Lab 1
Due 1/21, before class

We're going to start building up an API of our own design. We don't have a database yet, so you're welcome to store things on your VM for now.

Pick a topic (news, stocks, pictures, weather, anything you want) to be the theme of your data for this lab. You will need no less than 100 "objects." For example, if you are making a news site, you need no less than 100 articles.

Create a new Node project, name it Lab1 and put your name as the author. Install Express (and any other libraries you want).

You should create an API with Express that does the following. I am going to provide this example as if I am making a news site, but you feel free to modify the nouns of your API to match what you are doing (**document** your API in your README.md file):
1. GET / = retrieve an HTML/CSS/JS frontend of your creation (remember the app.use(express.static('public')) thing?)
2. GET /news = retrieve a listing of article numbers (hint: you can use headers for this if you don't want to send the list in the body of your response; hint2: consider pagination, since you have a lot of articles!)
3. GET /news/### = retrieve the specific article from the JSON object
4. POST /news = append a news article to the end of the JSON object
5. PUT /news = bulk update all your articles
6. PUT /news/### = update the specific article
7. DELETE /news/### = delete the specific article

(Please don't make DELETE /news)

Your frontend **_may not_** directly grab any news articles: it may only get articles via the API. You may need to create a form or some other interactive feature to POST, PUT, and DELETE articles.

The things your API serves _need not_ be JSON objects, though of course they can be (and you'll probably want some of them to be). You can mix and match the data you are working with.

You must serve your project on your VM at **https://<FQDN>/lab1**. In addition, the TA will run "npm i" on their local machine to see if your package.json file works correctly.

You may discuss the design and implementation of the lab as a group. You may also take the 2 hours in class on 1/14 to code together. After that, all individual work.

You need some sort of frontend to test things, but this lab isn't about the prettiness of the frontend you make, it's about creating your API and having your frontend talk to it. Of course,

you can make a pretty frontend for creativity. Document what you want us to consider for creativity in your README.md file.

If you don't document your API and/or cite your sources in your README.md file, you will get a 0 for the lab, no exceptions no apologies later.

Grading:

API creation	20 pts (this includes your POST, PUT, DELETE interactive feature)
Project.json	10 pts
Creativity	10 pts
README.md	10 pts
TOTAL	**50 pts**