1. Team details: Clearly state the names and net ids of your team members (there are 2 of you). **Netanya Trowers (nmt71) and David Dawoud (dbh72)**
2. Collaboration: Who did you collaborate with on this project? What resources and references did you consult? Please also specify on what aspect of the project you collaborated or consulted. **For implementing the ability to timeout out in ts1 and ts2 we referenced recitation slides. To not stall if nothing is received from either ts1 or ts2 or both, we raised an error which is also learnt from the notes. As a team we collaborated on ts1, ts2, rs and client implementation.**

1. Discuss how you implemented the RS functionality that tracks which TS responded to a given query or timing out if neither TS responded. Please be clear and specific.

**By using a try block that would send the query received from rs to ts1 and ts2, if there was an empty line then it would raise socket.error. Then, another try block to first receive from ts1 and save it and continue. However, if what was received was an error, it would raise the exception socket.timeout to timeout of ts1 then begins another try block that would send the query to ts2 and save what was received and continue if it was not an error. If it was an error however, another socket.timeout is raised and the last try block initiates as it sends to client the notification of timing out. Another final try block is used to send to client the saved result from when either ts1 or ts2 did in fact return a result and not an error.**

1. Is there any portion of your code that does not work as required in the description above? Please explain.

**It works accordingly.**

1. Did you encounter any difficulties? If so, explain.

**The difficulties lied in trying to timeout ts1 and ts2 as we weren’t sure how to enable it but with research on socket.timeout and select() we were able to.**

1. What did you learn from working on this project? Add any interesting observations not otherwise covered in the questions above. Be specific and technical in your response.

**We learned the distinction between non-blocking and blocking sockets by actually visualizing it due to this project and how in doing non-blocking affects the interaction between rs, ts1 and ts2, for example; ts1 and ts2 has their setblocking to false to enable the ability to keep sending them data from rs, in the buffer and thus needing to enable timeouts and raising errors.**