

WES-CS GROUP MEETING #7

Exercise 1: Where's Waldo? (Name Declarations and Uses)

A Java program can include many declarations and many uses of the same name. To understand what happens during execution, you need to know how to match declarations and uses. You should have learned some Java rules to help with this problem in class, so let's try using those rules on code that is intentionally confusing.

The classes `Waldo` and `TestWaldo` are defined below (with line numbers included for reference). Both classes declare and use the name `waldo`, sometimes incorrectly.

```
1. public class Waldo {
2.     private int waldo;
3.
4.     public Waldo(int init) {
5.         waldo = init;
6.     }
7.
8.     public void setWaldo(int val) {
9.         int waldo;
10.        waldo = val;
11.        this.waldo = waldo;
12.    }
13.
14.    public void printWaldo(int waldo) {
15.        for (int k=0; k<waldo; k++) {
16.            System.out.println(this.waldo);
17.        }
18.    }
19.
20.    public void waldo() {
21.        printWaldo(waldo);
22.    }
23. }

24. public class TestWaldo {
25.     public static void main(String[] args) {
26.         Waldo waldo = new Waldo(11);
27.         waldo.waldo = 10;
28.         helper(20);
29.         waldo.waldo();
30.     }
31.
32.     public static void helper(int k) {
33.         waldo.setWaldo(k);
34.     }
35. }
```

Part (a).

Start by finding every declaration of a `waldo` in the code. Draw a square box around the declaration, including the name (`waldo`) and the type it's declared to have.

Part (b).

Now annotate each declaration that you found with one of the following numbers:

1. A declaration of a *field* of a class.
2. A declaration of a *method* of a class.
3. A declaration of a *formal parameter* of a method.
4. A declaration of a *local variable* of a method.

Part (c).

Now find and *circle* each use of the name `waldo` in the code (that is, each occurrence of the name `waldo` that is not a declaration of `waldo`).

Match each use of `waldo` with the corresponding declaration of `waldo` by drawing an arrow from the use to the declaration. If there is no correct matching declaration (in other words, if that use of `waldo` would cause a compile-time error), cross out the line of code that includes that use of `waldo`.

Exercise 2: Jeopardy (Exam Review)

Now you're going to play a game like Jeopardy to help review for the Midterm Exam.

Divide into teams and decide on a team name. Your group leader will draw a Jeopardy grid on the board, with 100, 200, and 400-point questions in each of four categories:

1. If, Switches, and Booleans (or *Conds* for short)
2. Loops
3. Expressions
4. Program Design and Testing (or *Design* for short)

Pick a team to go first. That team will select an unused category and point value. Then your group leader will reveal the question. For each question, the team that grabs the flashing light thingy first gets to try to answer the question. If the answer is right, the team gets the points and gets to choose the next question. But beware: wrong answers will make you lose those points! If a group answers incorrectly, another team can grab the flashing light thingy and give an answer.

The game continues until all questions are answered, followed by a *Final Jeopardy* round, where all teams first decide how many points they'd like to wager on their answer. Then everyone sees the final question, and has a minute to write down the answer.

Good luck!