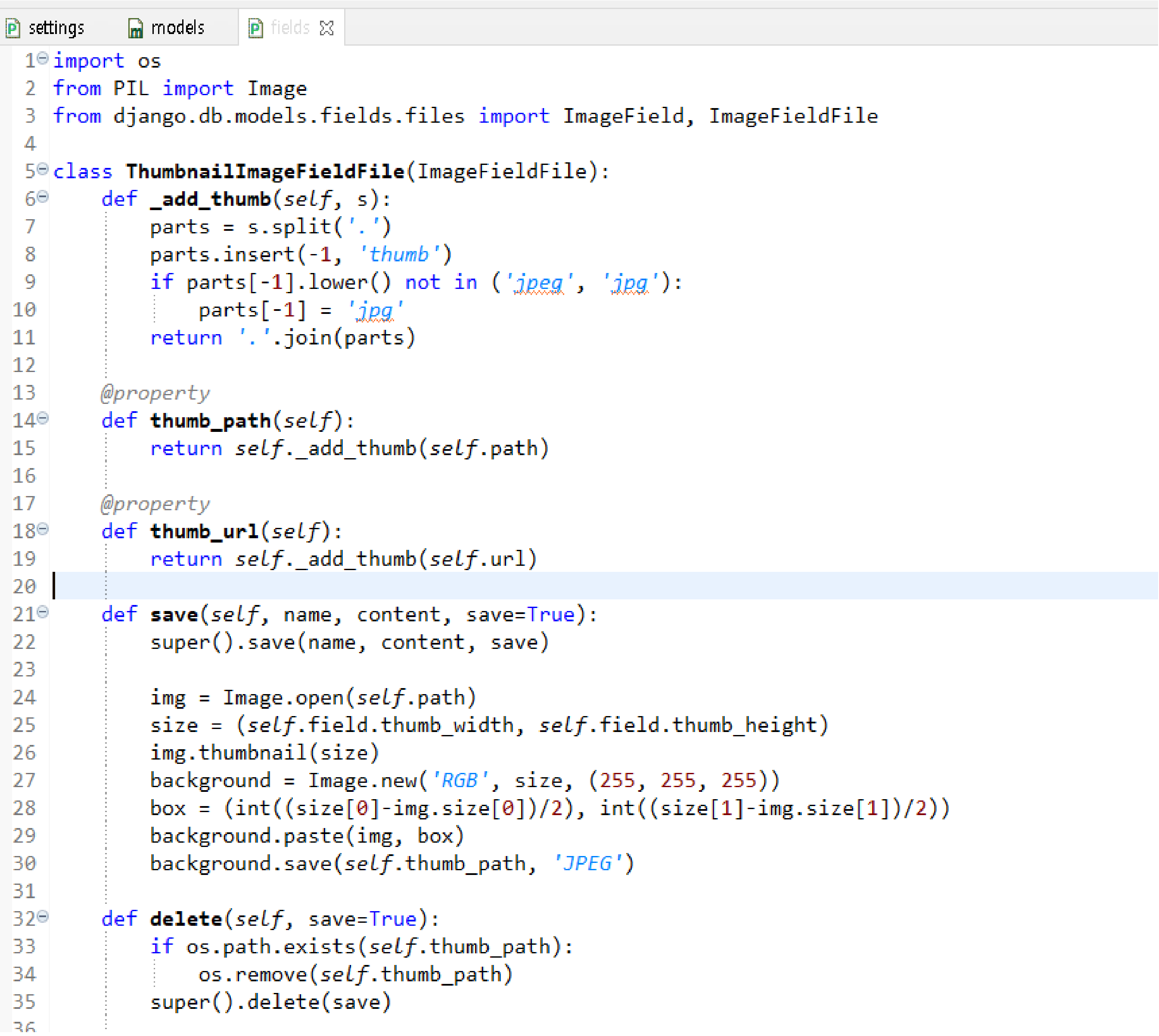
```python
from django.db import models
from django.urls import reverse

from photo.fields import ThumbnailImageField

class Album(models.Model):
    name = models.CharField(max_length=30)
    description = models.CharField('One Line Description', max_length=100, blank=True)

    class Meta:
        ordering = ('name',) #이름으로 오름차순

    def __str__(self): #객체표현양식
        return self.name

    def get_absolute_url(self):
        return reverse('photo:album_detail', args=(self.id,))

class Photo(models.Model):
    album = models.ForeignKey(Album, on_delete=models.CASCADE) #일련의 관련자료 전부 삭제, 고아레코드 발생방지
    title = models.CharField('TITLE', max_length=30)
    description = models.TextField('Photo Description', blank=True)
    image = ThumbnailImageField(upload_to='photo/%Y/%m')
    upload_dt = models.DateTimeField('Upload Date', auto_now_add=True)

    class Meta:
        ordering = ('title',)

    def __str__(self):
        return self.title

    def get_absolute_url(self):
        return reverse('photo:photo_detail', args=(self.id,))
```

④ photo – admin.py 코딩(p.199)

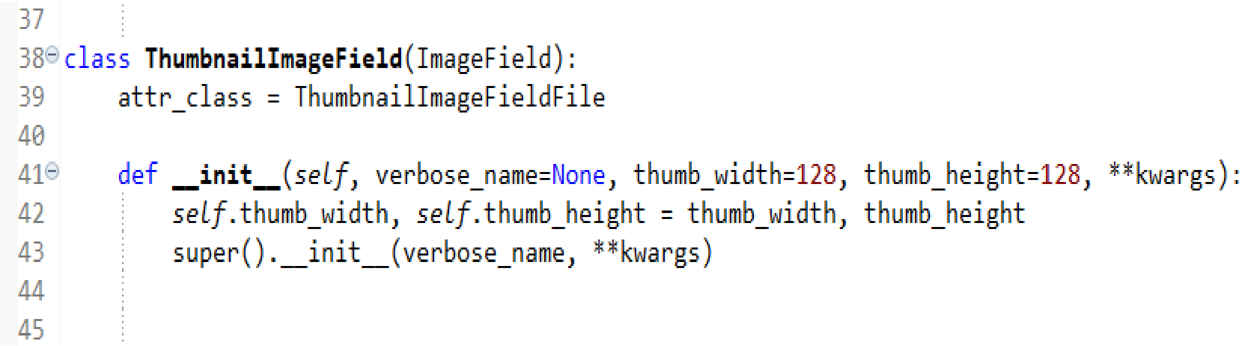


```python
from django.contrib import admin
from photo.models import Album, Photo

class PhotoInline(admin.StackedInline):
    model = Photo
    extra = 2

@admin.register(Album)
class AlbumAdmin(admin.ModelAdmin):
    inlines = (PhotoInline,)
    list_display = ('id','name','description')

@admin.register(Photo)
class PhotoAdmin(admin.ModelAdmin):
    list_display = ('id','title','upload_dt')
```

