Alec Zackarian
Topics In Computer Science A
5/22/20

Programming Final Project Writeup

My favorite project this year was our Final GUI project because it gave me near unlimited creative control. Because of this, I decided to give one of my older, console-based programs some GUI interactivity. I decided to upgrade my old quiz project because I liked the concept of a wacky quiz which would build off of the primitive and cumbersome original design. Through this project, I not only learned about new methods and new skills, but also learned how to make more efficient and organized code as well as troubleshoot problems rather than immediately asking for help.

For the group project, the class decided to make a bagel based game comprised of miniature widgets. My widget was a bagel themed quiz which would track your score and give you bagels based on how many questions you answered correctly. In order to build upon the old quiz project, I created a more simplified Question class which relies more on storing the components as parameters rather than using set methods. Then, for the quiz itself, rather than having you type options presented, I provide answers via buttons you can click. By using graphics displaying buttons rather than having the user type their answer, the risk of answering correctly but not getting credit due to capitalization is eliminated. By incorporating a graphical user interface, not only is the quiz more intriguing visually, but also easier to expand upon. In the event that I wanted to add more questions to the quiz, it would be much easier to add questions to the GUI compared to the original console based edition.

In my Individual Final Project, I was able to more fully develop the GUI quiz and add more features. For the individual project, because I had more time, I was able to add unique images for each question and make the “try again” option more interactive by using a new frame and sounds. By building upon the questions and “try again” feature, I was able to make the project feel more complete and polished. Rather than simply seeing a print statement with the amount of questions you answered correctly, or simply seeing your bagel score increase, for this project I added win/loss screens. These further help to make the game seem more polished but also make the game more interactive. On top of the visual component, I was also able to make the code more efficient. Rather than making four unique action listeners for each answer choice, I simplified the process and pared it down to one. Simple fixes like this make code easier to understand for outside parties and are in general a good habit to develop as simpler code is easier to understand and build up by other programmers. Through the process of creating this code, I also developed new skills in order to troubleshoot issues on my own. Rather than relying on help from other classmates, I learned to better address issues in my code and only ask Dr. Frewen for help once truly unsure. Through this experience, I learned about methods such as getMessage() which allowed me to further understand issues found by my try-catch. Rather than just seeing an IO exception, I learned that I had to use a .wav sound rather than an mp3; something I never would have known had it not been for the getMessage() method.

Through this final project, I not only honed new skills regarding efficient code and troubleshooting, but also had a lot of fun in the process. Adding more interactivity through a graphical user interface was not only a fun project, but also a great way for me to learn more about exceptions, troubleshooting, and developing more efficient code.

Thank you for the great class Dr. Frewen