

## Assignment 2 Analysis

1. My programming partner is Casey Nordgran, and he will be submitting the source code for our program.
2. I like driving better because I know a lot of shortcuts and fancy tricks for the IDE that we use and love sharing them with Casey and showing him how the environment works. Additionally, I really appreciate his advice where programming flow is concerned, and I like the way he thought to implement certain methods.
3. I liked working with Casey and plan on continuing to work with him. He is very amiable and easy to talk to, but more importantly he's a good teammate. He's good with compromise and addressing issues in programming or us as a team (from what I can tell so far, we haven't had enough time together to issue a full evaluation by any means).
4. The difference between the Java `Comparable` and `Comparator` classes is that the `Comparable` class implements the `x.compareTo(y)` method and is used for the natural ordering of objects, whereas the `Comparator` method implements the `compare(x, y)` method and is used for ordering objects in ways apart from their natural ordering. For example, if you simply wanted a sorted list of `LibraryGenericBooks` the `Comparable compareTo()` method should be overwritten so that a list of `LibraryGenericBooks` could be sorted against each other, whereas the `Comparator` class' `compare()` method could be used to create other sortings of the books, perhaps by ISBN or author.

Thus, the `Comparable` interface could *not* be used to implement the extra features in the `LibraryGeneric` class because we implemented multiple types of sorts with the different fields, and the `Comparable` interface only allows for the definition of one type of sort for the objects.

5. Casey and I were quite efficient in our time usage, though that was due in part to the straight-forward nature of the assignment. When we got together to program it seemed that we were always being productive, even if we were looking up how to do certain things in IntelliJ or talking about specifics of the assignment requirements or implementation techniques. Furthermore, we got together multiple times for only a few hours, which is a marked improvement upon the team techniques I had with my previous partner wherein we were sometimes bunkered together for almost a day straight (however, this is undoubtedly another symptom of having a very clear assignment with transparent requirements). I honestly cannot think of a way to improve our time together, though perhaps better techniques will become apparent as our team carries into the semester.
6. I estimate that we spent about two hours the first time we were together, and perhaps four hours the second day, and maybe 1.5 hours between today and yesterday. In total I believe we spent 7.5 hours together on this assignment, though a significant part of that was initializing GitHub access and setting up IntelliJ and the like.