**Assignment 9:**

**Display Dog Data on a Website**

**Master of Science**

**Information and Communications Technology Program**

**Nichel Emmanuel John**

**University of Denver University College**

**June 3, 2020**

**Faculty**: **Steven Russell**

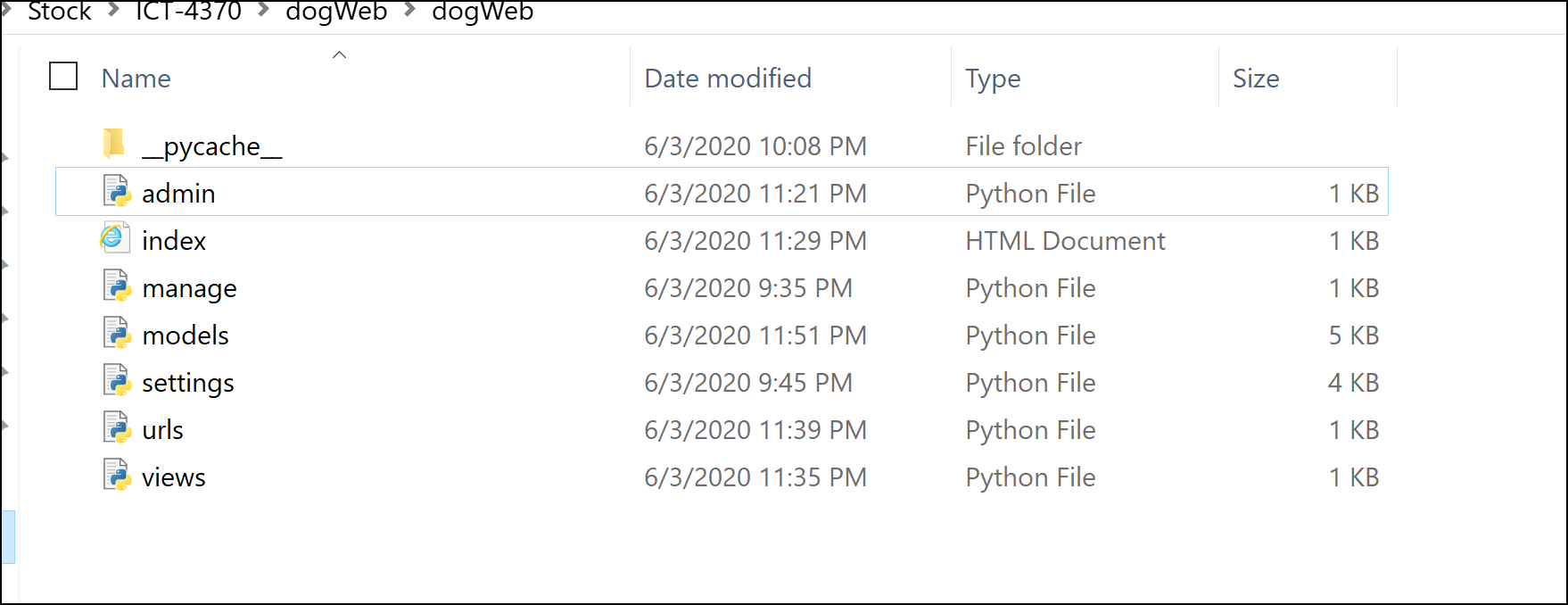
Programming Assignment 9:

This paper shows the steps to follow to build a Django application. The steps follow the blue print outlined in ‘Python Lesson 9’ as well as practices described in Chapter 18 of Python Crash Course by Eric Matthews.

The screenshots were on my machine running Windows 10. The applications used in the screenshots were in Windows command prompt and Python IDLE 3.8.0.

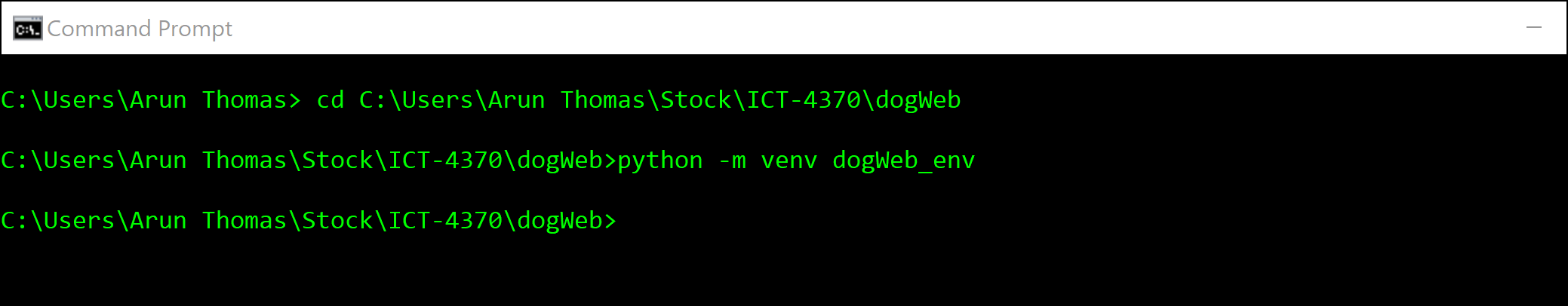
The program itself uses Python and Django to read from an external database of dogs and dog owners and display a table listing the names and statistics of all dogs owned by a single dog owner in a web page on the user’s local machine.

