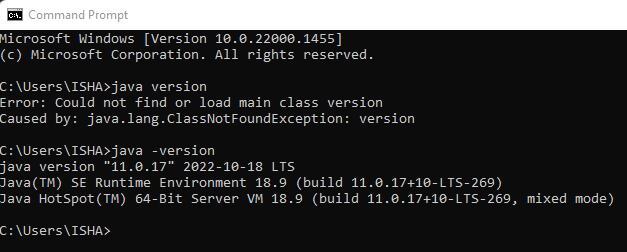
**NodeJS-MongoDB Signup APPLICATION**

***Isha Choudhary |*** [***shahooda637@gmail.com***](mailto:shahooda637@gmail.com)

***Github Link |*** [***https://github.com/Sha-98/MongoDb-Basic-Login***](https://github.com/Sha-98/MongoDb-Basic-Login)

1. Download Java and check its version in cmd to confirm the download.



1. Download java development kit from Oracle and install it. Add the JAVA\_HOME path in the environment variables.

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application

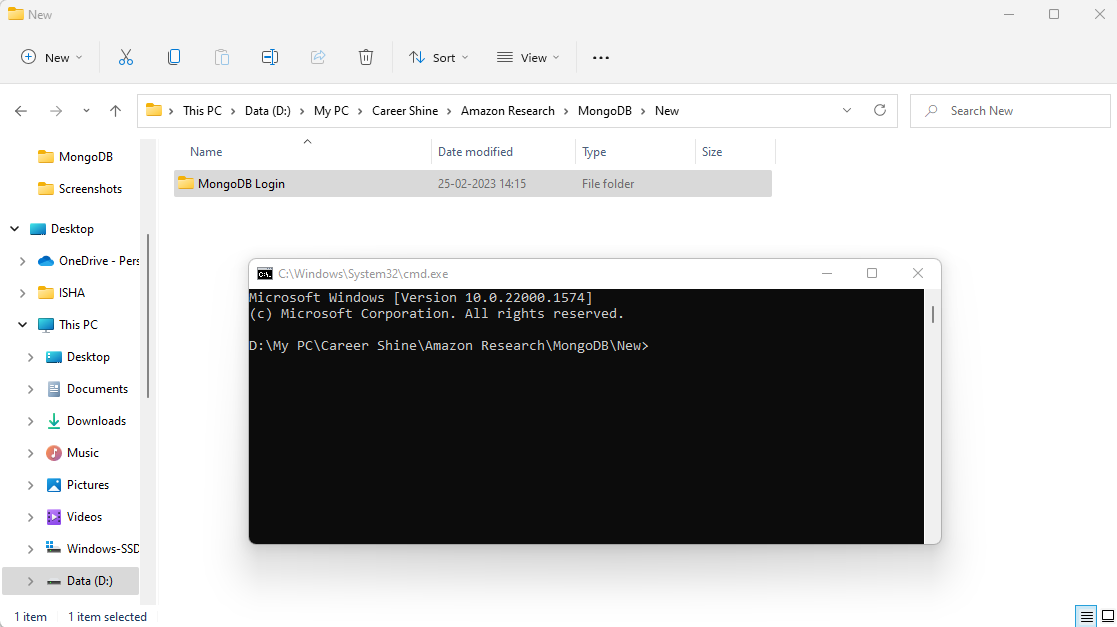
Description automatically generated

1. Set up java development in VS Code by downloading

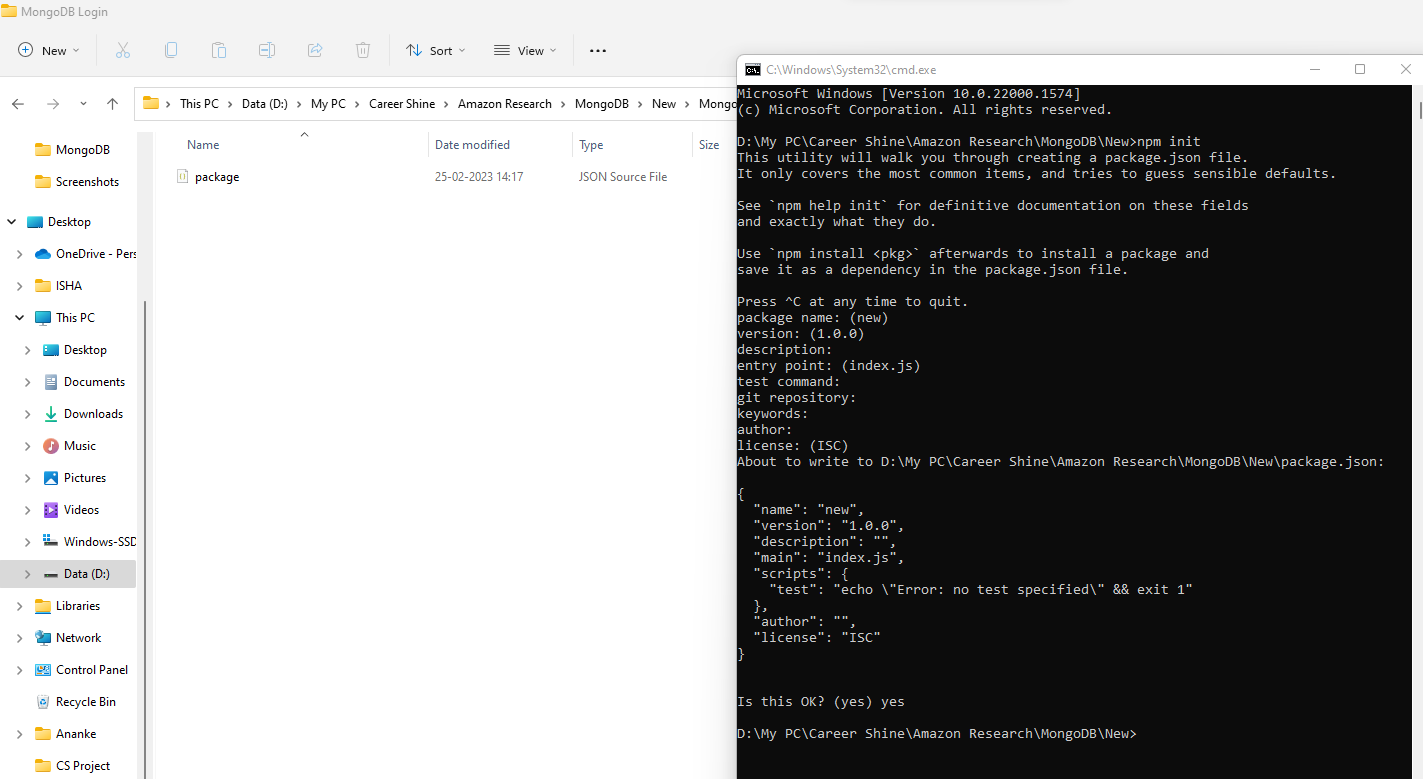
Graphical user interface, text

Description automatically generated

1. Create a new folder for your application. Open cmd on this path.



1. Create “***package.json***” file by typing “***npm init***” in cmd. Keep pressing “***Enter***” until the very end in cmd. This will create the desired .json file for us.



1. Open the browser and sign in on “***MongoDB Atlas***”.

Graphical user interface, website

Description automatically generated

