1. My evaluation metric implementations reside in the main method of the Evaluation java file located in *src*. There are some helper methods below the main method.
2. I used java for the entirety of this assignment. The data set requires pooling multiple pieces of information together in a set. I found the easiest way to accomplish this would be to implement two tuple classes. The first accepted two data types and was used as a value mapping to a query integer. This way I could parse and store the qrels information to determine what documents are relevant to what queries. The second tuple class accepts three data types and was used as a value to map again to query id integers. The difference between the two is that the latter allowed me to store the document id, rank and score all to a query. The key mapping feature that allowed me to map duplicate keys to different values was the LinkedHashMultiMap, Using these two tuples and the map, I was able to manipulate the given data to calculate the various metrics.
3. The LinkedHashMultiMap described above was taken from the google-collections-1.0-rc2.jar file. It was crucial in providing a feasible, efficient approach to storing the query and relevance information where multiple queries of the same value needed to be represented as duplicate keys mapping to different values. This is effectively accomplished by treating the values ascribed to the duplicate keys a set or collection of values mapping to a single key in the LinkedHashMultiMap. Other than this external library, I used java's io and util libraries to open and operate on the data files to read them in efficiently using a BufferedReader.
4. 
5. 
6. This is a puzzling question because the underlying factors at play are asymptotic in that the denominator results in values of 0 when determining precision. Therefore, it is unclear as to how to reevaluate the metric. In other words, should the proportion of this query's metric score with respect to the retrieved documents be 0 or infinity. The latter would more severely impact the average when calculating MAP.