Shelby Tubbs

1/25/2020

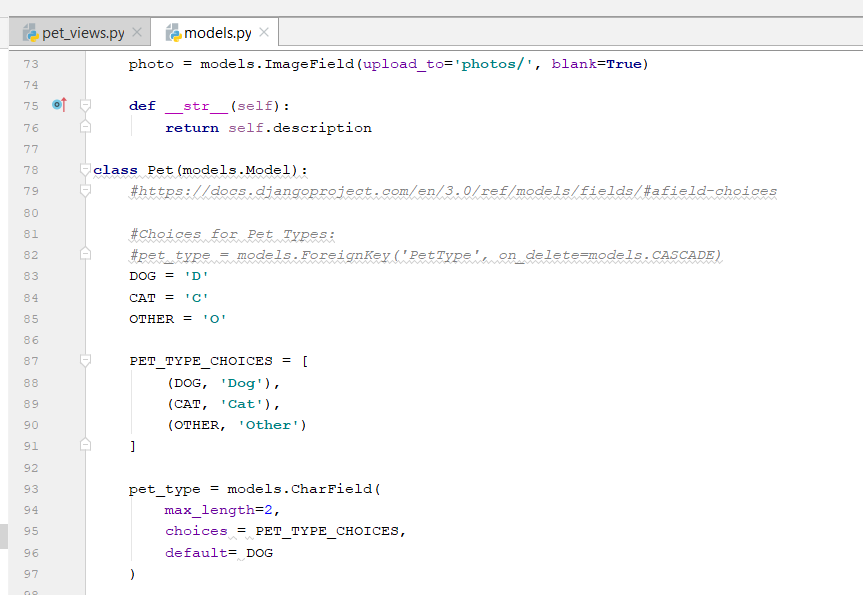
CS 467

Info regarding model changes for pet/user POST:

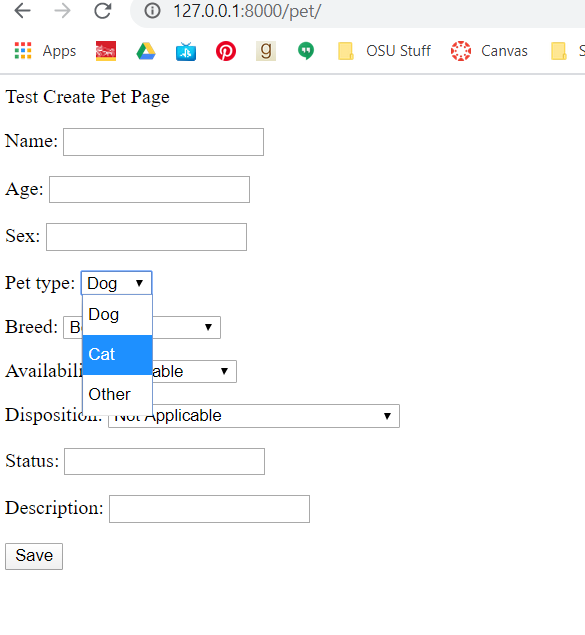
In order to take advantage of some of Django’s features, I made changes to the models that were originally in our project plan. The first change was adding choices into the pet object. This allows choices to be stored (can be displayed as a dropdown) in code, so that we don’t have to make changes to the database every time we set up our separate coding environments.

Example – Pet Type:

Instead of defining the Pet Type as a foreign key, it has been changed to a CharField with PET\_TYPE\_CHOICES as options. We can change this to an integer if we would rather identify Dog with 1, Cat with 2, etc.



This shows as follows on the create pet form (form can be customized):



The second change was made to override the existing User setup that Django has written. This is called an AbstractUser and allows us to use Django’s built in authentication instead of building our own. It validates the user info and saves it correctly, including password encryption. When a user logs in, it sends their user information with the request object, which will help us with determining permissions. If a user is not logged in, it sends Anonymous user. There is built in login page functionality, but we can build our own.

