

2023

응용소프트웨어개발 10주차 장고로 미디어 파일 관리하기

JEONBUK NATIONAL UNIVERSITY



전북대학교
JEONBUK NATIONAL UNIVERSITY

포스터에 이미지 올리기

01단계 이미지 폴더 지정하기

장고는 이미지 업로드를 위한 `ImageField`를 제공합니다. `ImageField`를 사용하려면 사용자가 업로드한 이미지를 어디에 저장할지 먼저 설정해야 합니다. 그리고 업로드된 이미지들이 모여 있는 폴더의 URL을 어떻게 할지도 설정해야 하죠. 일단 `do_it_django_prj/settings.py`를 열어서 맨 아래에 두 줄을 추가해 주세요. `os` 모듈을 사용하므로 `import os`도 추가하세요.

실습 파일: `do_it_django_prj/settings.py`

```
import os
from pathlib import Path
(...생략...)
STATIC_URL = '/static/'

MEDIA_URL = '/media/'
MEDIA_ROOT = os.path.join(BASE_DIR, '_media')
```

포스터에 이미지 올리기

이제 blog/models.py를 수정합시다. blog/models.py 파일을 열어 `imageField`로 `head_image` 필드를 추가합니다.

실습 파일: blog/models.py

```
from django.db import models

class Post(models.Model):
    title = models.CharField(max_length=30)
    content = models.TextField()

    head_image = models.ImageField(upload_to='blog/images/%Y/%m/%d/', blank=True)
    created_at = models.DateTimeField(auto_now_add=True)
    updated_at = models.DateTimeField(auto_now=True)
    # author: 추후 작성 예정
    (...생략...)
```

포스터에 이미지 올리기

The screenshot shows the Django administration interface for adding a new post. At the top, the header reads 'Django administration' and 'WELCOME, JAMES. VIEW SITE / CHANGE PASSWORD / LOG OUT'. Below the header, a breadcrumb trail shows 'Home > Blog > Posts > Add post'. The main heading is 'Add post'. A red-bordered box highlights a message: 'Please correct the error below.' Below this, the 'Title' field is shown with a red error message: 'This field is required.' The 'Content' field is a large text area with a message in Korean: 'title은 필수 요소인데 공백으로 입력하면 예러 메시지를 보여줍니다.' (The title is a required element, and if it is entered as a space, an error message will be shown).

그림 9-14 title 필드를 blank=True로 지정하지 않으면 비워 두고 저장할 경우 경고 메시지가 출력됨

포스터에 이미지 올리기

```

λ Cmler

C:\github\do_it_django_a_to_z (main -> origin)
(venv) λ python manage.py makemigrations
SystemCheckError: System check identified some issues:

ERRORS:
blog.Post.head_image: (fields.E210) Cannot use ImageField because Pillow is not
installed.
HINT: Get Pillow at https://pypi.org/project/Pillow/ or run command "python -m pip
install Pillow".
    
```

```

λ Cmler

C:\github\do_it_django_a_to_z (main -> origin)
(venv) λ pip install Pillow
C:\github\do_it_django_a_to_z (main -> origin)
(venv) λ python manage.py makemigrations
C:\github\do_it_django_a_to_z (main -> origin)
(venv) λ python manage.py migrate
    
```

포스터에 이미지 올리기

04단계 미디어 파일을 위한 URL 지정하기

오류가 발생한 이유는 아직 urls.py에서 media URL에 대한 설정을 하지 않았기 때문입니다. 프로젝트의 URL 설정을 해야 하므로 do_it_django_prj/urls.py 파일을 열어 맨 아래에 다음과 같이 한 줄을 추가하세요. static 폴더와 settings.py 파일을 사용해야 하므로 위에 from django.conf.urls.static import static과 from django.conf import settings도 추가해주세요.