Files Attached in Zip file

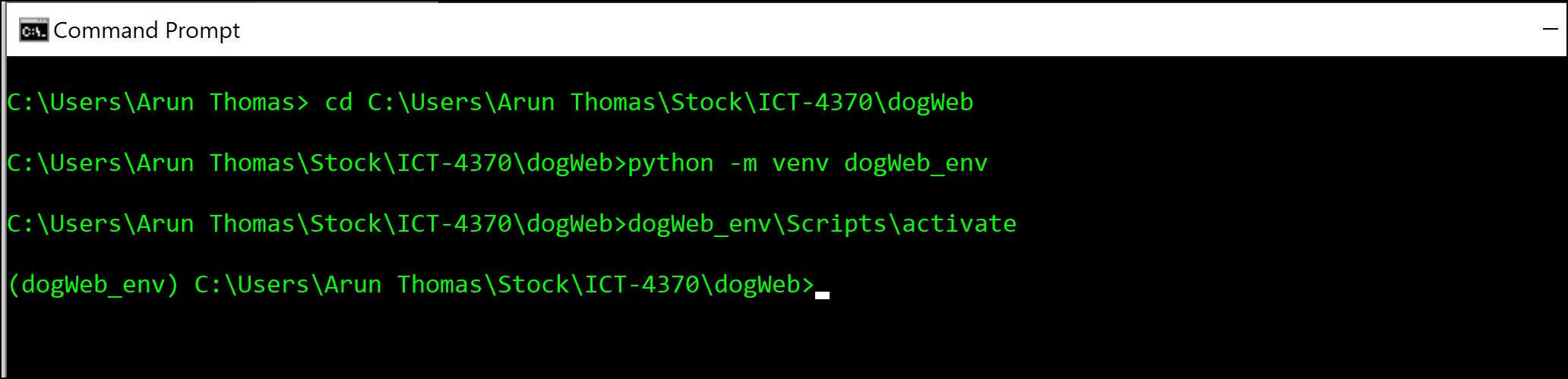


**Steps to follow:**

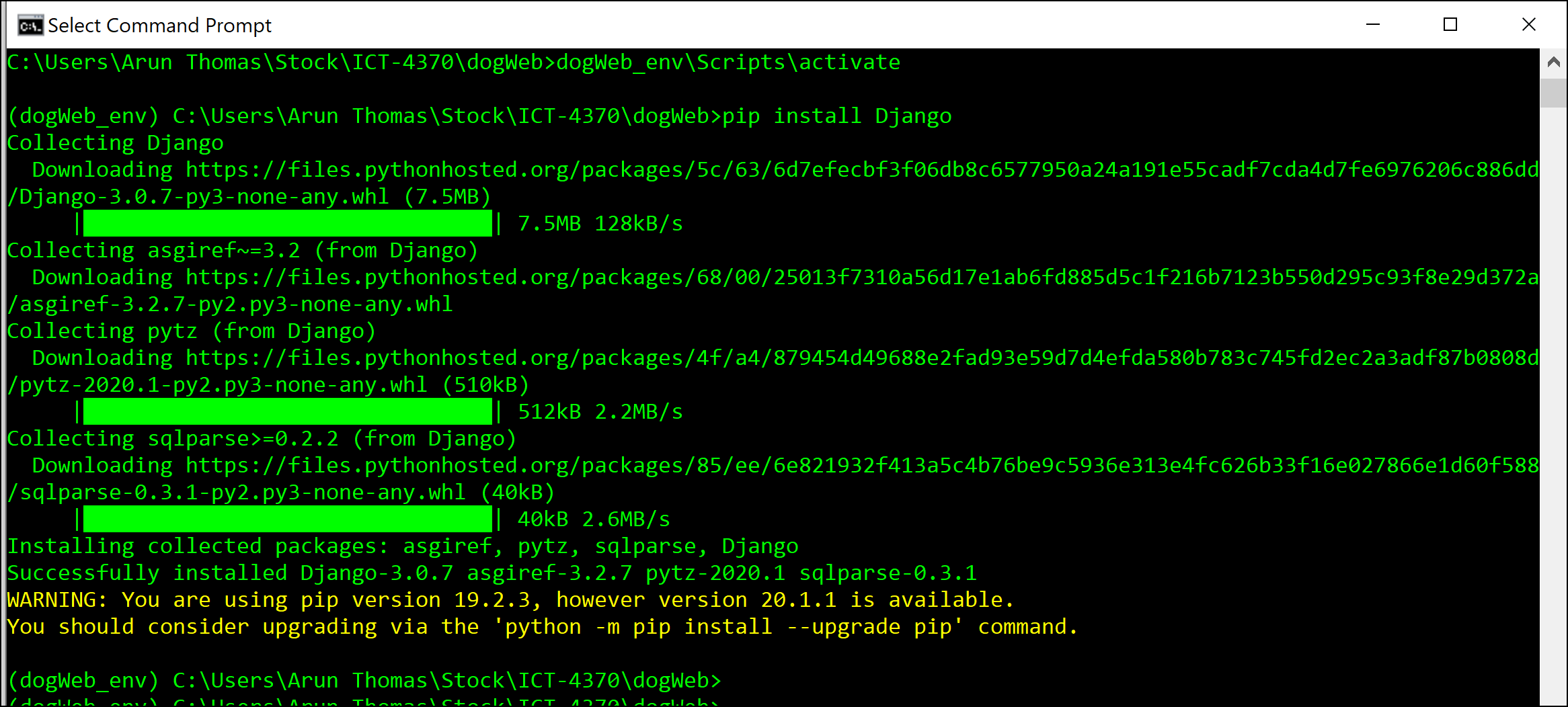
1. Create a new directory for the project called ‘dogWeb,’ switch to that directory in a terminal, and create a virtual environment with the command: python -m venv dogWeb\_env



