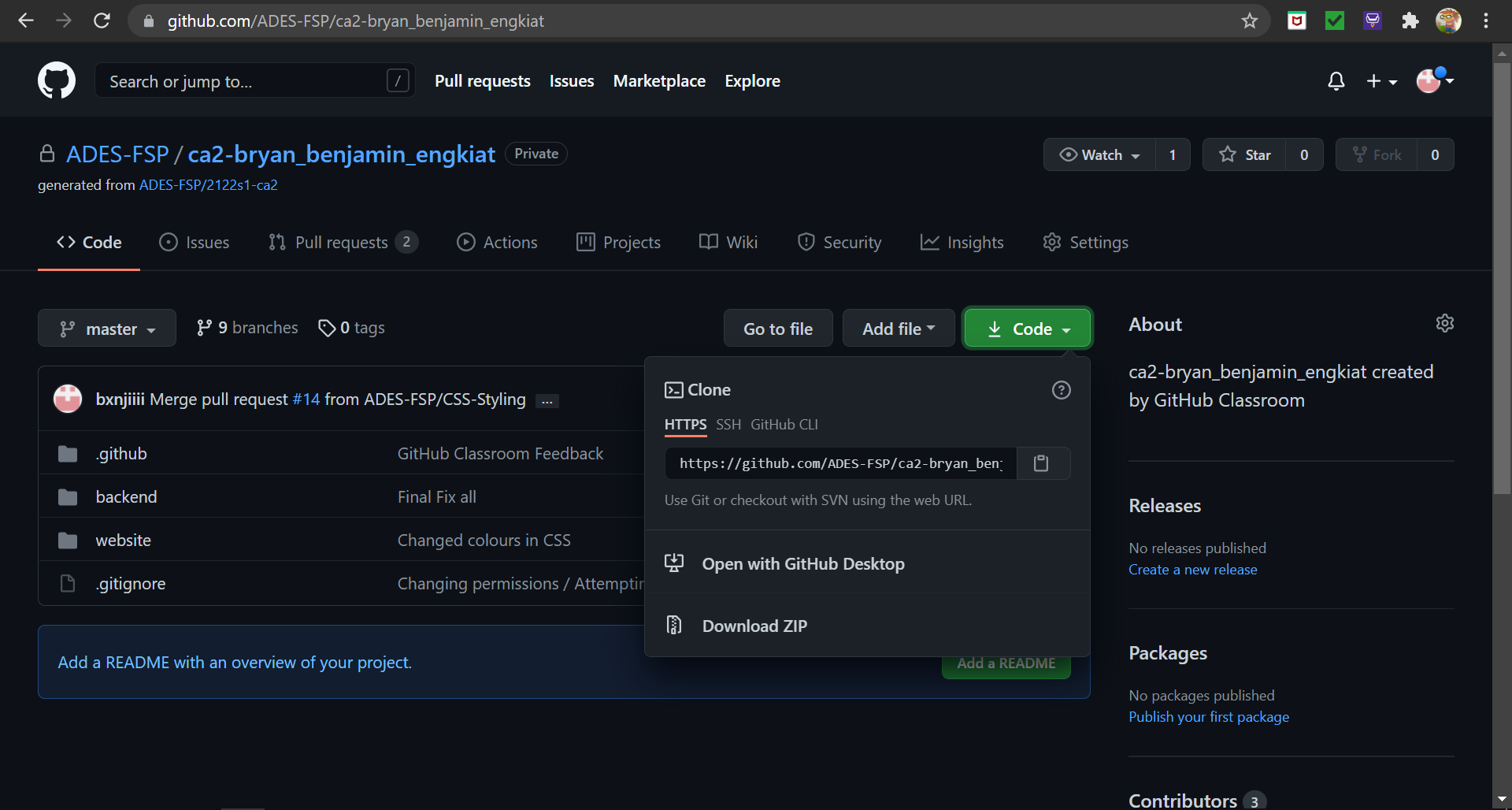
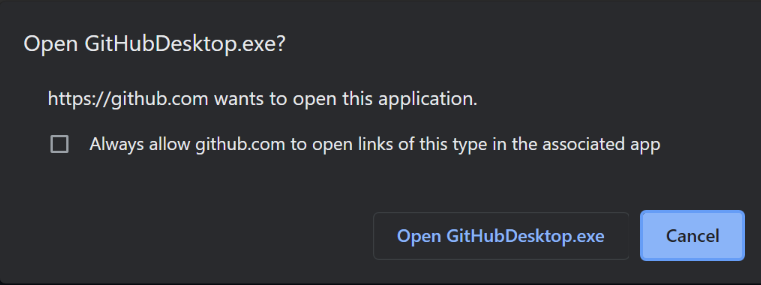
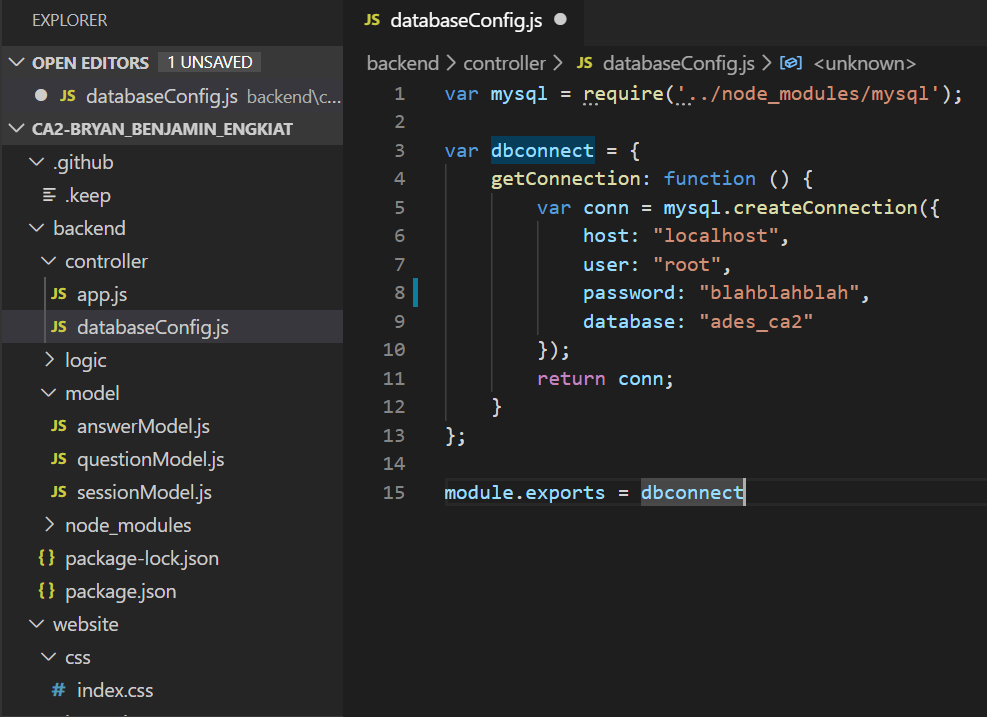
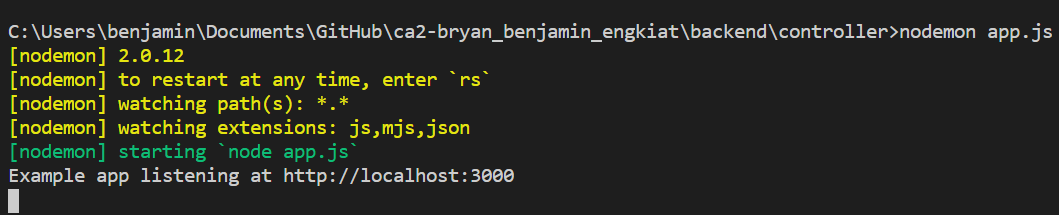
**Setup Guide**