⑤ photo – fields.py 만들고 코딩



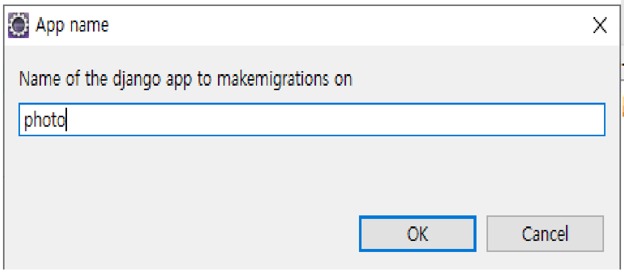
```python
import os
from PIL import Image
from django.db.models.fields.files import ImageField, ImageFieldFile

class ThumbnailImageFieldFile(ImageFieldFile):
    def _add_thumb(self, s):
        parts = s.split('.')
        parts.insert(-1, 'thumb')
        if parts[-1].lower() not in ('jpeg', 'jpg'):
            parts[-1] = 'jpg'
        return '.'.join(parts)

    @property
    def thumb_path(self):
        return self._add_thumb(self.path)

    @property
    def thumb_url(self):
        return self._add_thumb(self.url)

    def save(self, name, content, save=True):
        super().save(name, content, save)

        img = Image.open(self.path)
        size = (self.field.thumb_width, self.field.thumb_height)
        img.thumbnail(size)
        background = Image.new('RGB', size, (255, 255, 255))
        box = (int((size[0]-img.size[0])/2), int((size[1]-img.size[1])/2))
        background.paste(img, box)
        background.save(self.thumb_path, 'JPEG')

    def delete(self, save=True):
        if os.path.exists(self.thumb_path):
            os.remove(self.thumb_path)
        super().delete(save)
```



```python

class ThumbnailImageField(ImageField):
    attr_class = ThumbnailImageFieldFile

    def __init__(self, verbose_name=None, thumb_width=128, thumb_height=128, **kwargs):
        self.thumb_width, self.thumb_height = thumb_width, thumb_height
        super().__init__(verbose_name, **kwargs)

```

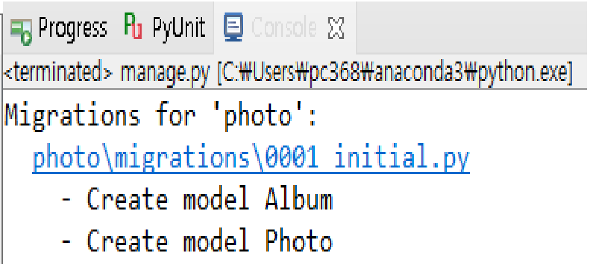
⑥ 데이터베이스 작업 두 가지

(1) Make Migrations ~


App name

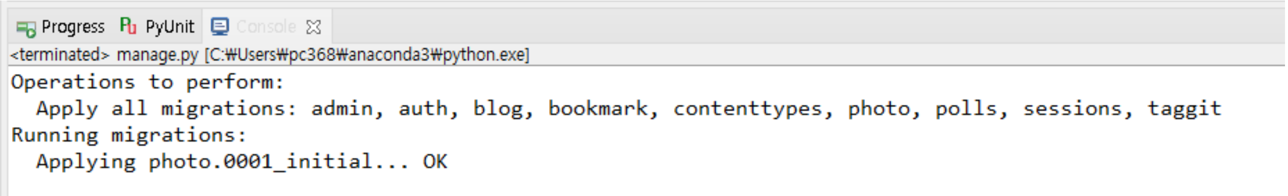
Name of the django app to makemigrations on

photo



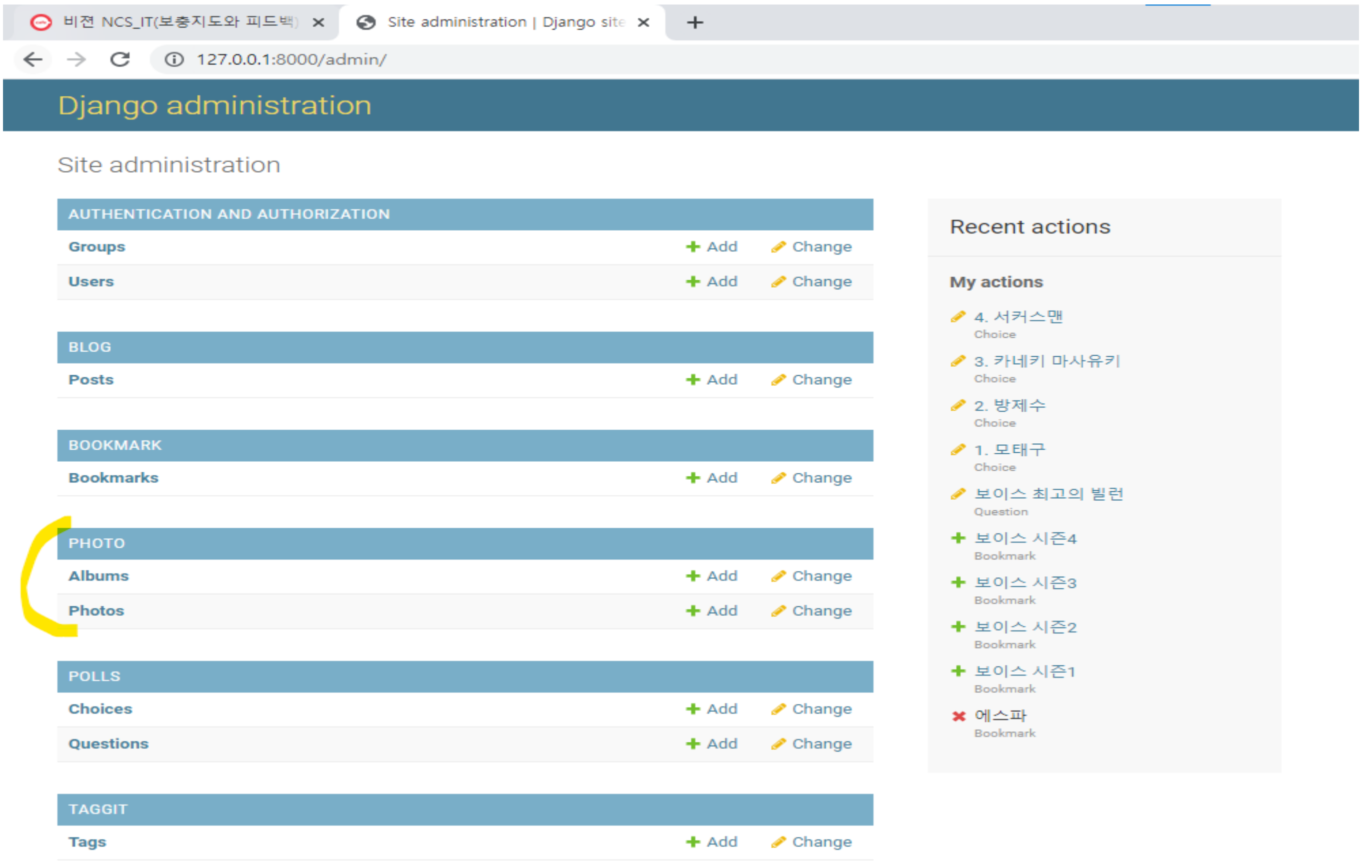
```
<terminated> manage.py [C:₩Users₩pc368₩anaconda3₩python.exe]
Migrations for 'photo':
  photo\migrations\0001_initial.py
    - Create model Album
    - Create model Photo
```