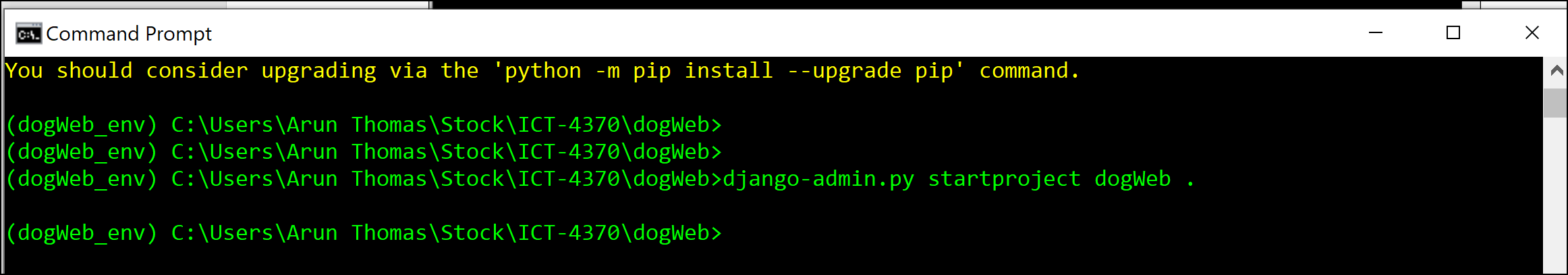
1. Activate the dogWeb virtual environment with the command: dogWeb\_env\Scripts\activate



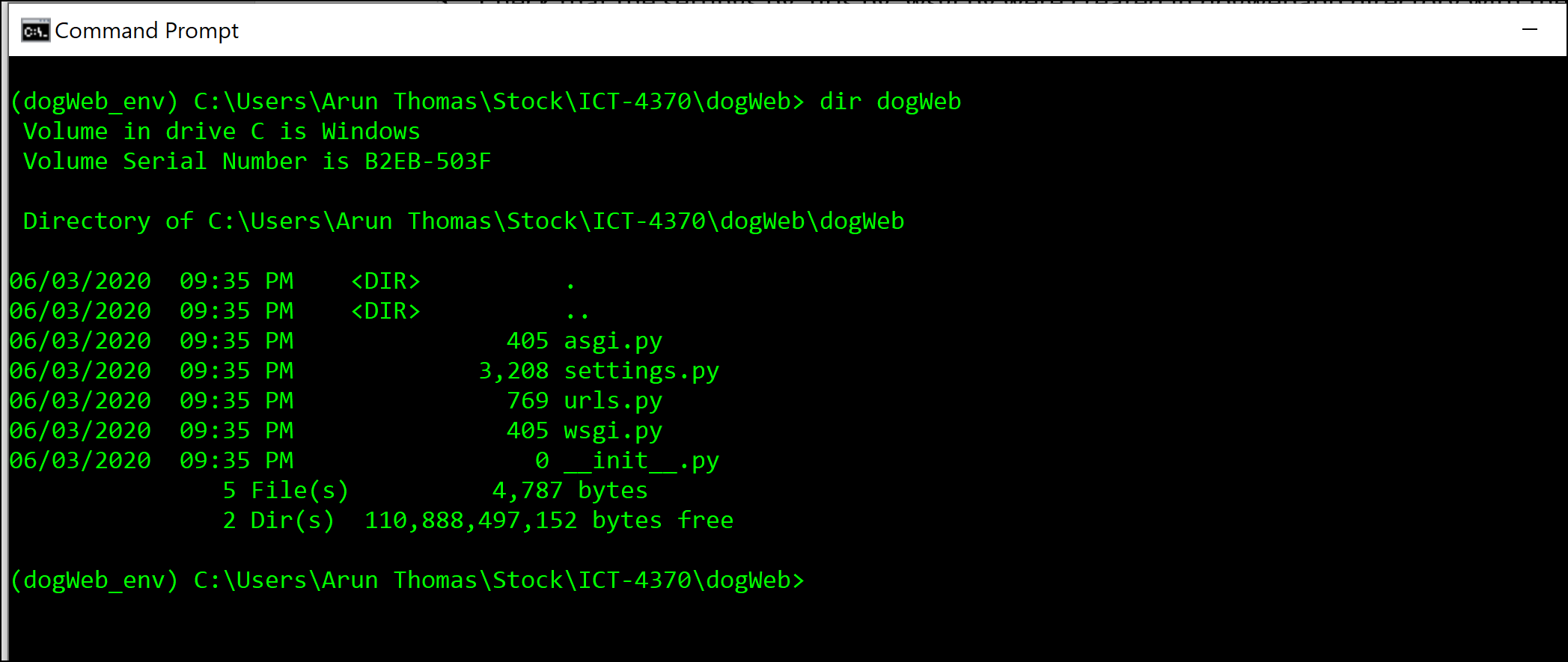
1. Install Django in the dogWeb virtual environment with the command: pip install Django



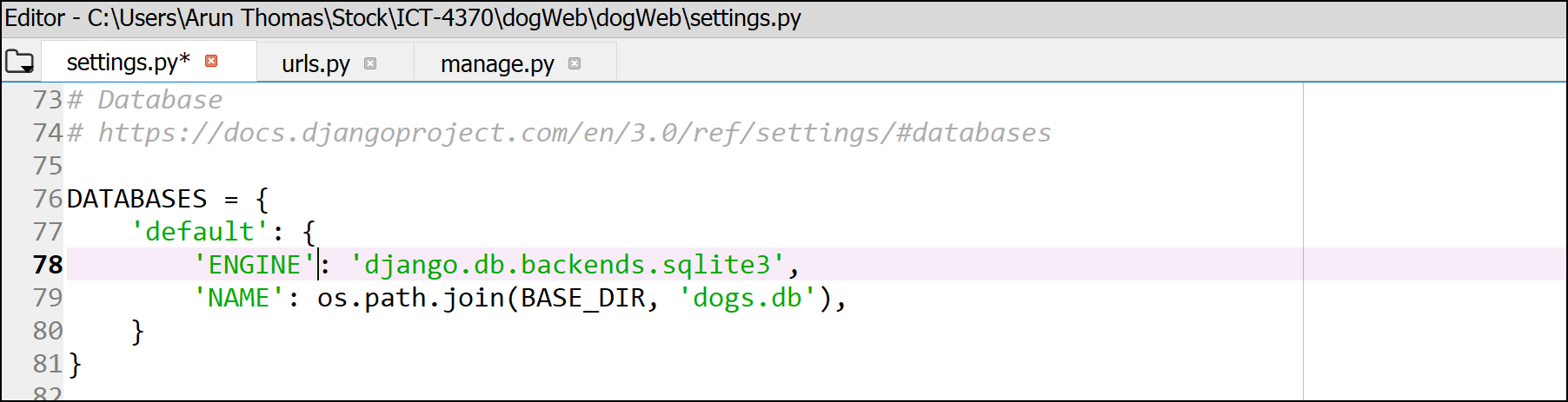
1. Create the new dogWeb project with the command: django-admin.py startproject dogWeb . (This command includes a period after dogWeb.)