***Instructions:***

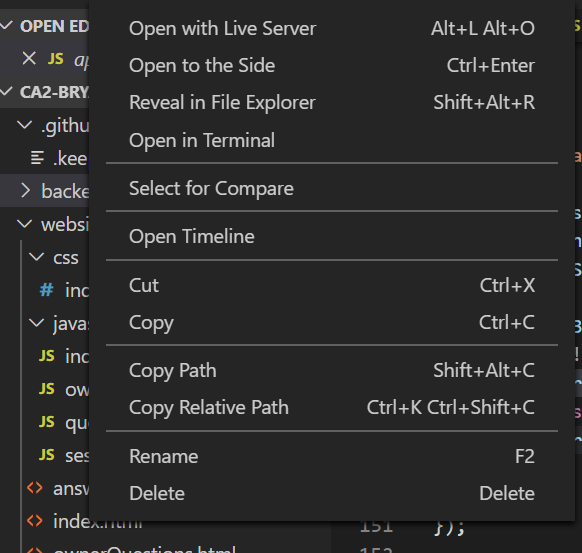
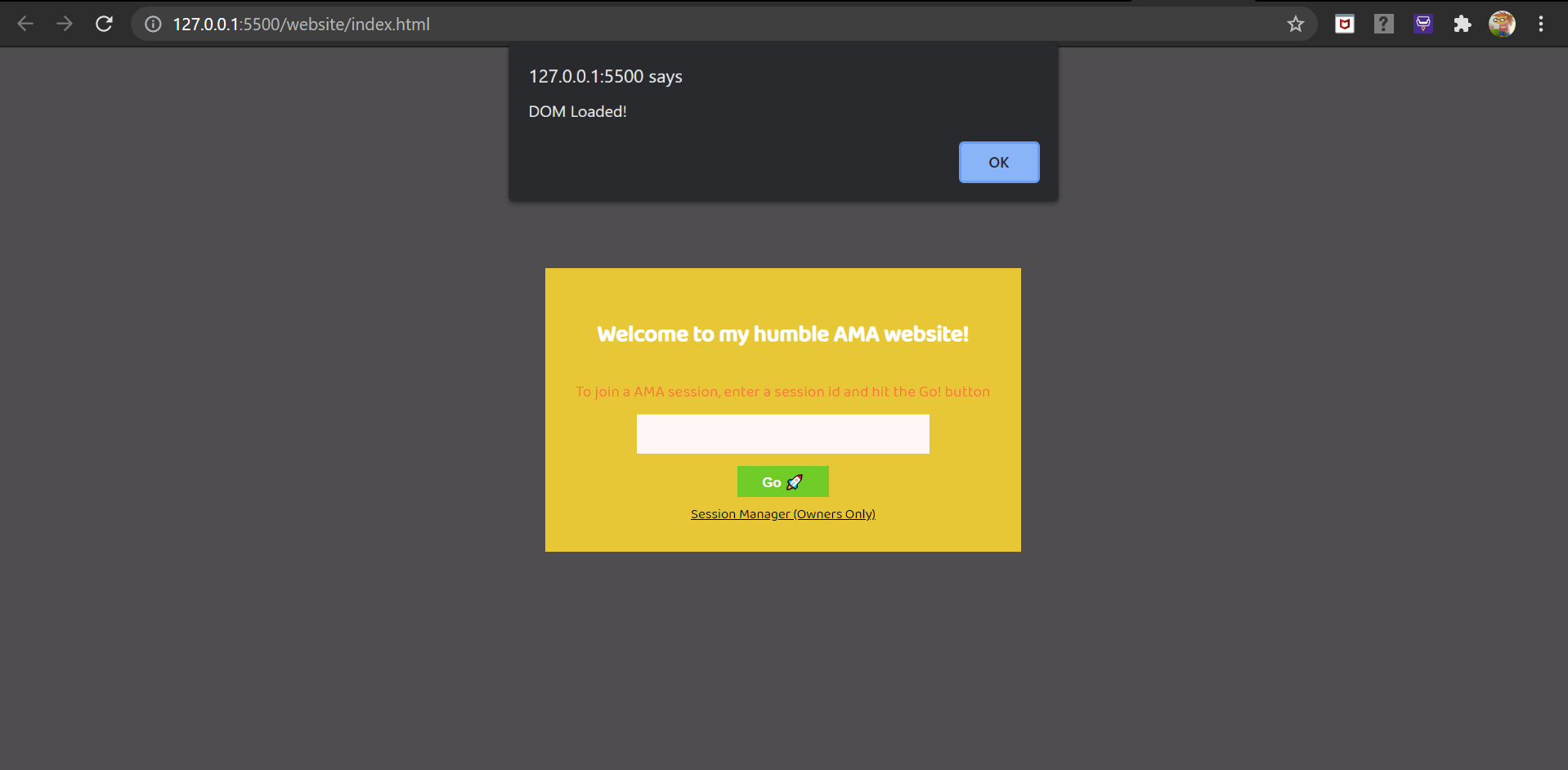
Go to <https://github.com/ADES-FSP/ca2-bryan_benjamin_engkiat>