1. Create a new cluster on MongoDB Atlas by following all the steps. Remember the username and password for the new cluster, it will be required at the time of connecting the application to the database.

Graphical user interface, application

Description automatically generated

1. Create a new database and collection in this cluster.

Graphical user interface, text, application, email

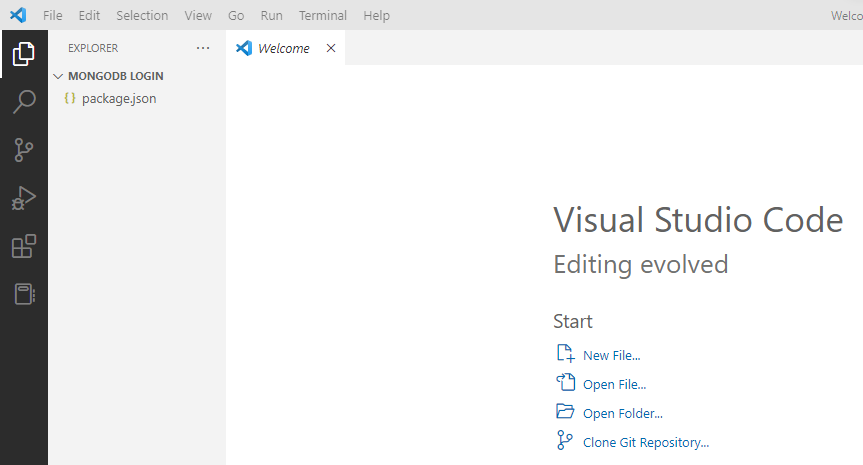
Description automatically generated

1. Allow the access to this cluster to desired IP addresses. To keep this application accessible with any device, we are allowing any IP address. You can keep it confined to selective IP addresses also.

Graphical user interface, text, application, email

Description automatically generated

1. Open this folder with IDE. Here we are using VS Code.



1. Open terminal in VS Code and download the necessary dependencies here. Type “***npm install express mongoose nodemon body-parser mongodb***”, and press “***Enter***”. Let the dependencies download.

Text

Description automatically generated

1. Now, you will be able to see the new folder named “***node\_module***” inside your folder and a new file “***package\_lock.json***”. At this point, we are ready to write our code for our node.js application.