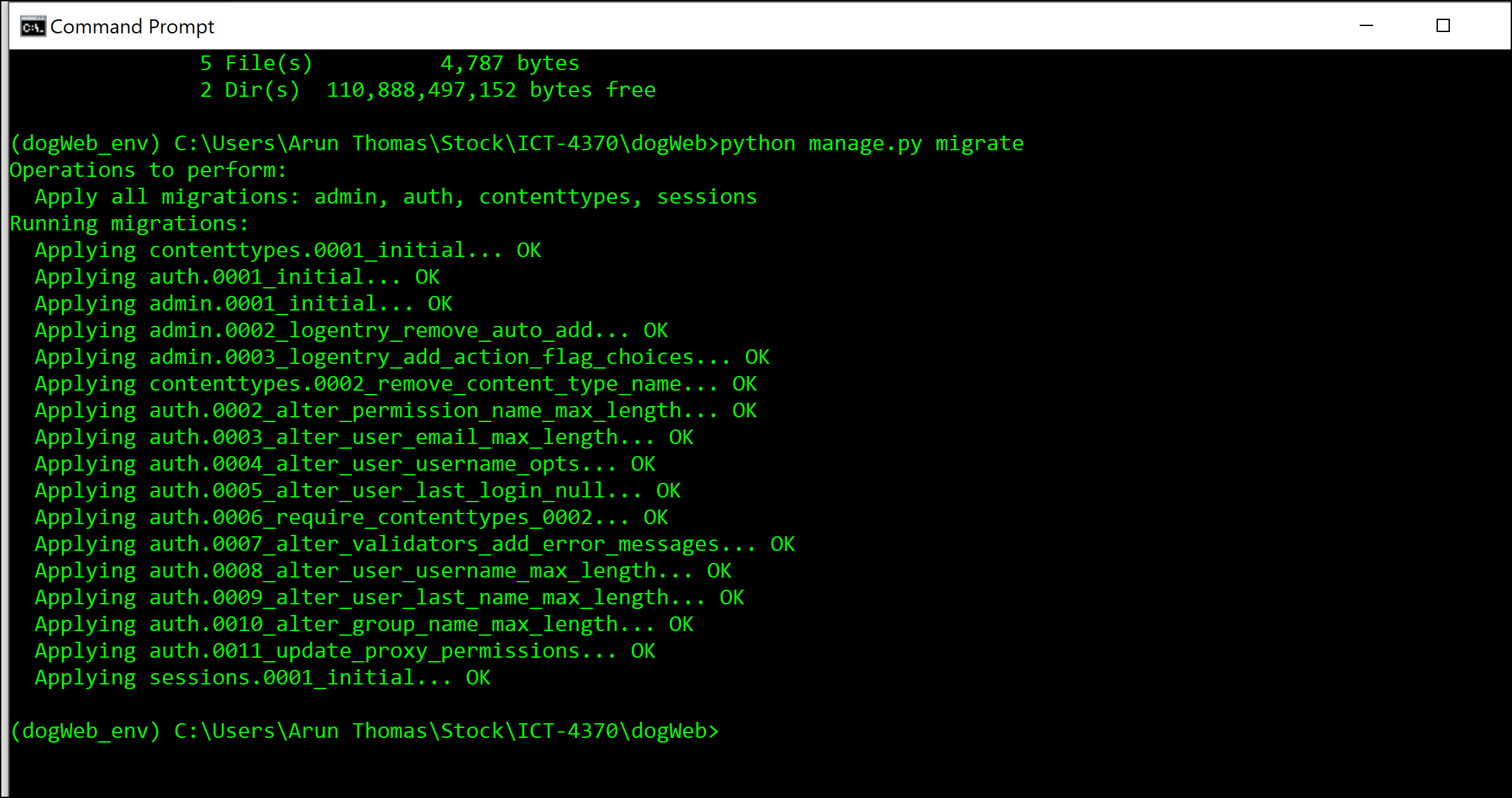
1. Check that the settings.py, urls,py, wsgi.py were created in dogwebapp directory with the input: dir dogwebapp



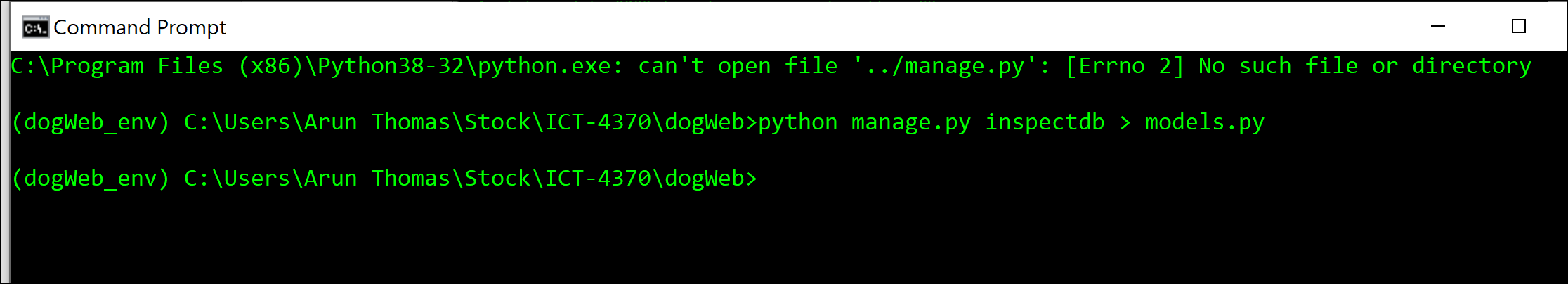
1. Open settings.py and change the default database to point to the ‘dogs.db’ under the database section, then save.