1. Click on “Code” and select “Open with GitHub Desktop”  
   
2. A pop up will appear and select “Open GitHubDesktop.exe”

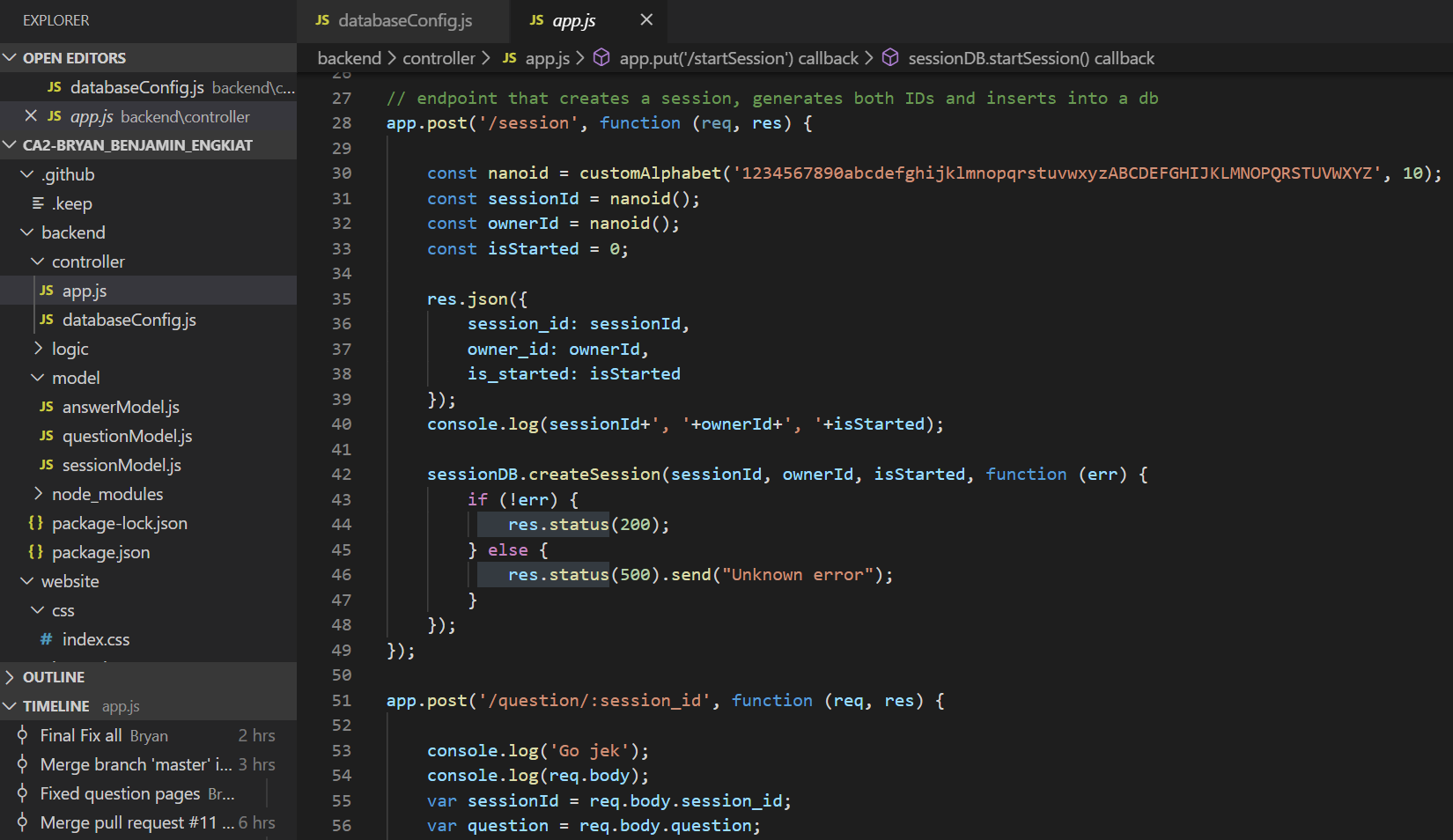


1. Clone the repository and select “Open in Visual Studio Code”
2. Open a terminal in the backend folder and install:   
   a) Express (<https://www.npmjs.com/package/express>)  
   b) nanoid (<https://www.npmjs.com/package/nanoid>)   
   c) cors (<https://www.npmjs.com/package/cors>)   
   d) mysql (<https://www.npmjs.com/package/mysql>)   
   e) nodemon (<https://www.npmjs.com/package/nodemon>)
3. Double check your package.json file to ensure that they have been installed
4. Open backend/controller/databaseConfig.js and change the password to yours  
   
5. Open the terminal in the controller folder and type nodemon app.js to run the backend  
   

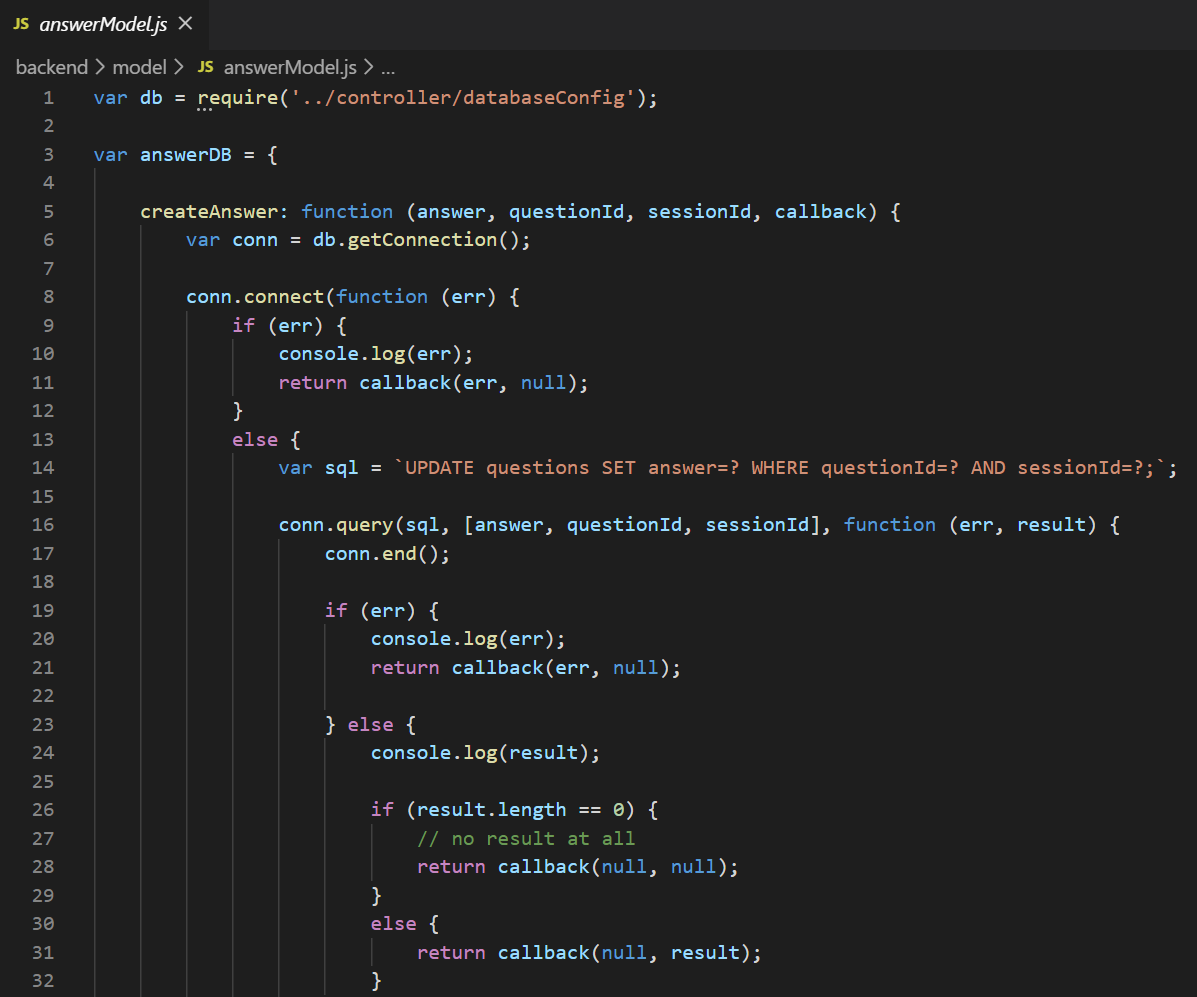
**This should pop up to show that your backend is running correctly.**

