



< MonsterTruck

Main Page → Problems → Solve a Problem

Butterfly >



Show Header

Language/Type:

Java <u>classes constructors Critters fields implementing inheritance</u>

instance methods

Related Links: <u>Critter.java</u>

Author: Marty Stepp (on 2010/05/30)

("Critter" classes come from the University of Washington's CSE 142 Critters homework assignment. See the <u>assignment spec</u> for more information.)

The following Critter class named Skunk is an attempt to make a critter that goes W, W, N and repeats, unless he eats food, in which case he will start going W, W, S and repeats. But the code contains several errors. Download the code and fix the errors so it compiles and behaves properly.

Revert

```
Type your solution here:
```

```
public class Skunk extends Critter {
    private int moves;
    private boolean hungry;

public void Skunk() { // constructor
        hungry = false;
        moves = 0;
    }

public boolean eat() {
        hungry = true;
        return true;
    }

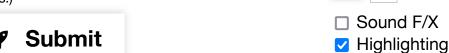
public Direction getMove() {

        // reset
        if (moves >= 3) {
```

1 of 4 2/21/24, 4:22 PM

```
19
                  moves = 0;
             }
20
21
             if(hungry == true) {
22
23
                  if (moves < 2){
24
                       moves++;
25
                       return Direction.WEST;
26
                  } else {
27
                       moves++;
28
                       return Direction.SOUTH;
29
                  }
             }
30
31
32
             if (moves < 2) {
33
                  moves++;
34
                  return Direction.WEST;
35
             } else {
36
                  moves++;
37
                  return Direction.NORTH;
38
39
40
        }
41 }
This is an inheritance problem. Write a Java class using inheritance. (You do not
                                                                         Indent
```

need to write any import statements.)



You passed 4 of 4 tests.

Go to the next problem: Butterfly

```
constructor
       test #1:
console output:
       result:
              pass
              getMove without eating
       test #2:
              Skunk getMove 1: "WWNWWNWWNWWNWWNWWN"
console output:
               Skunk getMove 2: "WWNWWNWWNWWNWWNWWN"
       result:
              pass
              getMove with eating
       test #3:
```

2 of 4 2/21/24, 4:22 PM

```
console output: Skunk getMove 1: "WWNWWNWWSWWSWWSWWS"
              Skunk getMove 2: "WWNWWNWWNWWNWWNWWN"
       result:
              pass
      test #4:
              eat
              Skunk 0 eat #1: true
console output:
              Skunk 1 eat #1: true
              Skunk 2 eat #1: true
              Skunk 0 eat #2: true
              Skunk 1 eat #2: true
              Skunk 2 eat #2: true
              Skunk 0 eat #3: true
              Skunk 1 eat #3: true
              Skunk 2 eat #3: true
              Skunk 0 eat #4: true
              Skunk 1 eat #4: true
              Skunk 2 eat #4: true
              Skunk 0 eat #5: true
              Skunk 1 eat #5: true
              Skunk 2 eat #5: true
              Skunk 0 eat #6: true
              Skunk 1 eat #6: true
              Skunk 2 eat #6: true
              Skunk 0 eat #7: true
              Skunk 1 eat #7: true
              Skunk 2 eat #7: true
              Skunk 0 eat #8: true
              Skunk 1 eat #8: true
              Skunk 2 eat #8: true
              Skunk 0 eat #9: true
              Skunk 1 eat #9: true
              Skunk 2 eat #9: true
              Skunk 0 eat #10: true
              Skunk 1 eat #10: true
              Skunk 2 eat #10: true
              Skunk 0 eat #11: true
              Skunk 1 eat #11: true
              Skunk 2 eat #11: true
              Skunk 0 eat #12: true
              Skunk 1 eat #12: true
              Skunk 2 eat #12: true
              Skunk 0 eat #13: true
              Skunk 1 eat #13: true
              Skunk 2 eat #13: true
              Skunk 0 eat #14: true
              Skunk 1 eat #14: true
Skunk 2 eat #14: true
              Skunk 0 eat #15: true
              Skunk 1 eat #15: true
              Skunk 2 eat #15: true
              Skunk 0 eat #16: true
              Skunk 1 eat #16: true
              Skunk 2 eat #16: true
              Skunk 0 eat #17: true
```

3 of 4 2/21/24, 4:22 PM

```
Skunk 1 eat #17: true
Skunk 2 eat #17: true
Skunk 0 eat #18: true
Skunk 1 eat #18: true
Skunk 2 eat #18: true
Skunk 0 eat #19: true
Skunk 1 eat #19: true
Skunk 2 eat #19: true
Skunk 0 eat #20: true
Skunk 1 eat #20: true
Skunk 2 eat #20: true
Skunk 2 eat #20: true
```

If you do not understand how to solve a problem or why your solution doesn't work, please contact your TA or instructor.

If something seems wrong with the site (errors, slow performance, incorrect problems/tests, etc.), please contact us.

Is there a problem? Contact a site administrator.

© University of Washington 2019

4 of 4 2/21/24, 4:22 PM