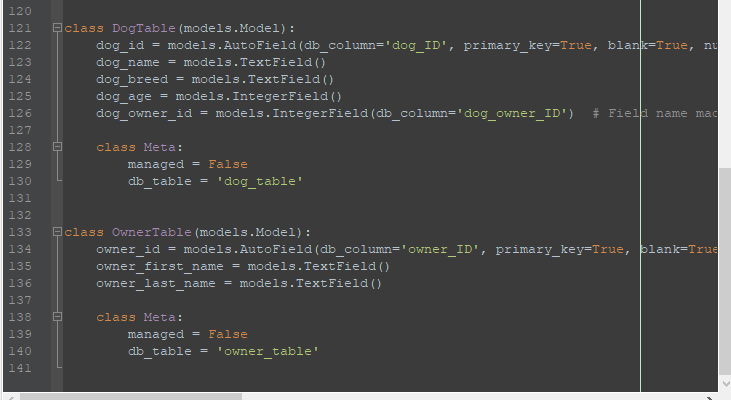
1. Run in the CML: python manage.py migrate



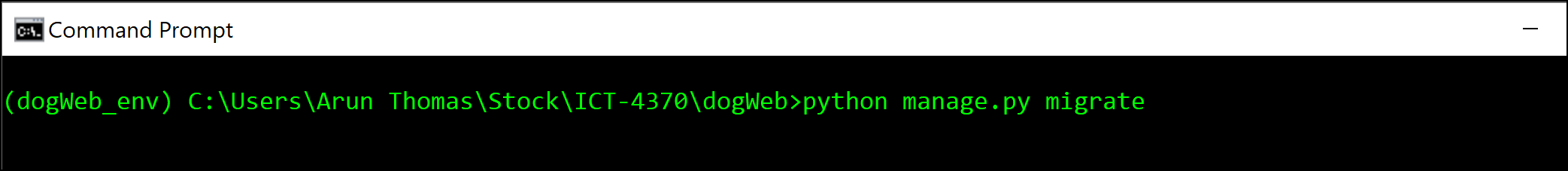
1. CD into the dogwebapp directory and input: python manage.py inspectdb > models.py



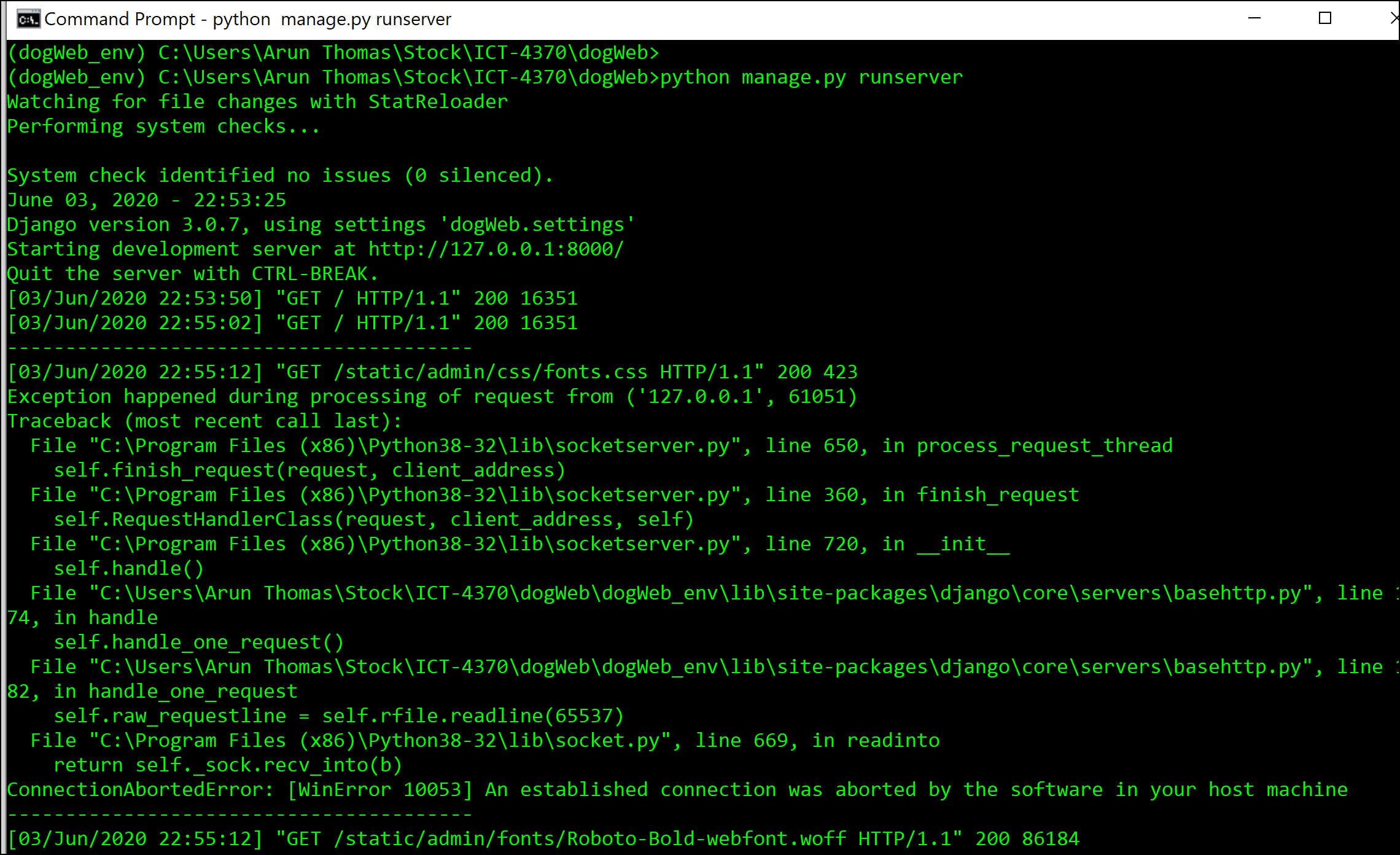
1. Check the models.py code and see that the dog and owner tables were added



