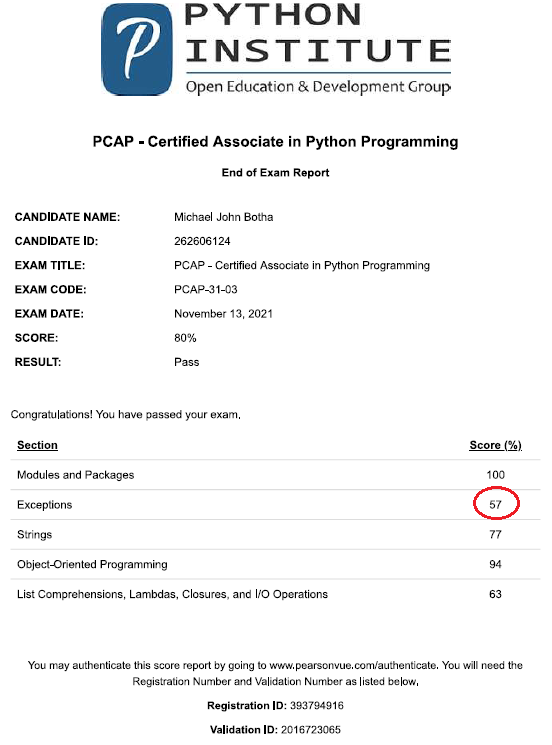
Week 1

This week was challenging because I was completing my preparation and examination for a Python associate level certification in unison with my studies. This I felt necessary to make myself more marketable as a junior software developer, but also to further my knowledge of the Python language and its associated technologies, enabling me to engage at a deeper level with my MSc studies. The final exam was expansive and challenging, yet fair. However, according to the test result seen in Figure 1, I do need to work specifically on my understanding of exceptions and their related classes and abilities. Although the required level of exception handling has not been that extensive in the masters thus far, this weakness will need to be improved in order to make my code more secure. With regards to the direct MSc content, it was very helpful to learn about the OWASP standards and the support framework that it provides for programmers. It will definitely aid me in terms of a base with regards to the common vulnerabilities to look for in my code and the standard means to rectify them.

Week 2

I needed to further research the Scrum Agile methodology, because the notes provided were not sufficient. However, I found some interesting content online and now have a basic understanding of the framework and relevant methodologies, which will hopefully help with the coding project planning. Reading the required chapters regarding the architectural attributes pertaining to good code was very insightful, however I will need a fair amount of practice to implement them wholistically.



**Figure 1**

Week 3

Having to get through many pages of reading has reiterated that at the MSc level one needs to be able to read quickly and effectively through large amounts of new information. Additionally, learning what areas of content not to focus on and making good notes has become a key attribute of success in this domain. Delving further into the key attributes of a well architectured system as well as documented system models of code is really helping me to be aware of more standardised and formal coding patterns.

Understanding the dangers of buffer overflows reignited my interest in Operating Systems (OS) and encouraged me to research further into memory allocations with regards to programs during execution. I will need to delve deeper into this subject, perhaps through learning more C code. However, my understanding is sufficient for now especially considering Python has a memory manager. Although an awareness of such is still relevant to the Python environment.

Week 4