1. Right click on index.html and click “Open with Live Server”  
   
2. **You should be successfully redirected to the website**  
   

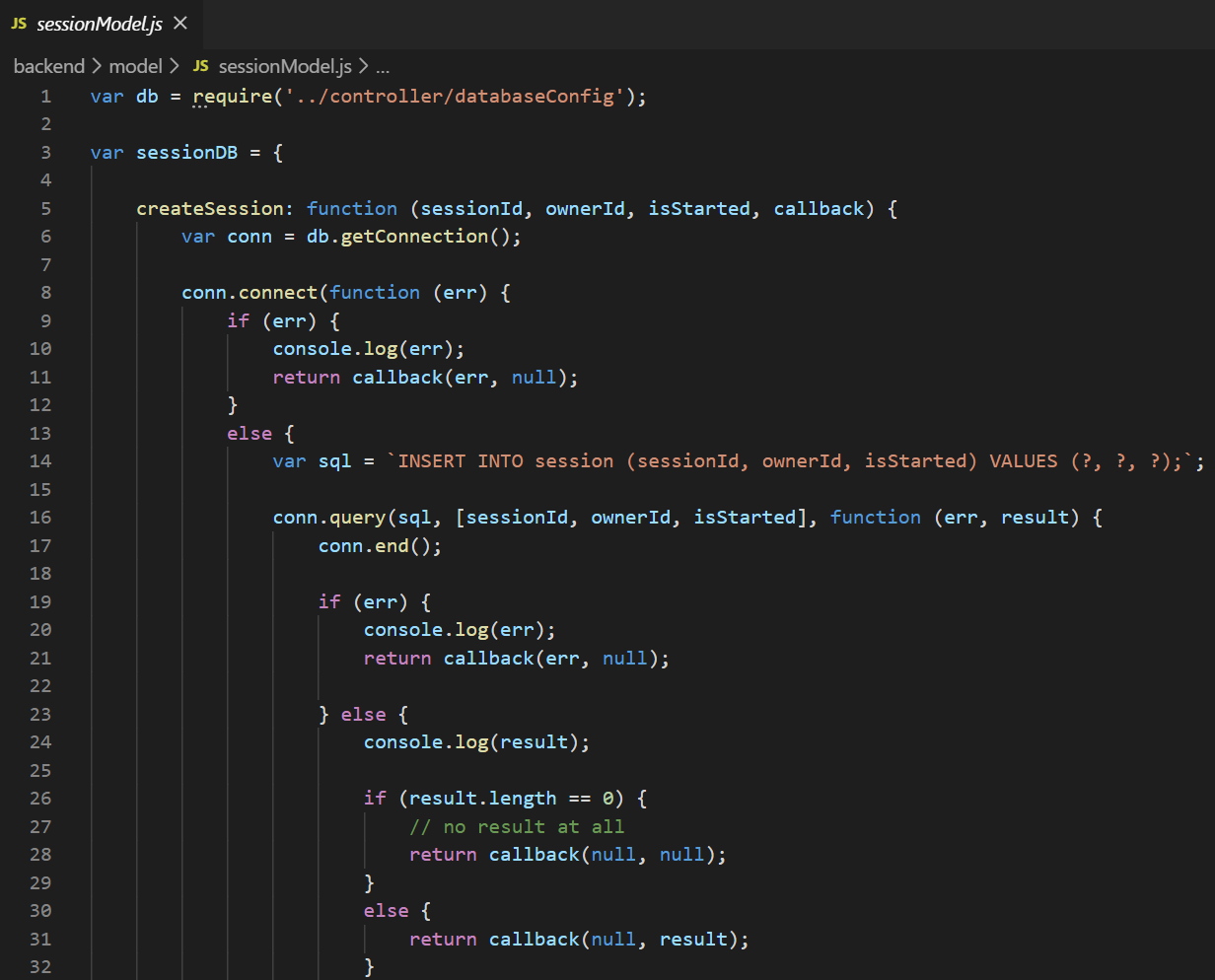
**Basic Code Organization**

All the APIs are in backend/controller/app.js  


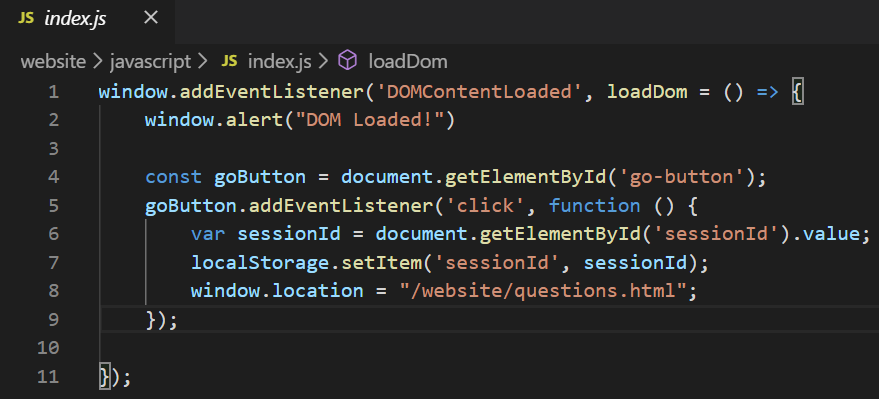
The backend application logic is separated into 3 models in the model folder

