Web Science
Quiz 1: March 1, 2022

Edward Shin
Enter your answers directly into this document (with the exception of #2 and #3).
All answers should be In Your Own Words, using complete sentences with proper spelling and grammar.

Save this document as: answers.docx (or .odf or .pdf) (-5 if wrong name). For all questions other than #2 and #3, you will not receive any credit for answers not placed in this document.

When finished with the quiz, put everything you wrote (this document, all code, etc.) on GitHub into a branch in your lab repo named: quiz1 (-5 if submitted incorrectly). **Do not submit your node_modules folder! (-15 if you submitted the node_modules folder)**

1. **Short answers** (25 points): (Answer in complete sentences, explain your answers)

    a. (5) How can I determine the type of device that my page is being displayed on? Give two examples of why I might care.

Using CSS media queries, we can first find out the device's screen size/resolution by utilizing min-width and max-width. Furthermore, I could use the navigator userAgent property in Javascript, as that returns a string containing information about the device the user is using.

One example that the type of device the page is being displayed might be important is because some devices have smaller/larger screens, and it may cause a conflict with the frontend properties being displayed if some sizes are not configured properly.
Another example is so that the developer can differentiate between pc (Windows/ macOS) and mobile (ios or android), both can have different properties and styling. The main goal, in the end, is to enhance the user experience.

    b. (5) What is a package-lock.json file? What is it used for? Is it required?

It serves multiple purposes, but the main purpose of the file is to track the exact version of every npm package installed so that the product is 100% reproducible in the same way even if the packages are updated by someone else.

    c. (5) What is npm? How does it work? Why is it used?

npm is an abbreviation for node package manager. It is used to keep track of, and also install, packages built for Node. These packages can be installed globally with a command line, such as npm install -g.
npm is used for managing dependencies of various server side dependencies.

d. (10) Describe **in detail** the sequence(s) of transaction(s) for a frontend to request data from some external entity via Node.

2. **Coding question**: (40 points) Create a webserver in node.js, name your server: server.js. You may use Express, but you *may not use a generator* – (i.e., NOT express-generator), which will serve a simple frontend (in the technologies of your choosing). The frontend will provide an input field for ZIP code and a series of buttons that issue GET and/or POST requests when clicked to the Node server. (frontend: 10 points)

   Upon entering a ZIP code and clicking the "Temperature" button, your application should send a POST request to http://localhost:3000/temperature. Node should then get the current temperature for that ZIP code (I bet you have an API for that!) and send the frontend back that information. The frontend should then output a sentence that says the name of the location and whether it is Freezing (<33F), Cold (between 33 and 50), Warm (between 51 and 80) or Hot (>80) – display the corresponding message in a unique color for each category. (temperature sequence: 10 points)

   Upon clicking the "Is RPI windy?" button, your application should send a GET request to http://localhost:3000/wind. Node should get wind speed information for Troy, NY, via that API and send that information back to the frontend. Have the frontend display this information in a unique color. (wind sequence: 10 points)

   Creativity matters; don't just give me an empty white page with a text entry form box and two buttons. Go beyond the minimum (but remember that creativity doesn't have to be visual). If you need to, write a short README file that tells me what I should consider for creativity. (creativity: 10 points)

   You may use any and all libraries you want for this coding question.

3. (15) Ensure the package.json file for Q2 has no errors when I run npm install & run your code.

4. (20) Provide **two** different explanations of the code below. The first explanation should be a high-level explanation (no less than four complete sentences) outlining what this code does to someone who has no coding experience. The second explanation should be a *detailed* one explaining line-by-line what the code does. If there are any errors in the code, fix them.

Basically this is a code for sockets, it allows you to exchange information between processes on the same machine or across a network, distribute work to the most efficient machine, and they easily allow access to centralized data.

```
var net = require('net');

var sockets==[];

var s = net.Server(function(socket) {
   sockets.push(socket);

   socket.on('data', function(d) {
      for(var i=0; i<sockets.length;i++) {
         if (sockets[i]==socket) continue;
         sockets[i].write(d);
      }
   });
   socket.on('end', function() {
      var i=sockets.indexOf(socket);
      sockets.splice(i,1);
   });
});

s.listen(8080);
```

The first line uses the net module, which contains functions for creating server and client streams

The second line creates a list for open websockets

The next line is a function that creates a new ‘server’

Adds some sockets into the list of websockets

In socket.on function, it checks to see if the data was transmitted to the socket, and then adds the data to all the open sockets, excluding the one itself to prevent duplicates.

In ‘end’, it ensures that when a socket’s connection is ended it will be removed from the list of sockets.

The last line makes sure that the server is listening to all incoming connections in localhost port 8080

5. (+5) What is the name of the RPI-developed chat protocol popular in the 1990s?

Hipchat