

# HOMEWORK: HOMEWORK 2.3 (MANUAL VALIDATION VERSION)

## ***Preface to Homework 2.3 (Manual Validation Version)***

There are two versions of homework 2.3. You only need to do one. This is the version that previous students have used. It involves running a manual validation script on your local machine in order to verify that your blog is working, and so this version will include instructions on running that script.

***You only need to do one version of homework 2.3. Do not do this version if you've already done the MongoProc version.***

If you choose to do this version, you will need to download the homework packet from the Download Handout link above, and unpack. This will include a *validate.js* file in the hw2-3/validate/ directory that we will use to verify on your local machine (port 8082) that the signup and login pages of the blog work properly. You will not yet be building functionality for displaying posts. Don't worry, that will come later.

## ***Blog User Sign-up and Login***

Download hw2-3.zip from the Download Handout link and uncompress it.

You should see four files in the 'blog' directory: app.js, users.js, posts.js and sessions.js. There is also a 'views' directory which contains the templates for the project and a 'routes' directory which contains our express routes.

If everything is working properly, you should be able to start the blog by typing:

```
npm install
node app.js
```

Note that this requires Node.js to be correctly installed on your computer. After you run the blog, you should see the message:

```
Express server listening on port 3000
```

If you goto <http://localhost:3000> you should see the front page of the blog. Here are some URLs that must work when you are done.

```
http://localhost:3000/signup  
http://localhost:3000/login  
http://localhost:3000/logout
```

When you login or sign-up, the blog will redirect to <http://localhost:3000/welcome> and that must work properly, welcoming the user by username.

We have removed parts of the code that uses the Node.js driver to query MongoDB from `users.js` and marked the area where you need to work with "TODO: hw2.3". You should not need to touch any other code. The database calls that you are going to add will add a new user upon sign-up and validate a login by retrieving the right user document.

The blog stores its data in the blog database in two collections, users and sessions. Here are two example docs for a username 'sverch' with password 'salty'. You can insert these if you like, but you don't need to.

```
> use blog  
switched to db blog  
> db.users.find()  
{ "_id" : "sverch", "password" : "$2a$10$w14bNB/5CqwWx4bB66PoQ.lmYvxUHigM1ehlJyW  
QBupen3uCclDOW" }  
> db.sessions.find()  
{ "username" : "sverch", "_id" : "8d25917b27e4dc170d32491c6247aabbba7598533" }  
>
```

Once you have the the project working, the following steps should work:

```
go to http://localhost:3000/signup  
create a user
```

It should redirect you to the welcome page and say: welcome username, where username is the user you signed up with. Now:

```
Goto http://localhost:3000/logout  
Now login http://localhost:3000/login
```

Ok, now it's time to validate you got it all working.

Download `hw2-3.zip` from the Download Handout link, uncompress and you can validate it using the validation script we've provided. Just run the following commands:

```
cd hw2-3validate  
npm install  
node validate.js
```

For those who are using MongoHQ, MongoLab or want to run the webserver on a different host or port, there are some options to the validation script that can be exposed by running:

```
node validate.js --help
```

If you got it right, it will provide a validation code for you to enter into the box below. Enter just the code, no spaces. Note that your blog must be running when you run the validator.

```
1 A7foYWqTpKqRftI2d1D8
```

[SHOW ANSWER](#)

*Submission deadline has passed.*

