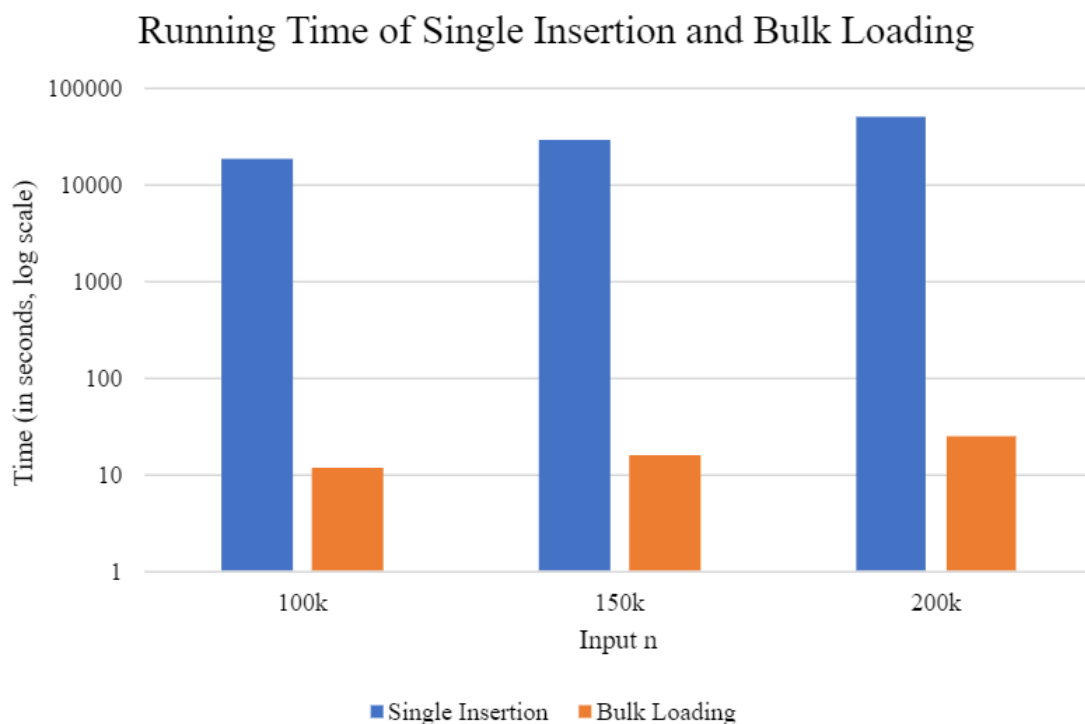


In my java file called Driver I have multiple methods that are used throughout the interface. Every method has its own connection to the MySQL data base. To access a different data base the jdbc (Java Database Connectivity) must be changed. The first method is called insert1 which is used within the file application2 to preform single insertion. The second method is called insert2 which is used within the file application3 to preform bulk loading. The third method is called delete and is used in application4 to delete all data from a table. The last method is called query which is used in application5 to send a query to MySQL and get the results back. To complete most of these methods I needed to use the java SQL imports. The file called application is used as the main page for the application and allows for the user to choose any of the four actions. To create the GUI, I used JFrame for all the files. To print out any errors I simply request the error that is sent from MySQL to be printed out on a label in the application the user is on. I also have it show how long each action took in the application. The methods for single insertion, bulk loading and delete all used the java SQL import method called executeUpdate to send the correct code to MySQL. The query method however used the java SQL import method called executeQuery to complete its task.



As can be seen in the graph above the Single Insertion took way longer than the Bulk Loading. The Single Insertion took hours with 100k taking 5 hours, 150k taking 8 hours and 200k taking 14 hours. The Bulk loading took only seconds with 100k taking 12 seconds, 150k taking 15 seconds and 200k taking 25 seconds. From this data we can safely say that if we are going to be inserting large amounts of data it would be better to use the bulk loading instead of the single insertion.