# React-Django Starter Code Setup Guide

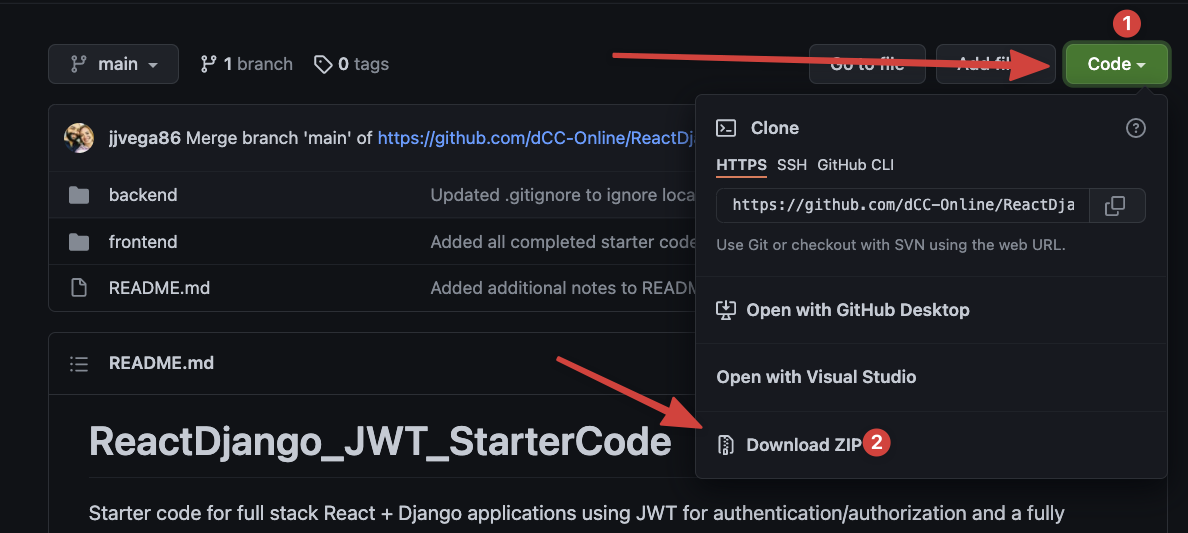
NOTE: This checklist is written using a Macintosh computer, but command substitutions for Windows machines have been included as needed.

💡 DO NOT MODIFY OR ADD TO THE PROVIDED EXAMPLE ‘CARS’ APP!!!! THIS IS FOR REFERENCE ONLY

# Initial Setup Steps (Getting Starter Code, creating GitHub Repository, setting up project)

Navigate to the GitHub repository for the starter code: <https://github.com/dCC-Online/ReactDjango_JWT_StarterCode>

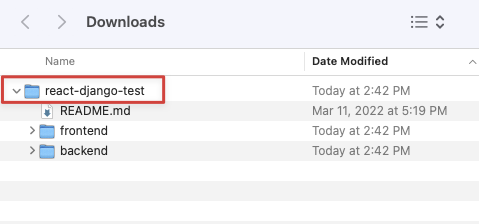
Download the starter code as a .zip file



First click the Code button, then download the ZIP file

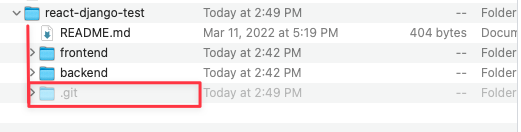
Once downloaded, **unzip into a location of your choice.** *You may re-name the project folder that is unzipped depending on what project you are working on.*

There are three items inside the main directory for the project: a backend folder, a frontend folder, and a README. Be sure your project structure looks similar (with the exception of a different name for your project).



At this point, it would be a good idea to create a GitHub repository on your own GitHub account, clone it down to your computer, and move the invisible .git folder over from your cloned repository to the parent directory of the starter code. The invisible .git folder should be on the same level as frontend, backend, and README.md once you have made that move.