(2) Migrate



```
<terminated> manage.py [C:₩Users₩pc368₩anaconda3₩python.exe]
Operations to perform:
  Apply all migrations: admin, auth, blog, bookmark, contenttypes, photo, polls, sessions, taggit
Running migrations:
  Applying photo.0001_initial... OK
```

▶▶▶ 중간점검 ◀◀◀


비전 NCS_IT(보충지도와 피드백)
Site administration | Django site
127.0.0.1:8000/admin/

Django administration

Site administration

AUTHENTICATION AND AUTHORIZATION		
Groups	+ Add	Change
Users	+ Add	Change

BLOG		
Posts	+ Add	Change

BOOKMARK		
Bookmarks	+ Add	Change

PHOTO		
Albums	+ Add	Change
Photos	+ Add	Change

POLLS		
Choices	+ Add	Change
Questions	+ Add	Change

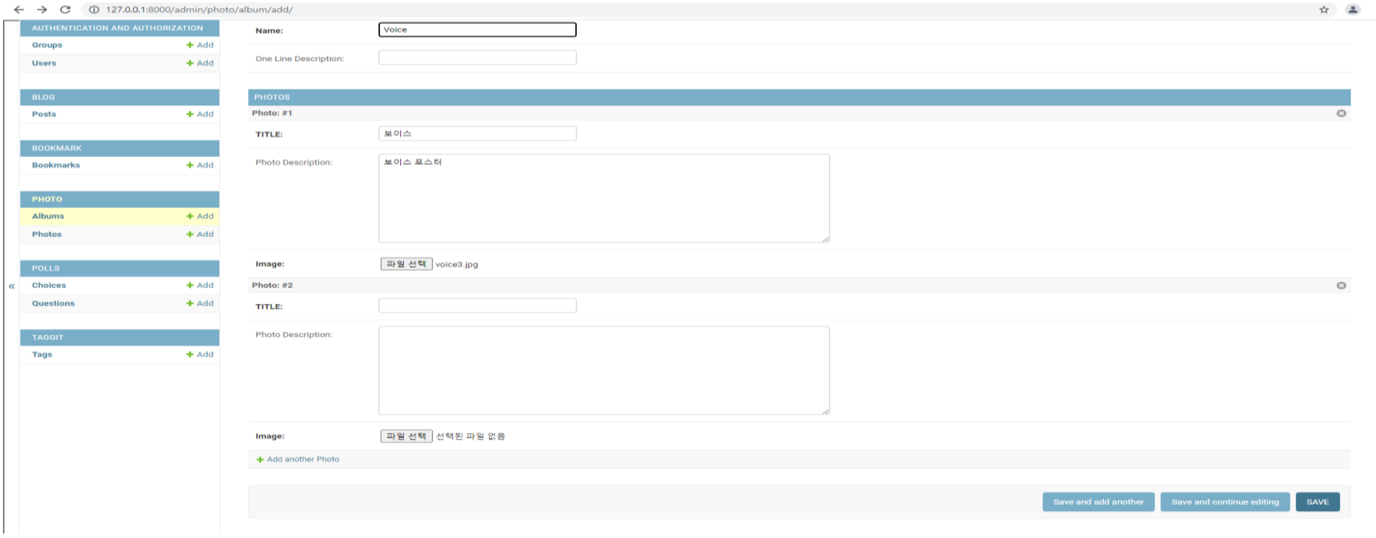
TAGGIT		
Tags	+ Add	Change

Recent actions

My actions

- 4. 서커스맨
Choice
- 3. 카네키 마사유키
Choice
- 2. 방제수
Choice
- 1. 모태구
Choice
- 보이스 최고의 빌런
Question
- + 보이스 시즌4
Bookmark
- + 보이스 시즌3
Bookmark
- + 보이스 시즌2
Bookmark
- + 보이스 시즌1
Bookmark
- ✖ 에스파
Bookmark

- Albums에 추가해보기



Name: Voice

One Line Description:

Photo: #1

TITLE: 보이스

Photo Description: 보이스 포스터

Image: 파일 선택 voice3.jpg

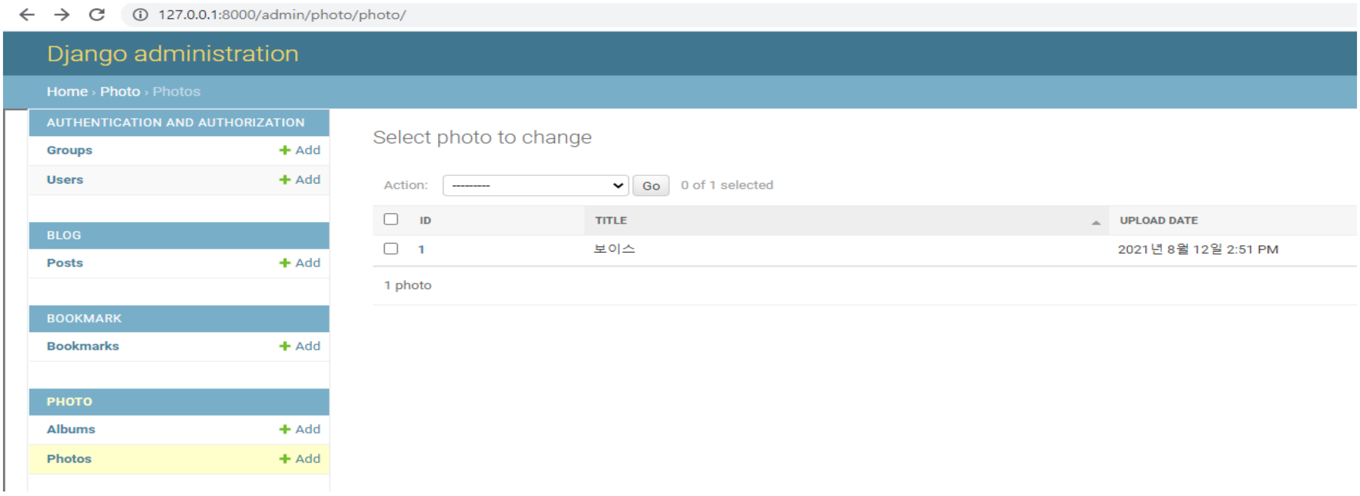
Photo: #2

TITLE:

Photo Description:

Image: 파일 선택 선택된 파일 없음

- Photos 확인



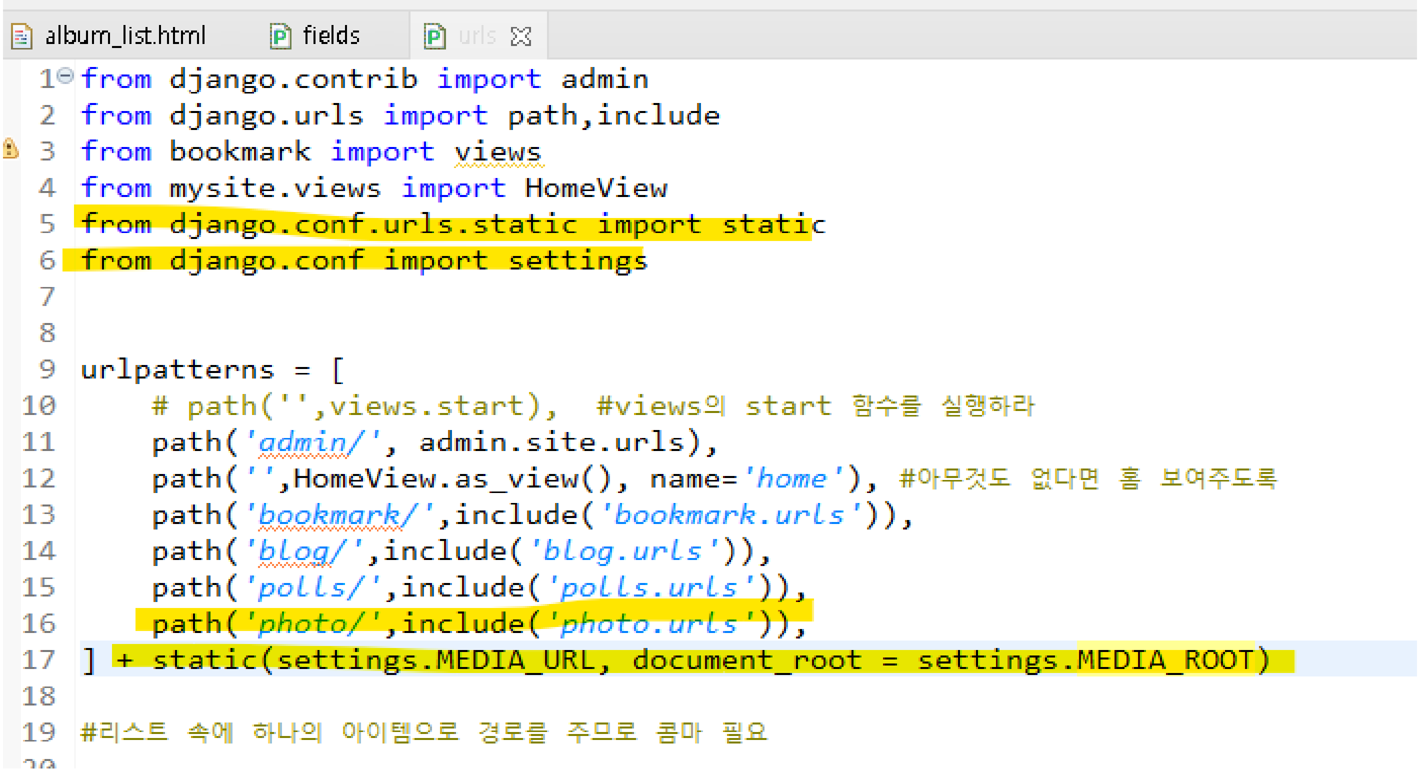
Select photo to change

0 of 1 selected

ID	TITLE	UPLOAD DATE
1	보이스	2021년 8월 12일 2:51 PM

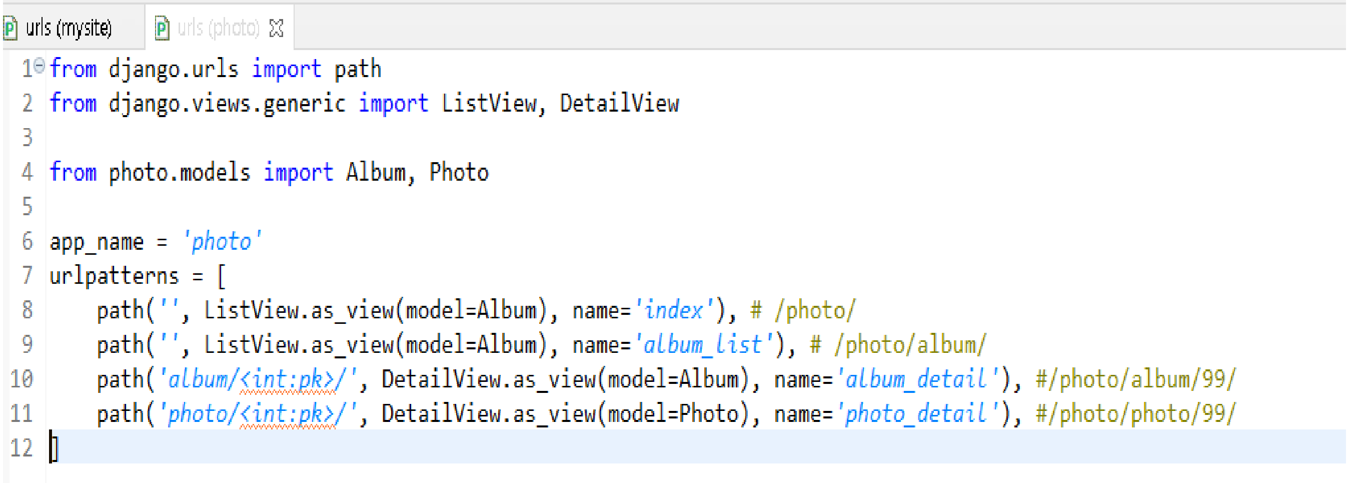
1 photo

⑦ mysite – urls.py 코딩 추가



```
from django.contrib import admin
from django.urls import path,include
from bookmark import views
from mysite.views import HomeView
from django.conf.urls.static import static
from django.conf import settings


urlpatterns = [
    # path('',views.start),  #views의 start 함수를 실행하라
    path('admin/', admin.site.urls),
    path('',HomeView.as_view(), name='home'), #아무것도 없다면 홈 보여주도록
    path('bookmark/',include('bookmark.urls')),
    path('blog/',include('blog.urls')),
    path('polls/',include('polls.urls')),
    path('photo/',include('photo.urls')),
] + static(settings.MEDIA_URL, document_root = settings.MEDIA_ROOT)

#리스트 속에 하나의 아이템으로 경로를 주므로 콤마 필요
```

