

```
1 import java.util.Scanner;
2
3 public class SoccerGameScoreBoard
4 {
5
6     static int numTeams;
7
8     static {
9         numTeams = 0;
10    }
11
12    String sTeamName;
13
14    int goalsInMatch;
15
16    int goalsInExtraTime;
17
18    int goalsInPenaltyShootOut;
19
20    /*
21     * include constructor chaining so that you can create an abject with only the
22     * team name, later
23     *
24     * in chaining, you can use a read method to read the rest of the field value
25     *
26     * hint: while calling the next constructor, you can place a call to read
27     * function as
28     *
29     * argument in the constructor call
30     *
31     */
32
33    SoccerGameScoreBoard(String sTeamName, int goalsInMatch, int goalsInExtraTime,
34        int goalsInPenaltyShootOut)
35    {
36
37        this.sTeamName = sTeamName;
38
39        this.goalsInMatch = goalsInMatch;
40
41        this.goalsInExtraTime = goalsInExtraTime;
42
43        this.goalsInPenaltyShootOut = goalsInPenaltyShootOut;
44
45    }
46
47    SoccerGameScoreBoard(String teamName) {
48        this(teamName, read("goals in match"), read("goals in extra time"),
49            read("goals in penalty shoot out"));
50    }
51
52    public static int read(String name) {
```

```
53     Scanner input = new Scanner(System.in);
54     System.out.print("How many "+name + ": ");
55     return input.nextInt();
56 }
57
58 @Override
59
60 public String toString() {
61
62     return "SoccerGameScoreBoard [sTeamName=" + sTeamName + ", goalsInMatch=" +
goalsInMatch + ", goalsInExtraTime="
63
64         + goalsInExtraTime + ", goalsInPenaltyShootOut=" +
goalsInPenaltyShootOut + "];"
65 }
66 }
67
68 }
69
```