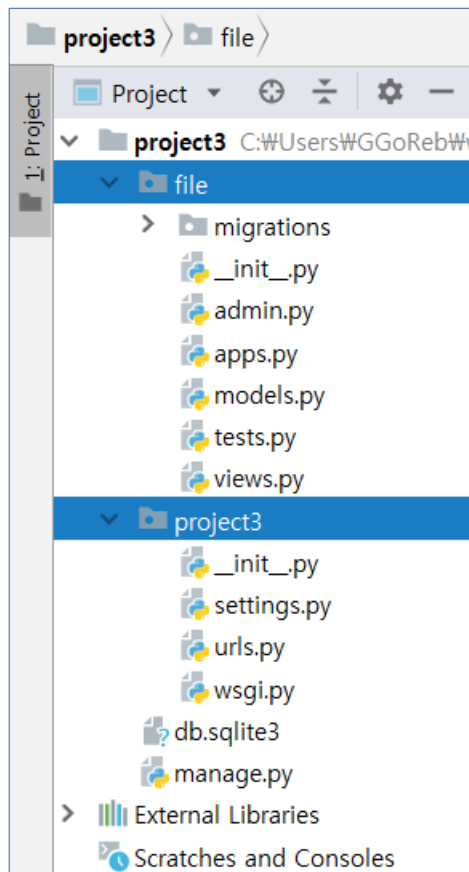


## ■ 파일 업로드 / 다운로드

### ● 프로젝트 / 앱 생성

- `django-admin startproject project3`
- `python manage.py startapp file`



## ■ 파일 업로드

### ● HTML 직접 사용

– file/views.py

```
from django.http import HttpResponse
import os

def upload1(request):
    if request.method == 'GET':
        return render(request, 'file/upload1.html', {})
    else:
        upload_file = request.FILES['my_file']
        try: # 디렉토리 생성
            os.mkdir('upload1')
        except FileExistsError as e:
            pass
        file_name = upload_file.name
        with open('upload1/' + file_name, 'wb') as file:
            for chunk in upload_file.chunks():
                file.write(chunk)
        return HttpResponse('파일 업로드 완료')
```

– file/templates/file/upload1.html

```
<form action="" method="post"
    enctype="multipart/form-data">
    <input type="file" name="my_file">
    <input type="submit" value="완료">
    {% csrf_token %}
</form>
```

## ■ 파일 업로드

### ● HTML 직접 사용

– project3/urls.py

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

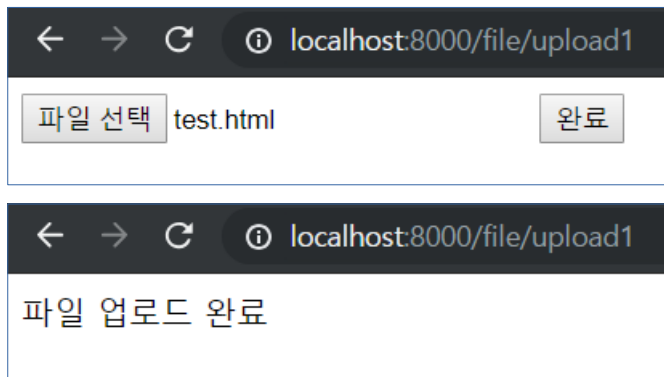
urlpatterns = [
    path('admin/', admin.site.urls),
    path('file/', include('file.urls'))
]
```

– file/urls.py

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

urlpatterns = [
    path('upload1', views.upload1),
]
```

