

# MongoDB Atlas Setup

In order to set up MongoDB Atlas (which is what is recommended for this course), I have made a really simple walkthrough. I have tried to be thorough, but if there is any confusion, please send me a message or post on Piazza and I'll fix it up as quickly as possible.

1. First, you need to create an account on MongoDB Atlas:  
<https://www.mongodb.com/cloud/atlas> . You should NEVER have to provide a credit card for this service.
2. When it asks you for your organization name, project name and preferred language, click "Skip" at the bottom.
3. On the next page, select "Shared Clusters", which is the free option.
4. Select "AWS" as the provider and "us-west-2" as the region (or whatever city is closest to you). The cluster tier should be set to M0 (note that you can only have one M0 cluster size per account, so you will need to share the cluster with your final project or your other classes.) You also might want to rename your cluster to something easy to memorize (mine is SeaWebDev). Then click continue
5. If you are brought to the "Security Quickstart" page, set a username and password (something simple like the username can be your name and the password can be 'banana1234'.) Then for the IP address, you can click "add my IP address" or just do "0.0.0.0/0" (note that this is a VERY INSECURE approach but is fine for the purposes of this class.) Note that you will need to come back here when you setup your Render page. Also, "0.0.0.0/0" lets Mongo know that anyone can connect to your app from anywhere.
6. Wait a few minutes for the cluster to be built.
7. When it is done, click "Connect"
8. Click on "Choose a Connection Method"
9. Click on "Connect your application"
10. Copy the text in step 2 (there's even a handy "copy" button to the right side.) Save this mongo db connection string in a notepad as you'll need to copy it a lot. Make sure that you replace the password with the correct password and the "myFirstCollection" bit, so it should look something like this:  
`mongodb+srv://myFirstName:banana2@myClusterName.ykjok.mongodb.net/pokemon_a  
pp?retryWrites=true&w=majority` Going forward, I will refer to this string as the **MongoAtlasURLString**
11. Now, go to your backend, Node code, go to your "server.js" (or wherever you have configured your MongoDB server" and set your MongoDB connection URL to the MongoAtlasURLString. So, if you're using the code from class, it originally looked like:

```
const mongoDBEndpoint = 'mongodb://127.0.0.1/collection_name';
```

And now should look like:

```
const mongoDBEndpoint =  
"mongodb+srv://myFirstName:myPassword@myClusterName.ykjok.mongodb.net/<dbname>?retryWrites=true&w=majority"
```

Now, run the code locally and make sure that you can add/remove entries from the database.

One issue, however, is that you never want this kind of database connection string in your code base as if anyone found it, they could do a lot of damage. So after testing, set the line to:

```
const mongoDBEndpoint = process.env.MONGODB_URI ||  
'mongodb://127.0.0.1/collection_name';
```

We'll now be setting the String as an environment variable in Render.

1. Log into Heroku and select your app that you're using for this project.
2. Select "Environment"
3. Click on "Add Environment Variable", and that should open up a simple form.
4. On the key side, type in "MONGODB\_URI". On the right side, paste the `MongoAtlasURLString`.
5. Click "Save Changes"

Now, when you push your code to Heroku, your app will connect with MongoDB Atlas, and you can still use your local MongoDB instance when testing locally. You'll notice that the `MONGODB_URI` text should match the end of the "process.env" in the line where you set the Mongo DB Connection String: this is because it will try to find this variable on the actual host itself.

As before, if something here isn't clear, please let me or the TA's know.