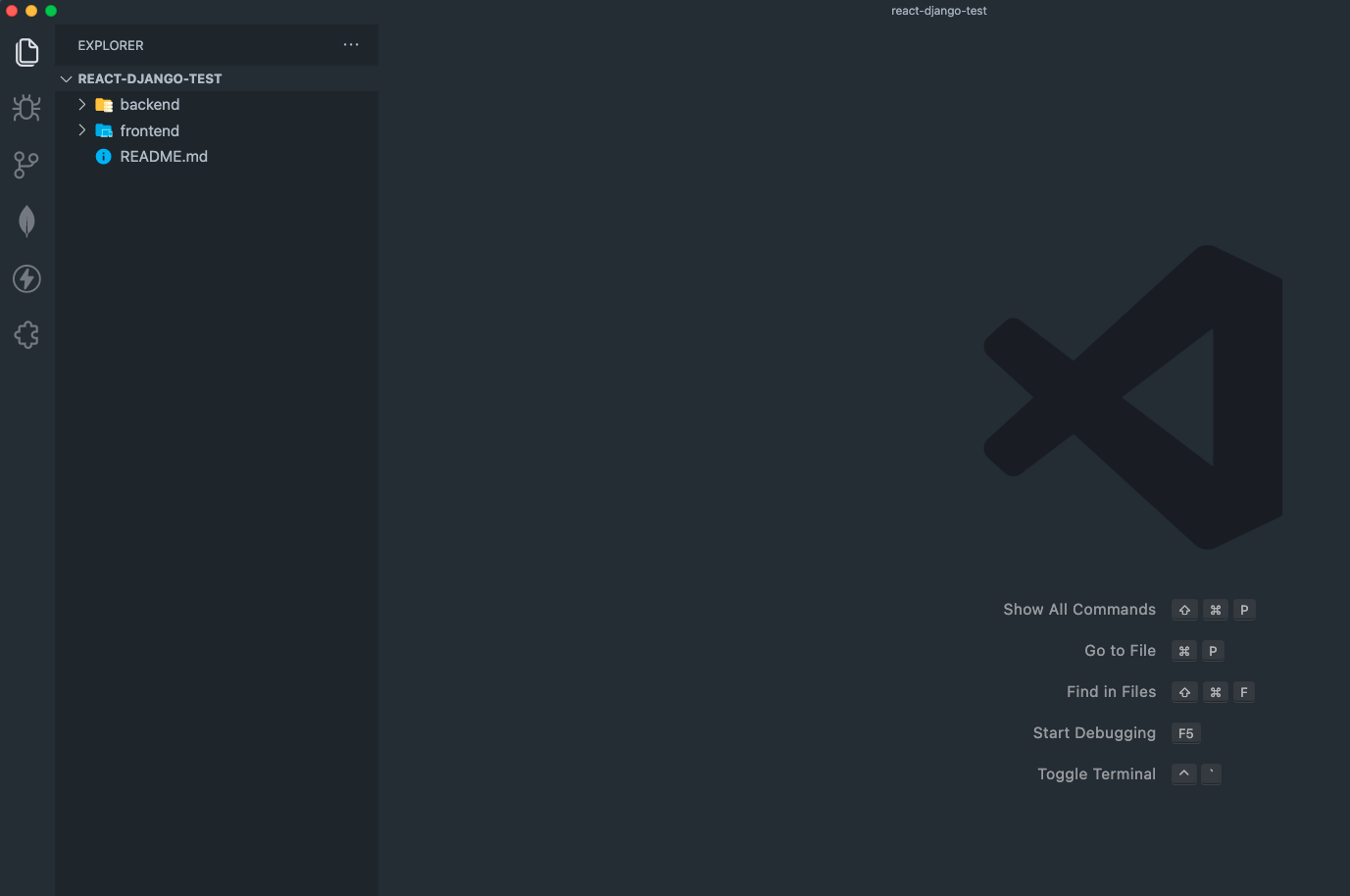
NOTE: You will not need to add a .gitignore for your new GitHub repository, both the React frontend and the Django backend each have their own .gitignore files set up already.



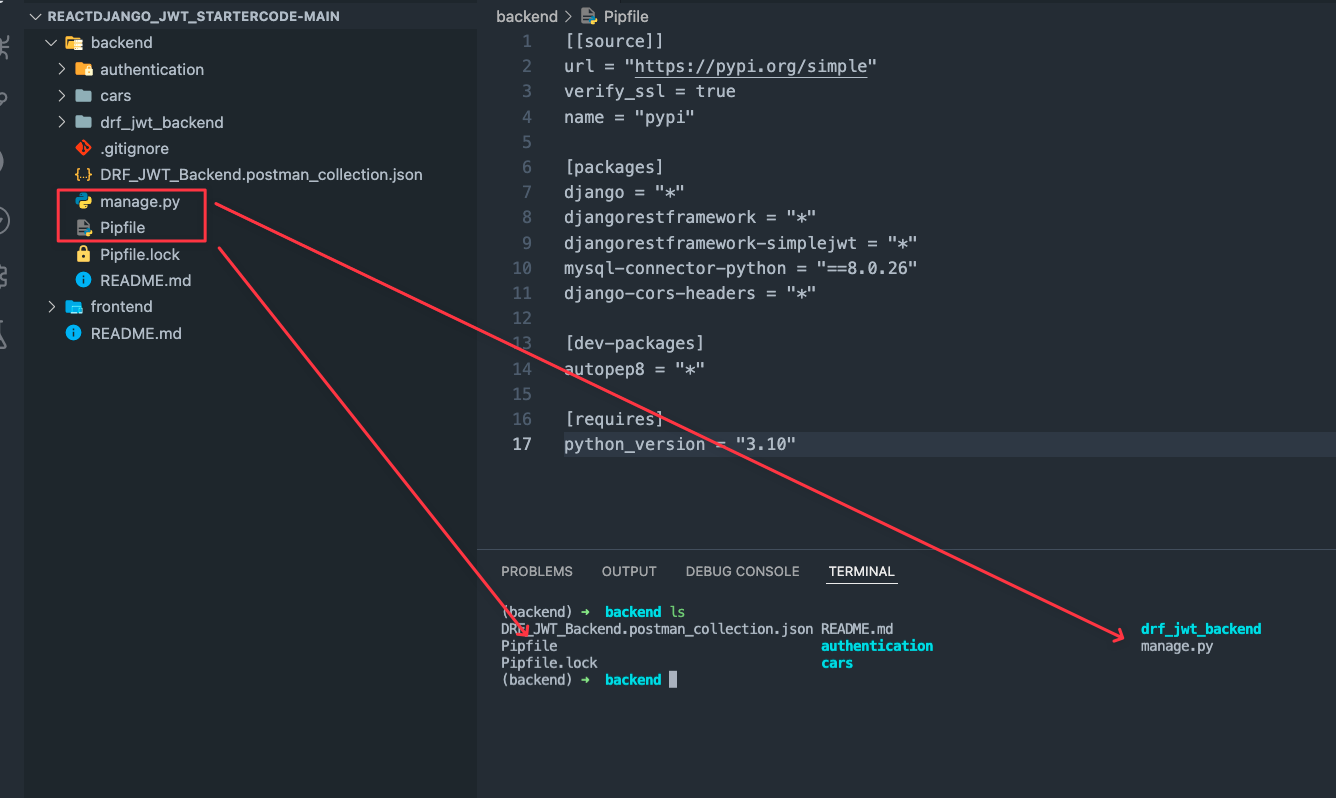
After the .git folder has been moved into the correct directory from your cloned repository