– <http://localhost:8000/file/upload1>



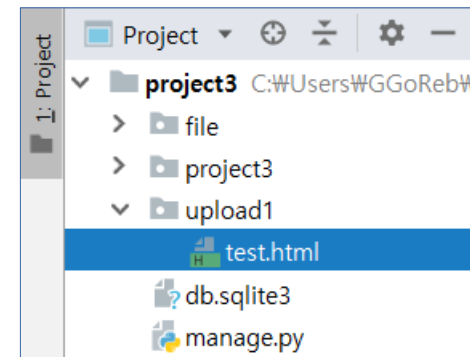
← → ↻ ⓘ localhost:8000/file/upload1

파일 선택 test.html 완료

← → ↻ ⓘ localhost:8000/file/upload1

파일 업로드 완료

– 업로드 경로



## ■ 파일 업로드

### ● forms.py 사용

– file/views.py

```
def upload2(request):
    if request.method == 'GET':
        form = forms.UploadForm
        return render(request, 'file/upload2.html', {'form': form})
    else:
        form = forms.UploadForm(request.POST, request.FILES)
        if form.is_valid():
            try:
                os.mkdir('upload2')
            except FileExistsError as e:
                pass
            upload_file = request.FILES['my_file']
            file_name = upload_file.name
            with open('upload2/' + file_name, 'wb') as file:
                for chunk in upload_file.chunks():
                    file.write(chunk)
            return HttpResponseRedirect('파일 업로드 완료')
```

– file/templates/file/upload2.html

```
<form action="" method="post"
    enctype="multipart/form-data">
    {{ form.as_p }}
    <input type="submit" value="완료">
    {% csrf_token %}
</form>
```

## ■ 파일 업로드

### ● forms.py 사용

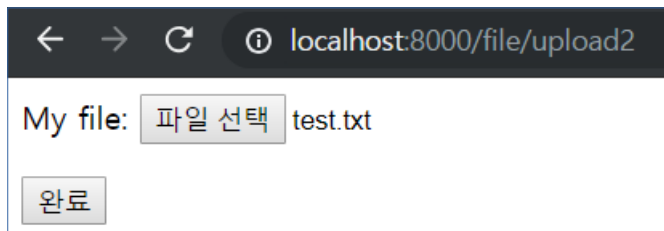
– file/forms.py

```
class UploadForm(forms.Form):  
    my_file = forms.FileField()
```

– file/urls.py

```
from django.urls import path  
import file.views as views  
  
urlpatterns = [  
    path('upload2', views.upload2),  
]
```

– http://localhost:8000/file/upload2



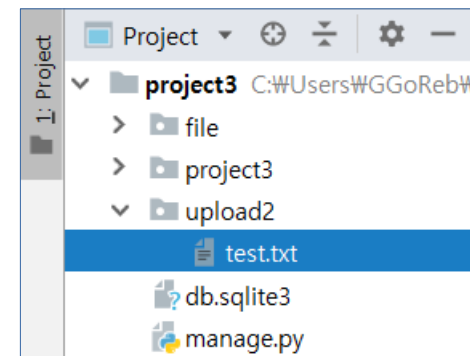
localhost:8000/file/upload2

My file:  test.txt

localhost:8000/file/upload2

파일 업로드 완료

– 업로드 경로



## ■ 파일 업로드

### ● HTML 직접 사용 - 다중 업로드

- file/views.py

```
def upload3(request):
    if request.method == 'GET':
        return render(request, 'file/upload3.html', {})
    else:
        upload_files = request.FILES.getlist('my_file')
        try:
            os.mkdir('upload3')
        except FileExistsError as e:
            pass
        for upload_file in upload_files:
            file_name = upload_file.name
            with open('upload3/' + file_name, 'wb') as file:
                for chunk in upload_file.chunks():
                    file.write(chunk)

        return HttpResponseRedirect('파일 업로드 완료')
```

- file/templates/file/upload3.html

```
<form action="" method="post"
    enctype="multipart/form-data">
    <input type="file" name="my_file"
        multiple>
    <input type="submit" value="완료">
    {% csrf_token %}
</form>
```

