|  |  |  |
| --- | --- | --- |
| Number of Orders | Processing time with single thread (msec) | Processing time with multi-thread  (msec) |
| 1 | 47 | 44 |
| 50 | 273 | 302 |
| 100 | 422 | 403 |
| 250 | 813 | 552 |
| 500 | 1149 | 607 |

Using the DataFilesGenerator class, I tried different numbers of orders to observe the benefits of multi-threading in increasing speed of processing. The settings I used to create the data sets were as follows:

* 500 Orders
* 100,000 Maximum number of orders
* 50 Minimum number of orders
* 100 Change in number of orders

Based on my results, we can see that initially, at lower number of orders(1 & 50), the single threaded method was faster in processing all the orders than the multi-threaded process. This could be attributed to the memory require to manage threads being more than just handling each process one by one. But beginning from 100 orders and more, the processing time required for the single thread process increases at a larger interval in comparison to the processing time with the multithreaded approach. As a result, it can be concluded that at higher numbers orders, the multithreaded approach is more efficient in order processing.