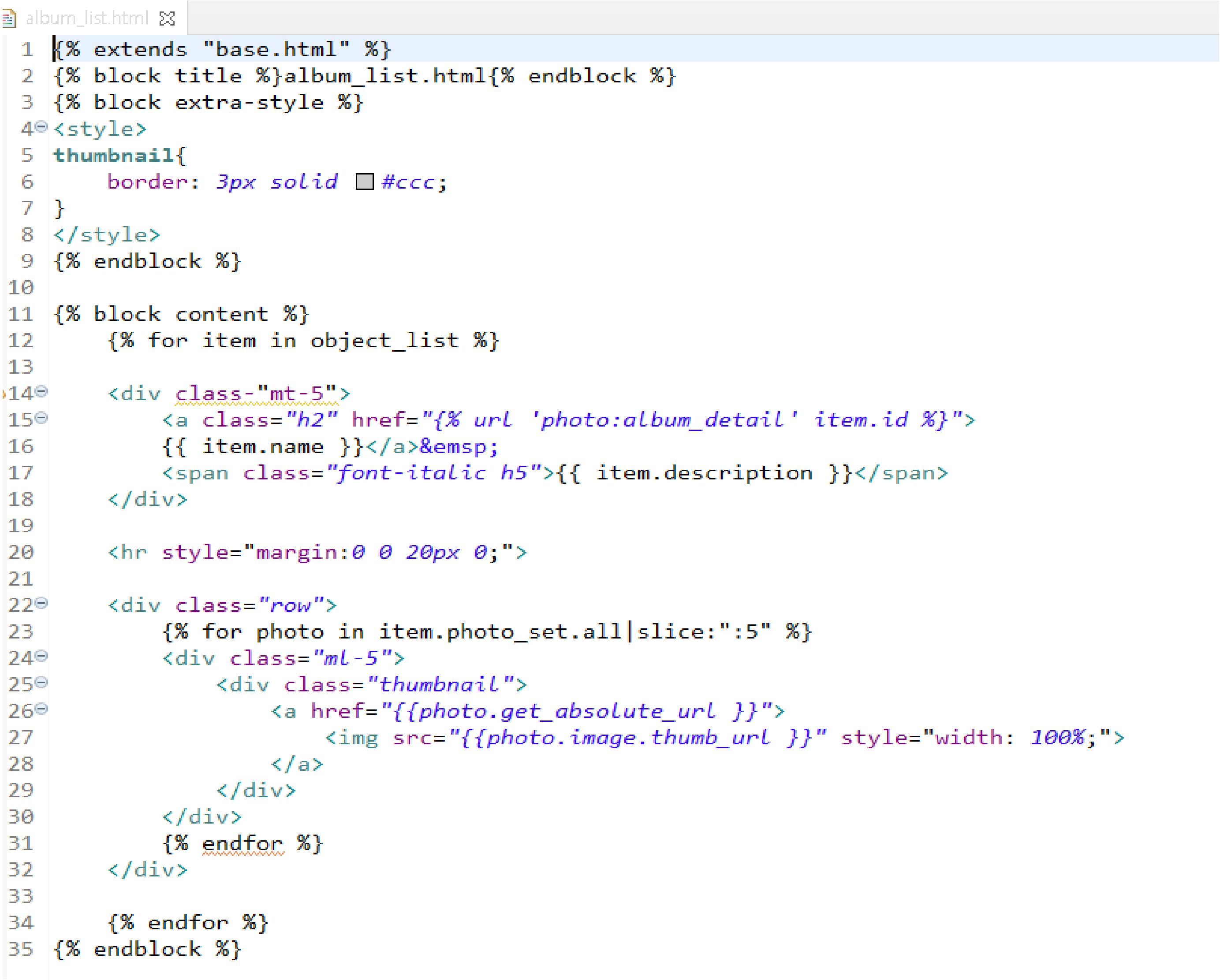
⑧ photo – urls.py 만들고 코딩(뷰 정의가 간단하므로 urls.py에 뷰를 같이 지정하는 방법 사용, p.207~208)



```python
from django.urls import path
from django.views.generic import ListView, DetailView

from photo.models import Album, Photo

app_name = 'photo'
urlpatterns = [
    path('', ListView.as_view(model=Album), name='index'), # /photo/
    path('', ListView.as_view(model=Album), name='album_list'), # /photo/album/
    path('album/<int:pk>/', DetailView.as_view(model=Album), name='album_detail'), #/photo/album/99/
    path('photo/<int:pk>/', DetailView.as_view(model=Photo), name='photo_detail'), #/photo/photo/99/
]
```

