Corey Dean Collins Jr

CPT 206 A01S

James Fowler

First, I like to address the designer of each program. The design of this project is very clean and easy to understand. The only thing I would say to add cosmetically would be maybe a mild color background or mild color tools. Just makes it prettier for the user and catches their attention. I would also say the author should maybe slide the database load to the right just a little bit, with the placement of the box it looks like it may be cutting off more data. From our point of view, it's fine. But once this loads to a user it could seem like it is cutting off.

Addressing the code, I’m not sure if it is something happening with my laptop, but it says there is an error with the code. Sometimes when I run it, it runs fine. Other times, it does not. One thing I noticed is that the author does not have any Try/Catch’s in their code allowing user error. If any of the buttons are selected with no data in the text boxes, the program crashes. I try to have the habit of using them for everything just in case because we could never know how everyone will process our form after it is created. Adding a Try/Catch along with a message will let users know if they may have accidentally entered incorrect data in a box and give them a hint on what data is allowed there. I am also catching an error at line 60. I’m not sure what the code “Convert.ToInt32” does, but I feel like the author was attempting to parse a value. If that is the case, I believe a “Int.TryParse” would have been fine.

After looking at both, if I had to steal something from the author's code it would be the load of the actual database into the designer. That was something I could not figure out. When it comes to the code itself, I think the authors were missing some stuff here and there. The program does crash when invalid info is entered in the text boxes. I know for sure that attaching a Try/Catch to each button click would cancel out about half of these crashes. My biggest feedback here would be to try and imagine that you do not know how this code works and troubleshoot your design by imputing data that should not be entered.