File upload

2.关于文件上传

2.1 基本操作

```html
<form method="post" enctype="multipart/form-data">
    {% csrf_token %}
    <input type="text" name="username">
    <input type="file" name="avatar">
    <input type="submit" value="提交">
</form>
```

```python
# 'username': ['big666']
print(request.POST)
# {'avatar': [<InMemoryUploadedFile: 图片 1.png (image/png)>]}
print(request.FILES)
```

python

```python
def upload_list(request):
    if request.method == "GET":
        return render(request, 'upload_list.html')

    # # 'username': ['big666']
    # print(request.POST)  # 请求体中数据
    # # {'avatar': [<InMemoryUploadedFile: 图片 1.png (image/png)>]}
    # print(request.FILES)  # 请求发过来的文件 {}

    file_object = request.FILES.get("avatar")
    # print(file_object.name)  # 文件名: WX20211117-222041@2x.png

    f = open(file_object.name, mode='wb')
    for chunk in file_object.chunks():
        f.write(chunk)
    f.close()

    return HttpResponse("...")
```

1. Upload Excel

```python
import pandas as pd
def upload_list(request):
    file_object = request.FILES.get('exc')
    print(file_object)
    df = pd.read_csv(file_object)
    print(df['type'][2])
    for i in range(2, 4):
        text = df['type'][i]
        exist = models.Department.objects.filter(title=text).exists()
        if not exist:
            models.Deparment.object.create(title=text)

    # return render(request, 'upload_list.html')
    # return HttpResponse("^-^")
    return redirect('/depart/list/')
```
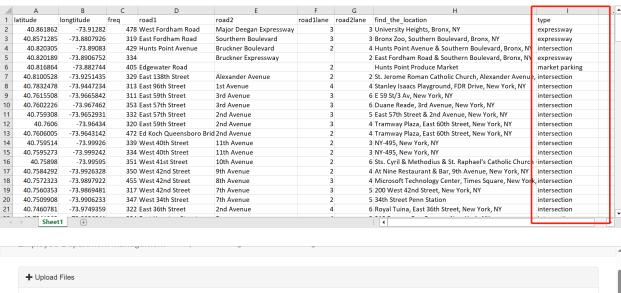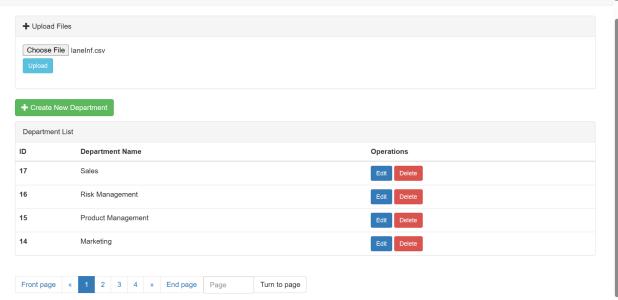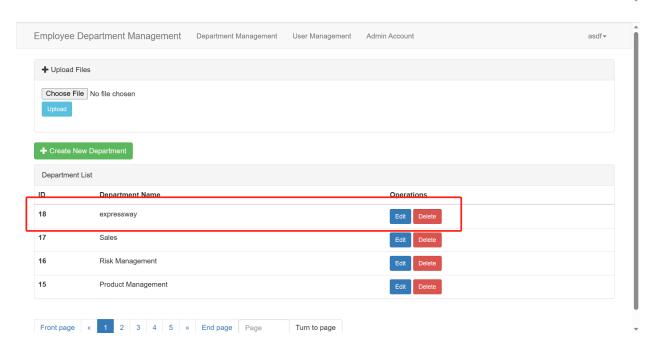
```html
<div class="panel panel-default">
        <div class="panel-heading">
            <span class="glyphicon glyphicon-plus" aria-hidden="true"></span>
            Upload Files
        </div>
        <div class="panel-body">
            <form method="post" enctype="multipart/form-data" action="/upload/list/">
                {% csrf_token %}
                <div class="form-group">

                    <input type="file" name="exc">
                    <input type="submit" value="Upload" style="margin-top: 5px;" class="btn btn-info btn-sm">
                </div>
            </form>
        </div>
    </div>
```

excel

The spreadsheet contains the following data:

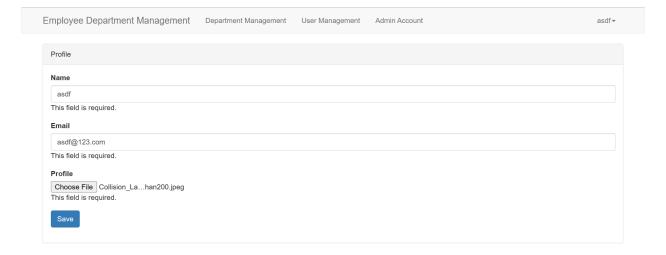|	A	B	C	D	E	F	G	H	I
1	latitude	longtitude	freq	road1	road2	road1lane	road2lane	find_the_location	type
2	40.861862	-73.91282	478	West Fordham Road	Major Deegan Expressway	3	3	University Heights, Bronx, NY	expressway
3	40.8571285	-73.8807926	319	East Fordham Road	Sourthern Boulevard	3	3	Bronx Zoo, Southern Boulevard, Bronx, NY	expressway
4	40.820305	-73.89083	429	Hunts Point Avenue	Bruckner Boulevard	2	4	Hunts Point Avenue & Southern Boulevard, Bronx, NY	intersection
5	40.820189	-73.8906752	334		Bruckner Expressway		2	East Fordham Road & Southern Boulevard, Bronx, NY	expressway
6	40.816864	-73.882744	405	Edgewater Road		2		Hunts Point Produce Market	market parking
7	40.8100528	-73.9251435	329	East 138th Street	Alexander Avenue	2	2	St. Jerome Roman Catholic Church, Alexander Avenue,	intersection
8	40.7832478	-73.9447234	313	East 96th Street	1st Avenue	4	4	Stanley Isaacs Playground, FDR Drive, New York, NY	intersection
9	40.7615508	-73.9665842	311	East 59th Street	3rd Avenue	3	6	E 59 St/3 Av, New York, NY	intersection
10	40.7602226	-73.967462	353	East 57th Street	3rd Avenue	3	6	Duane Reade, 3rd Avenue, New York, NY	intersection
11	40.759308	-73.9652931	332	East 57th Street	2nd Avenue	3	5	East 57th Street & 2nd Avenue, New York, NY	intersection
12	40.7606	-73.96434	320	East 59th Street	2nd Avenue	3	4	Tramway Plaza, East 60th Street, New York, NY	intersection
13	40.7606005	-73.9643142	472	Ed Koch Queensboro Brid	2nd Avenue	2	4	Tramway Plaza, East 60th Street, New York, NY	intersection
14	40.759514	-73.99926	339	West 40th Street	11th Avenue	2	3	NY-495, New York, NY	intersection
15	40.7595273	-73.999242	334	West 40th Street	11th Avenue	2	3	NY-495, New York, NY	intersection
16	40.75898	-73.99595	351	West 41st Street	10th Avenue	2	6	Sts. Cyril & Methodius & St. Raphael's Catholic Church	intersection
17	40.7584292	-73.9926328	350	West 42nd Street	9th Avenue	2	4	At Nine Restaurant & Bar, 9th Avenue, New York, NY	intersection
18	40.7572323	-73.9897922	455	West 42nd Street	8th Avenue	3	4	Microsoft Technology Center, Times Square, New York,	intersection
19	40.7560353	-73.9869481	317	West 42nd Street	7th Avenue	3	5	200 West 42nd Street, New York, NY	intersection
20	40.7509908	-73.9906233	347	West 34th Street	7th Avenue	2	5	34th Street Penn Station	intersection
21	40.7460781	-73.9749359	322	East 36th Street	2nd Avenue	4	6	Royal Tuina, East 36th Street, New York, NY	intersection

