Assignment 4 – AK/ITEC1620 3.0

**Problem**

In a certain sports league, a group of teams plays through a Schedule of Games (see APIs below -- Schedule.html and Game.html). At the end of this Schedule, they want to determine the winner.

To determine the winner, you will have to determine for each team how many Games they won, lost, and tied.  Assuming 2 points for a win, 1 point for a tie, and 0 points for a loss, you can then determine how many points they scored during the season.  The team(s) with the most points is then the winner.

**Required Classes (save the class files below to your working directory):**

Game.class   
Schedule.class

**Sample Program Run:**

You should build your application to take no input and produce output similar to the following.  If your application works correctly, then your program output should look like the following:

Vancouver - 3 wins, 2 losses, 0 ties = 6 total points   
Calgary - 2 wins, 3 losses, 2 ties = 6 total points   
Edmonton - 1 wins, 3 losses, 2 ties = 4 total points   
Toronto - 2 wins, 3 losses, 2 ties = 6 total points   
Ottawa - 4 wins, 3 losses, 1 ties = 9 total points   
Montreal - 4 wins, 2 losses, 1 ties = 9 total points

The season winner(s) with 9 points   
Ottawa   
Montreal

Games’ Results

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Vancouver 5  Calgary 3  Vancouver 2pt  Calgary 0pnt | Calgary 4  Edmonton 4  Calgary 1pt  Edmonton 1pt | Edmonton 3  Toronto 3  Edmonton 1pt  Toronto 1pt | Ottawa 2  Toronto 4  Ottawa 0pt  Toronto 2pt | Ottawa 6  Montreal 3  Ottawa 2pt  Montreal 0pt |
| Montreal 5  Calgary 4  Montreal 2pt  Calgary 0pt | Vancouver 4  Toronto 3  Vancouver 2pt  Toronto 0pt | Calgary 3  Ottawa 4  Calgary 0pt  Ottawa 2pt | Edmonton 2  Vancouver 3  Edmonton 0pt  Vancouver 2pt | Toronto 1  Calgary 4  Toronto 0pt  Calgary 2pt |
| Ottawa 5  Edmonton 3  Ottawa 2pt  Edmonton 0pt | Montreal 4  Toronto 4  Montreal 1pt  Toronto 1pt | Calgary 3  Ottawa 3  Calgary 1pt  Ottawa 1pt | Edmonton 2  Montreal 4  Edmonton 0pt  Montreal 2pt | Toronto 1  Montreal 3  Toronto 0pt  Montreal 2pt |
| Ottawa 5  Vancouver 4  Ottawa 2pt  Vancouver 0pt | Montreal 4  Ottawa 3  Montreal 2pt  Ottawa 0pt | Vancouver 3  Toronto 4  Vancouver 0pt  Toronto 2pt | Montreal 2  Edmonton 3  Montreal 0pt  Edmonton 2pt | Ottawa 1  Calgary 4  Ottawa 0pt  Calgary 2pt |
|  |  |  |  |  |
| Vancouver  Wins 3  Loss 2  Ties 0  Total pts 6 | Calgary  Wins 2  Loss 3  Ties 2  Total pts 6 | Edmonton  Wins 1  Loss 3  Ties 2  Total pts 4 | Toronto  Wins 2  Loss 3  Ties 2  Total pts 4 | Ottawa  Wins 4  Loss 3  Ties 1  Total pts 9 |
| Montreal  Wins 4  Loss 2  Ties 1  Total pts 9 |  |  |  |  |
|  |  |  |  |  |

import java.lang.\*;

public class LabAssignment4

{

public static void main(String[] args)

{

int topScore = 0;

Schedule seasonSchedule = new Schedule(); // a new schedule object that store results for GAMES games

int numOfGamesInSchedule = seasonSchedule.GAMES; // number of games in the schedule

String [] teamsInSeason = seasonSchedule.getTeams(); // Names of teams in schedule

teamData[] teamRecord = new teamData [teamsInSeason.length]; //object to store "database for each team

for (int i=0;i<teamRecord.length && i<teamsInSeason.length ;i++ ) //Process to store data in teamRecord

{

teamRecord[i] = new teamData();

teamRecord[i].teamName = teamsInSeason[i];

teamRecord[i].numOfWins = 0;

teamRecord[i].numOfLosses = 0;

teamRecord[i].numOfTies = 0;

teamRecord[i].totalPoints = 0;

}

Game [] gameResults = new Game [seasonSchedule.GAMES]; // print outs all games results

for (int i =0; i < seasonSchedule.GAMES; i++)

{

gameResults[i] = seasonSchedule.getGame(i);

String homeTeam = gameResults[i].getHomeTeam();

int homeScore = gameResults[i].getHomeScore();

String awayTeam = gameResults[i].getAwayTeam();

int awayScore = gameResults[i].getAwayScore();

if (homeScore > awayScore)

{

{

int k = 0;

while (homeTeam != teamRecord[k].teamName)

k++;

teamRecord[k].numOfWins += 1;

}

{

int k = 0;

while (awayTeam != teamRecord[k].teamName)

k++;

teamRecord[k].numOfLosses += 1;

}

}

else if (homeScore == awayScore)

{

{

int k = 0;

while (homeTeam != teamRecord[k].teamName)

k++;

teamRecord[k].numOfTies += 1;

}

{

int k = 0;

while (awayTeam != teamRecord[k].teamName)

k++;

teamRecord[k].numOfTies += 1;

}

}

else

{

{

int k = 0;

while (awayTeam != teamRecord[k].teamName)

k++;

teamRecord[k].numOfWins += 1;

}

{

int k = 0;

while (homeTeam != teamRecord[k].teamName)

k++;

teamRecord[k].numOfLosses += 1;

}

}

}

for (int i=0;i<teamRecord.length ;i++ )// calculating total points

{

teamRecord[i].totalPoints = (2\*teamRecord[i].getNumOfWins()) + (teamRecord[i].getNumOfTies());

}

for (int i=0;i<teamRecord.length ;i++ )

{

if (teamRecord[i].getTotalPoints() > topScore)

{

topScore = teamRecord[i].getTotalPoints();

}

}

for (int i=0;i<teamRecord.length ;i++ )

{

System.out.println(teamRecord[i].getTeamName() + " - " + teamRecord[i].getNumOfWins() + " wins, "

+ teamRecord[i].getNumOfLosses() + " losses, " + teamRecord[i].getNumOfTies() + " ties = "

+ teamRecord[i].getTotalPoints() + " total points");

}

System.out.println("The season winner(s) with " + topScore+ " points");

for (int i=0; i<teamRecord.length;i++ )

{

teamRecord[i].getTotalPoints();

if (teamRecord[i].getTotalPoints() == topScore)

{

System.out.println(teamRecord[i].getTeamName());

}

}

}

}