

## Exercise 2.2: Django Project Set Up

### Reflection Questions

1. **Suppose you're in an interview. The interviewer gives you their company's website as an example, asking you to convert the website and its different parts into Django terms. How would you proceed? For this question, you can think about your dream company and look at their website for reference.**  
*(Hint: In the Exercise, you saw the example of the CareerFoundry website in the Project and Apps section.)*
  - a. The first thing I would do is separate all of the pieces as they relate to MVT. You need to break down the website, and its function, into parts so you can see what fits where and how many apps you're going to create. Once you understand that, you can work within the Django template to build the site. Your customer-facing html will be in the template file, which will be managed by the VIEW file. The View file takes in HTTP requests from URLs and forwards that request to the appropriate view, which is then rendered. Any data that needs to be read or written goes from the View to the Model and then back again. Once all of that data is compiled, the View sends out the correct HTTP Response, and the user is presented with the desired website.
2. **In your own words, describe the steps you would take to deploy a basic Django application locally on your system.**
  - a. To deploy a basic Django application, you're first going to need to make sure you can create a virtual environment, and make sure you have the correct commands depending on your operating system. Once that is established, you'll install Django using pip, create your virtual environment, and then create your project using the CLI and "manage.py startapp <project>". From there you can navigate to the files to see that everything has been populated, open the venv inside of that folder via your CLI and use migrate and runserver to set up your SQLite db and turn on the server. That will generate your URL from which you can see everything is working properly. Before that, or after you shut the server back down, you can create a superuser via the manage.py createsuperuser command and set up your admin functions, which can be checked again by running the server again.
3. **Do some research about the Django admin site and write down how you'd use it during your web application development.**
  - a. The primary use of the Django admin site, in its early stages of building an application, can be summed up by its ability to allow the user to perform CRUD options quickly and easily. You're able to test models quickly to ensure that the data you're processing is the correct data and that your functions are acting as expected. Additionally, the admin site can be used to create users under the superuser, in the event that you wish to test the site to other, outside, testers. The admin site is customizable, allowing you to configure list

views, or add filters, for example, and can be set up with advanced configurations; but should only be modified as necessary for the completion of the application.