**Capstone Project Preliminary Analysis Template**

**Name: Julian Bombard**

**Project Title:** *ATP Tour Analysis*

**Project Objective:** *My goal is to discover which statistical and categorical variables are the most significant predictor of winning matches. I want to provide insight to tennis players, coaches, and pundits/journalists. Ideally, I would like to be able to predict who will win a match before it even starts based on their previous outcomes, but I’m not sure I know enough about machine learning to build those models in the next two weeks.*

**Data Collection Source:** [**https://github.com/JeffSackmann/tennis\_atp**](https://github.com/JeffSackmann/tennis_atp)

* Downloaded the last 10 years of match data from this repository

**Methodology:**

*Each year of matches was an individual csv file. Each csv file had the exact same column names, so I built a single table in SQL and imported all 10 csv into that table. Then, I loaded that data into a Jupyter notebook using pd.read\_sql\_query. I ran my custom data\_check function on the dataframe to gather information about the size of the dataframe, duplicates and missing values, data types, and descriptive statistics. I had to change many data types, including most of the statistical columns, which were originally being read as objects after importing from SQL. I also had to normalize the date format and change it to a datetime data type. I added additional columns to the dataframe, such as month and year (which I extracted from the date column), and created a custom and unique match\_id column, combining tournament ID with match number. In some years, the US Open was titled as Us Open. I converted all of them to US Open. Since each row is a single match with a winner, loser, winning players statistics, and losing players statistics, I wanted to build some more tables. I built a win-loss table, where each player gets one row that includes their win loss record, matches played, and winning percentage. I also made a player\_match\_stats table, where I took each match and effectively split it in two, getting two rows for each match, with just one player and his performance, as well as a ‘won’ boolean column. This helps me compare statistics between players regardless of whether they won or lost. I dropped all rows that included any missing values from this table. I then built a player\_summary table by condensing the player\_match\_stats table by grouping by player to get each player's average statistics from all matches. Since each row was a single player, I merged this table with my win-loss table.*

**Findings:**

1. **Data Size:** *There are 27672 rows and 49 columns in the original data set*
2. **Missing Values:** *There are missing statistics from matches that are walkovers (when a player has to surrender the match before it even begins, therefore no statistics are recorded). There are also missing statistics from Davis Cup matches, which is a team tournament with a different format. I am excluding both walkovers and Davis Cup matches from my statistical analysis. When I filter those categories out, there are 1737 rows that do not have the number of minutes in the match. There are 118 rows that have none of the match statistics. In my custom table ‘player\_match\_data’ and ‘player\_summary’, i have dropped any rows with missing data.*
3. **Duplicates:**  *There are 0 duplicate rows. This was detected using df.duplicated().sum() within my data\_check function that I built*
4. **Dates:***The dates range from 1/4/2015 to 12/18/2024 (the start of the 2015 season through the end of the 2024 season)*
5. **Additional Findings:** *I ran several correlations between certain statistics and winning, and then I plotted them. I will continue to observe these findings next week*

**Attach any SQL or Python code in a separate file**