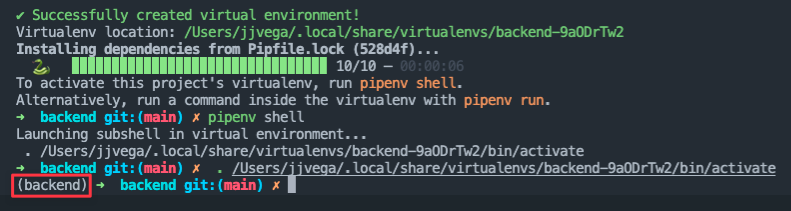
Commit the newly added files! Please refer to previous projects and the Basic Git Operations document for a refresher of that workflow.

Open your project in Visual Studio Code. Once open, it should look like this:



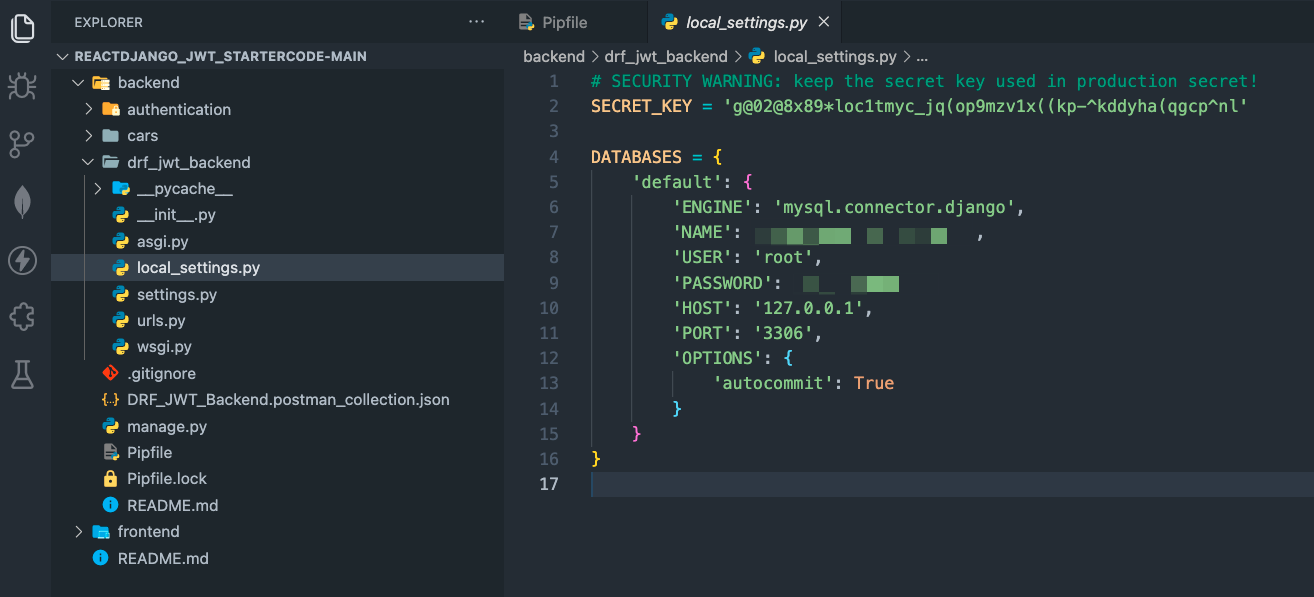
# Setting up Django Backend

1. Open the Integrated Terminal in VSCode
2. Change directory into the backend folder by running the command cd backend
3. Examine the contents of the current folder
   1. For Mac users, type the ls command
   2. For Windows users, type the dir command
4. Confirm that you see the Pipfile in your project folder. Run the pipenv install commandAfter running the ”ls” or ”dir” command, you should see both the manage.py and Pipfile in your Terminal – this is how you’ll know that you can now run any Django or pipenv commands
5. Run the pipenv shell command to activate your virtual environment. After doing so, you should see that the name of your virtual environment shows up behind the directory on the current Terminal line.
   1. Pay close attention to where you are now in the Terminal! You may need to change directory (cd) into the proper folder to run your manage.py commands



