CAREERFOUNDRY

Python for Web Developers Learning Journal

Objective

We find that the students who do particularly well in our courses are those who practice metacognition. Metacognition is the art of thinking about thinking; developing a deeper understanding of your own thought processes. With the help of this Learning Journal, you'll broaden your metacognitive knowledge and skills by reflecting on what you learn in this course.

Thanks to this Learning Journal, when you finish the course you'll have a complete and detailed record of your learning journey and progress over time. We really recommend that you take the time to complete this Journal; students do better in CF courses and in the working world as a result!

Directions

First complete the pre-work section before you start your course. Then, once you've begun learning, take time after each Exercise to return to this Journal and respond to the prompts.

There will be 3 to 5 prompts per Exercise, and we recommend spending about 10 to 15 minutes in total answering them. Don't overthink it—just write whatever comes to mind!

Also make sure that, once you've started filling this document in, you upload it as a deliverable on the platform. This is so that your mentor can also see your Journal and how you're progressing over time. Don't worry though—what you write here won't affect how you're graded for the Exercise tasks. The learning journal is mostly for you and your self-evaluation!

Pre-Work: Before You Start the Course

Reflection questions (to complete before your first mentor call)

1. What experiences have you had with coding and/or programming so far? What other experiences (programming-related or not) have you had that may help you as you progress through this course?

    a. **My recent experience with programming so far is in the Career Foundry bootcamp for JavaScript. I believe that my previous certification in web design from Design Labs might be beneficial for me to work through this course when it comes down to designing.**
2. What do you know about Python already? What do you want to know?
    a. **I honestly don't know too much about Python other than a lot of people say it's easier to understand once you've understood another language. Based on some opinions from my colleagues, they claim it's easier to comprehend compared to JavaScript. What I would like to know is how much of the world is running Python compared to JavaScript.**
3. What challenges do you think may come up while you take this course? What will help you face them? Think of specific spaces, people, and times of day of week that might be favorable to your facing challenges and growing. Plan for how to solve challenges that arise.
    a. **I think a big challenge for me is that since I'm using Python 3.12 and the achievement is using 3.8. I'm concerned about things that might be obsolete or may have changed. I believe a good way to combat this challenge is to probably read up on the documentation/release notes to see if there are any changes, and using Stack Overflow to see if there are any workarounds if something might not work. Similar to how I was working in the JavaScript course; I ran into an issue where certain operations in Angular 10 had changed in Angular 18, and I had to use Stack Overflow as a resource.**

Remember, you can always refer to Exercise 1.4 of the Orientation course if you're not sure whom to reach out to for help and support.

Exercise 1.1: Getting Started with Python

Learning Goals

- Summarize the uses and benefits of Python for web development
- Prepare your developer environment for programming with Python

Reflection Questions

1. In your own words, what is the difference between frontend and backend web development? If you were hired to work on backend programming for a web application, what kinds of operations would you be working on?

    a. **Front-end development focuses on the visible aspects of a site, including design and interactivity, using technologies like HTML, CSS, and JavaScript. Conversely, backend development handles the server-side processes, managing databases, servers, and application logic. Backend programmers engage in tasks such as database management, server-side coding, API development, security implementation, performance optimization, external service integration, and error handling.**

2. Imagine you're working as a full-stack developer in the near future. Your team is asking for your advice on whether to use JavaScript or Python for a project, and you think Python would be the better choice. How would you explain the similarities and differences between the two languages to your team? Drawing from what you learned in this Exercise, what reasons would you give to convince your team that Python is the better option?
   (Hint: refer to the Exercise section "The Benefits of Developing with Python")
    a. **Python stands out over JavaScript for web development due to its readability, dynamic typing, extensive package ecosystem, built-in essentials, and strong community support. With Python, developers find code easier to understand, variables more flexible, and access to a wealth of pre-built solutions for various tasks like mathematical calculations and database connections. Additionally, Python frameworks streamline common web operations, reducing development time, while its supportive community offers resources for troubleshooting and continuous improvement.**

3. Now that you've had an introduction to Python, write down 3 goals you have for yourself and your learning during this Achievement. You can reflect on the following questions if it helps you. What do you want to learn about Python? What do you want to get out of this Achievement? Where or what do you see yourself working on after you complete this Achievement?
    a. **I want to build some real world projects with Python and compare it to JavaScript to see what might be my preferred language.**
    b. **I want to gain a solid understanding of the fundamentals concepts of Python so that I can write clean, efficient and maintainable code.**
    c. **I also want to explore possible specialization in the language itself.**

Exercise 1.2: Data Types in Python

Learning Goals

- Explain variables and data types in Python
- Summarize the use of objects in Python
- Create a data structure for your Recipe app

Reflection Questions