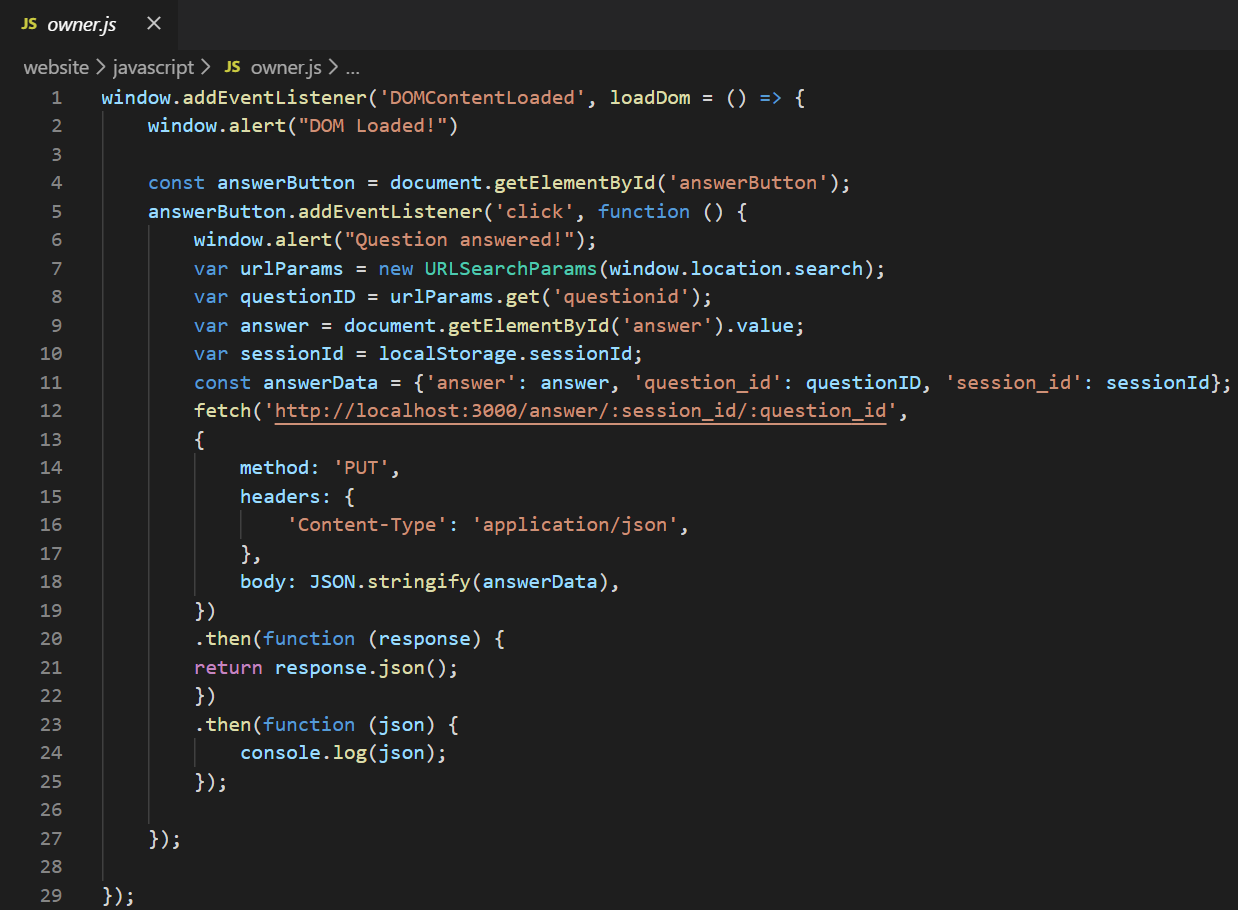
Backend application logic for answering questions is in answerModel.js  


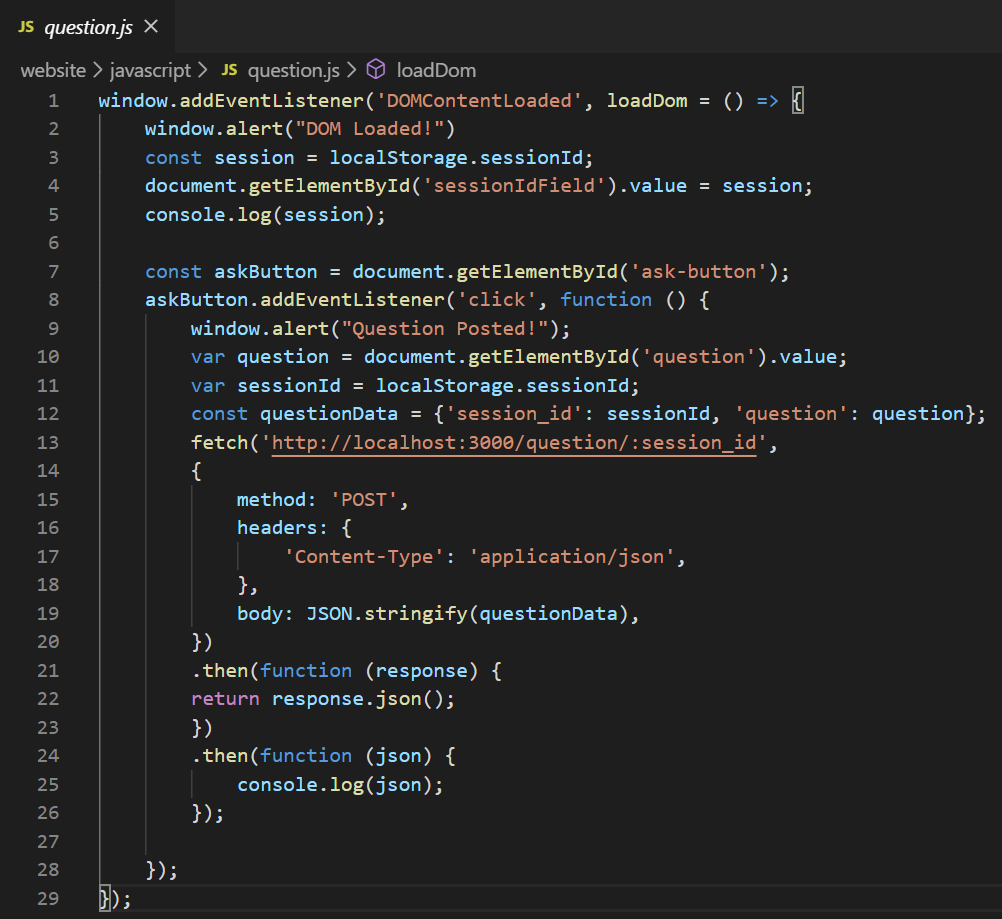
Backend application logic relating to questions is in questionModels.js  


