**Preamble**

What makes our team unique and great at TEKSystems is our focus on code quality. We pride ourselves in having a great team with strong engineering practices skills. This is why customers choose us as their partner. They don’t want us to just develop some code for them, they want us to build an enterprise quality solution that will support the growth of their business. This is why we believe that you have to start with a great foundation that is based on strong engineering practices, code quality and object oriented design.

We would like to give you a small challenge so we can see how you think about code quality. We know you are probably very busy and we appreciate the time you will be investing on this but we believe this is the best way for us to get to know the great problem solver you are.

We are very excited to see what you will come up with.

Sam Malek

Practice Director - Application Development and Management

TEKsystems.

The problem to resolve: Order calculation



The attached zipped code is a program that compiles but which contains bugs preventing from calculating the expected cost of 3 orders.

You have to fix all those bugs providing appropriate documentation and writing unit tests.

**Environment prerequisite**

The code compiles with any version of Java. You can use the IDE of your choice as well your preferred framework for the unit tests but the calculation of the order cost should be performed by your own code.

**What we are looking for**

We are looking for a code with 0 bugs but you will be also evaluated on how much your code is easy to read as well as on the quality of its documentation and design.

We would like you to provide 2 versions of the code;

1. The first version would fix all the program **bugs**, both in the code and its **documentation**;
2. In the second version we would ask you to refactor the code to make it easier to maintain, writing [**clean**](http://www.amazon.com/Clean-Code-Handbook-Software-Craftsmanship/dp/0132350882) **code** with meaningful names, OOP principle etc.. and writing **unit tests** to provide a good coverage of the code;

Be sure that the submitted project is something you are proud of and showcases your software development skill’s mastery.

Do not take for granted anything in the code provided, the code is purposefully bad and flawed in many aspects to give you the opportunity to show your knowledge, some flaws are readly evident, some are conceptual and subtle.

Submit your code

Send to your recruiter a zip file with the git repository of your code and anything else that you deem relevant to explain your approach, the repository should have a commit tagged where your changes to fix the bugs are and the tip of the repository should be the final version.

Once the code is written, the source code must be submitted without its byte code; please remove any files archiving byte code (jar, war, "\*.class” and the bin/target directory, those binary files should not be versioned in your repository.

The delivered project must be compilable “from the box” no changes in directory structure should be necessary or creation of anything besides a simple import from IDE or execution of a tool from commandline like: ant, maven or gradle.