Graphical user interface, application

Description automatically generated

1. Graphical user interface, text, application

   Description automatically generatedText

   Description automatically generatedCreate the “***index.js***” file and add the code for loading the dependencies into our app.

1. Connect the node.js application to the mongodb atlas cluster.



1. Copy the connection link from MongoDB Atlas and alter the desired credentials like “***<password>***” and “***Database name***” in the link.
2. Connect the application with a port to display and run our application. Here, we are connecting with “***localhost:3000***”.

Graphical user interface, text

Description automatically generated

1. Create the “***app.post***” function for connecting the application to the database and enable the posting of the data from application into our database collections. This function connects the “***index.html***” to “***signup.html***”.

Text

Description automatically generated

1. Create the public folder and create the html and css files for your application inside it. The 4 files in this folder will be “***index.html***”, “***index.css***”, “***signup.html***” and “***signup.css***”.

Graphical user interface, text, application

Description automatically generated

1. Edit the “***index.html***” file to create the home page of your application. Add the form on this page.

Timeline

Description automatically generated

1. Edit the “***signup.html***” file. This page will have the message of successful signup.

Text

Description automatically generated

1. Beautify the “***index.html***” and “***signup.html***” files using css files for both the pages.

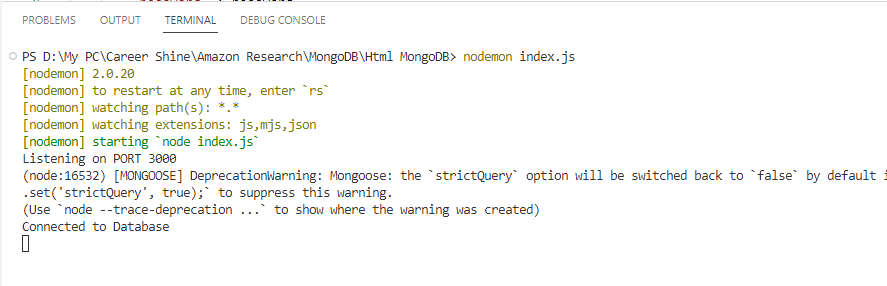
A picture containing text

Description automatically generated

Scatter chart

Description automatically generated with low confidence

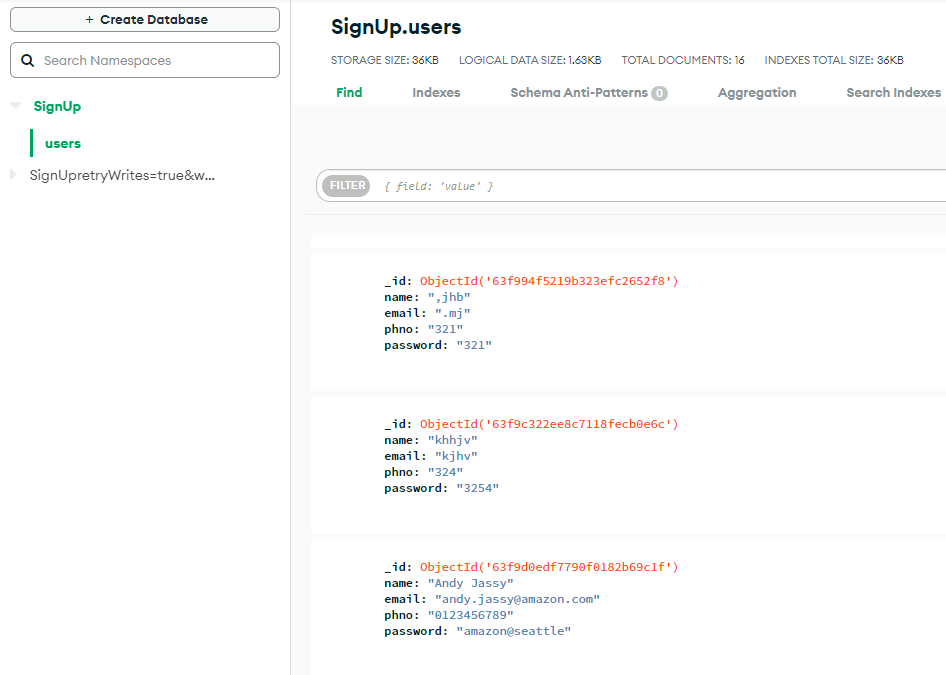
1. Add the code for connecting the html and css file.
2. Run the application by typing “***nodemon index.js***” in the terminal. If your application is running fine your will get this message.



1. Open the browser and go to port:3000. You will find the final application here.

Graphical user interface, website

Description automatically generated

1. Graphical user interface, application

   Description automatically generatedText

   Description automatically generatedAdd the required details and click in “***Submit***” to add your data into the database directly.