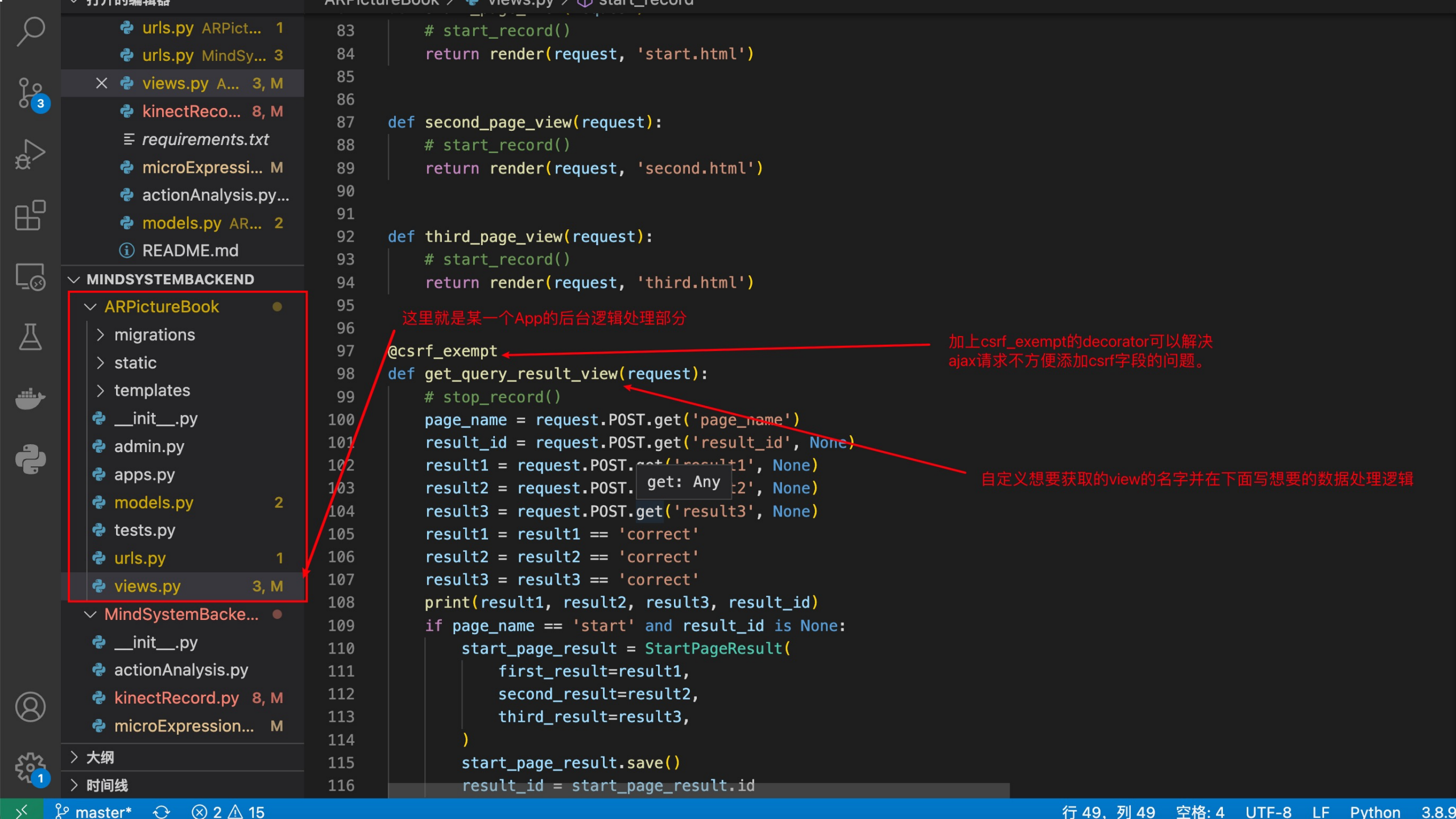
Django Create API Tutorial




```
83      # start_record()
84      return render(request, 'start.html')
85
86
87  def second_page_view(request):
88      # start_record()
89      return render(request, 'second.html')
90
91
92  def third_page_view(request):
93      # start_record()
94      return render(request, 'third.html')
95
96
97  @csrf_exempt
98  def get_query_result_view(request):
99      # stop_record()
100     page_name = request.POST.get('page_name')
101     result_id = request.POST.get('result_id', None)
102     result1 = request.POST.get('result1', None)
103     result2 = request.POST.get('result2', None)
104     result3 = request.POST.get('result3', None)
105     result1 = result1 == 'correct'
106     result2 = result2 == 'correct'
107     result3 = result3 == 'correct'
108     print(result1, result2, result3, result_id)
109     if page_name == 'start' and result_id is None:
110         start_page_result = StartPageResult(
111             first_result=result1,
112             second_result=result2,
113             third_result=result3,
114         )
115         start_page_result.save()
116         result_id = start_page_result.id
```

```
149         start_page_result = StartPageResult.objects.filter(id=result_id)
150         if start_page_result:
151             start_page_result = start_page_result.first()
152         else:
153             return JsonResponse({"status": "error", "errormessage": 'can not find the id in start page!'})
154         third_page_result = ThirdPageResult.objects.filter(
155             start_page_result=start_page_result)
156         if third_page_result:
157             third_page_result = third_page_result.first()
158         else:
159             third_page_result = ThirdPageResult(
160                 start_page_result=start_page_result)
161         third_page_result.first_result = result1
162         third_page_result.second_result = result2
163         third_page_result.third_result = result3
164         third_page_result.save()
165     else:
166         return JsonResponse({"status": "error", "errormessage": "UnExpected Error... Maybe dismiss id..."})
167 
168     return JsonResponse({"status": "success", "result_id": result_id})
169 
```