Backend application logic relating to sessions is in sessionModel.js  


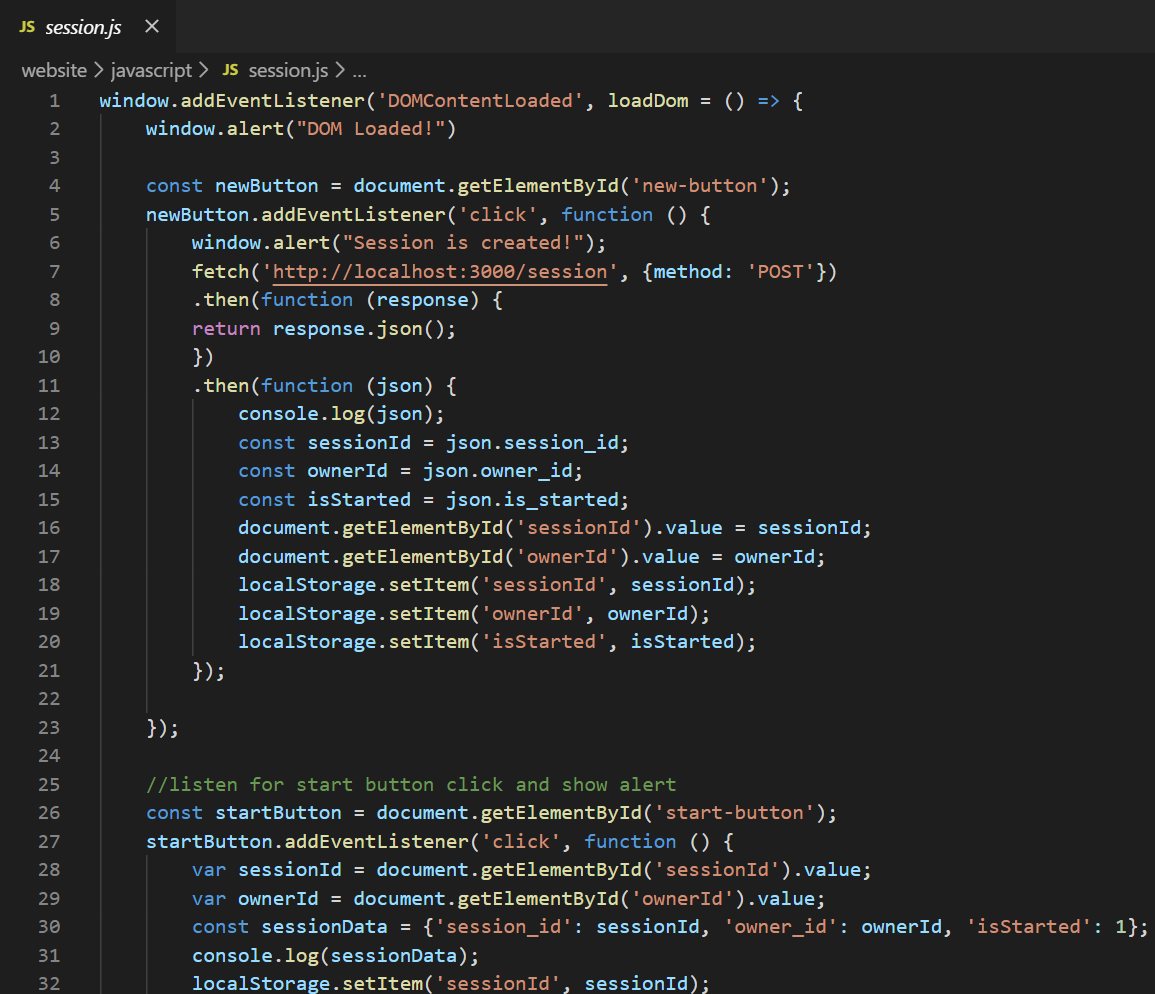
The website sends HTTP requests through 4 javascripts files:

**Index.js** handles going to the questions page of the specified session as a regular user  


**Owner.js** handles the owner answering a question  


**Question.js** handles posting a question  


**Session.js** handles posting, starting, stopping a session and going to the questions page as a owner



**Database Setup**

MySQL Setup and installation

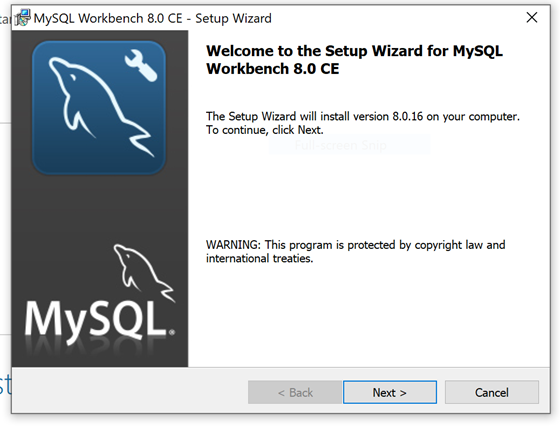
We will be installing the following:

1. MySQLWorkbench (<https://dev.mysql.com/downloads/workbench/>)

2. MySQL (<https://dev.mysql.com/downloads/installer/>)

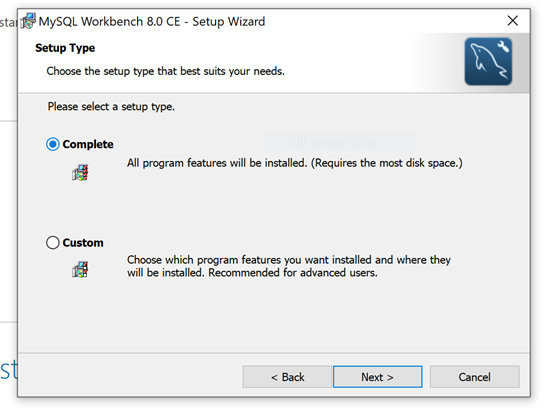
Installing MySQLWorkbench

Download and open the file named “mysql-workbench-community-8.0.16-winx64”. You should see the following:



Click Next and use the default configurations for the rest of the setup.

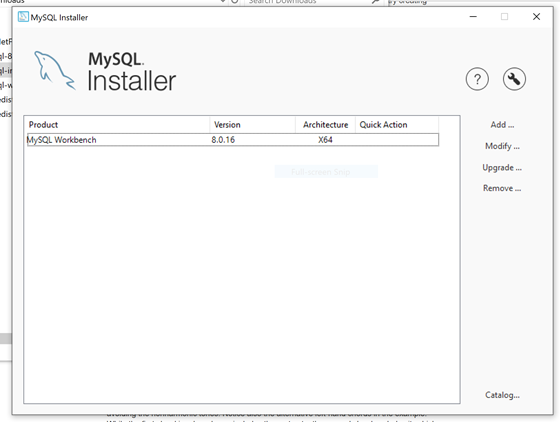
Ensure that the setup type is Complete instead of Custom:



Once the installation is done, close the setup wizard.

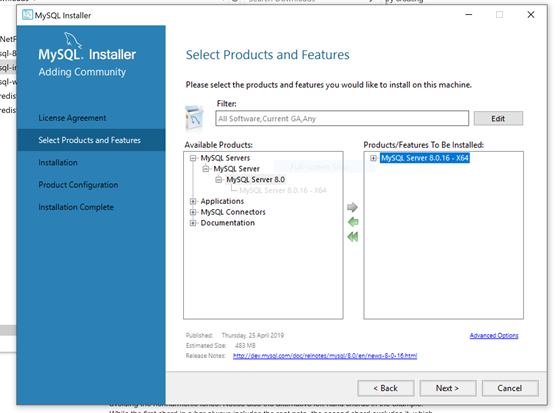
**Installing MySQL**

Download and open the file named “mysql-installer-web-community-8.0.16.0”. You should see the following:

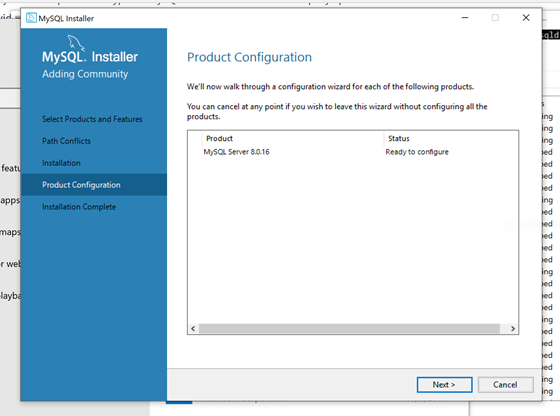


Click on Add…\*. Accept the license agreement.

