/\*\* =======================================================================

\* Class:RugbyPlayer ExT.T Pg.N/A Author: Yin Linhai

\* Version:001Date:Mar 19, 2014

\*

\* A program which helps manage a rugby team

\*

\* Course:Computer Science 201Teacher:Mr Blakey

\* School:Sir Winston Churchill High School, Calgary, Alberta, Canada

\* Language: Java SE 7.0Target Operating System: Java Virtual Machine

\* System:Intel Celeron 3GHz running under Windows 7 IDE: Eclipse 4.2

\*========================================================================\*/

**RugbyPlayer Class**

**package** test\_8;

**import** java.util.ArrayList;

**public** **class** RugbyPlayer {

//constructor

**public** RugbyPlayer(**int** number, String first, String last, **boolean** injured, ArrayList<Integer> points) {

studentNumber=number;

firstName=first;

lastName=last;

injury=injured;

pointsScored = points;

**for**(**int** x = 0; x<pointsScored.size(); x++) {

totalPoints+=pointsScored.get(x);

}

}

//get methods

**public** **int** getNumber() {

**return** studentNumber;

}

**public** **int** getPointsGame(**int** x) {

**return** pointsScored.get(x);

}

**public** **int** getTotalPoints() {

**return** totalPoints;

}

//add points

**public** **void** addPoints(**int** points) {

pointsScored.add(points);

}

//set points

**public** **void** setPoints(**int** points, **int** game) {

**try** {

pointsScored.set(game, points);

}

**catch**(RuntimeException e) {

}

}

//more get methods

**public** String getFirstName() {

**return** firstName;

}

**public** String getLastName() {

**return** lastName;

}

**public** **boolean** getInjured() {

**return** injury;

}

**public** **int** getGamesPlayed(){

**return** pointsScored.size();

}

//set injury state (true for injured)

**public** **void** setInjured(**boolean** injured) {

injury=injured;

}

**private** **int** studentNumber, totalPoints=0;

**private** ArrayList<Integer> pointsScored = **new** ArrayList<Integer>();

**private** String firstName;

**private** String lastName;

**private** **boolean** injury;

}

**Tester Class**

**package** test\_8;

**import** java.util.ArrayList;

**import** java.util.Scanner;

**public** **class** Main {

**public** **static** **void** main(String[] args) {

**int** topNumber=0, lowNumber = 0;

**double** topPoints=0, lowestPoints=9999999;

String topFirst=**null**, topLast=**null**;

String lowFirst=**null**, lowLast=**null**;

**boolean** end = **false**, tie = **false**;

ArrayList<RugbyPlayer> csTeam = **new** ArrayList<RugbyPlayer>();

Scanner scan = **new** Scanner(System.*in*);

Scanner scan1 = **new** Scanner(System.*in*);

System.*out*.println("Create the team");

//fill team loop

**do** {

**int** number;

String first, last;

**boolean** status;

ArrayList<Integer> points = **new** ArrayList<Integer>();

**try** {

//get player number

System.*out*.println("What is the number of the new Player?\nPress Q to quit");

number=scan.nextInt();

//get player name

System.*out*.println("What is the Player's Name?\n(First Last)\nPress Q to quit");

first = scan.next();

last = scan.next();

//get whether the player is injured

System.*out*.println("What is the players Status?\n(injured or uninjured)");

System.*out*.println("Press Q to quit");

//decision on status

**if**(scan.next().equalsIgnoreCase("injured")) {

status=**true**;

} **else** {

status = **false**;

}

**int** z=1;

**boolean** end2 = **false**;

**while**(!end2) {

**try** {

System.*out*.println("How many points did the player score for game " + z + "?");

System.*out*.println("Press Q to stop adding scores");

points.add(scan.nextInt());

z++;

}

**catch**(RuntimeException e) {

**if**(scan.next().equalsIgnoreCase("q")) {

end2=**true**;

}

}

}

//make and add to arraylist

RugbyPlayer play = **new** RugbyPlayer(number, first, last, status, points);

csTeam.add(play);

}

**catch**(RuntimeException e) {

**if**(scan.next().equalsIgnoreCase("q")) {

end=**true**;

}

}

}**while**(!end);

end = **false**;

//loop for updating every game;

System.*out*.println("Updating player data for a game");

**for**(**int** x = 0; x<csTeam.size(); x++) {

**while**(!end) {

RugbyPlayer temp = csTeam.get(x);

**boolean** status;

//try to add points

**try**{

System.*out*.println("How many points did player number " + temp.getNumber() + " get?");

temp.addPoints(scan.nextInt());

System.*out*.println("Did the player get injured this game?\n(y/n)");

//decision on status

**if**(scan.next().equalsIgnoreCase("y")) {

status=**true**;

} **else** {

status = **false**;

}

temp.setInjured(status);

end = **true**;

}

**catch**(RuntimeException e) {

String bug = scan1.nextLine();

}

}

end = **false**;

}

//ask if you want to remove a player

System.*out*.println("Would you like to remove a player?\n(y/n)");

//if you want to remove someone

**if**(scan.next().equalsIgnoreCase("y")) {

System.*out*.println("Who would you like to remove?");

**while**(!end) {

System.*out*.println("(Player Number please)");

**try** {

**for**(**int** x = 0; x<csTeam.size(); x++) {

RugbyPlayer temp = csTeam.get(0);

**if**(temp.getNumber()== scan.nextInt()) {

csTeam.remove(x);

}

}

end = **true**;

}

**catch**(RuntimeException e) {

String bug = scan1.nextLine();

}

}

end = **false**;

}

**while**(!end) {

System.*out*.println("Would you like to add a player?\n(y/n)");