실습 파일: do_it_django_prj/urls.py

```
from django.contrib import admin
from django.urls import path, include

from django.conf import settings
from django.conf.urls.static import static

urlpatterns = [
    path('blog/', include('blog.urls')),
    path('admin/', admin.site.urls),
    path('', include('single_pages.urls')),
]

urlpatterns += static(settings.MEDIA_URL, document_root=settings.MEDIA_ROOT)
```

포스터에 이미지 올리기

실습 파일: blog/templates/blog/post_list.html

```
(...생략...)
{% for p in post_list %}
<!-- Blog Post -->
<div class="card mb-4">
    
(...생략...)
```

실습 파일: [blog/templates/blog/post_list.html](#)

Server time: Sun, 2 Feb 2020 23:55:43 +0900

이 경우 대표 이미지가 있는 경우에만 이미지를 출력하도록 수정하면 되지만, 아직은 템플릿 파일에서 **if** 문을 사용하는 방법을 배우지 않았습니다. 지금은 다음 2가지 방법을 사용할 수 있습니다. 모든 **Post** 레코드에 이미지를 업로드하거나 아니면 이미지를 업로드하지 않아 대표 이미지가 비어 있는 포스트를 지워주면 됩니다. 여기서는 모든 포스트에 대표 이미지를 넣어주겠습니다. 관리자 페이지에서 지금까지 작성한 모든 포스트로 들어가 Head image에 이미지를 채워주세요. 그리고 웹 브라우저에서 새로고침합니다.

포스터에 이미지 올리기

실습 파일: blog/templates/blog/post_detail.html

(...생략...)

<!-- Preview Image -->

(...생략...)

포스터에 다른 파일 올리기

01단계 file_upload 필드 만들기

blog/models.py에 FileField로 file_upload 필드를 다음과 같이 추가합니다. 사용법은 ImageField와 거의 동일합니다. upload_to에는 ImageField에 사용한 경로에서 images 폴더를 files 폴더로만 바꿔줍니다.

실습 파일: blog/models.py

```
from django.db import models

class Post(models.Model):
    title = models.CharField(max_length=30)
    content = models.TextField()

    head_image = models.ImageField(upload_to='blog/images/%Y/%m/%d/', blank=True)
    file_upload = models.FileField(upload_to='blog/files/%Y/%m/%d/', blank=True)

    created_at = models.DateTimeField(auto_now_add=True)
    updated_at = models.DateTimeField(auto_now=True)
    # author: 추후 작성 예정
    (...생략...)
```

포스터에 다른 파일 올리기

새로운 필드를 만들었으니 마이그레이션을 해서 데이터베이스에 변화된 내용을 적용합니다.



```
C:\github\do_it_django_a_to_z (main -> origin)
(venv) λ python manage.py makemigrations
C:\github\do_it_django_a_to_z (main -> origin)
(venv) λ python manage.py migrate
```

포스터에 다른 파일 올리기

02단계 파일 업로드 테스트하기

이제 서버를 실행하고 관리자 페이지에서 포스트를 열어보면 `file_upload` 필드가 반영된 File upload 입력란이 추가되어 있습니다. txt 파일을 만들어 적당한 내용을 입력한 후 업로드합니다. 파일이 제대로 올라오면 저장하세요.



그림 9-22 txt 파일 업로드 후 저장

저장을 하고 나면 `_media` 폴더 아래에 `blog/files/년/월/일/` 경로가 생기고, 그 아래에 파일이 저장됩니다. 일단은 Post 레코드 중 하나만 `file_upload` 필드를 채우겠습니다.

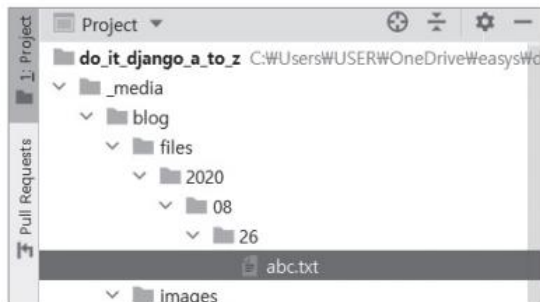


그림 9-23 폴더 자동 생성 후 파일 저장

포스터에 이미지 없을 때

Change post
HISTORY
VIEW ON SITE >

Title:

Content:

첫번째 포스트입니다. 무슨 말을 써야할지 모르겠네요. 책 내용 일부를 붙여 넣겠습니다.

우리가 models.py에 CharField로 지정했던 title, TextField로 지정했던 content, DateTimeField로 지정했던 create 필드들이 각각의 양식에 맞게 입력란이 만들어졌습니다. 블로그에 글 쓰듯이 여러분이 원하는 내용을 채워넣고, created 필드는 Today, Now를 클릭하면 됩니다. 그리고 SAVE 버튼을 누르세요.