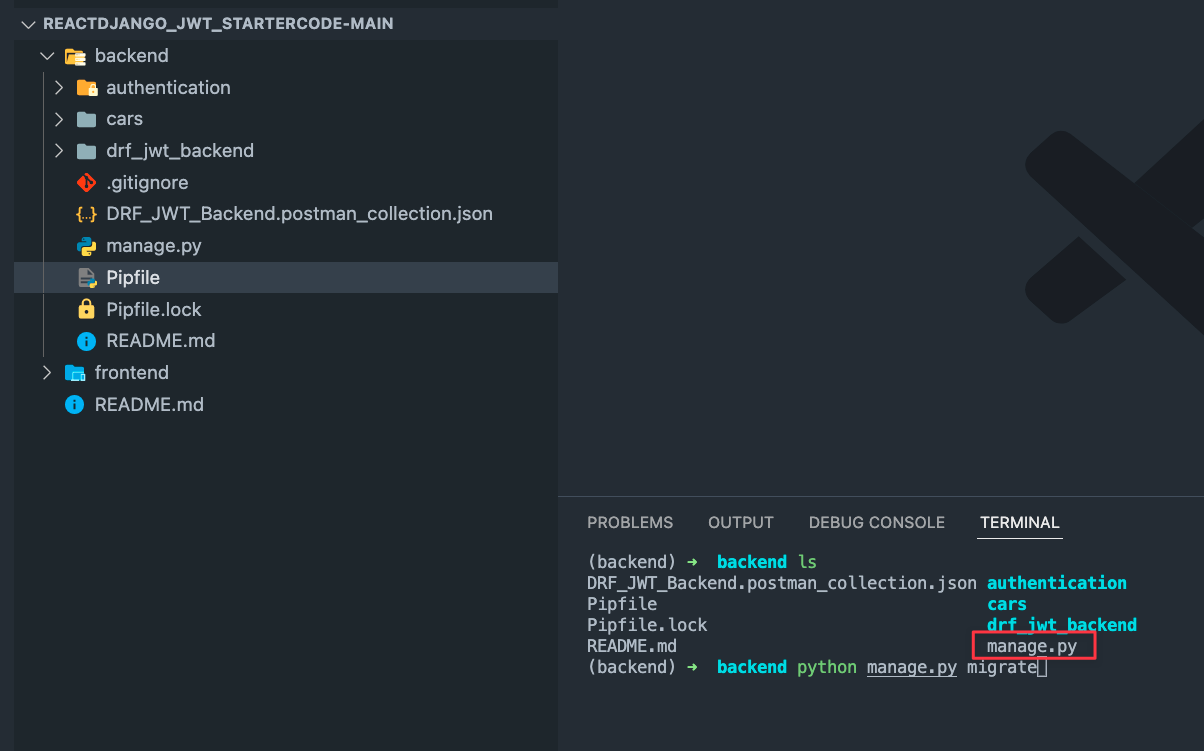
With the virtual environment activated

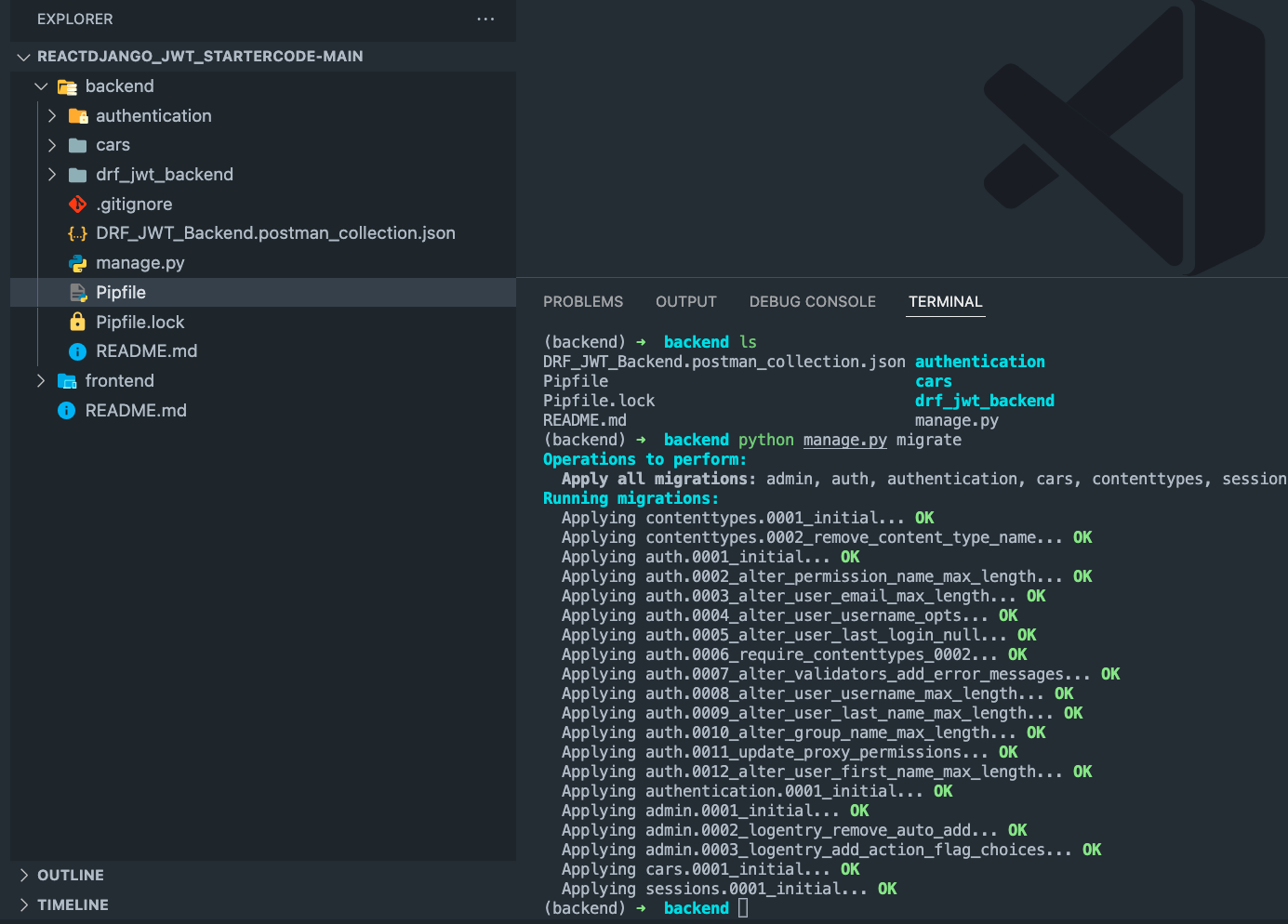
1. **NOTE:** You may need to reload your Visual Studio Code and select the correct interpreter before proceeding
   1. CTRL/CMD + SHIFT + P to open the Command Pallette
   2. Search for “Developer: Reload Window”, highlight, and press enter
   3. Once reloaded, open the Command Pallette again and choose “Python: Select Interpreter”, press enter
   4. Search for the correct virtual environment by name and either click it or press enter to select
      1. To find out which one to select, you can run the Terminal command pipenv --venv
2. Create a local\_settings.py file inside the <project-name>/backend/drf\_jwt\_backend/directory.
   1. Add the SECRET\_KEY and DATABASES dictionary to it. Be sure to fill in the name of your MySQL database and the password you use for your MySQL connection.
      1. The SECRET\_KEY can be any combination of characters and numbers, so long as they are complex and at least 32 characters long (see screenshot below).
      2. You will find examples of these values from past projects!
   2. If you have not already, make sure to run the CREATE DATABASE command in your MySQL Workbench for the database you are creating for the Django project.



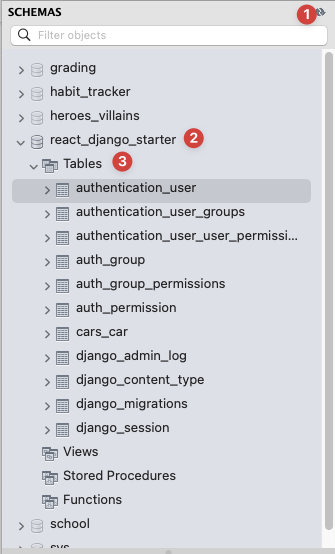
With local\_settings.py in the correct location. Note that we need to replace the ‘NAME’ and ‘PASSWORD’ values according to how our database is set up

1. Now, it’s time to migrate the current Django models to your MySQL Database. Be sure to double check that you are in the same directory as your manage.py file and run the command python manage.py migrate