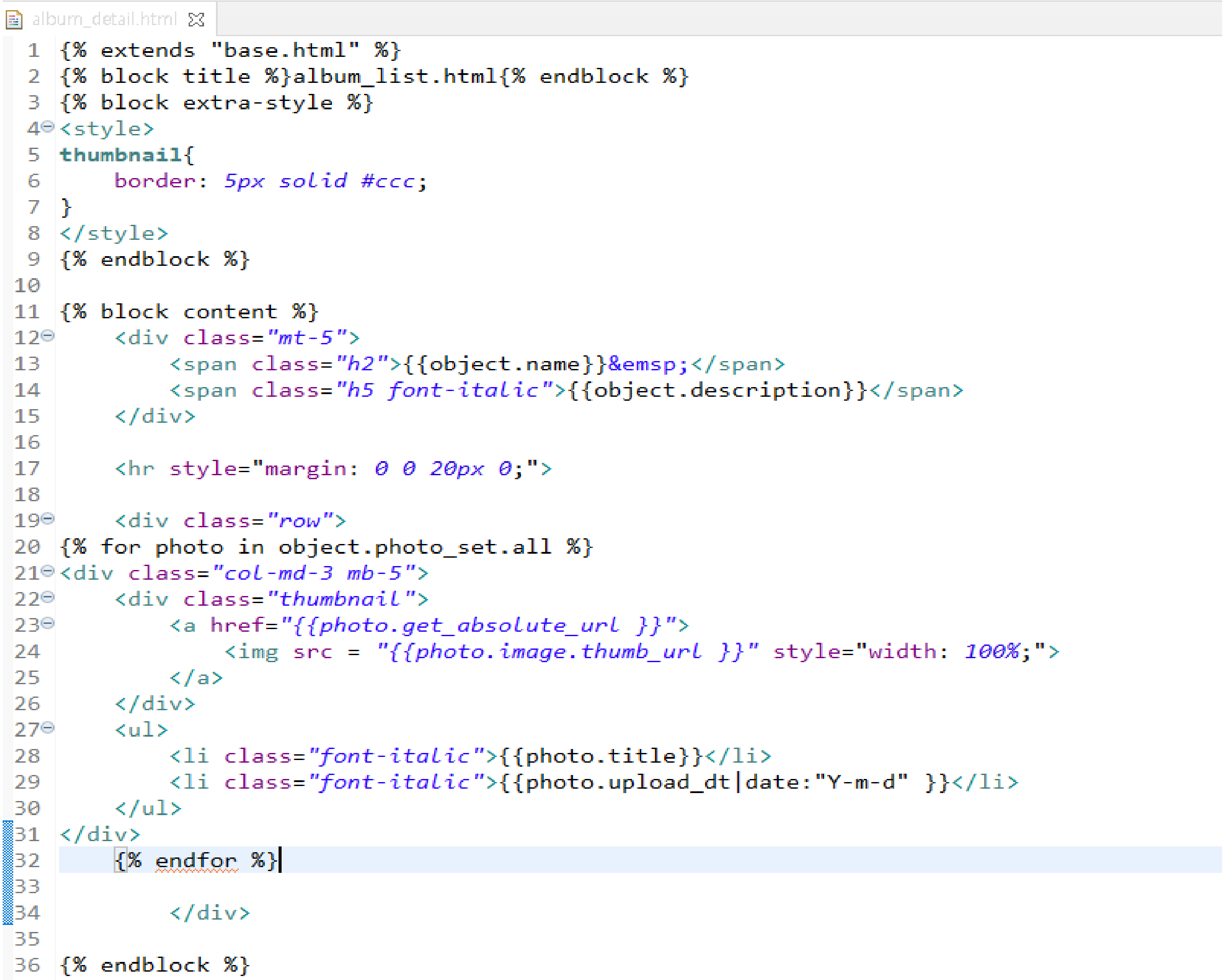
⑨ photo에 templates 만들고 photo만들고 html 작성

-album_list.html



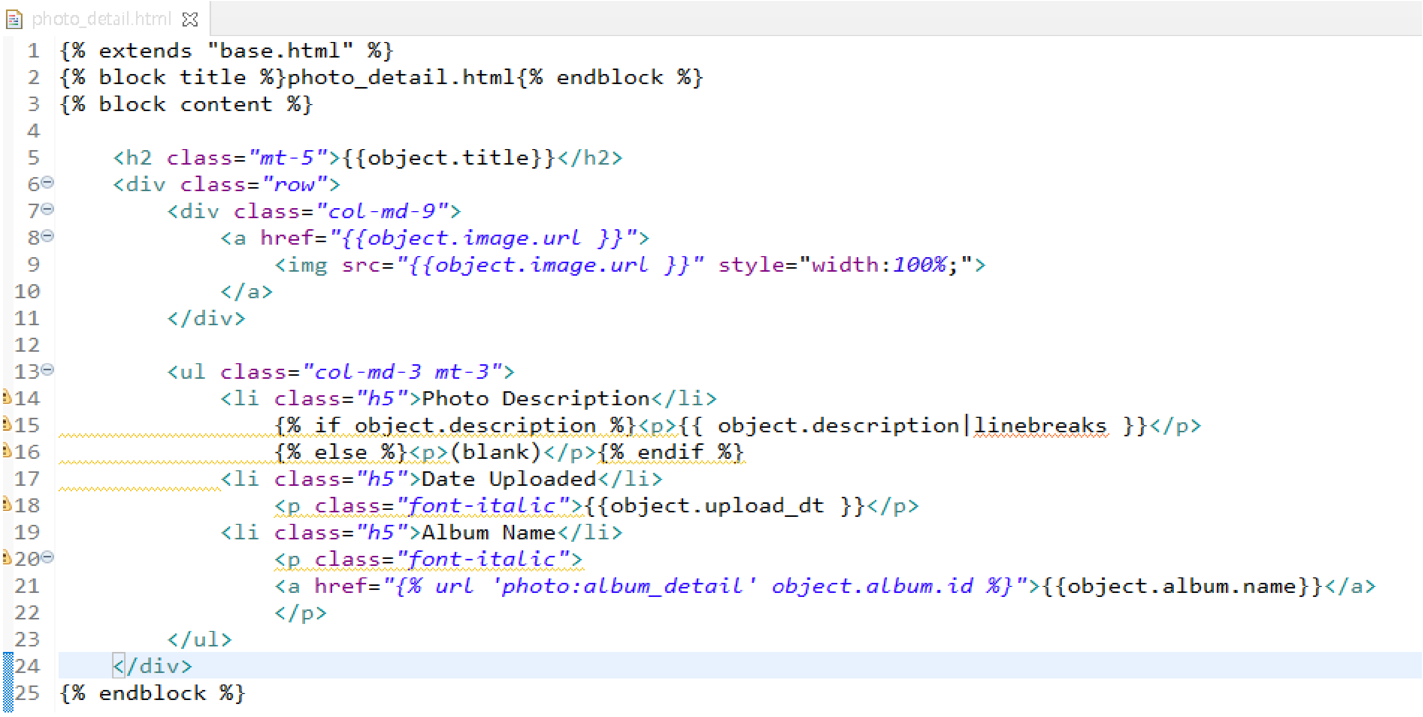
```html
{% extends "base.html" %}
{% block title %}album_list.html{% endblock %}
{% block extra-style %}
<style>
thumbnail{
    border: 3px solid #ccc;
}
</style>
{% endblock %}

{% block content %}
    {% for item in object_list %}

    <div class-"mt-5">
        <a class="h2" href="{% url 'photo:album_detail' item.id %}">
        {{ item.name }}</a>&emsp;
        <span class="font-italic h5">{{ item.description }}</span>
    </div>

    <hr style="margin:0 0 20px 0;">

    <div class="row">
        {% for photo in item.photo_set.all|slice:":5" %}
        <div class="ml-5">
            <div class="thumbnail">
                <a href="{{photo.get_absolute_url }}">
                    <img src="{{photo.image.thumb_url }}" style="width: 100%;">
                </a>
            </div>
        </div>
        {% endfor %}
    </div>

    {% endfor %}
{% endblock %}
```