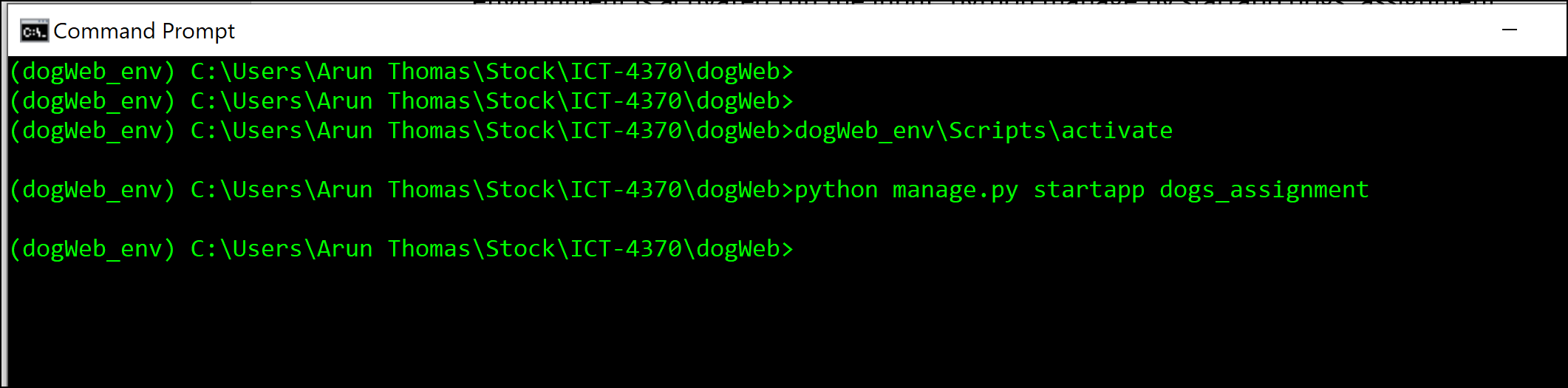
1. While in the dog\_assignment directory input: python manage.py inspectdb > models.py



1. Try to start the project using the input: python manage.py runserver

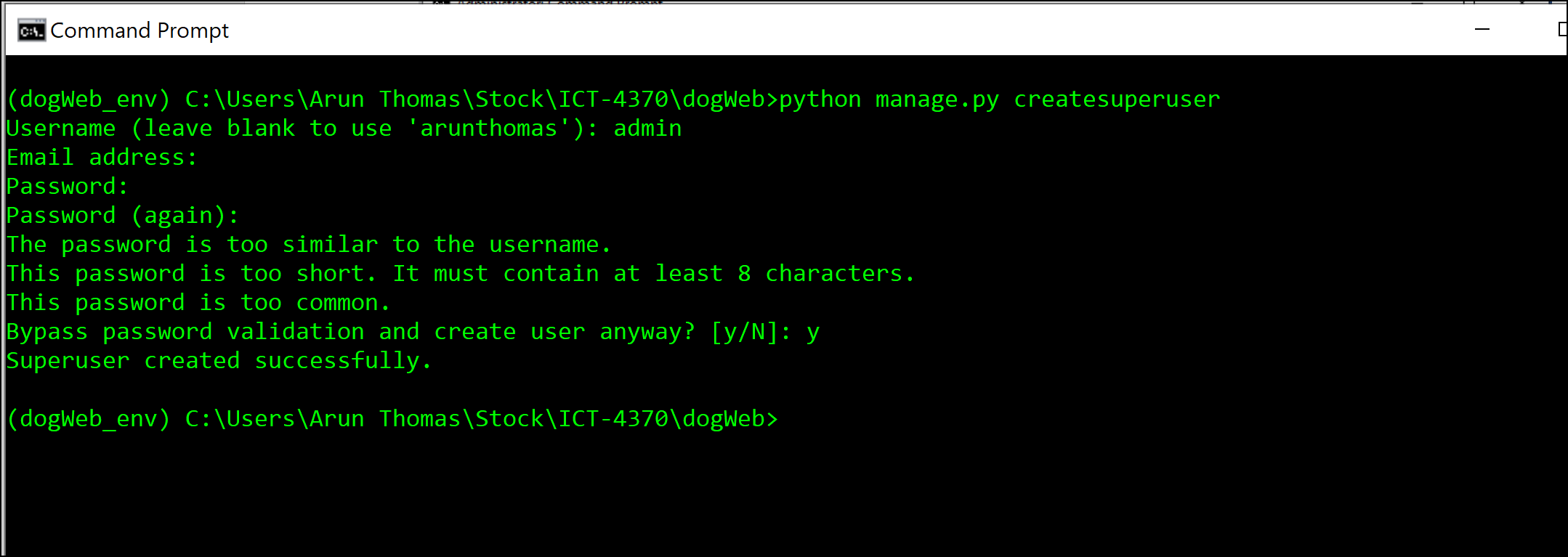


1. Open up a second CLI and cd into the dogassignment directory and activate the virtual environment with the input: .\dogassignment\_env\Scripts\activate. Once the virtual environment is activated run the input: python manage.py startapp dogs\_assignment

