

Overview

This week's assignment will be a review exercise to get you comfortable with OOP and variables. In the first exercise, you will fix some common mistakes people tend to make. The second part gets you comfortable with declaring objects, working with object arrays, using constructors, and calling (overloaded) member methods.

Answer the assessment questions as you encounter them in the next section. This is especially true for this assignment, as each question will help you finish and understand the point of that particular exercise. The prompts for answering assessment questions are placed immediately following the material to which the listed questions relate.

Getting Started

After following the import instructions in the assignment page, you should have a Java project in Eclipse titled **Lab21_9**. This PDF document is included in the project in the **doc** directory. The Java files you will use in this lab are in the **src** directory, as usual.

Part 1: Fix Lab21_Vars.java

You will implement 7 fixes that are indicated in the code. The following 7 issues (and 10 assessment questions) correspond to the 7 fixes you need to implement. Your changes **must** follow the convention below:

```
System.out.println("This is corrected statement"); // inserted double quotes at
                                                    // the end of output
                                                    // string
// System.out.println("This is old erroneous statement);
```

You'll turn erroneous lines of code inside the files into comments (**required**) and then have the **corrected line precede the commented lines**, as shown above. You're free to add more comments as notes to yourself to explain the nature of the error(s) you fixed. This way it will be easy to recall the changes you made when the TA asks you to demonstrate, or when you subsequently review this lab.

Issue 1: 5.0 is a double but we want to assign it to an integer variable instead. We could of course change it to 5 but what if we couldn't? Remember, we can always type cast any type to any other type.

[Answer assessment question 1]

Issue 2: We want to declare an array of integers, but instead have the following code:

```
int arri0 = {10, 9, 8, 7, 6, 5, 4, 3, 2, 1};
```

[Answer assessment question 2]

Issue 3: We want to share a variable in the if-clause and else-clause as attempted by the following code:

```
if (i < j) {
    int temp = 0;
    System.out.println("Temp is " + temp);
} else {
    temp = 1;
```

```

        System.out.println("Temp is " + temp);
    }

```

[Answer assessment question 3]

Issue 4: Another common construct we use is a variable to store the total of some calculation. Suppose we want to do square of sums and want it in **total** which we later print, as attempted by the following code:

```

    for (int i = 0; i < 10; i++) {
        int total = 0;
        total += i*i;
    }
    System.out.println("i value is " + i);
    System.out.println("Total is " + total);

```

[Answer assessment questions 4 and 5]

Issue 5: We can create objects using **new**, and create pointers that references these objects. Multiple pointers can point to the same object or different objects. Analyze the following code to figure out the logical error.

```

Cheese jack;
Cheese monterey = new Cheese("Monterey");
jack = monterey;
System.out.println("Monterey name is " + monterey.getName());
jack.setName("Jack");
System.out.println("Jack name is " + jack.getName());
System.out.println("Monterey name is still " +
monterey.getName());

```

[Answer assessment questions 6 and 7]

Issue 6: Sometimes we write code that duplicates or is redundant in different parts of our conditional statements. The following code illustrates this:

```

Scanner input = new Scanner(System.in);
System.out.print("Enter first number: ");

if (input.nextInt() > var3) {
    System.out.print("Enter second number: ");
    int num2 = input.nextInt();
    System.out.println("First is greater");
    if (num2 < var3)
        System.out.println("Second is Less than");
    else
        System.out.println("Second is Greater or equal");
} else {
    System.out.print("Enter second number: ");
    int num2 = input.nextInt();
    System.out.println("First is Less than or equal");
    if (num2 < var3)
        System.out.println("Second is Less than");
    else
        System.out.println("Second is Greater or equal");
}

```

}

[Answer assessment questions 8 and 9]

Issue 7: Suppose we want to print out the first number the user entered after all the lines given above.

[Answer assessment question 10]

Part 2: Fill-in Lab21_Objects.java

Have a look at **Dummy.java**. It has 11 constructors and 11 overloaded **display()** method. Your job is to call every one of them from the **main** method of **Lab21_Objects**, and fill in each of the **dlist** (Dummy list) entries with the appropriate objects. We have included the first two constructor calls and first two **display()** calls in the file. Fill in 9 more constructor calls and 9 more **display()** calls.

DO NOT DECLARE NEW VARIABLES. YOU MUST USE ONLY THE GIVEN VARIABLES AS ARGUMENTS TO THE METHOD CALLS. DO NOT USE ANY CONSTANTS OR NUMBERS AS ARGUMENTS (e.g., 5, 5.0, etc.)

You can use individual indices of the arrays for the arguments. For example, using **dlist[1].display(iarr[0]);** is okay, but **dlist[1].display(0);** is not (as that is using the number 0 directly as an argument).

[Answer assessment questions 11, 12 and 13]

Part 3: (Assessment) Logic Check and Level of Understanding

Create a Word document or text file named **Part3** that contains answers to the following:

- 1) How do you type cast a **double** into an **int**?
- 2) How do you declare an array of **int** that goes from **10** to **1**?
- 3) What is the scope of the variable **temp** declared at line 20 of **Lab21_Vars.java**? Where does it need to be declared if it is to be used for the if-clause and else-clause?
- 4) What is the scope of the variable **total** declared in line 29 of **Lab21_Vars.java**?
- 5) What is **X** in the output “**i value is X**” produced by line 32 of **Lab21_Vars.java**? And why is it that value?
- 6) What is the logical error in the code in lines 36-42 of **Lab21_Vars.java**? (How do you fix it?)
- 7) How many pointers and objects are created in your fixed version of code at lines 36-42 of **Lab21_Vars.java**?
- 8) What parts are redundant in the code at lines 46-66 of **Lab21_Vars.java**?
- 9) How do reduce or combine the redundant code at lines 46-66 of **Lab21_Vars.java** so we have no redundancy?
- 10) How can we figure out what was the first number for code at lines 46-65 of **Lab21_Vars.java**? What is the **println** statement to print the first number?
- 11) Give two distinct characteristics of a constructor.
- 12) What is the purpose of ‘.’ in **System.out.println();** or **dlist[1].display();**?
- 13) What happens if you swap the order of the two lines in **Lab21_Objects.java**? (and why?)

```
dlist[0].display();    // Goes first
dlist[0] = new Dummy(); // Goes after
```

What to hand in

When you are done with this lab assignment, submit all your work through CatCourses.

Before you submit, make sure you have done the following:

- Attached the file named **Part3** containing answers to Assessment questions (1 – 13).

- Attached the `Lab21_Vars.java` and `Lab21_Objects.java` files
- Filled in your collaborator's name (if any) in the "Comments..." text-box at the submission page.

Also, remember to demonstrate your code to the TA or instructor before the end of the grace period.