Professional Behavior

1. **Interacting Professionally with the Instructor:**
   My interaction with the Instructor were limited. The main feedback I received from the Instructor were when he graded my fluency topics. I did post a question in the Technology board where I posted a question on JSON. Here was the question I asked: *"For the JSON topic, do we need to actually send a request from a website that is entirely JSON, or can we just ping a JSON file that is hosted online?"* I followed that up with the following question *"Ok, I've got it reading a JSON file... Are we ok just to include the JSON file for the fluency review?"* The reason I posted the question on the discussion board was so others in the class that might have the same question would also be able to see the response from the Instructor.

   Another example of an interaction was some confusion around where to upload the final personal application. I had let him know of all the topics included in my project as well as where he could find my fluency template once it was time to grade the final. He asked that I move the template to the Code Topic assignment folder, and I let him know that I had uploaded to that assignment folder, as well as the Project Assessment assignment folder.

2. **Interacting Helpfully and Professionally with Other Teams:**
   This was a difficult element of the class. With the format of the class being divided into teams, it made it difficult to really interact with other teams. With that said, I found the Technology discussion board a great way to interact with other people in the class who were not a part of your team. I was able to do this by posted a question about the JSON topic, as well as posting a website that included some helpful information about HQL queries. I did response to two other threads where class members were asking questions about Hibernate as well as help setting up Tomcat. In both cases I was able to provide some helpful hints to get their environment setup.

3. **Supporting the Team by Teaching, etc.**
   With a team size of just 3, I was provided multiple times to teach my fellow teammates on my assigned topics. Here are the following topics and videos where I presented to the team:
   JSON - http://youtu.be/0S3pvUVN9EU (Starts at 0:00)
   Collections - https://www.youtube.com/watch?v=IxZu5VG4ku8 (Starts at 0:00)
   Servlets - https://www.youtube.com/watch?v=qHtma-LxPOw (Starts at 13:34)
   HttpURLConnection - http://youtu.be/z2hrd2HgCh4 (Starts at 0:00)
   JUnit - https://www.youtube.com/watch?v=o2ORJzsXzrc (Starts at 0:00)

Besides the videos, I would often provide links to our Google+ community site where the team could find help information on topics. Here is the link to our site where you can see my participation: https://plus.google.com/u/1/communities/109721001854630278851/stream/84e470f8-b306-4706-9c51-b614a4c02d08

4. **Self-Reflection/Meta-Cognition:**
   Reading through my weekly journal, I see how putting your mind towards a goal can help you achieve the impossible. In the previous Java class I did not feel like I learned a whole lot, not even the basics. However, after taking this class, and having to really learn topics on my own, I felt like I was pushed to the point that I actually learned more than just the basics from this course. While I don't feel like I could go to work as a full time Java developer, I do feel like I was able to learn a lot of the concepts and basic skills to at least understand and have a conversation with a full time Java developer! Also, seeing all the coding topics complete was a huge win for me and gives me a sense of pride and accomplishment.

   Now as I take on a new project or assignment, I approach the required learning differently. Having gained the skills and perspective of learning the topic on my own, I can apply those same principles, and learn new things at a much deeper level. For example, when working on the Hibernate topic, I had to figure out how to setup a MySQL server and have that integrate with Netbeans and Apache/Tomahawk to complete the assignment and have my code actually write data to my database. That entire process taught me how to dive into a problem, find a solution, and achieve a suitable outcome. I can then apply those same principles/discoveries to any other aspect in my life and achieve the same successful outcome.

   When it came time to work on the final project, a big aspect of the application needed to be built in Android Studio. Since we had no training or any fluency reviews using that tool, this was a huge ask that caused a moment of panic. However, after watching the video from the instructor and watching other YouTube videos, I was able to put together a UI front end that would interact with my web server. After thinking about why this was added to the final, it dawned on me that this was done to test our ability to work on a new technology with a looming deadline. This is done all the time in the business world and is something we can count on doing if we work in the technology field. I actually found that I was able to take on this challenge, find success with it, and ultimately found something I could add as a resume builder.

5. **Team's Agile Methodology:**
   Our team used the Scrumban methodology. This methodology utilized weekly sprints from scrum, allowing us the flexibility to what we wanted to do in each weekly sprint cycle. We also took the concept of "to-do", "doing", and "done" tasks from Kanban. This allowed us to pick from our own theoretical "to-do" list and focus on one topic at a time to avoid multi-tasking and time wasting.

   How I have adapted the team methodology of Scrumban is by implementing the "to-do", "doing", and "done" tasks while completing my fluency topics. I kept all unfinished topics in my "to-do" list, moving one topic over to the "doing" list at a time and working on that topic until I could turn in topic and move it over to the "done" list. If a topic needed a revision, this would be moved to the top of the "to-do" list, where it would then be worked on in the "doing" list and then moved back to "done" once complete.

   We also meet weekly in our team sprint meeting and used those opportunities to discuss outstanding topics where others may need additional help.

6. **Professional Scheduling Group:**
   Originally our team consisted of 5 members. Through the course of the semester, this number quickly decreased to 3. Instead of skipping topics, each of us took on a large presentation size, often leaving little to no preparation time. However, each week a proper presentation was given, and everyone had the opportunity to teach and learn all the topics. My first presentation I had to do from a hotel room since I was traveling for work. I was not able to meet with the team during the normal meeting times due to my involvement with work, but I was able to record my presentation later that night and still post the content to our community board only a couple of hours after our normal meeting time. Here is a link to our topic assignment sheet:
   https://docs.google.com/spreadsheets/d/13z76ml_1y6SwY4i9IHT069WhsgjSre00XqOVB0wk0a4/edit?usp=sharing