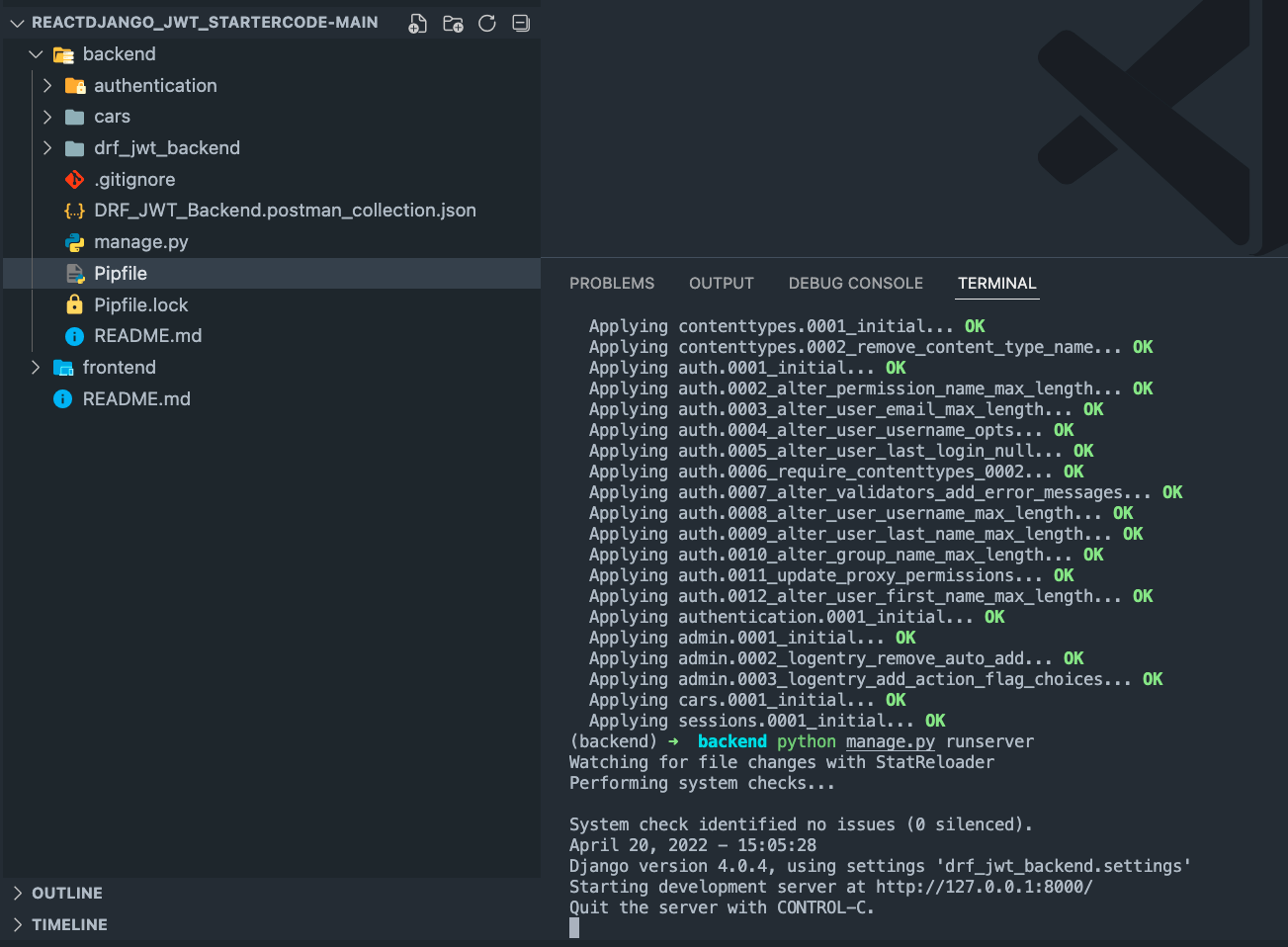


Success!

1. Verify that your tables have migrated to your MySQL database in MySQL Workbench (don’t forget to refresh your schemas!)



1. Create a superuser for your project. Make sure you are still in the same directory as your manage.py file – python manage.py createsuperuser
2. Now, it’s time to run our server! Without changing your current directory from the last command – python manage.py runserver