## ■ 파일 업로드

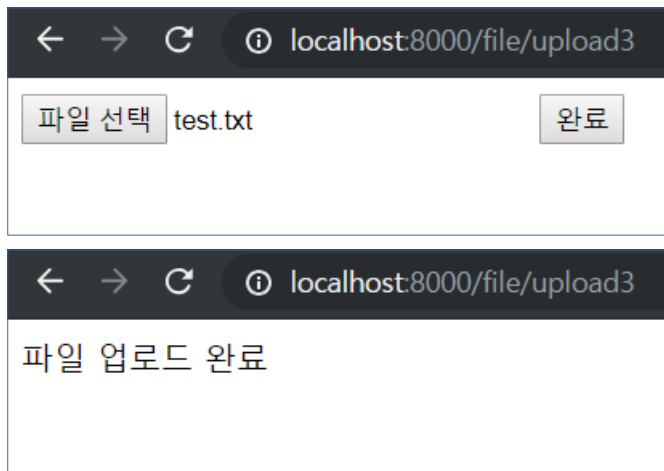
### ● HTML 직접 사용 - 다중 업로드

- file/urls.py

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

urlpatterns = [
    path('upload3', views.upload3),
]
```

- http://localhost:8000/file/upload3



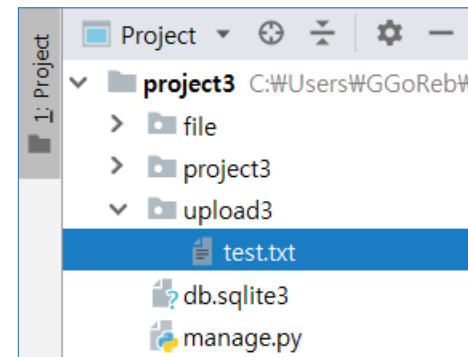
← → ↻ ⓘ localhost:8000/file/upload3

파일 선택 test.txt 완료

← → ↻ ⓘ localhost:8000/file/upload3

파일 업로드 완료

- 업로드 경로



## ■ 파일 업로드

### ● forms.py 사용 - 다중 업로드

- file/views.py

```
def upload4(request):
    if request.method == 'GET':
        form = forms.MultiUploadForm
        return render(request, 'file/upload4.html', {'form': form})
    else:
        upload_files = request.FILES.getlist('my_file')
        try:
            os.mkdir('upload4')
        except FileExistsError as e:
            pass
        for upload_file in upload_files:
            file_name = upload_file.name
            with open('upload4/' + file_name, 'wb') as file:
                for chunk in upload_file.chunks():
                    file.write(chunk)
        return HttpResponse('파일 업로드 완료')
```

- file/templates/file/upload4.html

```
<form action="" method="post"
    enctype="multipart/form-data">
    {{ form.as_p }}
    <input type="submit" value="완료">
    {% csrf_token %}
</form>
```



## ■ 파일 업로드

### ● forms.py 사용 - 다중 업로드

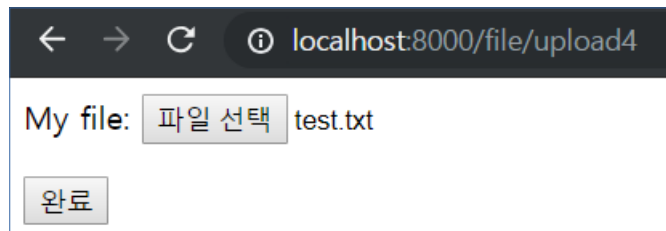
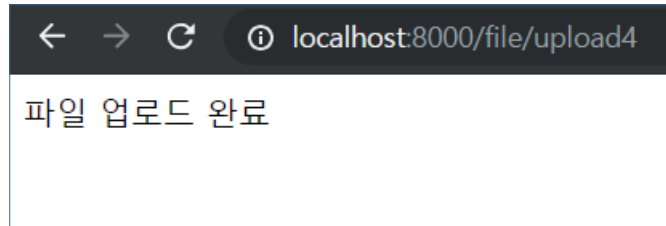
- file/forms.py

```
class MultiUploadForm(forms.Form):  
    my_file = forms.FileField(  
        widget = forms.ClearableFileInput(  
            attrs={'multiple': True}  
        )  
    )
```

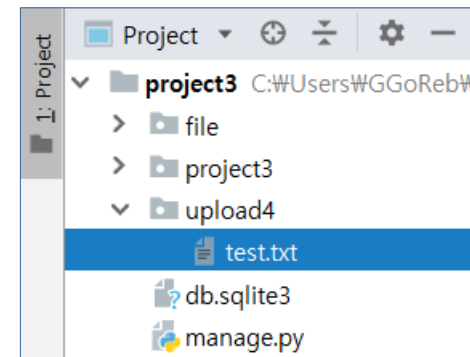
- file/urls.py

