Chris Jantzen

CS 4003

Typing Trainer

A problem I have had in my life was that I wasn’t typing as fast as I would have liked to and it was slowing down my productivity when writing papers. To start this process, I switched from typing in the standard QWERTY keyboard layout in favor of the Dvorak layout because it is much more efficient and better on the fingers, but since I had to learn this new muscle memory, I need a means to train. My solution to this was to create my own typing trainer that allowed for me to increase the accuracy and speed of my keystrokes.

To create the main functionality of this project, I used an object-oriented approach because I thought it would be best to treat each letter in the text block as an object to keep track of the correctness and visual properties all in one place that would be easy to keep track of. I used JavaScript to implement this because it is a language that I am comfortable with because I used JavaScript and the NodeJS framework very frequently at my Co-op job, however I treated it more as a functional language than an object-oriented language at work. I elected to implement this in JavaScript with an HTML file instead of using NodeJS because doing this allows me to create an interface with HTML and CSS in the future, once I learn more about designing web interfaces. While working on this project, I learned a decent amount about how to use JavaScript to implement objects because it is a bit unorthodox in its implementation compared to some languages that are a bit more conventionally structured, like C++ or Java, for example. Learning to use the “this” property of a class in JavaScript is somewhat difficult, but it seems to work roughly as a public method or variable, and whenever an object that belongs to that object is used within the class, you must use “this.variable,” whereas variables defined with “let” or “const” can be used throughout the class without “this.variable” but are not accessible from the outside directly. I really like working in the object-oriented paradigm because everything is easier to keep track of and test because it is very organized in implementation. You can define helper functions within the class without making them publicly accessible, so it’s very easy to know which attributes of the class you can see when working elsewhere in the codebase. This gets to be especially helpful when working in very large code bases because you can place one class per file and organize the files with namespaces, or other similar organization methods. I think that JavaScript was the right language for me to have chosen because it is a good language for working with objects and it gives me a good opportunity to continue on with the project and create an interface with HTML and CSS.

For the logical portion of this program, I chose to use MySQL to make a database to be able to theoretically tie my JavaScript and Python code together to be able to interact with each other properly. I used the database to store the pieces of text that would be getting typed along with the amount of characters in each, and an ID for each in one table. In a second table, I stored a reference to which text block was being typed, the time it took, and the accuracy of the typing for that attempt. This second table would continuously expand as each attempt was taken. I had never made a database before this project, so I feel that learning how to create one is going to be very useful for my future projects. In implementing this, I learned how to make SQL queries and how those will work to get the types and amounts of data that I want to be displayed and from where. I learned that the logical paradigm is meant to define what the program is meant to accomplish with the code written, but not how to accomplish it. I think this was the best logical programming method I could’ve chosen for a real implementation of the project, however, I feel that I could’ve maybe chosen a language that taught me a bit more about logical programming because I think I still only understand the paradigm in a very basic sense and I would like to get more experience with it to see if it really is something that I would have any interest in further pursuing.

For my imperative portion, I used Perl to get user input on deciding which block of text that they would like to type. I chose Perl because this seemed like the best task to take advantage of the imperative programming paradigm. I learned a lot about Perl in doing this, having only seen the very basics of the language before starting this. Learning when to use the dollar sign, at sign, and percent was difficult because sometimes when dealing with arrays, you use the array with the dollar sign to index into that array. I also had a lot of issues with getting the syntax correct because things like loops don’t always work as I would think they would, but after some visits to Stack Overflow, the issues become a bit clearer. I think that Perl was a good choice for this implementation because it provides all of the tools that were needed to complete this portion of the project without too much trouble, and that’s really all that was needed because it wasn’t a very complicated bit of code. Much like my experience with logical programming, I feel that this was the correct choice for the project, but I would like to get a bit more experience coding with the paradigm to get experience for my own knowledge.

To implement my functional programming portion of my assignment, I chose to use Python because it is a very versatile interpreted scripting language that would allow me to do a lot easily. I performed my data calculations in this portion of the program in regard to the amount of time taken to complete the typing and the accuracy. To do this in practice, I would have the program read any new data from the database to get the start and end times typing session, the number of characters typed correctly the first time, number of characters that were corrected, and the number of characters left incorrect, and the total number of characters in the text the user had typed. I created several functions to perform small tasks and used those small tasks to be combined and make larger tasks to produce the desired result, as is my understanding of the correct way to implement the functional programming paradigm. I support my choice of using Python for this because Python is a language that supports first-class functions and anonymous functions (lambdas in Python), so I could create functions and pass them as variables to the other functions to be applied to the data in order to streamline the process of writing more complicated functions because I wouldn’t have to worry about altering the data inside of that code block and I could just take the data that function was given and do what is needed right away. I liked using Python because I was already comfortable with the syntax and how the language worked, which made it easier to focus on using the functional programming paradigm instead of trying to both learn the language and the methods of implementing functions at the same time.

I learned a good amount about the different paradigms in this project and I think that this knowledge will be good to bring to my future major projects. Being able to properly implement functional and object-oriented programming seems to be a very common thing in enterprise-level software development, so having a basis in these will be very useful as my professional career develops. A side benefit to this project was being able to spend some time setting up a git repository on GitHub and gaining experience with a source management system.