Every if-statement must have an else.

If this else should never run because it doesn't make sense, then you must use a die function in the else that prints out an error message and dies, just like we did in the last exercise. This will find many errors.

Never nest if-statements more than two deep and always try to do them one deep.

Treat if-statements like paragraphs, where each if-elif-else grouping is like a set of sentences. Put blank lines before and after.

Your boolean tests should be simple. If they are complex, move their calculations to variables earlier in your function and use a good name for the variable.

Most of the uses of inheritance can be simplified or replaced with composition, and multiple inheritance should be avoided at all costs.

**Tips for Debugging**

1. Do not use a "debugger." A debugger is like doing a full-body scan on a sick person. You do not get any specific useful information, and you find a whole lot of information that doesn't help and is just confusing.
2. The best way to debug a program is to use print to print out the values of variables at points in the program to see where they go wrong.
3. Make sure parts of your programs work as you work on them. Do not write massive files of code before you try to run them. Code a little, run a little, fix a little.

The best way to work on a piece of software is in small chunks like this:

1. On a sheet of paper or an index card, write a list of tasks you need to complete to finish the software. This is your to do list.
2. Pick the easiest thing you can do from your list.
3. Write out English comments in your source file as a guide for how you would accomplish this task in your code.
4. Write some of the code under the English comments.
5. Quickly run your script so see if that code worked.
6. Keep working in a cycle of writing some code, running it to test it, and fixing it until it works.
7. Cross this task off your list, then pick your next easiest task and repeat.

This process will help you work on software in a methodical and consistent manner. As you work, update your list by removing tasks you don't really need and adding ones you do.