**Demo:**

1. Walk the class through installing Node.js
2. Create a project folder and open it in VS Code.
3. Open a terminal in VS Code.
4. Create a package.json using npm
5. Show the package.json
6. Install express as a dependency.
7. Show the updated package.json
8. Create a modules folder and create a file inside called settings.js
9. Have it export the port number for the server to listen on. (80)
10. Create a file called app.js
11. Require express in the app.js
12. Require settings.js in the app.js
13. Create the “app” variable and set the server to listen on the port from the settings.js module. (Make sure they have MAMP turned off)
14. Create a route for the home page.
15. Create a folder called views.
16. Create an index.html inside the views folder with an h1 that says home. (Tell them about using “!” for boilerplate)
17. Send index.html as the response in their route.
18. Create a route using a route parameter that will return the parameter value as a response. (response.send(request.params.id))
19. Show them that when the value of the route parameter changes so does the value on screen.
20. Create a folder called assets and add a stylesheet to it.
21. Create a rule in the stylesheet that will change the color of the h1 on the index.html
22. Link the stylesheet to the index.html page and show them what happens when trying to do this without static routing setup.
23. Setup static routing for the assets folder.
24. Explain that the string in app.use is the route and that the path in express.static is the folder path.
25. Show that the styles are now applied.
26. Create a form.html page containing a form with First Name, and Last Name fields.
27. Install body-parser as a dependency. (Remind them that this will now be added to their package.json)
28. Create a get route for accessing the form.html page.
29. Require body-parser.
30. Create a urlEncodedParser.
31. Create a post route that will return the response.body object as a response and log the first and last names to the console individually.
32. Install EJS as a dependency.
33. Install the EJS Language Support extension for VS Code.
34. Set the view engine to ejs.
35. Create a file in the views folder called index.ejs using the code from index.html
36. Remove the linked stylesheet and replace it with bootstrap cdn tags.
37. Add the defer attribute to all the bootstrap scripts and make sure they are placed in the head.
38. Comment out the route that returns index.html and have it render index.ejs instead.
39. Pass a message to the index view and output it in an h3 tag.
40. Create a folder called partials and create a file inside called nav.ejs
41. Copy a bootstrap navbar and place it inside the nav.ejs file.
42. Get rid of any unnecessary links and create links for the home and form pages.
43. Include the nav.ejs in the index view.