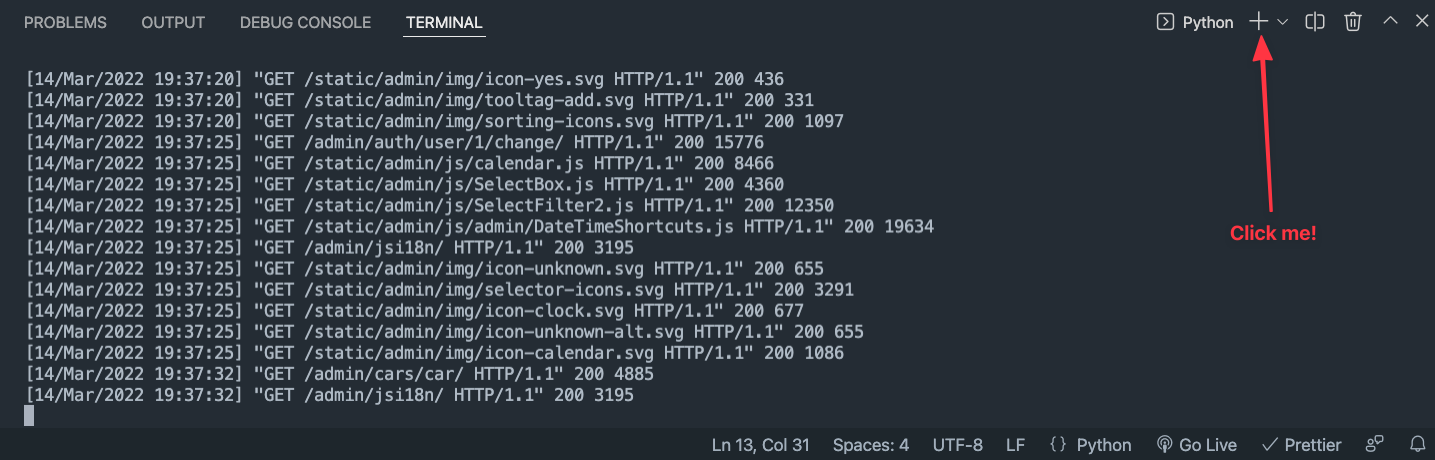


Server is running successfully!

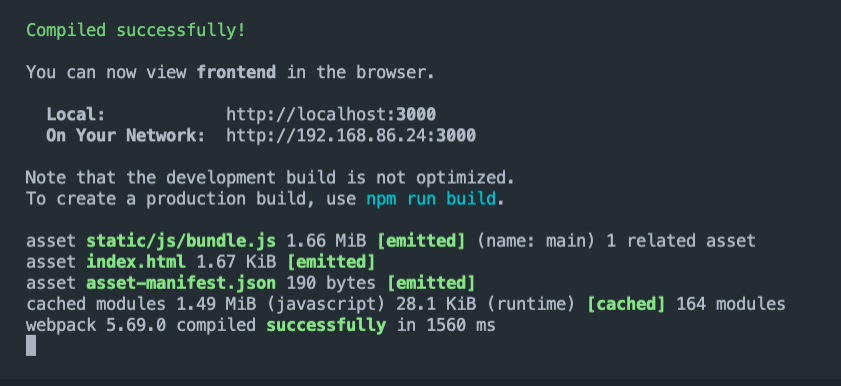
1. If we have successfully run our server, your backend is set up! This would be a great time to use your Admin console to set up a few users and seed data for other models if needed by entering the following into your browser while the server is running: <http://127.0.0.1:8000/admin/>
2. We recommend that you make a GitHub commit now – if you already committed the initial files, then there should be nothing to commit. If there is, then the local\_settings.py file was NOT set up correctly to be ignored by Git.

# Setting Up React Frontend

1. Create a new Instance of a Terminal in Visual Studio Code (DO NOT SHUT DOWN THE RUNNING INSTANCE FOR YOUR BACKEND!)

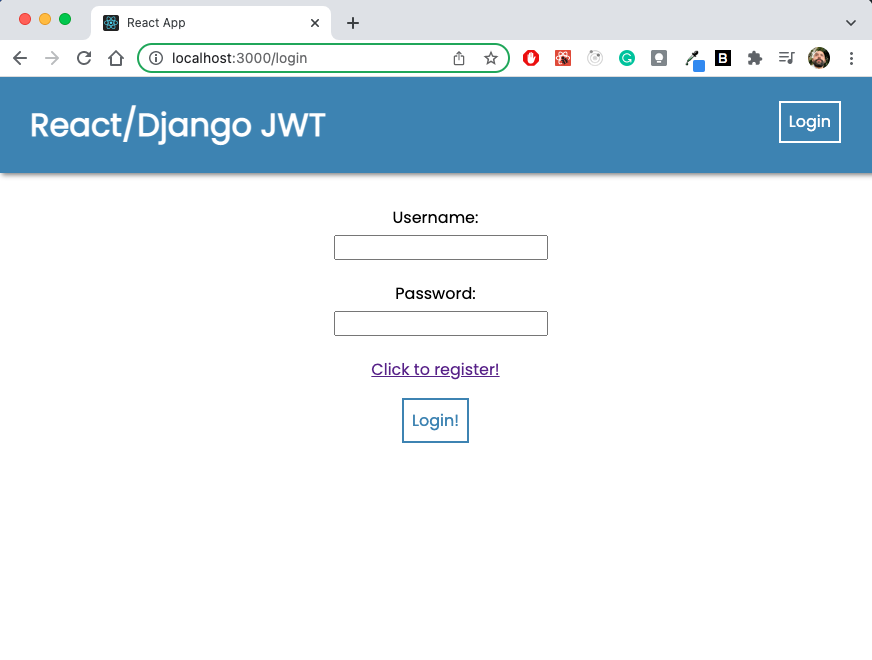


1. Your new Terminal instance will start at the parent directory of the project. Change directory into the frontend directory.
2. Run the npm install command to install React dependencies
3. Run the npm start command to start your React application!