- album_detail.html



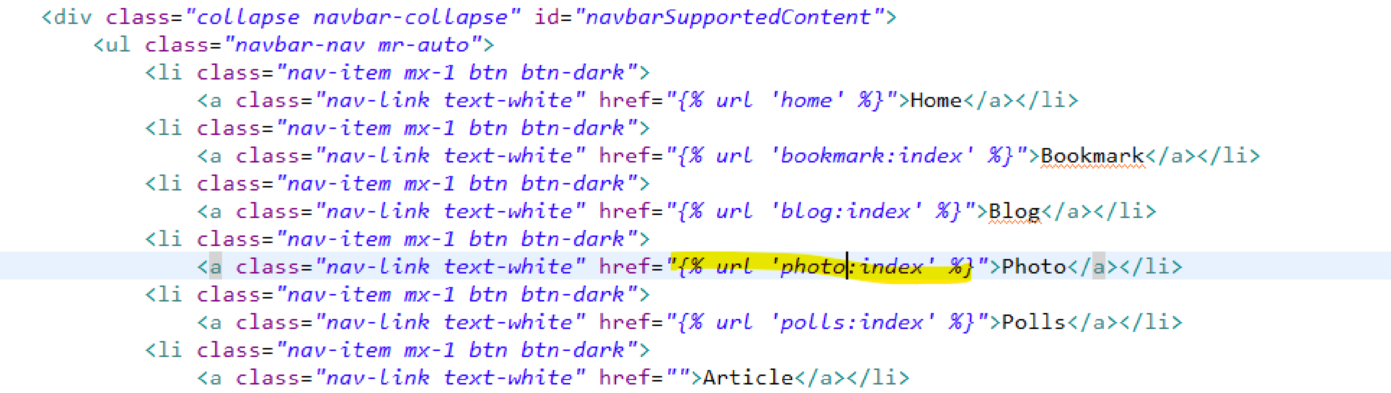
```
{% extends "base.html" %}
{% block title %}album_list.html{% endblock %}
{% block extra-style %}
<style>
thumbnail{
    border: 5px solid #ccc;
}
</style>
{% endblock %}

{% block content %}
    <div class="mt-5">
        <span class="h2">{{object.name}}&emsp;</span>
        <span class="h5 font-italic">{{object.description}}</span>
    </div>

    <hr style="margin: 0 0 20px 0;">

    <div class="row">
{% for photo in object.photo_set.all %}
<div class="col-md-3 mb-5">
    <div class="thumbnail">
        <a href="{{photo.get_absolute_url }}">
            <img src = "{{photo.image.thumb_url }}" style="width: 100%;">
        </a>
    </div>
    <ul>
        <li class="font-italic">{{photo.title}}</li>
        <li class="font-italic">{{photo.upload_dt|date:"Y-m-d" }}</li>
    </ul>
</div>
    {% endfor %}

        </div>

{% endblock %}
```

- photo_detail.html



```
{% extends "base.html" %}
{% block title %}photo_detail.html{% endblock %}
{% block content %}

    <h2 class="mt-5">{{object.title}}</h2>
    <div class="row">
        <div class="col-md-9">
            <a href="{{object.image.url }}">
                <img src="{{object.image.url }}" style="width:100%;">
            </a>
        </div>

        <ul class="col-md-3 mt-3">
            <li class="h5">Photo Description</li>
                {% if object.description %}<p>{{ object.description|linebreaks }}</p>
                {% else %}<p>(blank)</p>{% endif %}
            <li class="h5">Date Uploaded</li>
                <p class="font-italic">{{object.upload_dt }}</p>
            <li class="h5">Album Name</li>
                <p class="font-italic">
                <a href="{% url 'photo:album_detail' object.album.id %}">{{object.album.name}}</a>
                </p>
        </ul>
    </div>
{% endblock %}
```

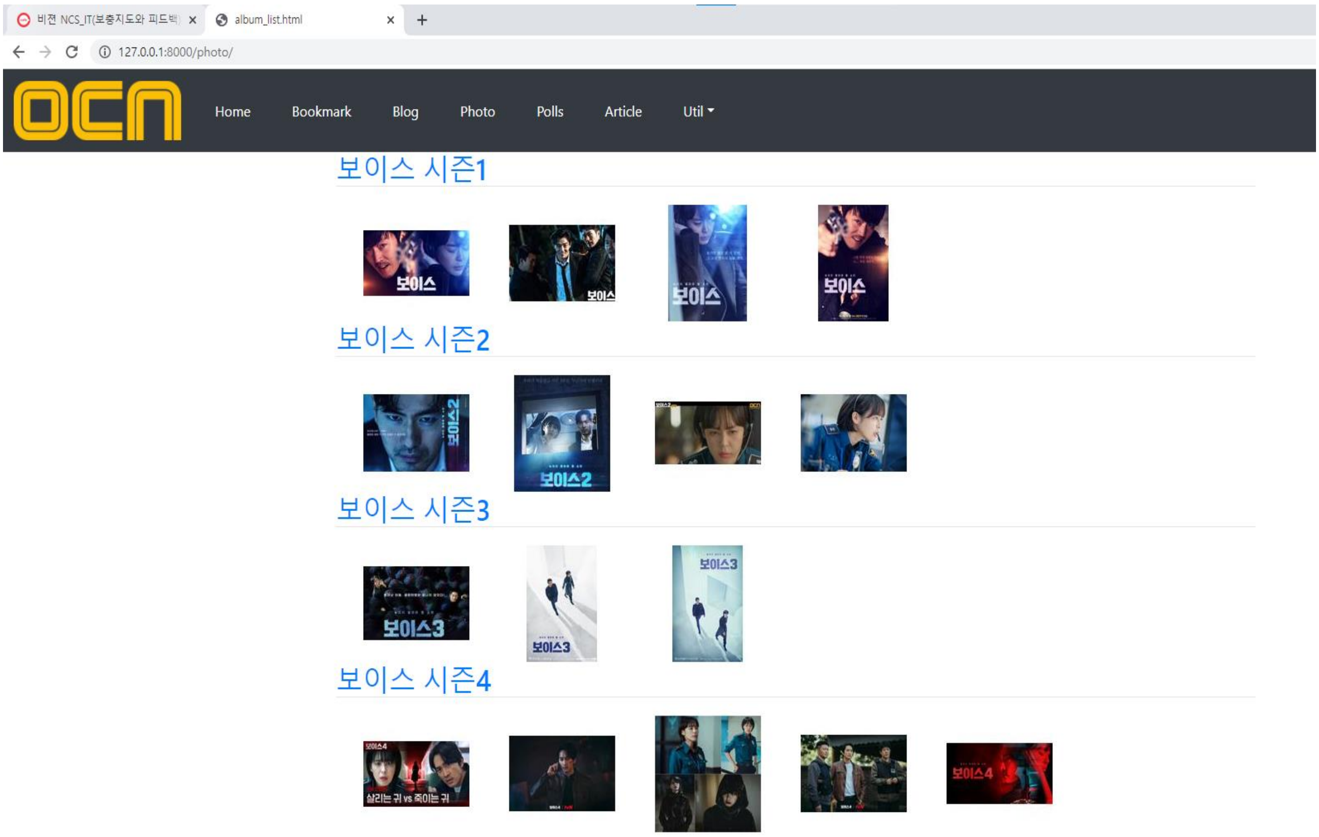
⑩ base.html 수정



```
<div class="collapse navbar-collapse" id="navbarSupportedContent">
    <ul class="navbar-nav mr-auto">
        <li class="nav-item mx-1 btn btn-dark">
            <a class="nav-link text-white" href="{% url 'home' %}">Home</a></li>
        <li class="nav-item mx-1 btn btn-dark">
            <a class="nav-link text-white" href="{% url 'bookmark:index' %}">Bookmark</a></li>
        <li class="nav-item mx-1 btn btn-dark">
            <a class="nav-link text-white" href="{% url 'blog:index' %}">Blog</a></li>
        <li class="nav-item mx-1 btn btn-dark">
            <a class="nav-link text-white" href="{% url 'photo:index' %}">Photo</a></li>
        <li class="nav-item mx-1 btn btn-dark">
            <a class="nav-link text-white" href="{% url 'polls:index' %}">Polls</a></li>
        <li class="nav-item mx-1 btn btn-dark">
            <a class="nav-link text-white" href="">Article</a></li>
```

