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

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

\* Version:001Date:Mar 6, 2014

\*

\* A program which uses arraylists to store objects for future use, as well as test the methods

\*

\* 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

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

**Rugby Player Class**

**package** test\_7;

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

//constructor

**public** RugbyPlayer(**int** number, String first, String last, **boolean** injured, **int** points) {

studentNumber=number;

firstName=first;

lastName=last;

injury=injured;

pointsScored=points;

}

//get methods

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

**return** studentNumber;

}

**public** **int** getPoints() {

**return** pointsScored;

}

//add points

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

pointsScored+=points;

}

//set points

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

pointsScored=points;

}

//more get methods

**public** String getFirstName() {

**return** firstName;

}

**public** String getLastName() {

**return** lastName;

}

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

**return** injury;

}

//set injury state (true for injured)

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

injury=status;

}

**private** **int** studentNumber;

**private** **int** pointsScored;

**private** String firstName;

**private** String lastName;

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

}

**Tester Class:**

**package** test\_7;

**import** java.util.ArrayList;

**import** java.util.Scanner;

**public** **class** Tester {

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

**int** topNumber=0, topPoints=0;

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

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

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

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

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

//fill team loop

**do** {

**int** number, points;

String first, last;

**boolean** status;

**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**;

}

//get number of points player has

System.*out*.println("How many points does the Player have?\nPress Q to quit");

points=scan.nextInt();

//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 adding points in one game

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

**while**(!end) {

RugbyPlayer temp = csTeam.get(x);

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

//try to add points

**try**{

temp.addPoints(scan.nextInt());

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**;

}

//check for greatest points

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

RugbyPlayer temp = csTeam.get(x);

//if points is greater

**if**(temp.getPoints()>topPoints) {

topPoints=temp.getPoints();

topNumber=temp.getNumber();

topFirst=temp.getFirstName();

topLast=temp.getLastName();

tie = **false**;

//if a tie exists

} **else** {

**if**(temp.getPoints()==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 " + topPoints + " points!");

System.*out*.println();

}

//loop to print out uninjured

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

RugbyPlayer temp = csTeam.get(x);

//decide if injured or not

**if**(!temp.getInjured()) {

System.*out*.println("Player " + temp.getFirstName() + " " + temp.getLastName() + " Is uninjured");

}

}

scan.close();

scan1.close();

}

}

**Output:**

What is the number of the Player?

Press Q to quit

1

What is the Player's Name?

(First Last)

Press Q to quit

Bob Sandy

What is the players Status?

(injured or uninjured)

Press Q to quit

injured

How many points does the Player have?

Press Q to quit

10

What is the number of the Player?

Press Q to quit

3

What is the Player's Name?

(First Last)

Press Q to quit

Jake Blake

What is the players Status?

(injured or uninjured)

Press Q to quit

uninjured

How many points does the Player have?

Press Q to quit

15

What is the number of the Player?

Press Q to quit

q

How many points did number 1 get?

1222

How many points did number 3 get?

2

Would you like to remove a player?

(y/n)

n

Player Bob Sandy Number 1 Scored the most Points!

He scored 1232 points!

Player Jake Blake Is uninjured