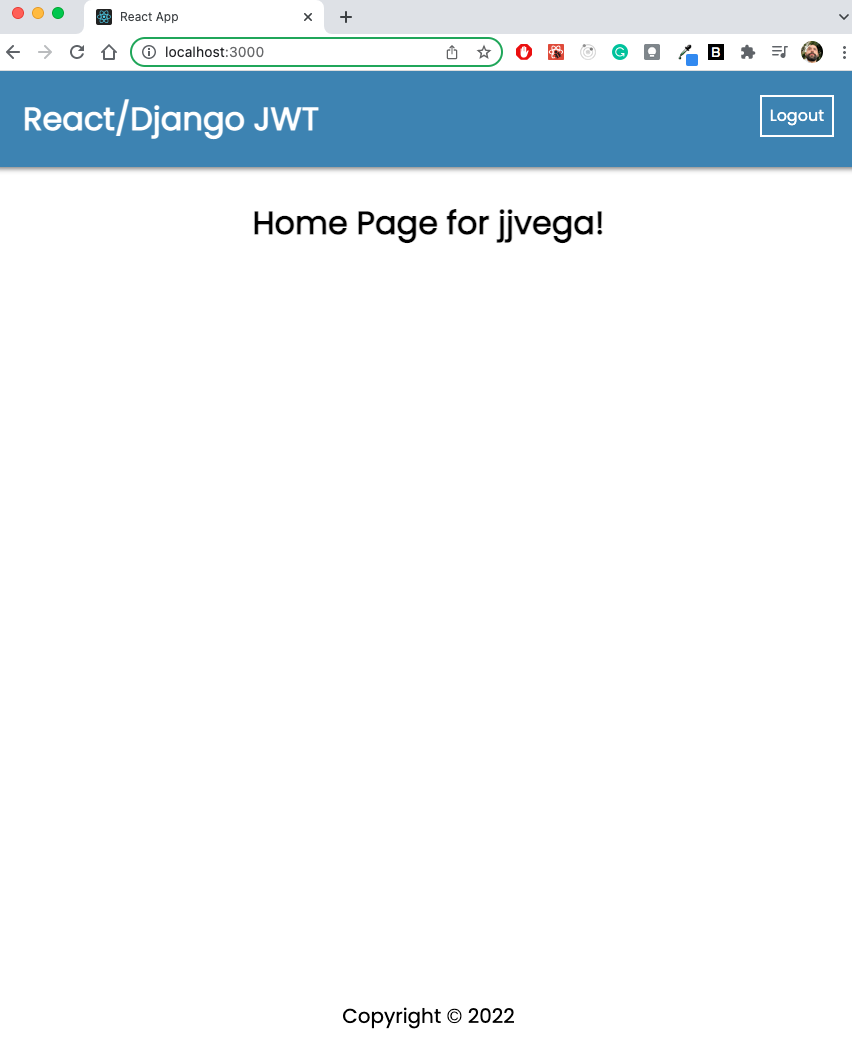


The React server running successfully

1. Your React application should open in your browser and automatically route to the Login screen.



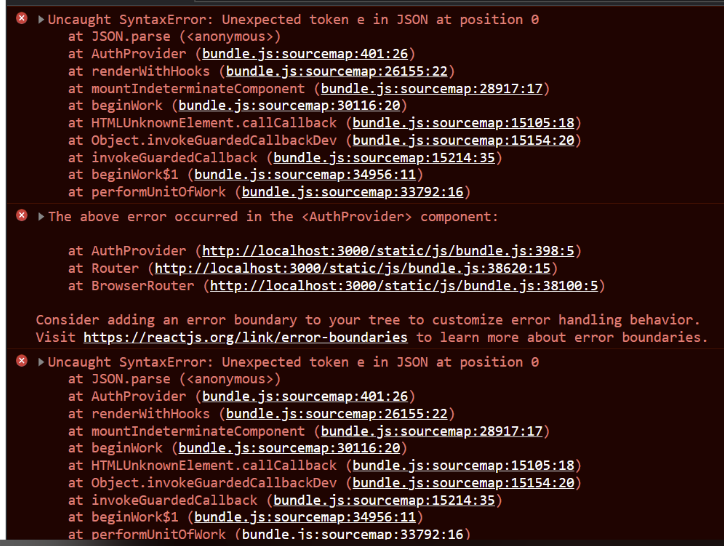
1. From here, enter the username and password you entered for the superuser you created when setting up your Django backend. If you logged in successfully, you should see a Home Page screen like this!



The logged in state is using the user’s username property to show the heading

NOTE: As you use the code, remember that you need to be in the frontend directory to use any React commands, and in the backend/drf\_jwt\_backend directory to use any Django commands.

NOTE: If you see the below message, make sure your Application’s Local Storage is clear! This article links to how to do this in Google Chrome, and will work similarly in other browsers: <https://www.leadshook.com/help/how-to-clear-local-storage-in-google-chrome-browser/>



Congratulations! Your starter code is now set up. Please see the additional video resources for a more in-depth look at how it is set up, and how to utilize JWT to make requests to protected endpoints from your React application!