▶▶▶ 결과확인 ◀◀◀

- /photo/ 일 경우 : 등록한 앨범들과 사진들이 뜬다



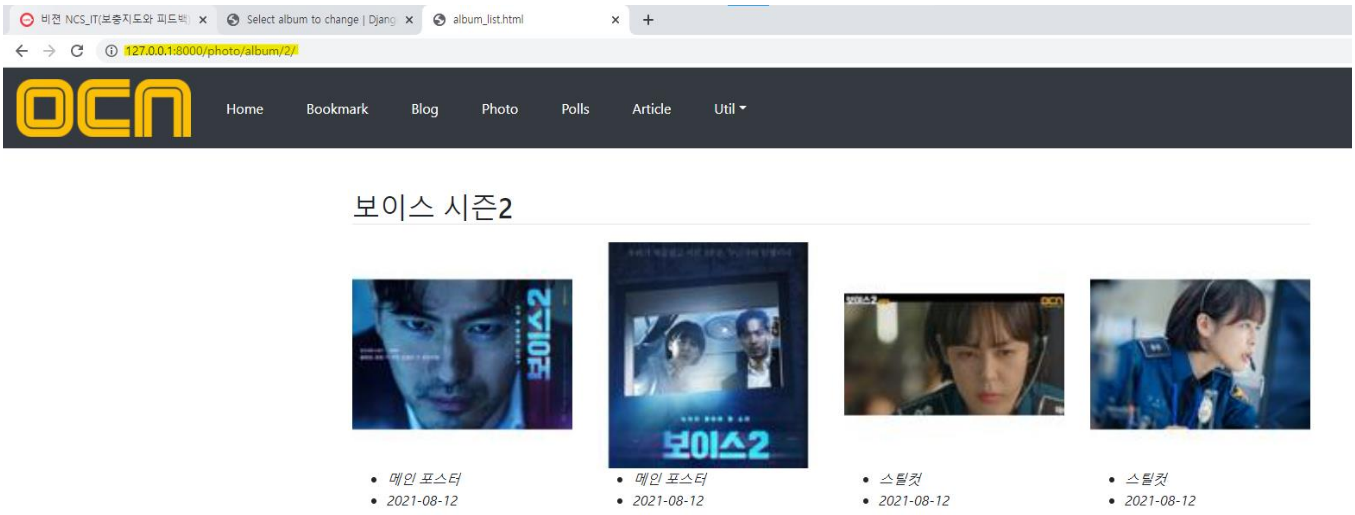
보이스 시즌1

보이스 시즌2

보이스 시즌3

보이스 시즌4

- /photo/album/2/ : 앨범 선택하면 사진과 제목, 등록 날짜가 보인다.


비전 NCS_IT(보충지도와 피드백
Select album to change | Djang
album_list.html
127.0.0.1:8000/photo/album/2/
Home
Bookmark
Blog
Photo
Polls
Article
Util

보이스 시즌2

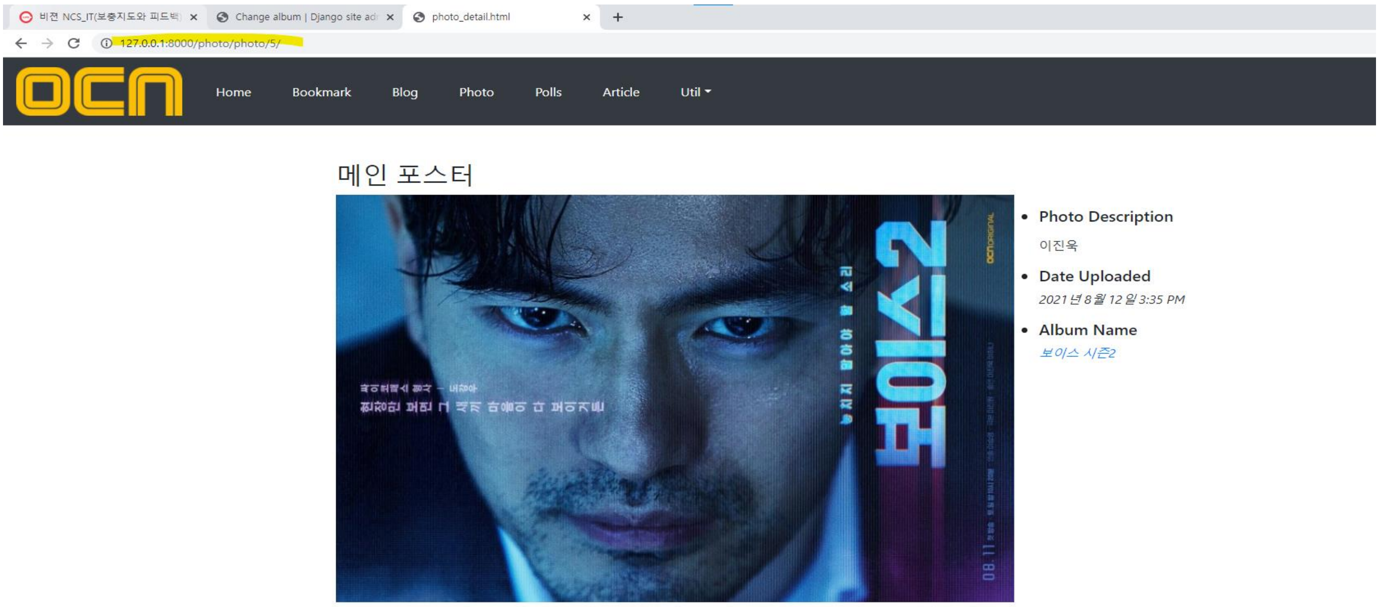
- *메인 포스터*
- *2021-08-12*

- *메인 포스터*
- *2021-08-12*

- *스틸컷*
- *2021-08-12*

- *스틸컷*
- *2021-08-12*

- /photo/photo/5/ : 사진 디테일 정보를 보여준다


비전 NCS_IT(보충지도와 피드백
Change album | Django site adr
photo_detail.html
127.0.0.1:8000/photo/photo/5/
Home
Bookmark
Blog
Photo
Polls
Article
Util

메인 포스터

- **Photo Description**
  이진욱
- **Date Uploaded**
  2021년 8월 12일 3:35 PM
- **Album Name**
  보이스 시즌2