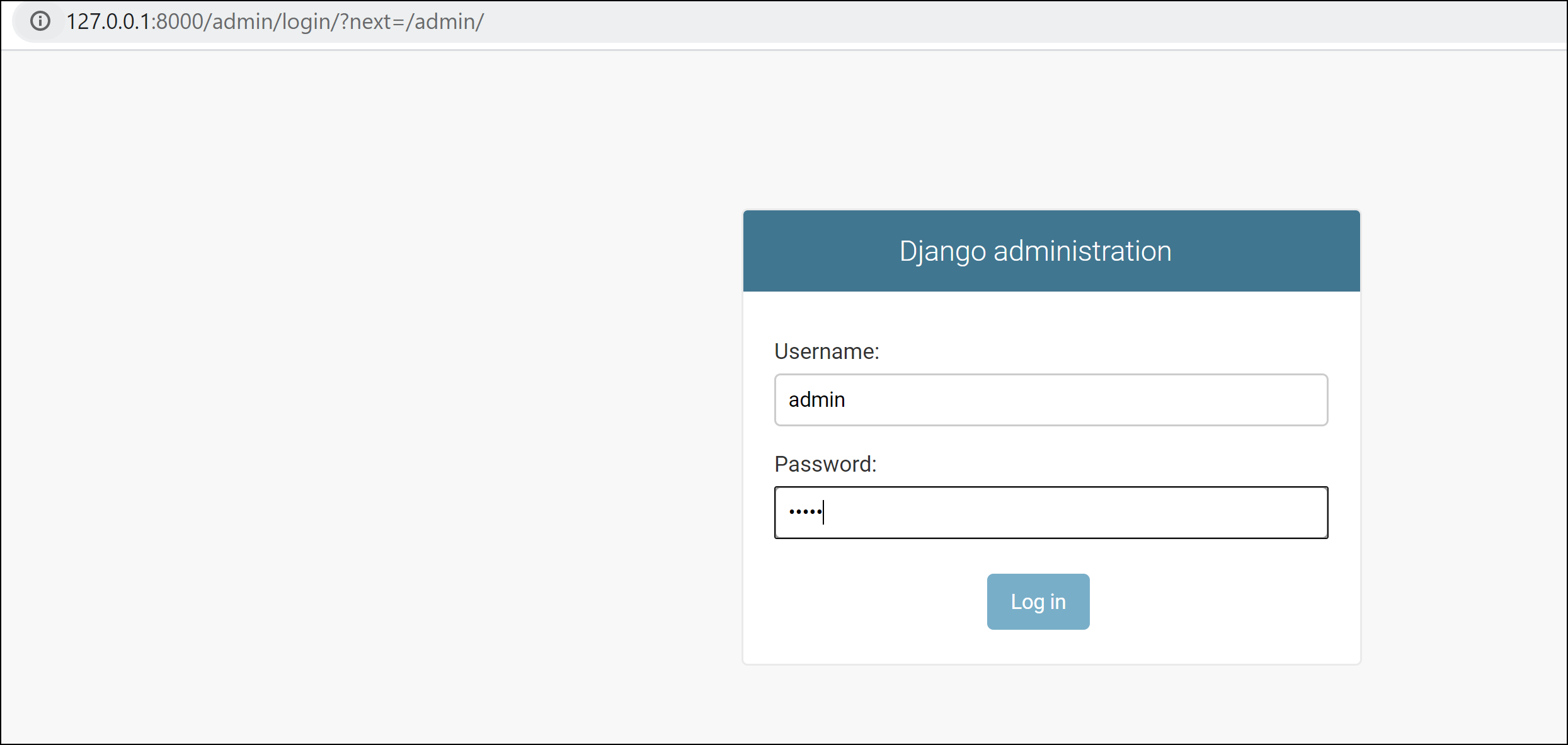


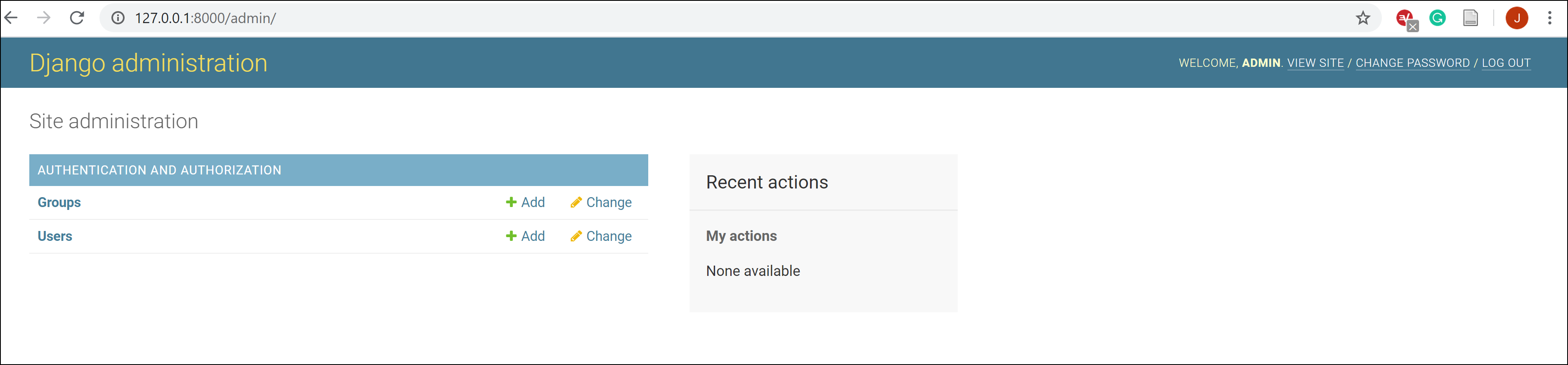
1. Create a super user with the input: python manage.py create super user

