4. Writeup (10 points)

Please describe and/or draw your database design. This is so we can understand your implementation as close to what you were thinking. Explain your design choices in creating new tables. Also, describe your thought process in deciding what needs to be persisted on the database and what can be implemented in-memory (not persisted on the database). Please be concise in your writeup (< half a page).

In the beginning I created tables Users, Itineraries, and Reservations. I actually did not know what attributes to be included in the table would work the best. I basically added attributes along when I was implementing the functions such as book, reservations, pay and cancel. Along the process, I decided to add a TempFlight table to keep track of the flights information after search. The TempFlight table would enable me to keep track of the capacity status since I should not modify the FLIGHTS table, and the TempFlight table would speed up my afterward SQL queries without having to get data from the humungous FLIGHTS table.

At first I just stored all the search results in a stringbuffer and showed the users the search results by returning the stringbuffer. Afterwards I realized that it's better to store them in a table, for example, itineraries, so that I can order them by travel time afterwards. It took me a while to figure out how to keep track of relationship between itinerary ID and the sequence it showed to users. I later used a in-memory Java map to keep track of the relationship. This map is used in book and reservation to retrieve the correct itinerary ID.