

Changeable Parts

Kalu Kalu

| | |
|--|---|
| Changeable Parts | 1 |
| Creating Our Project and App | 2 |
| Adding Our “taobao” App to settings.py | 2 |
| Project urls.py | 3 |
| App urls.py | 3 |
| views.py | 3 |

Hopefully you have noticed that the majority of the code we type can simply be copied and pasted from https://github.com/KaluEmeKalu/intro_to_cs_midterm_review . The complete app made in the review is at the repository; https://github.com/KaluEmeKalu/intro_to_cs_midterm_review/tree/master/retail_store. The majority of the code you will need to write for the final exam app will be almost exactly the same.

Since your mid-term exam will ask you to create a different app from this review, you should know how to use the skeleton of the review to create another app. In this tutorial we will learn the only parts that change from project to project.

In this version, instead creating a `retail_store` project, we will pretend that we are creating a `barbershop` project. Furthermore, instead of creating a `taobao` app, we want to create a “`appointment`” app. Lastly, as opposed to taking the `item` and `address` input by the user, we instead want to take user input of 1) the barbershop customer’s name, `customer_name` and 2) the time they want to come to get a haircut, `time`.

Let’s get started.

Creating Our Project and App

```
KaludeMacBook-Pro:~ kalukalu$ cd ~/Documents/kalu/
KaludeMacBook-Pro:kalu kalukalu$ source ~/
django_and_machine_learning_venv/bin/activate
(django_and_machine_learning_venv) KaludeMacBook-Pro:kalu kalukalu$
django-admin startproject retail_store 1
(django_and_machine_learning_venv) KaludeMacBook-Pro:kalu kalukalu$ cd
retail_store/
(django_and_machine_learning_venv) KaludeMacBook-Pro:retail_store
kalukalu$ python manage.py startapp taobao 2
```

1. Since the name of our project is barbershop we want to do `django-admin startproject barbershop` instead 1
2. And since our app name should be “appointment” we want to do `python manage.py startup appointment` instead. 2

Adding Our “taobao” App to settings.py

`your_name/retail_store/retail_store/settings.py`

1. `INSTALLED_APPS = [`
2. `'django.contrib.admin',`
3. `'django.contrib.auth',`
4. `'django.contrib.contenttypes',`
5. `'django.contrib.sessions',`

```
6. 'django.contrib.messages',
7. 'django.contrib.staticfiles',
8. 'taobao', 3
9. ]
```

Here 3 Because our app name is “appointment” we want to add “appointment” instead of taobao.

Project urls.py

your_name/retail_store/retail_store/urls.py

```
1. from django.contrib import admin
2. from django.urls import path, include
3.
4. urlpatterns = [
5.     path("", include('taobao.urls')), 4
6.     path('admin/', admin.site.urls),
7. ]
```

Because we want to include urls for our “appointment” app 4 , we should use include(‘appointment.urls’) instead.

App urls.py

your_name/retail_store/taobao/urls.py

```
1. from django.urls import path
2.
3. from . import views
4. app_name = 'taobao' 5
5. urlpatterns = [
6.     path("", views.home_page, name='home_page'),
7. ]
```

Because our app name is “appointment” 5 we want to say app_name = ‘appointment’ instead.

views.py

your_name/retail_store/taobao/view.py

```
1. from django.shortcuts import render
2.
```

```

3. def home_page(request):
4.
5.     if request.method == 'POST':
6.         item = request.POST.get('item') 6
7.         address = request.POST.get("address") 7
8.
9.         return render(request, 'taobao/home_page.html', {'item': item, 'address':
address}) 8
10.
11.     return render(request, 'taobao/home_page.html' ) 9