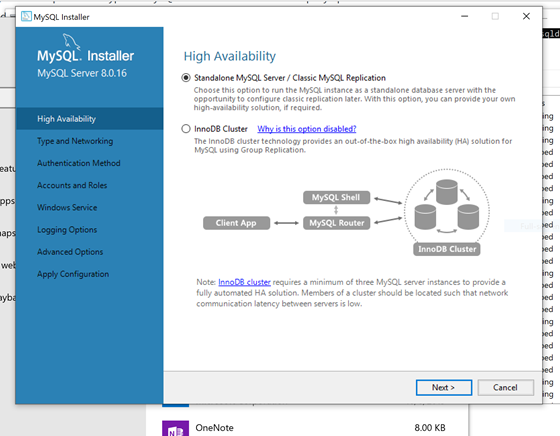
In the Select Products and Features, just the green right arrow to move MySQL Server 8.0 to the “Products/Features To Be Installed” section:



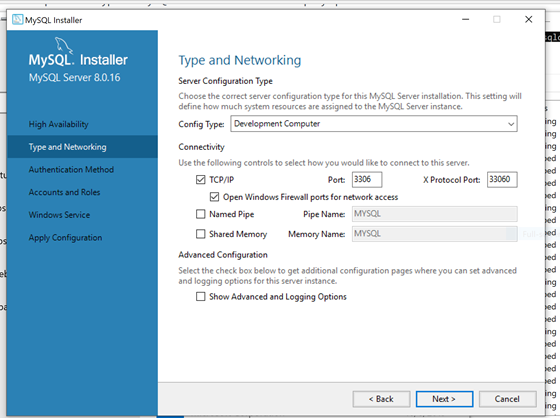
Click on Next to continue. The MySQL 8 installer will be downloaded. Follow the default configuration until Product Configuration:



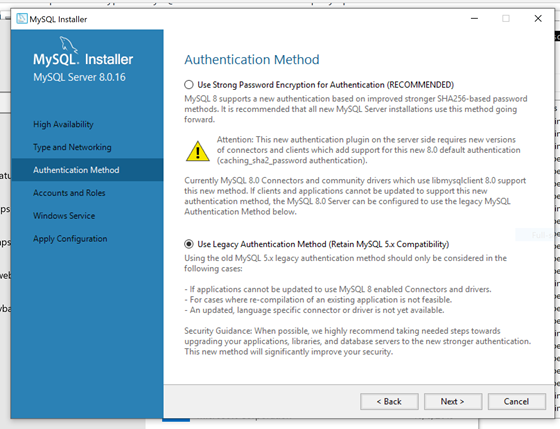
Click on Next.



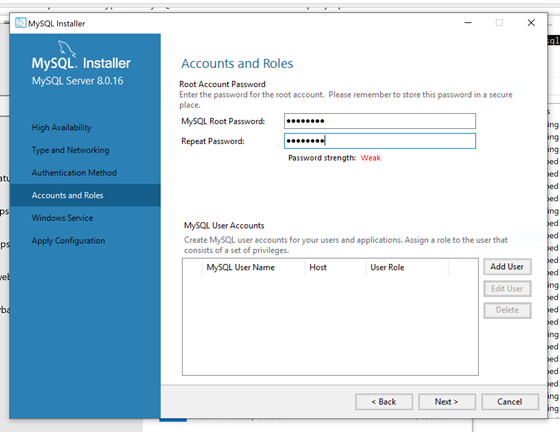
Ensure that the Standalone MySQL Server option is selected and click on Next.



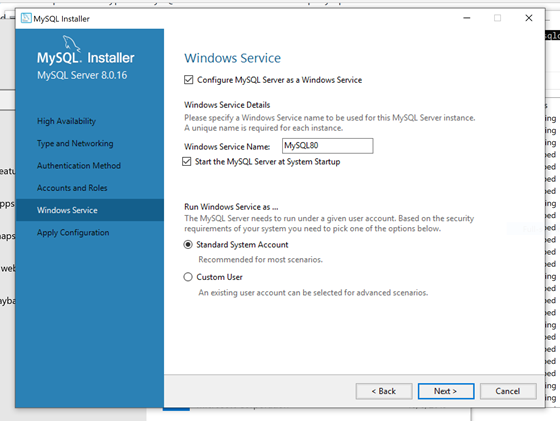
Go with the default configuration for Type and Networking and click on Next.



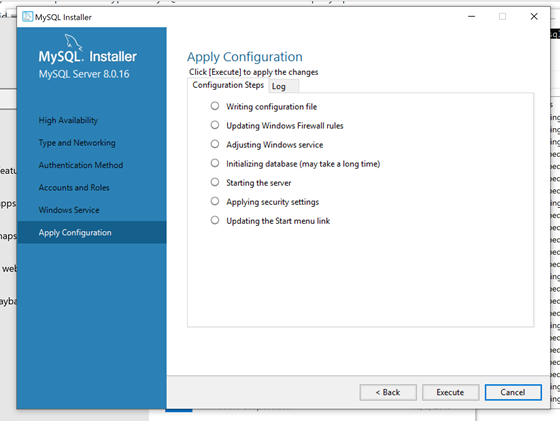
Select the legacy authentication method for Authentication Method and click on Next. We are using the legacy authentication because at the time this document is written, the new authentication method is still not supported by the default MySQL library for Node.js.



Make sure to type in a secure password and remember it! We will need the password to connect to the database.



Under Window Service, go with the default configuration and click on Next.

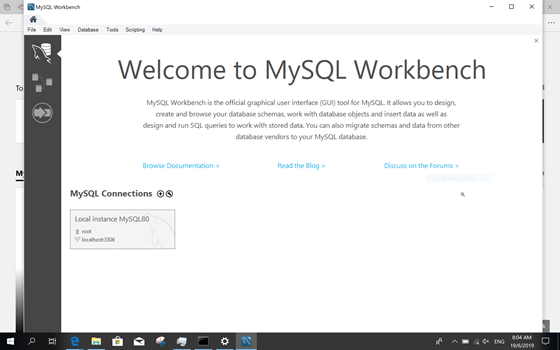


Click on Execute. Once it is done, click on Finish.