국민경제자문회의의 조직·직무범위 기타 필요한 사항은 법률로 정한다. 교육의 자주성·전문성·정치적 중립성 및 대학의 자율성은 법률이 정하는 바에 의하여 보장된다. 공무원인 근로자는 법률이 정하는 자에 한하여 단결권·단체교섭권 및 단체행동권을 가진다. 국회는 국민의 보통·평등·직접·비밀선거에 의하여 선출된 국회의원으로 구성한다.

1 내용 추가

Head image:

Currently: blog/images/2020/05/25/alizee-baudez-QElq3ldWzlQ-unsplash.jpg
☐ Clear

2 대표 이미지 삭제

Change:
선택된 파일 없음

File upload:
선택된 파일 없음

Delete

Save and add another

Save and continue editing

3

그림 10-1 본문으로 긴 글을 추가하고 대표 이미지 삭제

포스터에 이미지 없을 때



로렘 입숨이란?

테스트를 위해 매번 장문의 글을 작성하기는 어렵고 귀찮은 일입니다. 이럴 때 로렘 입숨 ^{Lorem Ipsum} 이라는 간단한 해결책이 있습니다. 로렘 입숨은 얼핏 보면 그럴싸하지만 막상 읽으려고 하면 아무런 의미도 없는 글을 무작위로 생성해주는 도구입니다. 그림 10-1에서 content 필드에 추가한 긴 글 역시 로렘 입숨 한글 버전을 이용해 작성한 것입니다.



그림 10-3 로렘 입숨 영문판(www.lipsum.com)과 한글판(hangul.thefron.me)

포스터에 이미지 없을 때

01단계 이미지가 있을 경우 보여주기

post_list.html에서 <!--Blog Post--> 부분을 찾으세요. 그리고 다음과 같이 {% if p.head_image %}와 {% endif %}로 태그를 감싸고 alt="Card image cap"를 alt="{{ p }} head image"로 수정합니다. HTML의 태그에서 alt는 이미지를 보여줄 수 없을 때 이미지 대신 나타나게 하는 텍스트를 의미합니다. 물론 제대로 웹 사이트가 동작하고 있다면 이 텍스트가 보일 일은 없겠지만 그렇지 않은 경우에도 대비해 두어야 하니까요.

실습 파일: blog/templates/blog/post_list.html

```
(...생략...)
{% for p in post_list %}
<!-- Blog Post -->
<div class="card mb-4">
    {% if p.head_image %}
        
    {% endif %}
    <div class="card-body">
    (...생략...)
```

포스터에 이미지 없을 때

02단계 이미지가 없을 경우 사용할 임의의 이미지 가져오기

이미지가 없을 때는 아무 그림도 보이지 않게 했더니 너무 행해 보이네요. 이런 경우 임의의 이미지를 보여주면 좋겠죠? 그런데 괜찮은 이미지를 어디서 구할 수 있을까요? 이런 경우에는 Lorem Picsum이라는 웹 사이트가 유용합니다(picsum.photos/). 앞서 더미 텍스트를 만들 때 사용했던 Lorem Ipsum과 이름이 비슷하죠?

이 웹 사이트는 재밌는 기능을 제공합니다. URL로 'https://picsum.photos/가로 픽셀수/세로 픽셀수'를 입력하면 임의로 지정한 크기의 이미지를 보내줍니다. 가로 800px, 세로 200px 이미지를 요청하기 위해 https://picsum.photos/800/200을 주소 창에 입력해 보겠습니다. 여러분도 재미삼아 웹 브라우저에서 여러 값을 입력해 보세요. 반복해서 입력할 때마다 무작위로 다른 이미지를 보여줍니다.


실습 파일: blog/templates/blog/post_list.html

```
(...생략...)
{% for p in post_list %}
<!-- Blog Post -->
<div class="card mb-4">
    {% if p.head_image %}
        
    {% else %}
        
    {% endif %}
    <div class="card-body">
        (...생략...)
```


포스터에 이미지 없을 때

03단계 임의로 나타나는 이미지 고정하기

Lorem Picsum은 이런 문제에 대처하기 위해 seed를 주는 방법도 제공하고 있습니다. 예를 들어 'https://picsum.photos/seed/id 값/200/300'을 입력하면 해당 id 값을 가지는 위치에 매번 동일한 이미지를 나타냅니다. Post 레코드마다 고유한 pk 값을 가지므로 id 값에 해당 포스트의 id를 넣으면 Post 레코드의 pk 값이 들어가면서 원하는 결과를 얻을 수 있습니다. 다음과 같이 수정하세요.

 id 값은 pk 값과 같은 것으로 생각하면 됩니다. 즉, p.id 대신 p.pk로 써도 결과는 동일합니다.

