

2019 NBA Hackathon Application – Basketball Analytics Prompt



Task: Use the attached data and instructions to calculate offensive rating and defensive rating for each player in each game from the 2018 Playoffs.

Offensive Rating is defined as the team points scored per 100 possessions while the player is on the court. Defensive Rating is defined as the number of points per 100 possessions that the team allows while that individual player is on the court. A possession is ended by (1) made field goal attempts, (2) made final free throw attempt, (3) missed final free throw attempt that results in a defensive rebound, (4) missed field goal attempt that results in a defensive rebound, (5) turnover, or (6) end of time period.

Note: When a player is substituted before or during a set of free throws but was on the court at the time of the foul that caused the free throw, he is considered to be on the court for the free throws for the purposes of offensive and defensive rating. A player substituted in before a free throw but after a foul is not considered to be on the court until after the conclusion of the free throws.

Ex: In a Bucks vs. Nets game, Kyle Lowry commits a shooting foul on Kristaps Porzingis. Porzingis shoots and scores the first free throw, and then Jonas Valanciunas is substituted in for Serge Ibaka. Porzingis makes the second free throw. For the purposes of offensive and defensive ratings, Ibaka (because he was on the court at the time of the foul) receives a value of 2 toward his defensive rating for those free throws while Valanciunas is not considered on the court until after the free throws are completed. Porzingis receives a value of 2 toward his offensive rating.

This folder includes three data sets: `Event_Codes.txt`, `Play_by_Play.txt`, and `Game_Lineup.txt`. Please note that each question is permitted a maximum of two file attachments. Please submit your answer in a `.csv` file and save your code, spreadsheets and all other work in a `.zip` file.

Please submit a `.csv` file titled “Your_Team_Name_Q1_BBALL.csv” substituting in the name of your team for “Your_Team_Name”. Please save as a `.csv`. The final product should have 4 columns. Column 1: `Game_ID`, Column 2: `Player_ID`, Column 3: `OffRtg`, Column 4: `DefRtg`.

Provided Data:

- `Event_Codes.txt`
 - o This dataset provides look up values for the event message types and action types found in the play by play dataset. Each code is converted to an English language description of the event.
- `Game_Lineup.txt`
 - o This dataset provides the start of period player availability.
 - `Game_id` – a unique game code for each game
 - `Period (Quarter)` – the associated period of the line up (overtime periods are indicated by values greater than 4)
 - `Person_id` – a unique identifier for each player
 - `Team_id` – a unique identifier for each team
 - `Status` – a variable indicating whether the player is active (A) or inactive (I)
- `Play_by_Play.txt`
 - o This dataset provides play by play information on the event level for each game.
 - o To properly sort the events in a game, use the following sequence of sorted columns: `Period` (ascending), `PC_Time` (descending), `WC_Time` (ascending), `Event_Num` (ascending)
 - `Event_Num` – an ordered counter for each event in a game. Note, this number may not be perfectly sequential so please use the sorting methodology outlined above
 - `Event_Msg_Type`, `Action_Type` – coded descriptions of what happened during the event (see the `Event_Codes.txt` dataset to see the codes)
 - `WC_Time` – the in-arena time of the event in Unix format. It is coded as tenths of a second
 - `PC_Time` – the time on the game clock in tenths of a second (e.g. 7200 corresponds to 720 seconds/12 minutes remaining in the quarter)
 - `Option1` – on a shot attempt, this column will tell you the point value of the shot
 - On free throw attempts, if the value in this column is 1, it means it was a made free throw, otherwise, it was missed
 - `Person1`, `Person2` – the `Person_id`'s of the players who are directly associated with the event (e.g. if the event is an assisted made basket, `Person1` is the shot maker and `Person2` is the player who assisted)
 - In the case of a substitution, the `Event_Msg_Type` will be 8, `Person1` will be the `Person_id` for the player leaving the game, and `Person2` will be the `Person_id` for the player entering the game
 - `Team_id` – in most scenarios, this is the `Team_id` associated with the `Person1` column. However, there are instances when this is not the case. To accurately and consistently identify a player's team, we suggest merging with the `Game_Lineup` dataset on the `Person1` and `Person2` columns.