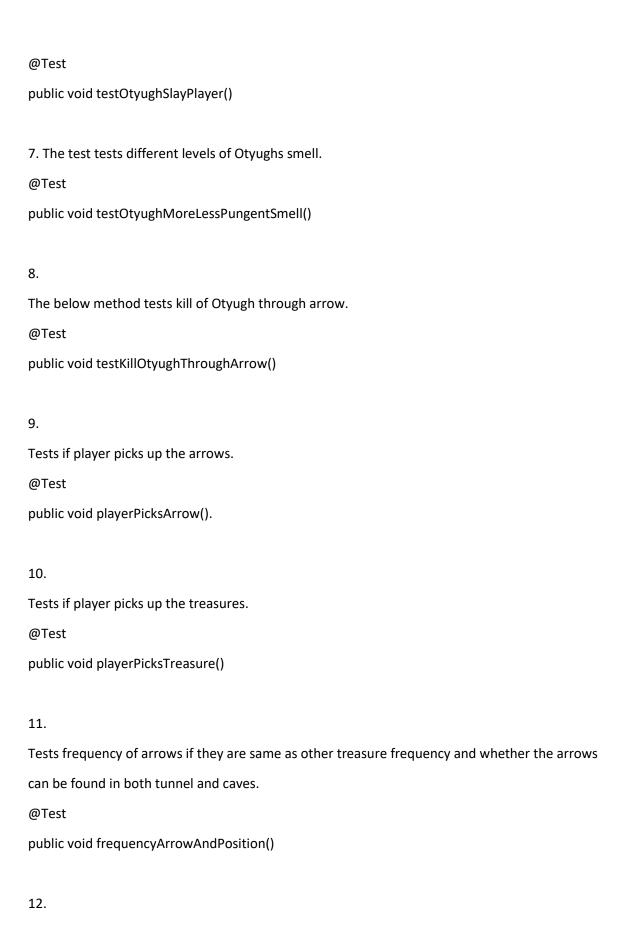
```
1.
The below tests if an otyugh gets set at the end point.
@Test
public void testOtyughEnd()
2.
The below tests if an otyugh never gets stored at the start point.
@Test
public void testOtyughStart()
3.
The below tests number of Otyughs.
@Test
public void testNumberOfOtyughs()
4.
The below tests that Otyughs stay only in cave.
@Test
public void testOtyughFoundInCaves()
5.
Test Otyughs with other treasures.
@Test
public void testOtyughsWithTreasures()
6.
The below test player getting killed by the otyugh.
```



The below test tests the presenec of arrow with

```
other artifacts and when they are found independently.
@Test
public void independentArrowWithTreasure()
13.
The below test tests killing of otyugh verifying players win.
@Test
public void testKillOtyughThroughArrowVerifyingWin()
14.
The below method tests the differences between the shot distances.
@Test
public void testExactDistanceDifferentDistanceShoot()
15.
Test Survival chance of player.
@Test
public void testPlayerSurvival()
16.
The below test tests if the player is not eaten by the injured Otyugh.
17.
The below test tests if the player is eaten by the injured Otyugh.
@Test
public void testEatenByInjuredOtyugh()
18.
Tests player can't move after getting eaten.
@Test(expected = IllegalArgumentException.class)
```

public void testIllgalmoveByPlayerAfterGettingEaten().
19.
The method checks if the arrow moves in the wrapping dungeon or not.
@Test
public void arrowMovingWrappingDungeon()
20.
The below method confirms that the arrow can only travel in a straight line in cave.
@Test
public void arrowNotTravellingInCave()
21.
The method tests if the arrow moves in the tunnel.
@Test
public void testArrowMovingInTunnel()
22.
The method simulates a snenario where the player wins the game using controller.
@Test
public void testPlayerWinNew()