1. Please list out changes in the directions of your project if the final project is different from your original proposal (based on your stage 1 proposal submission).
    a. From a high-level perspective, our final project was pretty similar to our original proposal.
2. Discuss what you think your application achieved or failed to achieve regarding its usefulness.
    a. Our application achieved the goal of providing the user's with recommendations of which airlines to take for their upcoming trip based on their travel details.
3. Discuss if you changed the schema or source of the data for your application
    a. We changed the schema for one table slightly (explained in the following question).
4. Discuss what you change to your ER diagram and/or your table implementations. What are some differences between the original design and the final design? Why? What do you think is a more suitable design?
    a. While working on the project, we changed our table implementation to index each user's flight information based on their first and last name instead of an auto-generated customer ID. We did this because it made it easier for us to code the backend that interfaced with the database. However, assigning a unique customer ID to each user would be better because it would allow our application to distinguish between users with the same first and last name. We also changed our original database design to contain an additional table to keep track of general information about each airline (avg delay, cancellation rate, reliability). We used this table for our stored procedure.
5. Discuss what functionalities you added or removed. Why?
    a. We added a functionality that allows users to delete all their flight information because we felt that we should allow users the ability to remove their data if they wanted to. We also added a function that displays the current most reliable airlines on the main page of our application. We added this function because we thought it would be nice to show the users the overall most reliable airlines based on our data.
6. Explain how you think your advanced database programs complement your application.
    a. Our advanced database programs provide the main functionality of our application, which is to display the airlines with the least average flight delay time filtered by user input (travel month, origin/destination airport). Our other advanced database programs serve our stored procedure, which updates our list of overall most reliable flights based on new data that the users provide.

7. Each team member should describe one technical challenge that the team encountered. This should be sufficiently detailed such that another future team could use this as helpful advice if they were to start a similar project or where to maintain your project.

   a. Wenyu: It's hard to debug the application while running it, so test each SQL query on MySQLWorkbench before adding that query to the backend of the application. Also log the output of each query in the console to help with debugging.
   b. Annie: When we set the data type for the "cancellation status" attribute of our flight data to BIT (0 or 1), it set all the values to 0 when we uploaded the data, even though some flights had a cancellation value of 1. When we changed the data type to INT, it fixed this issue.
   c. Gayathri: Make sure to pull from your repository every time before you start working on your project. If you don't do that, there will be merge conflicts when you push your commit, and the merge conflicts are annoying to resolve.

8. Are there other things that changed comparing the final application with the original proposal?

   a. We changed the layout of our application to include multiple pages, instead of having all the functionalities on one page. We created a navigation menu on the main page to help the users navigate to the other pages.

9. Describe future work that you think, other than the interface, that the application can improve on

   a. In the future, we should have a way of verifying if the data contributed by the users is legit. This is important because we don't want the users to upload lots of fake data to mess with our system. We can also try to use a MongoDB database design instead of SQL, because it might improve the efficiency of the queries in our application.

10. Describe the final division of labor and how well you managed teamwork.

   a. Every team member worked on a similar number of functionalities. Both the backend and frontend component of each functionality was programmed by the same person. We decided how to divide up the work during our meetings.