

Forum:[Hack Forums](#)

Thread: [Need Help with Java and Junit](#)

legenda008 - January 29, 2016, 12:54 am

Ok guys I needed quick help with my assignment I think it is not something difficult
You see I have to write Bowling Code kata

and I have written the code for it

Here is my Bowling Class

```
Code public class BowlingGame {  
    private int roll = 0;  
    private int [] rolls = new int[21];
```

```
  
    public void roll(int...rolls){  
        for(int pinsDown:rolls){  
            if(pinsDown>10){  
                throw new IllegalArgumentException();  
            }else{  
                roll(pinsDown);  
            }  
        }  
    }  
}
```

```
  
    public void roll(int pinsDown){  
        rolls[roll++] =pinsDown;  
    }  
}
```

```
  
    public int score(){  
        int score = 0;  
        int cursor = 0;  
  
        for(int frame = 0;frame<10;frame++){  
            if(rolls[cursor]==10){ // check if it is strike  
                score+=10 + rolls[cursor+1] + rolls[cursor+2];  
                cursor++;  
            }else if(rolls[cursor] + rolls[cursor+1]==10){ // check if it is spare  
                score+=10 + rolls[cursor+2];  
                cursor+=2;  
            }  
        }  
    }  
}
```

```

        }else{
            score+=rolls[cursor] + rolls[cursor+1];
            cursor+=2;
        }
    }
    return score;
}
}

```

and here is my Test Class

```

Code public class TestBowling {
    private BowlingGame game;

```

```

    @Before
    public void setingUp(){
        game = new BowlingGame();
    }

```

```

    @Test
    public void canScore60(){
        game.roll(3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3);
        assertEquals(game.score(), 60);
    }

```

```

    @Test
    public void canScorePerfect(){
        game.roll(10,10,10,10,10,10,10,10,10,10,10,10,10,10,10,10);
        assertEquals(game.score(), 300);
    }

```

```

    @Test
    public void canScore67(){
        game.roll(3,3,3,3,3,3,3,4,6,3,3,3,3,3, 3,3, 3,3);
        assertEquals(game.score(), 67);
    }

```

```

    @Test
    public void canScore75(){
        game.roll(3,3,3,3,3,3,3,4,6,4,6,3,3,3,3, 3,3, 3,3);
        assertEquals(game.score(), 75);
    }

```

```
@Test
public void canScore70(){
    game.roll(3,3,3,3,3,3,3,10,3,3,3,3,3, 3,3, 3,3);
    assertThat(game.score(), is(70));
}
```

```
@Test
public void canScore87(){
    game.roll(3,3,3,3,3,3,3,10,10,3,3,3,3,3, 3,3, 3,3);
    assertThat(game.score(), is(87));
}
```

```
@Test
public void canScore70with10(){
    game.roll(3,3,3,3,3,3,3,3,3,3,3,3,3,3,10,3,3);
    assertThat(game.score(), is(70));
}
```

```
@Test
public void canScore84(){
    game.roll(3,3,3,3,3,3,3,3,3,3,3,3,3,3,10,10,10);
    assertThat(game.score(), is(84));
}
```

```
@Test
public void canScoreWith3and7(){
    game.roll(3,7,3,7,3,7,3,7,3,7,3,7,3,7,3,7,3,7,3);
    assertThat(game.score(), is(130));
}
```

//this tests should fail

```
@Test (expected = IllegalArgumentException.class)
public void testing15() {
```

```

int a = 15;
game.roll(a);

}

```

```

@Test (expected = IllegalArgumentException.class)
public void testingLetter() {
    char c = 'c';
    game.roll(c);
}
}

```

I need to do this or My Question is how can I test this

Test wrong input with 15 for one throw
 with two throws in the same frame bigger than 10
 with a letter input

csrk - January 29, 2016, 9:17 am

When using JUnit to test your code you want to create a new instance of BowlingGame in every test.

Code @Test

```

public void methodname() {
    BowlingGame bg = new BowlingGame(); // new one in every test, important
    bg.roll... // do your rolls here. How you do them are irrelevant, the point is to get your game to a
    desired state.
    assertTrue(bg.score() == 69); // finally, assert that the state of the game is what you expect it to
    be
}

```

Unit testing means that for your input you state explicitly what you expect the state to be like. I am not going to do your homework for you but there are a lot of other methods. I.e
 assertEquals(array 1, array 2); assertFalse(bool);.

Also, remember to write a lot of tests. You never do more than 1 assert in a single test. If the first assert fails and you have another one after that, the second assert will never be evaluated. You might think the 2nd one is the one that fails and spend time debugging a method that works. What you do is write two tests, use the exact same input and just assert for different things.

legenda008 - January 29, 2016, 4:19 pm

Ok i figured it out ,,but I have a new problem

how do i make that if two values in frame cannot be bigger than 10 .or

that the sum of throws in one frame cannot be bigger than 10

I have made this code

```

for(int frame = 0;frame<10;frame++){
    if(rolls[cursor] + rolls[cursor+1]>10){
        throw new Illegal Argument Exception
    }

    if(rolls[cursor]==10){ // check if it is strike
        score+=10 + rolls[cursor+1] + rolls[cursor+2];
        cursor++;
    }else if(rolls[cursor] + rolls[cursor+1]==10){ // check if it is spare
        score+=10 + rolls[cursor+2];
        cursor+=2;
    }
    else {
        score+=rolls[cursor] + rolls[cursor+1];
        cursor+=2;
    }
}
return score;

}

```

but when I test it with Junit it is not working

```

@Test (expected = IllegalArgumentException.class)
public void testFrames(){
    game.roll(8,8);
}

```

csrk - January 30, 2016, 3:12 am

What do you mean is the problem, is the test not succeeding?

From what i understand about your code, if any of the inputs in rolls is above 10 (11 or higher) it will throw an exception. game.roll(8,8) should not throw an exception? I've not gone through your code entirely.