Module 12 Career Connection

Introduction

Congratulations! This week you learned how to work with Plotly, `D3.json()`, as well as how to parse data in JSON format, using JavaScript to create and deploy interactive charts to GitHub Pages. Feeling a little overwhelmed? That's okay and is to be expected. In this section, we're going to review how all of this new material is applicable to your new career.

You might remember from an earlier Career Connection that most of our students who obtain **Employer Competitive** status are successful in getting positions after their boot camp has ended. How can you use the material you've learned this week to become more employer competitive?

1. **Deploy your own interactive chart to GitHub.** Pick a topic that you are interested in and research an API that will give you the data that you'll need. A great place to get started is **Rapid API** **(https://rapidapi.com/? utm_source=google&utm_medium=cpc&utm_campaign=1674315309_7653 9154269&utm_term=rapid%20api_e&utm_content=1t1&gclid=CjwKCAjwxt_t BRAXEiwAENY8hYl80EA4_3yhyyHyL6r5SKp6M2vwXPCtGXomEZLF4UERT6 aL8LAlOBoCzQQQAvD_BwE)** or **Data.gov (https://www.data.gov/developers/apis)** to find some cool data to play with. Once you have the API you need, use what you've learned in this module to pull the data, format it, and create some interactive charts. Deploy it to GitHub Pages and add in some text to explain what you're doing and why.

2. **Link a PDF of your resume to your portfolio or GitHub page.** Use your portfolio as an opportunity to market yourself. If you haven't done so already, add a button to your portfolio that allows prospective employers to download a copy of your resume. Also, you might

consider adding a section on how to contact you and/or get your resume on your newly deployed interactive chart page.

3. **Update your resume with your new skillset.** If you haven't done so already, add Plotly and D3 to your resume in the "Technical Skills" section. Display it loud and proud!

Technical Interview Preparation

Technical interviewing is a skill that requires constant practice and development. Moreover, being tested on your knowledge of a particular topic in an interview can be a nerve-wracking experience. Here are some common interview questions to consider. Before viewing the answer to each question, jot down some things you might say in response.

Common Technical Interview Questions

Q: How would you create a bar chart with Plotly.js to display data?

A: Remember, you aren't expected to remember the exact syntax—employers know that you can look this up when you're working. But you should have an idea of how to conceptually sketch out how this might work. For example, you might touch on the following points:

- How you would fetch data from the API (`d3.json()` to send an HTTP request and load `.json` file)
- Use JavaScript methods and Math object to massage the data into the appropriate JSON format
- Use Plotly.js `newPlot` method to create the bar chart, passing in the JSON for the data

While this isn't a complete answer, it does give the employer a solid idea that you know what you're talking about. Remember, concepts are better than syntax details, but the details are a great bonus if you can remember them.

Q: How would you use the JavaScript method `.filter()` to manipulate data?

A: JavaScript's `.filter()` method is used to essentially filter out the data you don't want. It will return all items from the array that match the given filter statement.

Case Study: NBA Shot Data

If you've ever watched a sports game on TV, you may have noticed the prevalence of data. Before, during, and after the game, viewers are bombarded with enormous amounts of data, such as predictions based on data trends or performance comparisons of individual players and teams.

Imagine that you've just been hired on by your favorite NBA team, your dream job. Your new boss points out that a lot of draft decisions and game strategies have been based on the coaches' gut instincts: who they feel is a good player, who makes the best shots, and who is a better player in comparison to another on the team.

For example, Player X is regularly played on the court as a starter because he usually scores 30–40 points per game. However, a quick look at the data shows he is actually making only about 35% of all the shots he takes —so, in reality, he's missing a lot of opportunities! Corporate management has been pushing for more data-driven decision making in order to maximize the team's performance.

After reading this prompt, the technical interviewers ask the following questions:

Q: What data points would you want to track in order to push data-driven decision making for the coaches?

A: There's no one correct answer for this question, but some data points that would probably be worth tracking are percentage of shots made, the spots where shots are made from, and the percentage of shots made from those various spots.

Q: How would you visually display this data in an attractive and useful format for corporate management that are not data scientists?

A: Again, this isn't the kind of question that has a right or wrong answer. The goal here is to think about the various ways in which you might visualize the data using an interactive chart and your justifications for doing so. In this scenario, you might consider an interactive view of a basketball court with the points mapped out where the shots are taken from. Each of those points might vary in color intensity—the lighter the point, the lower the percentage of shots made.

Continue to Hone Your Skills

If you're interested in learning more about the technical interviewing process and practicing algorithms in a mock interview setting, check out our **upcoming workshops.** **(https://careernetwork.2u.com/? utm_medium=Academics&utm_source=boot_camp)**