```

1. Because the user inputted data we want to get is the “time” of the appointment we want to change the variable name 6 , and more importantly we also have to change the .get() argument to “time”. In our home_page.html we must also include “time” as the value of an input or textarea name attribute. Therefore it should be time = request.POST.get('time') instead of item = request.POST.get('item').
2. Similarly, because the user inputted data we want to get is the “customer_name” of the customer we want to change the variable name 7 , and more importantly we also have to change the .get() argument to “customer_name”. In our home_page.html we must also include “customer_name” as the value of an input or textarea name attribute. Therefore it should be customer_name = request.POST.get("customer_name") instead of address = request.POST.get("address") .
3. Because our app name is “appointment” and where we will save our template is inside appointment/home_page.html , we must change the 'taobao/home_page.html' 8 to 'appointment/home_page.html'.
4. Also, because the variable names we set up in line number 6 was changed from “item” to “time”, we want to change our context dictionary keyword and object from item 8 to “time”. Similarly, because the variable names we set up in line number 7 was changed from “address” to “customer_name”, we want to also change our context dictionary keyword and object from “address” 8 to “customer_name”. Therefore it should be render(request, 'appointment/home_page.html', {'time': time, 'customer_name': customer_name}) instead of render(request, 'taobao/home_page.html', {'item': item, 'address': address})
5. Lastly, again because our app name is “appointment” and where we will save our template is inside appointment/home_page.html , we must change the 'taobao/home_page.html' 9 to 'appointment/home_page.html'.

your_name/retail_store/taobao/templates/taobao/home_page.html 10

```

1. <!DOCTYPE html>
2. <html>
3. <head>
4.     <title>Kool Karen Taobao</title>
5. </head>

```

```

6. <body>
7.   <h1>Kool Karen Retail Store</h1>
8.   <p>Your convenience is our duty</p>
9.
10.  <h2>Here are all our items!</h2>
11.
12.  <!-- Item 1 -->
13.  <p>Water: 12RMB</p>
14.  
15.  <br>
16.
17.  <!-- Item 2 -->
18.  <p>Toothbrush: 7RMB</p>
19.  
20.
21.  <!-- Item 3 -->
22.  <p>Chips: 10RMB</p>
23.  
24.
25.  <!-- Order Form -->
26.  <h2>Place your Order</h2>
27.  <form action="{% url 'taobao:home_page' %}" method= "POST"> 11
28.      {% csrf_token %}
29.      <input type="text" name="item" placeholder="Please Type Which Item You
    Want"><br> 12
30.      <textarea name="address" placeholder="Please Type Your Address" ></
    textarea><br> 13
31.      <input type="submit" value="Place Your Order" >
32.  </form>
33.
34.  <!-- Show Order -->
35.  <h1>Your Order:</h1>
36.  <p>Item Ordered: {{item}}</p> 14
37.  <p>Delivery Address: {{address}}</p> 15
38.
39.</body>
40.</html>

```

1. Because our app name is “appointment” instead of “taobao” we should create our home_page.html template at your_name/retail_store/appointment/templates/appointment/

home_page.html instead of your_name/retail_store/taobao/templates/taobao/home_page.html 10 .

2. Because our app_name that we set in our app urls.py is "appointment" and not taobao, we should change "{% url 'taobao:home_page' %}" to "{% url 'appointment:home_page' %}" .
3. Because in our views we had request.POST.get("time") we should change our input tag name attribute from name="item" to name="time" 12 .
4. Similarly because in our views we had request.POST.get("customer_name") we should change our textarea tag name attribute from name="address" to name="customer_name" 13 .
5. Because in our views.py we passed our return render() function a context dictionary with the key {'time': time} instead of {'item': item}, we have to use the {{ time }} instead of {{ item }} 14 .
6. Similarly, because in our views.py we passed our return render() function a context dictionary with the key {'customer_name': customer_time} instead of {'item': item}, we have to use the {{ customer_name }} instead of {{ item }} 15 .
7. Lastly of course you have to change the pictures and the text of home_page.html. But that stuff will only affect the look of your website, it won't affect the ability for your website to work.