Nathan Conn
Capstone Project 1: Data Wrangling

1. **Create a Google Doc (1-2 pages) describing the data wrangling steps you took to clean the dataset. Include answers to these questions in your submission:**
    1. **What kind of cleaning steps did you perform?**
    2. **How did you deal with missing values, if any?**
    3. **Were there outliers, and how did you handle them?**
2. **Submit a link to the document.**
3. **Discuss it with your mentor at the next call.**
4. **Revise and resubmit if needed.**
5. **Convert the final document to a .pdf and add it to your GitHub repository for this project. This document will eventually become part of your milestone report.**

As the raw data for my capstone project was found via Kaggle, I was fortunate to inherit a very clean dataset. The set does however require a large amount of additional munging before it can be trained. This document outlines the steps I've taken to complete this aspect of the project.

The first data wrangling challenge I faced was the joining and onehot encoding of the JSON file containing individual hero data with the match data. This allowed me to identify which heroes were used by each player in each match.

Another important step in the data wrangling process for this project was transposing the hero json file, (by first importing it and converting to data frame), so that it could be merged with the match data. Each match contained the ten ids of the heroes used, which correlate to the hero JSON file, which contains twenty nine attributes of each hero. To later identify which hero attributes most impacted match outcomes, I'd need to insert the twenty nine respective attributes of each of the ten heroes used. This required the following steps:

1. Rerun the read methods for the match and hero data, to start from a fresh dataframe perspective, (pd.read_csv() for the match data and pd.read_json() for the hero data).
2. I then created a list of simplified player labels so that the hero attributes could be more easily associated and evaluated based on the player and team they represented, which I did by creating a simple list:

list_of_all_players = ['r1', 'r2', 'r3', 'r4', 'r5', 'd1', 'd2', 'd3', 'd4', 'd5']

3. I then created a loop which would:

a. Iterate over the player labels in the list I created, “list_of_all_players” and print a string confirming that each individual player column from the match data was being merged with hero data and that the new player label had been applied:

```
for player in list_of_all_players:
  print('Joining data for player: '+ player)
```

b. Create a local copy of the hero data that had the current loop iteration(player) labels added

```
hero_local = hero_df.copy()
  hero_local.columns = player + '_' + hero_df.columns
```

c. Merge the local hero copy with the match data:

```
df = df.merge(hero_local, left_on=[player + '_hero'], right_on=[player + '_id'], how='left')
```

These manipulation steps allowed me to begin the analysis and storytelling steps of my project.