Parsing Optimization (SAX instead of DOM)

1st technique:

* One of the thing we notice that when we are inserting values into the database and printing out statement in the console, it causes the program to run relatively slow. Once we got rid of the print statement, it ran quickly. As a result, we got rid of the print statement in the program after we ensure everything is working fine and finish debugging the program.
* Parsing was never an issue since it ran relatively fast. We notice that creating a connection to the database and executing the query was what making the program slow. Since we know that this was the problem, we decided to connect to the database as less as possible. One of the way that we try to optimize is by not checking duplicate or anything. If we check duplicate for every item, we need to create a new connection to the database which will create delay. Thus, when we are unsure due to missing data, we simply disregard it and move on to the next item.

2nd technique:

* Another thing that we did was that when we parse name from the xml, we split the name to tokens with the delimiter of space and set the limit as 2. Since we know that some actors have middle name while other don’t, it creates inconsistent if we don’t limit by two so we decided to combine the middle name with the last name. Setting the limit by two, the splitter will only spilt on the first space, leaving the first token as first name and second token as last name. Doing so, save us from worrying if the Actor has a irregular pattern of name and from using an extra if loop. Although the time optimization for this technique is not significant but it will make a difference in the long run.