api处理完成后记得返回数据
给前端处理完成的反馈。

urls.py — MindSystemBackend
资源管理器
打开的编辑器
urls.py ARPict... 1
urls.py MindSy... 3
views.py A... 3, M
kinectReco... 8, M
requirements.txt
microExpressi... M
actionAnalysis.py...
models.py AR... 2
README.md
MINDSYSTEMBACKEND
ARPictureBook
migrations
static
templates
__init__.py
admin.py
apps.py
models.py 2
tests.py
urls.py 1
views.py 3, M
MindSystemBacke...
__init__.py
actionAnalysis.py
kinectRecord.py 8, M
microExpression... M
大纲
时间线
urls.py ARPictureBook 1
urls.py MindSystemBackend 3
views.py 3, M
kinectRecord.py 8, M
requirements.txt
ARPictureBook > urls.py > ...
1 from django.urls import path
2 from ARPictureBook.views import (
3 start_page_view,
4 second_page_view,
5 third_page_view,
6 get_query_result_view,
7)
8
9 urlpatterns = [
10 path('start/', start_page_view),
11 path('second/', second_page_view),
12 path('third/', third_page_view),
13 path('get_query_result/', get_query_result_view),
14]
15
把刚才写好的View import过来
确定好自己的url路径
在这里确定前段访问的url

urls.py — MindSystemBackend
资源管理器
打开的编辑器
urls.py ARPict... 1
urls.py MindSy... 3
views.py A... 3, M
kinectReco... 8, M
requirements.txt
microExpressi... M
actionAnalysis.py...
models.py AR... 2
README.md
MINDSYSTEMBACKEND
models.py 2
tests.py
urls.py 1
views.py 3, M
MindSystemBacke...
__init__.py
actionAnalysis.py
kinectRecord.py 8, M
microExpression... M
settings.py
urls.py 3
videoRecord.py
wsgi.py
.gitignore
db.sqlite3
manage.py
大纲
时间线
urls.py ARPictureBook 1
urls.py MindSystemBackend 3
views.py 3, M
kinectRecord.py 8, M
requirements.txt
MindSystemBackend > urls.py > ...
6 Function views
7 1. Add an import: from my_app import views
8 2. Add a URL to urlpatterns: path('', views.home, name='home')
9 Class-based views
10 1. Add an import: from other_app.views import Home
11 2. Add a URL to urlpatterns: path('', Home.as_view(), name='home')
12 Including another URLconf
13 1. Import the include() function: from django.urls import include, path
14 2. Add a URL to urlpatterns: path('blog/', include('blog.urls'))
15 """
16 from django.contrib import admin
17 from django.urls import path, include
18
19 urlpatterns = [
20 path('admin/', admin.site.urls),
21 path('ARPicture/', include('ARPictureBook.urls'))
22]
23
把ARPictureBook这个app的url文件包含了进来
所以所有ARPictureBook这个app的url都被包含在了/ARPicture/
的这个目录以下。
这里是整个项目的url文件，可以包含多个app的子url文件

settings.py | start.js | second.js | third.js | urls.py

ARPictureBook > static > js > pagejs > start.js > scene_2_wrong

```
232
233  function jumpToNext() {
234    var currentHref = window.location.href;
235    let p=currentHref.split('?')[1]
236    let params=new URLSearchParams(p)
237    const resultId = params.get('id');
238    if (resultId){
239      resultData['result_id'] = resultId;
240    }
241    $.ajax({
242      url: '/ARPicture/get_query_result/',
243      type: 'POST',
244      data: resultData,
245      datatype: 'json',
246      success: function(response){
247        console.log(response);
248        if (response['status'] == 'error'){
249          alert("error!", response['errormessage']);
250          return;
251        }
252        alert('success, click to next page...');
253        url = "/ARPicture/second/"
254        if (response['result_id']) {
255          url += '?id=' + response['result_id']
256        }
257        window.location.href = url;
258      }
259    })
260  }
```

所以刚才新建的那个api的路径就是/ARPicture/get_query_result/

前端ajax直接就可以访问了