Insert multiple documents using NodeJS Script

1) Extract the "**06_Script to populate database**" folder from the **06_Script to populate database.zip** provided.

2) This folder contains

   a) the **data** folder having json files for each collection,

   b) **package.json** file having dependencies mentioned,

   c) **script.js** file to run the script to populate data from the json files to the MongoDB database.

3) In the script.js file at line no. 5, modify the <YOUR_CONNECTION_STRING> with your MongoDB Atlas connection string or Localhost connection string.

   a) You can get your MongoDB Atlas connection string as follows –

      i) Click the "Connect" option in your MongoDB cluster and Select "Drivers" option.

      ii) Select Node.js as the driver and you will see a connection string, copy and paste the connection string and make sure you replace the password.

      iii) If you don't provide database name in the connection string then it will by default create a database named "test"

      iv) Make sure you have clicked on "Add Current IP Address" button to whitelist the IP Address of your local machine so you can connect to the cloud from your local machine through the script.

   b) If you intend to use the localhost database then the connection string will be –

      mongosh "mongodb://localhost:27017/**DbName**"

      Replace the DBName with the actual database name. For example:

      mongosh "mongodb://localhost:27017/**EcommerceAppDb**"

4) Open the terminal and navigate to **06_Script to populate database** folder.

5) Install the dependencies using the following command - **npm install**

6) Run the following command to run the script.js file - **node script.js**

7) Check your MongoDB database in Atlas or MongoDB Compass, you should have all the documents inserted in all the collections.