실습 파일: blog/templates/blog/post_list.html

```
(...생략...)
{% for p in post_list %}
<!-- Blog Post -->
<div class="card mb-4">
    {% if p.head_image %}
        
    {% else %}
        
    {% endif %}
    <div class="card-body">
(...생략...)
```

포스터에 이미지 없을 때 - 상세페이지

실습 파일: blog/templates/blog/post_detail.html


```
(...생략...)
<!-- Preview Image -->
{% if post.head_image %}
    
{% else %}
    
{% endif %}

<hr>
(...생략...)
```

첨부파일 다운로드

01단계 첨부 파일 다운로드 버튼 만들기

if 문을 사용해 첨부 파일이 있는 경우에는 버튼이 보이도록 post_detail.html을 수정합니다. 버튼은 부트스트랩 공식 웹 사이트(getbootstrap.com)의 [Documentation > Components > Buttons]로 들어가 마음에 드는 버튼을 선택합니다. 그리고 소스를 복사해서 본문이 출력되는 위치 아래에 배치시킵니다. 마지막으로 <a> 태그 안에 Download를 꼭 명시하세요. 이렇게 하지 않으면 파일을 내려 받지 않고 웹 브라우저에서 바로 열려버릴 수도 있거든요.

 부트스트랩 웹 사이트의 버튼 중 disabled 혹은 area-disabled="true"가 명시되어 있는 버튼은 클릭이 안 되므로 선택하지 마세요.

실습 파일: blog/templates/blog/post_detail.html

```
(...생략...)
<!-- Post Content -->
<p>{{ post.content }}</p>

{% if post.file_upload %}
    <a href="{{ post.file_upload.url }}" class="btn btn-outline-dark" role="button"
download>Download</a>
{% endif %}

<hr>
(...생략...)
```

부트스트랩에서 원하는 버튼의 소스 코드를 가져오세요

첨부파일 다운로드

02단계 첨부 파일명과 확장자 아이콘 나타내기

지금은 사용자가 어떤 파일을 내려받는지 알 수 없네요. <Download> 버튼에 첨부된 파일 이름과 첨부 파일의 확장자를 알려주는 아이콘을 나타내겠습니다. 파일 경로는 빼고요.

실습 파일: blog/models.py

```
from django.db import models
import os

class Post(models.Model):
    title = models.CharField(max_length=30)
    content = models.TextField()

    def get_file_name(self):
        return os.path.basename(self.file_upload.name)

    def get_file_ext(self):
        return self.get_file_name().split('.')[-1]
```

첨부파일 다운로드

실습 파일: blog/templates/blog/post_detail.html

(...생략...)

<head>

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

<meta name="description" content="">

<meta name="author" content="">

<title>{{ post.title }} - Blog</title>

<!-- Bootstrap core CSS -->

<link rel="stylesheet" href="{% static 'blog/bootstrap/bootstrap.min.css' %}" media="screen">

<!-- Custom styles for this template -->

<link rel="stylesheet" href="{% static 'blog/css/blog-post.css' %}" media="screen">

<script src="https://kit.fontawesome.com/*****.js" crossorigin="anonymous"></script>

</head>

(...생략...)

본인의 kit code를 입력하세요

첨부파일 다운로드

실습 파일: blog/templates/blog/post_detail.html

```
(...생략...)
{% if post.file_upload %}
  <a href="{{ post.file_upload.url }}" class="btn btn-outline-dark" role="button">
    Download:

    {% if post.get_file_ext == 'csv' %}
      <i class="fas fa-file-csv"></i>
    {% elif post.get_file_ext == 'xlsx' or post.get_file_ext == 'xls' %}
      <i class="fas fa-file-excel"></i>
    {% elif post.get_file_ext == 'docx' or post.get_file_ext == 'doc' %}
      <i class="fas fa-file-word"></i>
    {% else %}
      <i class="far fa-file"></i>
    {% endif %}
    {{ post.get_file_name }}
  </a>
{% endif %}
(...생략...)
```