1. **What was the most surprising characteristic to you of a great software engineering, as described by the paper?**

   To me, the most surprising characteristic is risk-taking [willing to go into high-value areas even though they may not have knowledge or expertise]. I believe this is very gray-area to be spoken about and not covered in much details. We don't have to go far to look at financial industry when we assume risk-taking should not be a desired characteristic and it can put society at hardships, evident by many examples within the financial industry. Similar, we must don't forget this characteristic within tech or software industry and what kind of harm it can do to us, especially in the unseen area of work. There must be a proper procedure for one to exercise this characteristic, such as mitigation exercise or analysis, or budgeting on what's feasible before accepting a blanket trait of this characteristic.

2. **What are some things you might want to do on your own, after this course ends, to become a great software engineer?**

   My next steps in my own journey is risk management and analysis within software engineering space. Our industry is still trying to find ways to mitigate risk and yet deliver good quality software deliverables. For example, the rapid development concept of programing and how it has shaped over the industry without the realization of risk analysis. I want to study risk management where we plan ahead of time to provide further possible resources [engineering support, systems, headcount] to cover any and all risks. This could include assessment, identification, controlling, prioritization, resolution, etc. The impacted audience isn't just me or the company I work for, they are end-users/customers, contractors, government parties, non-direct personal, or your normal Joe around the street.