2. Mix Input

```python
class Profile(models.Model):
    name = models.CharField(verbose_name='Name', max_length=32)
    email = models.CharField(verbose_name='Email', max_length=64)
    img = models.CharField(verbose_name='Profile', max_length=128)
```
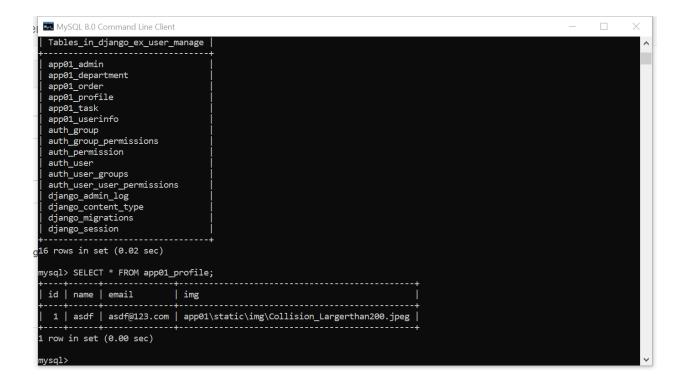
```html
{% extends 'layout.html' %}
{% load static %}
{% block css %}
    <link rel="stylesheet" href="{% static "plugins/bootstrap-datepicker-master/dist/css/bootstrap-datepicker.css" %}">
{% endblock %}

{% block content %}
    <div class="container">
```
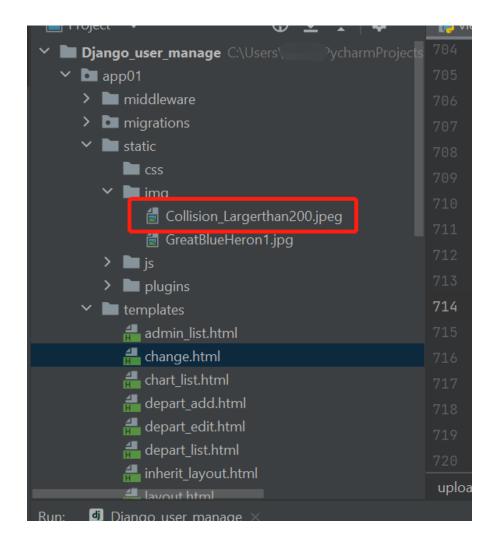
```html
        <div class="panel panel-default">
            <div class="panel-heading">{{ title }}</div>
            <div class="panel-body">
                <form method="post" enctype="multipart/form-data" novalidate>
                    {% csrf_token %}
                    {% for field in form %}
                        <div class="form-group">
                            <label>{{ field.label }}</label>
                            {{ field }}
                            <span style="color: red">{{ field.errors.0 }}</span>
                        </div>

                    {% endfor %}
                    <div class="form-group">
                        <button type="submit" class="btn btn-primary">Save</button>
                    </div>
                </form>
            </div>
        </div>

    </div>
{% endblock %}
{% block js %}
    <script src="{% static "plugins/bootstrap-datepicker-master/dist/js/bootstrap-datepicker.js" %}"></script>
<script>
    $(function () {
        $('#id_create_time').datepicker({
            format: 'yyyy-mm-dd',
            startDate: '0',
            autoclose: true
        });
    })
</script>
{% endblock %}
```
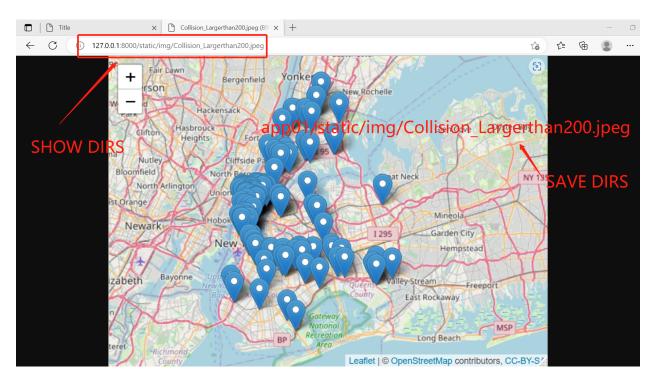
```python
class Upform(forms.Form):
    name = forms.CharField(label='Name')
    email = forms.CharField(label='Email')
    img = forms.FileField(label='Profile')
    exclude_files = ['img']

    def __init__(self, *args, **kwargs):
        super().__init__(*args, **kwargs)

        for name, field in self.fields.items():
            if name in self.exclude_files:
                continue
            if field.widget.attrs:
                field.widget.attrs['class'] = 'form-control'
                field.widget.attrs['placeholder'] = field.label
            else:
                field.widget.attrs = {"class": "form-control",
                                      "placeholder": field.label}
import os
def upload_form(request):
    title = 'Profile'
    if request.method == "GET":
        form = Upform()
        context = {
            'form': form,
            'title': title
        }
        return render(request, 'upload_form.html', context)

    form = Upform(data=request.POST, files=request.FILES)
    if form.is_valid():
        # save img
        # save img direction to database
        image_object = form.cleaned_data.get('img')
        # file path in Linux & windows are diff
        file_path = os.path.join("app01", "static", "img", image_object.name)
        f = open(file_path, mode='wb')
        for chunk in image_object.chunks():
            f.write(chunk)
        f.close()

        models.Profile.objects.create(
            name=form.cleaned_data['name'],
            email=form.cleaned_data['email'],
            img=file_path,
        )

        return HttpResponse("^-^")
    context = {
        'form': form,
        'title': title
    }
    return render(request, 'upload_form.html', context)
```

