Now that you can create one valid order and send it up to a route in node, the next step is to

[1] Edit your client code to add a new button, call it “Bulk” or “Do450” or some such. When the user clicks it, it should run a client JS method that creates 450 orders and sends them, **one at a time**, up to node (as it already does). You may NOT submit 450 orders as one post. You need to post 450 times and the posts must be spaced at least 5 seconds apart. However, you may cheat a little, and not really wait 5 seconds, instead, get real time for the first order, and then compute a new time stamp for the other 449 orders that is “now” + 5 or more seconds.

[2] Modify your existing route on node to stop writing the data to the console, but instead, insert the data as new documents in a new collection (call it orders or something) using the same mongo database we have been using in class for the two examples. You can either pre-create the collection name or just start writing to it, Mongo is friendly.

Each record (also called a document) must hold the data stored as key-value pairs

{

itemNumber: value

timePurch: value

storeNumber: value

pricePaid: value

salesPersonID: value

}

[3] Add a capability to your client to read all 450 records back from Mongo and display them.

[4] To submit your homework

- ZIP and submit your Node project including your Mongo connection string for your mongo data. Note, I will see your password, so make sure you use a unique to this project password. I suggest bcstudent.