

Raffi Dilanchian - 951811193016
Mohammad Hossein Komeili - 951811193033

Software design test phase

addScore function

addScore function takes teamId as an argument (parameter) and add 5 to its score

```
function addScore( teamID ){  
  
    //check the input validation  
    if (teamID is not integer || teamID is not positive)  
        throw error; //throw error and exit function  
  
    // getTeams() returns array of team objects  
    teams = getTeams();  
  
    // storing score of team having teamID;  
    score = teams[teamID].score;  
  
    // adding 5 units to prevoius score  
    newScore = score + 5;  
  
    // applying changes by calling setTeamScore function by teamID and newScore  
    setTeamScore(teamID, newScore);  
}
```

Unit test for addScore()

```
function AddScoreTest() {  
  
    // a sample teamID in order to test addScore  
    teamID = 1;  
  
    // calling getTeams() to get array of team objects  
    teams = getTeams();  
  
    // create team with teamID=1 if does not exist  
    if(teams[teamID] does not exist)  
        createTeam(teamID);  
  
    //storing score of team having teamID  
    score = teams[teamID].score;  
  
    // call addScore function on teamID  
    addScore(teamID);  
  
    // calling getTeams() agian to get updated teams array  
    teams = getTeams();  
  
    // storing updated team score  
    updatedScore = teams[teamID].score;  
  
    // checking equality of updatedScore and score + 5 by using Assert.AreEqual() and The output will be  
    True if addScore function does its job correctly  
    Assert.AreEqual(score + 5, updatedScore);  
}
```

توضیحات فارسی

تابع اصلی
تابع نوشته شده با گرفتن شماره یا آی دی تیم, به امتیاز آن تیم 5 واحد اضافه میکند. برای این منظور ابتدا تابع getTeams را که اطلاعات تیم ها را بازمیگرداند صدا میزند، سپس با استفاده از آرگومان ورودی، امتیاز تیم داده شده را 5 واحد اضافه میکند، و در نهایت با استفاده از تابع setTeamScore که امتیاز تیم ها را تغییر میدهد، امتیاز جدید را برای تیم ثبت میکند.

تابع تست
در قسمت تست ابتدا یک شماره فرضی به عنوان شماره آی دی تیم در نظر میگیریم.
با استفاده از تابع getTeams آرایه ی تیم ها را میگیریم، در صورت عدم وجود تیمی با شماره فرضی آن را میسازیم. امتیاز تیم دلخواه را ذخیره میکنیم.
تابع نوشته شده را برای تست کردن کارایی آن روی تیم صدا میزنیم و دوباره آرایه تیم ها را از تابع getTeams میگیریم تا امتیاز جدید تیم را بدست آوریم. در نهایت با استفاده از تابع Assert.AreEqual چک میکنیم که امتیاز قبلی به علاوه 5 با امتیاز جدید تیم برابر باشد و در اینصورت خروجی صحیح میباشد.