Employee Department Management   Department Management   User Management   Admin Account   asdf▾

Profile

Name

asdf

This field is required.

Email

asdf@123.com

This field is required.

Profile

[Choose File] Collision_La…han200.jpeg

This field is required.

[Save]

MySQL 8.0 Command Line Client   — ☐ ✕

```
| Tables_in_django_ex_user_manage |
+---------------------------------+
| app01_admin                     |
| app01_department                |
| app01_order                     |
| app01_profile                   |
| app01_task                      |
| app01_userinfo                  |
| auth_group                      |
| auth_group_permissions          |
| auth_permission                 |
| auth_user                       |
| auth_user_groups                |
| auth_user_user_permissions      |
| django_admin_log                |
| django_content_type             |
| django_migrations               |
| django_session                  |
+---------------------------------+
16 rows in set (0.02 sec)

mysql> SELECT * FROM app01_profile;
+----+------+--------------+----------------------------------------------+
| id | name | email        | img                                          |
+----+------+--------------+----------------------------------------------+
|  1 | asdf | asdf@123.com | app01\static\img\Collision_Largerthan200.jpeg |
+----+------+--------------+----------------------------------------------+
1 row in set (0.00 sec)

mysql>
```

BUT

we cannot show img using dirs we save in database.

Thus we can create two dirs, one for saving, one for showing to Users.

```
db_file_path = os.path.join("static", "img", image_object.name)
file_path = os.path.join("app01", db_file_path)
```

```
mysql> SELECT * FROM app01_profile;
+----+------+-------------+---------------------------------------------+
| id | name | email       | img                                         |
+----+------+-------------+---------------------------------------------+
|  1 | asdf | asdf@123.com | app01\static\img\Collision_Largerthan200.jpeg |
+----+------+-------------+---------------------------------------------+
1 row in set (0.00 sec)

mysql> SELECT * FROM app01_profile;
+----+------+-------------+---------------------------------------------+
| id | name | email       | img                                         |
+----+------+-------------+---------------------------------------------+
|  1 | asdf | asdf@123.com | app01\static\img\Collision_Largerthan200.jpeg |
|  2 | asdf | asdf@345.com | app01\static\img\BIRDS.png                  |
+----+------+-------------+---------------------------------------------+
2 rows in set (0.00 sec)

mysql> SELECT * FROM app01_profile;
+----+--------+----------------+--------------------------------------------+
| id | name   | email          | img                                        |
+----+--------+----------------+--------------------------------------------+
|  1 | asdf   | asdf@123.com   | app01\static\img\Collision_Largerthan200.jpeg |
|  2 | asdf   | asdf@345.com   | app01\static\img\BIRDS.png                 |
|  3 | asdf   | asdf@345.com   | static\img\BIRDS.png                       |
|  4 | ijnbhu | ijnbhu@7yg.com | static\img\MixedTern2.jpg                  |
+----+--------+----------------+--------------------------------------------+
4 rows in set (0.00 sec)

mysql>
```

Media

File upload   6