
상권분석

순천향대학교
컴퓨터시스템연구실

컴퓨터공학과/23.05.08/강민서

상권분석

1. ~~파일 업로드 페이지~~
 2. 파일 업로드 버튼 클릭 후 db에 회원 아이디 및 파일 저장 + 관리자 페이지 ←
 3. 서비스 관리자가 적용하기를 클릭하면 db에 국가명, 도시명, 파일명(파일 안 데이터를 꺼내서 같이 저장) 저장
 4. 분석된 결과가 제대로 전송되는지 postman을 통해 확인
- 4-1. 분석 주제별로 기능 확인

상권분석

+ 파일 업로드 기능

› html 파일이 전송 가능하게 수정

```
<form action="upload_file" method="POST" enctype="multipart/form-data">
  {% csrf_token %}
  <input class="txt_input" type="text" placeholder="국가명" name="country">
  <input class="txt_input" type="text" placeholder="도시명" name="city">
  <table id="table">
    <th width="40%">데이터 종류</th>
    <th>파일 올리기</th>
    <tr>
      <td>상권영역(필수)</td>
      <td><input type="file" name="file[]" id="district"></td>
    </tr>
    <tr>
      <td>점포</td>
      <td><input type="file" name="file[]" id="market"></td>
    </tr>
    <tr>
      <td>추정매출</td>
      <td><input type="file" name="file[]" id="sales"></td>
    </tr>
    <tr>
      <td>소득소비</td>
      <td><input type="file" name="file[]"></td>
    </tr>
  </table>
</form>
```

상권분석

+ 파일 업로드 기능

- › 프론트 서버
- › 파일 데이터, 업로드 된 파일명 전달

```
def upload_file(request):  
    file_name=["district", "market", "sales", "income", "district_change", "apart", "people"]  
    if request.method=="POST":  
        files=request.FILES.getlist('file[]')  
        # 파이썬 딕셔너리 형식으로 file 설정  
        upload = {}  
        upload_name=""  
        i=0  
        for file in files:  
            print(file)  
            if file!="":  
                upload[file_name[i]]=file  
                upload_name=upload_name+file_name[i]+","  
                i+=1  
  
        url="http://127.0.0.1:8000/api/upload_file"  
        print(upload_name) //파일명  
        print(upload) //파일  
        response=requests.post(url=url, files = upload, data={"name":upload_name})
```

상권분석

+ 파일 업로드 기능

- › 백엔드 서버
- › 전달받은 파일 데이터 몽고디비에 저장

```
@api_view(['POST'])
def upload_file(request):
    if request.method=='POST':
        name = request.POST['name']
        dict={i:request.FILES[i] for i in name.split(",") if i!=""}
        print(dict)
        print(type(request.FILES["district"]))
        serializer=File_dataSerializer(data=dict)
        if serializer.is_valid():
            serializer.save()
        else:
            print(serializer.errors)
    return JsonResponse({'message': 'Tutorials were deleted successfully!'}, status=status.HTTP_204_NO_CONTENT)
```

```
quit the server with ctrl-c
{'district': <InMemoryUploadedFile: 서울시 우리마을가게 상권분석서비스(상권영역).csv (
)>}>
<class 'django.core.files.uploadedfile.InMemoryUploadedFile'>
[]
```

상권분석

+ 파일 업로드 기능

>결과

```
_id: ObjectId('64589abc4728b5f493274258')
id: null
district: "서울시_우리마을가게_상권분석서비스상권영역_4sYrKsZ.csv"
market: ""
sales: ""
income: ""
district_change: ""
apart: ""
people: ""
```

```
_id: ObjectId('64589d4db5cc0965007c5f98')
id: null
district: "서울시_우리마을가게_상권분석서비스상권영역_FT0U6L0.csv"
market: ""
sales: ""
income: ""
district_change: ""
apart: ""
people: ""
```

```
<Class 'django.core.files.uploadedfile.InMemoryUploadedFile'>
<QuerySet [(<File_data: File_data object (None)>), (<File_data: File_data object (None)>)]>
[08/May/2023 15:57:17] "[m"POST /api/upload file HTTP/1.1" 204 51?["0m
```

정리

+ to do

1. ~~파일 업로드 페이지~~
 2. 파일 업로드 버튼 클릭 후 db에 회원 아이디 및 파일 저장 + 관리자 페이지
 3. 서비스 관리자가 적용하기를 클릭하면 db에 국가명, 도시명, 파일명(파일 안 데이터를 꺼내서 같이 저장) 저장
 4. 분석된 결과가 제대로 전송되는지 postman을 통해 확인
- 4-1. 분석 주제별로 기능 확인

Question?



Please contact :

강민서
순천향대학교 컴퓨터학부
멀티미디어관 M606

Email : a01063294905@gmail.com