String in = scan.next();

//if you want to remove someone

**if**(in.equalsIgnoreCase("y")) {

**int** number;

String first, last;

**boolean** status;

ArrayList<Integer> points = **new** ArrayList<Integer>();

**try** {

//get player number

System.*out*.println("What is the number of the new Player?");

number=scan.nextInt();

//get player name

System.*out*.println("What is the Player's Name?\n(First Last)");

first = scan.next();

last = scan.next();

//get whether the player is injured

System.*out*.println("What is the players Status?\n(injured or uninjured)");

//decision on status

**if**(scan.next().equalsIgnoreCase("injured")) {

status=**true**;

} **else** {

status = **false**;

}

**int** z=1;

**boolean** end2 = **false**;

**while**(!end2) {

**try** {

System.*out*.println("How many points did the player score for game " + z + "?");

System.*out*.println("Press q to quit");

points.add(scan.nextInt());

z++;

}

**catch**(RuntimeException e) {

**if**(scan.next().equalsIgnoreCase("q")) {

end2=**true**;

}

}

}

//make and add to arraylist

RugbyPlayer play = **new** RugbyPlayer(number, first, last, status, points);

csTeam.add(play);

}

**catch**(RuntimeException e) {

String bug = scan.nextLine();

}

} **else** {

**if**(in.equalsIgnoreCase("n")) {

end=**true**;

}

}

}

end = **false**;

//check for greatest average points

**for**(**int** x = 0; x<csTeam.size(); x++) {

RugbyPlayer temp = csTeam.get(x);

**double** averagePoints = (**double**) temp.getTotalPoints()/ (**double**) temp.getGamesPlayed();

//if points is greater

**if**(averagePoints>topPoints) {

topPoints=averagePoints;

topNumber=temp.getNumber();

topFirst=temp.getFirstName();

topLast=temp.getLastName();

tie = **false**;

//if a tie exists

} **else** {

**if**(averagePoints==topPoints) {

tie = **true**;

}

}

}

//decide what to print

//If it's a tie

**if**(tie) {

System.*out*.println("It's a tie");

//if someone has the greatest points

} **else** {

System.*out*.println("Player " + topFirst + " " + topLast + " Number " + topNumber + " Scored the most Points!");

System.*out*.println("He scored " + (**int**) topPoints + " points!");

System.*out*.println();

}

//check for lowest average points

**for**(**int** x = 0; x<csTeam.size(); x++) {

RugbyPlayer temp = csTeam.get(x);

**double** averagePoints = (**double**) temp.getTotalPoints()/ (**double**) temp.getGamesPlayed();

//if points is greater

**if**(averagePoints<lowestPoints) {

lowestPoints=averagePoints;

lowNumber=temp.getNumber();

lowFirst=temp.getFirstName();

lowLast=temp.getLastName();

tie = **false**;

//if a tie exists

} **else** {

**if**(averagePoints==lowestPoints) {

tie = **true**;

}

}

}

//decide what to print

//If it's a tie

**if**(tie) {

System.*out*.println("It's a tie");

//if someone has the greatest points

} **else** {

System.*out*.println("Player " + lowFirst + " " + lowLast + " Number " + lowNumber + " Scored the least average Points!");

System.*out*.println("He scored " + (**int**) lowestPoints + " points!");

System.*out*.println();

}

System.*out*.println("\t\t\tScore per Game");

//loop to print out uninjured

**for**(**int** x = 0; x<csTeam.size(); x++) {

RugbyPlayer temp = csTeam.get(x);

System.*out*.print(temp.getFirstName() + " " + temp.getLastName() + "\t");

**for**(**int** s = 0; s<temp.getGamesPlayed(); s++) {

System.*out*.print(temp.getPointsGame(s) + " ");

}

System.*out*.println("");

}

scan.close();

scan1.close();

}

}

Output:

Create the team

What is the number of the new Player?

Press Q to quit

12

What is the Player's Name?

(First Last)

Press Q to quit

sad nasda

What is the players Status?

(injured or uninjured)

Press Q to quit

uninjured

How many points did the player score for game 1?

Press Q to stop adding scores

132

How many points did the player score for game 2?

Press Q to stop adding scores

4768

How many points did the player score for game 3?

Press Q to stop adding scores

q

What is the number of the new Player?

Press Q to quit

123

What is the Player's Name?

(First Last)

Press Q to quit

sajd asd

What is the players Status?

(injured or uninjured)

Press Q to quit

injured

How many points did the player score for game 1?

Press Q to stop adding scores

1454

How many points did the player score for game 2?

Press Q to stop adding scores

765

How many points did the player score for game 3?

Press Q to stop adding scores

q

What is the number of the new Player?

Press Q to quit

q

Updating player data for a game

How many points did player number 12 get?

347

Did the player get injured this game?

(y/n)

y

How many points did player number 123 get?

4556

Did the player get injured this game?

(y/n)

n

Would you like to remove a player?

(y/n)

n

Would you like to add a player?

(y/n)

y

What is the number of the new Player?

13

What is the Player's Name?

(First Last)

asdfjk jksa

What is the players Status?

(injured or uninjured)

uninjured

How many points did the player score for game 1?

Press q to quit

7

How many points did the player score for game 2?

Press q to quit

q

Would you like to add a player?

(y/n)

n

Player sad nasda Number 12 Scored the most Points!

He scored 1633.3333333333333 points!

Player asdfjk jksa Number 13 Scored the least average Points!

He scored 7.0 points!

Score per Game

sad nasda 132 4768 347

sajd asd 1454 765 4556

asdfjk jksa 7