

Multiple Apps

Objectives:

- Practice creating Django apps
 - Get familiar with a Django project with multiple apps
-

With Django, it's easy to create multiple apps that can be used across multiple projects. Let's say that you're a freelancer and have worked with many clients in the past. Say that almost every single project you've worked on, the client/project wants their own blogs, surveys, and user management system. Instead of having to re-create these modules each time from scratch, you decide to create three independent *apps* that you can utilize across all of these projects.

Add 2 additional apps, **surveys** and **users** to your project from the First Django Project assignment. (If you've forgotten how to create just an app, refer to all of step 5 (a-e) in the Creating a Django Project module.)

- **blogs** - update the routes as follows (content is the same):
 - /blogs - display the string "placeholder to later display a list of all blogs" with a method named "index"
 - /blogs/new - display the string "placeholder to display a new form to create a new blog" with a method named "new"
 - /blogs/create - redirect to the "/blogs" route with a method called "create"
 - /blogs/<number> - display the string "placeholder to display blog number: {{number}}" with a method named "show"
 - /blogs/<number>/edit - display the string "placeholder to edit blog {{number}}" with a method named "edit"
 - /blogs/<number>/delete - redirect to the "/blogs" route with a method called "destroy"
- **surveys**
 - /surveys - display the string "placeholder to display all the surveys created"
 - /surveys/new - display "placeholder for users to add a new survey"
- **users**
 - /register - display the string "placeholder for users to create a new user record"
 - /login - display the string "placeholder for users to log in"
 - /users/new - have the same method that handles /register also handle the url request of /users/new
 - /users - display the string "placeholder to later display all the list of users"

Hint:

Notice how the blogs and surveys routes all begin with the same pattern, but the users app routes do not. This means our project urls.py file should look something like this:

```
project_name/project_name/urls.py  
urlpatterns = [  
    path('blogs/', include('apps.blog_app.urls')),  
    path('surveys/', include('apps.survey_app.urls')),  
    path("", include('apps.users_app.urls')),  
]
```

Then, in the respective blogs and surveys app urls.py files, we only need to include the remainder of the route to match!

A reminder about redirecting: you should always provide the whole path, starting with the first /.

1. Add a surveys app and a users app to your first Django project
2. Complete the /blogs route
3. Complete the /blogs/new route
4. Complete the /blogs/create route
5. Complete the /blogs/<number> route
6. Complete the /blogs/<number>/edit route
7. Complete the /blogs/<number>/delete route
8. Complete the /surveys route
9. Complete the /surveys/new route
10. Complete the /register route
11. Complete the /login route
12. Complete the /users/new route
13. Complete the /users route
14. NINJA BONUS: Have the root route utilize the same method as the /blogs route