

10 – LEAGUE TABLE

Quite often as a programmer you are given a set of data that requires processing. Although the subject of the data may not interest you, what is important is that the outcome is accurate.

In this challenge you have been provided with a csv file containing some of the data generated from the 16-17 premier league football season. CSV stands for Comma Separated Value, an example of which is below:

```
13/08/2016,Crystal Palace,West Brom,0,1,A,C Pawson,14,13,4,3,12,15,3,6,2,2,0,0
13/08/2016,Everton,Tottenham,1,1,D,M Atkinson,12,13,6,4,10,14,5,6,0,0,0,0
```

Programs can read these files easily into lists or arrays because the comma separates each value into an index. As an example, if each line was saved as a list called row then row[3] would be '0' on the first line and '1' on the second line.

To access the csv file and loop through it row by row, the following code could be used:

```
import csv

with open('Premier 16-17.csv') as csvfile:
    spamreader = csv.reader(csvfile, delimiter=',')
    for row in spamreader:
```

The main task of this challenge is to process the data file and produce the following pieces of information:

1. A league table showing: wins, draws, losses, goal difference and total points. Team with most points at the top.
 - a. Wins are three points for the winning
 - b. Draw is one point for both teams
 - c. Loss is 0 points for the losing team
 - d. Ties in total points use goal difference to separate (scored – conceded)
2. Single statistics for the following information points:
 - a. Most accurate team (Total shots on target / total shots)
 - b. Least accurate team (Total shots on target / total shots)
 - c. Dirtiest team (most fouls per game)
 - d. Cleanest team (least fouls per game)
 - e. Referee with highest card average per game (Yellows +1, Red+2)
 - f. Referee with lowest card average per game (Yellows +1, Red+2)