```
from django.urls import path  
import file.views as views  
  
urlpatterns = [  
    path('upload4', views.upload4),  
]
```

- http://localhost:8000/file/upload4

- 업로드 경로



## ■ 파일 업로드

### ● forms.py / models.py 사용 – 자동저장(양식지정)

– file/views.py

```
def upload5(request):  
    if request.method == 'GET':  
        form = forms.ModelFormWithFileForm  
        return render(request, 'file/upload5.html', {'form': form})  
    else:  
        form = forms.ModelFormWithFileForm(request.POST, request.FILES)  
  
        if form.is_valid():  
            instance = form.save(commit=False)  
            instance.save()  
  
        return HttpResponseRedirect('파일 업로드 완료')
```

– file/templates/file/upload5.html

```
<form action="" method="post"  
    enctype="multipart/form-data">  
    {{ form.as_p }}  
    <input type="submit" value="완료">  
    {% csrf_token %}  
</form>
```

## ■ 파일 업로드

### ● forms.py / models.py 사용 – 자동저장(양식지정)

#### – file/forms.py

```
class ModelWithFileForm(forms.ModelForm):  
    class Meta:  
        model = ModelWithFile  
        fields = ['my_file']  
        widgets = {  
            'my_file': forms.FileInput()  
        }
```

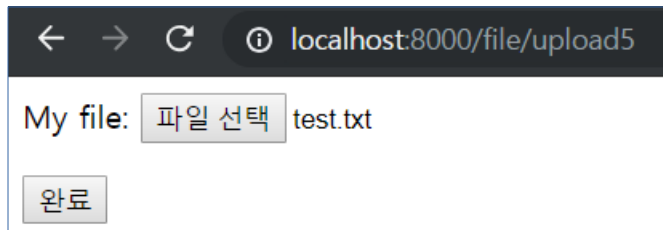
#### – file/models.py

```
class ModelWithFile(models.Model):  
    my_file = models.FileField(upload_to='%Y/%m/%d/%H/%M')
```

## ■ 파일 업로드

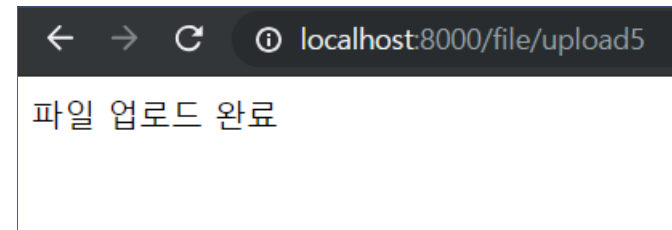
### ● forms.py / models.py 사용 – 자동저장(양식지정)

– <http://localhost:8000/file/upload5>



← → ↻ ⓘ localhost:8000/file/upload5

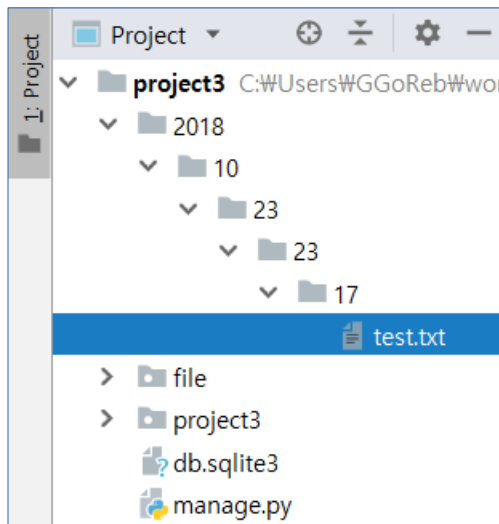
My file:  test.txt



← → ↻ ⓘ localhost:8000/file/upload5

파일 업로드 완료

– 업로드 경로



## ■ 파일 다운로드

### ● 템플릿 X

– file/views.py

```
def download(request):
    filepath = os.path.join(settings.BASE_DIR, 'upload/파일명.txt')
    filename = os.path.basename(filepath)

    with open(filepath, 'rb') as f:
        response = HttpResponse(f, content_type='application/octet-stream')
        response['Content-Disposition'] = 'attachment; filename="{ }".format(filename)
    return response
```

– file/urls.py

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

urlpatterns = [
    path('download', views.download),
]
```