Create superuser: Username: admin and Password: admin

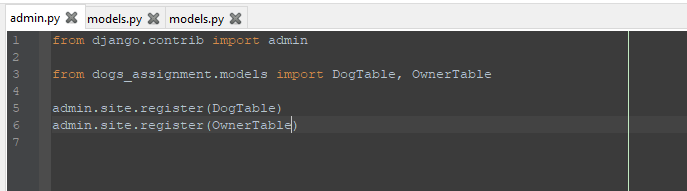


1. Login to the admin site at <http://127.0.0.1:8000/admin/>





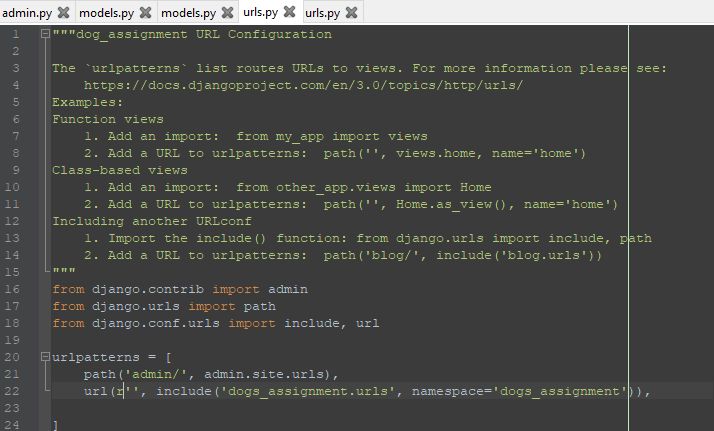
1. Open the admin.py file in the dogs\_assignment folder and add the code below:



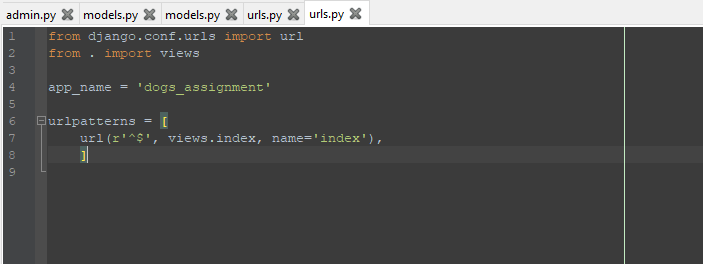
1. Check the admin site for to see if the Dog table and Owner table can be seen



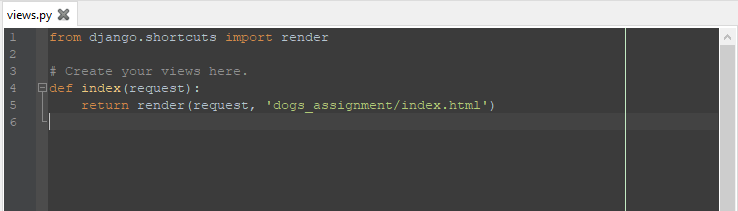
1. Open the urls.py file from the dog\_assignment directory and add the code in the screenshot below.



1. In the dogs\_assignment directory create a urls.py file and add the code in the screenshot below.



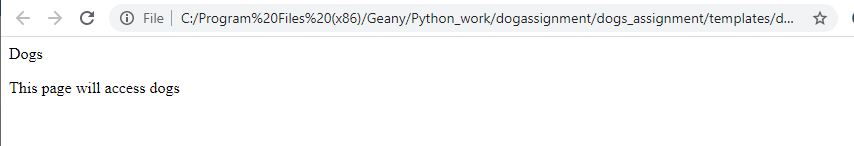
1. In the dogs\_assignment directory open the views.py file and add the code below.



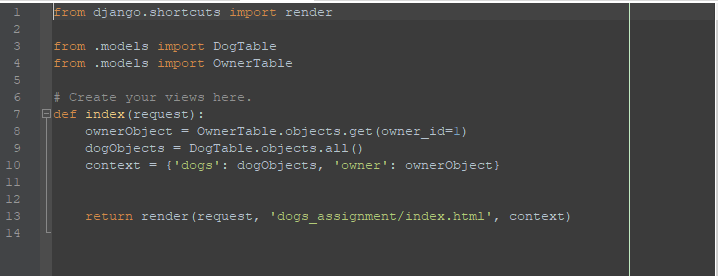
1. Create a file in the dogs\_assignment folder called ‘templates’. The create another folder called ‘dogs\_assignment’. Within the newly created dogs\_assignment folder, using notepad input: <p>Dogs</p>

<p>This page will access dogs</p>

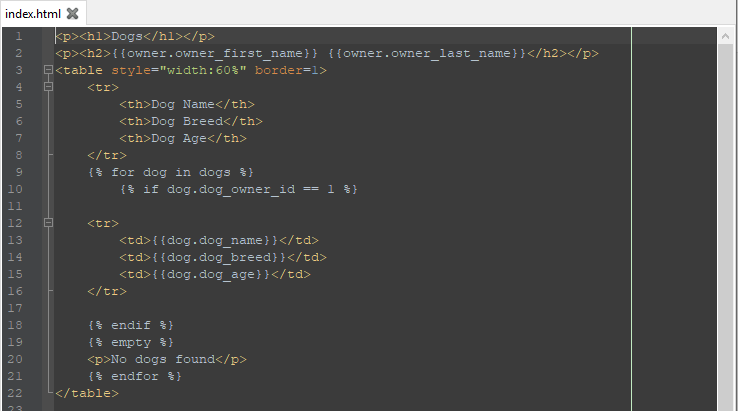
Save this text document as an HTML document



1. In the views.py file edit add the following to the code below.



1. Edit the index.html file to have the following content:



1. With the local host activated, open a web page with localhost:8000 and the mellow information should be displayed