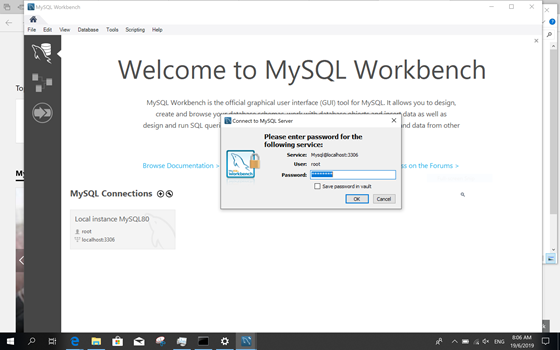
Connecting to the database from MySQLWorkbench

We will now connect to the MySQL database server we just installed.

Open MySQLWorkbench. You should see the following:

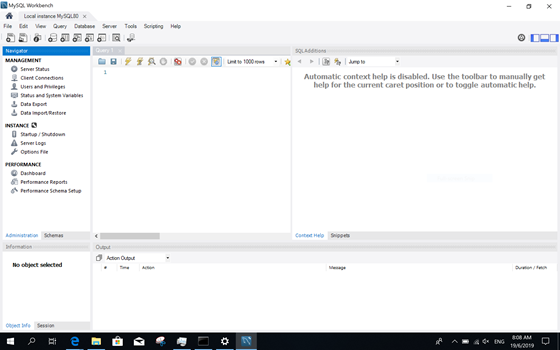


Click on the Local instance MYSQL80 connection. You will be prompted to enter a password:



Enter the password you previously set for the MySQL server and click on OK.

Once the login is successful you will be brought here:

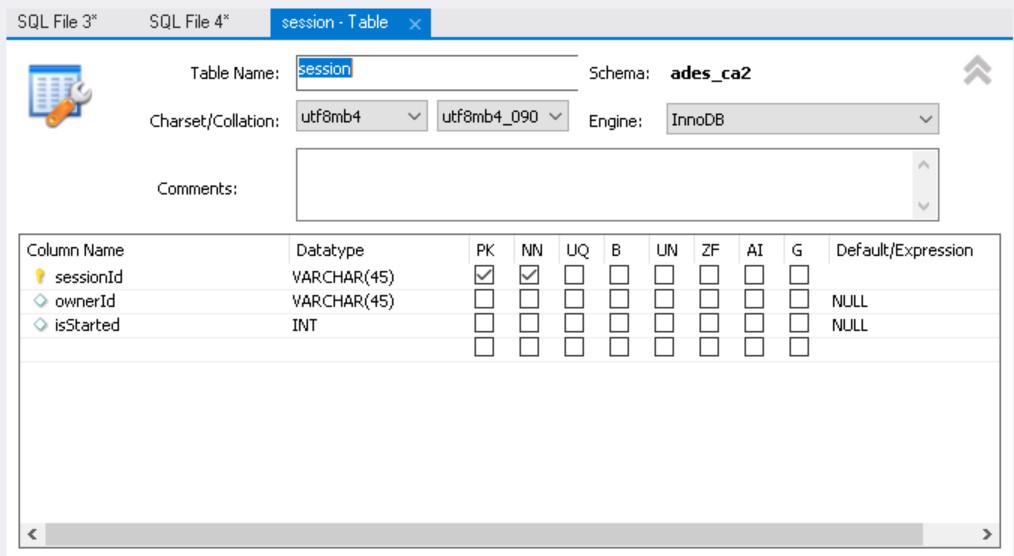


We have now successfully connected to our database server from the workbench.

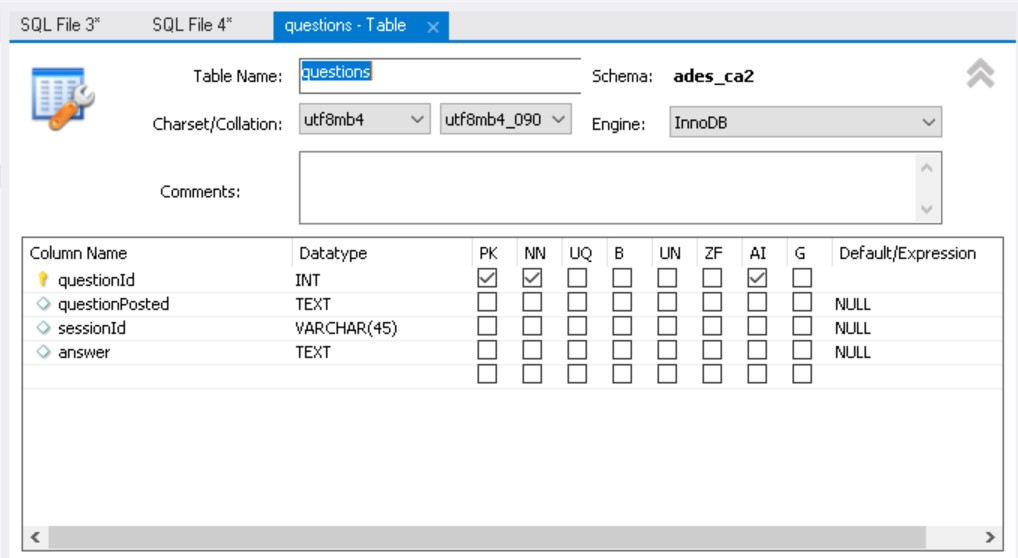
**Create Schema and Tables**

Create a schema and give it any name you want.

Create a table called session:



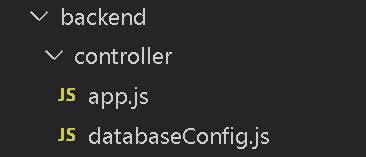
Create table called questions:



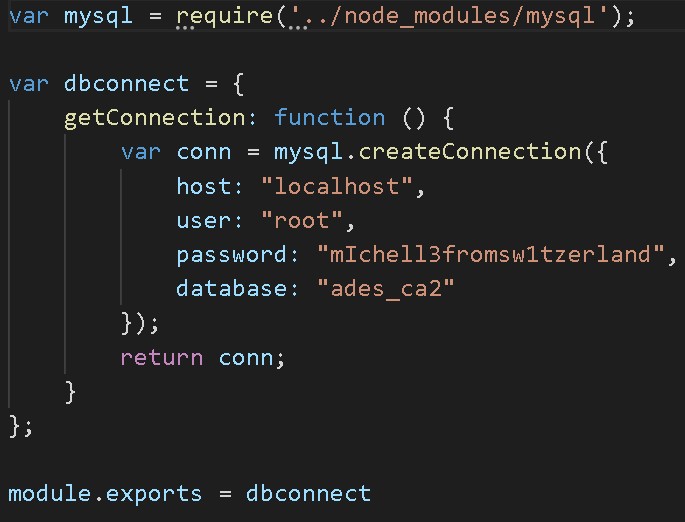
Once created, you will have a database to link to the backend portion of our codes.

**Link to backend**

To link the backend to the database, go to the databaseConfig.js file located in the backend controller folder in our project.



Click into the file and change the host, user, password and database fields that you used for your database.





Now you can test our project.