Name: *(as it would appear on official course roster)*		
Umail address:	**@umail.ucsb.edu**	**section** **5pm, 6pm, 7pm**
Optional: name you wish to be called if different from name above.		
Optional: name of "homework buddy" (leaving this blank signifies "I worked alone"		

1

h02: HFJ 3,4; JN7 Ch2 pp 19-32; Primitives, References, Instance Variables, Methods

h02

CS56 F19

ready?	assigned	due	points
true	Mon 09/30 05:00PM	Thu 10/03 05:00PM	100

Printable PDF *You may collaborate on this homework with AT MOST one person, an optional "homework buddy".*

MAY ONLY BE TURNED IN IN THE LECTURE/LAB LISTED ABOVE AS THE DUE DATE, OR IF APPLICABLE, SUBMITTED ON GRADESCOPE. There is NO MAKEUP for missed assignments; in place of that, we drop the three lowest scores (if you have zeros, those are the three lowest scores.)

Reading Assignment:

- Read HFJ Chapter 3 (especially pages 59-62) and the online reading notes.
- Read HFJ Chapter 4 and the online reading notes.
- As you read, also consult the online reading notes linked above. To do so, go to the online version of this homework, at https://ucsb-cs56.github.io/f19/hwk/h02/ and click the links.
- Then, do the problems below.

Please:

- **No Staples**.
- **No Paperclips**.
- **No folded down corners**.

1. (10 pts) Please fill in the information at the top of this homework sheet, including your name and umail address. Put the time your discussion section starts (5pm, 6pm, 7pm) in the space indicated (the one you are registered for—even if you usually attend a different one.) If the other two items apply, please fill them in as well. Please do this every single time you submit homework for this class.

2. Based on your reading in HFJ Chapter 3:

   a. (5 pts) In Java, a variable can store a *primitive* or a *reference*. Briefly: What's the difference?

   b. (5 pts) If I write `3.4`, is that of type `double`, or `float`?

   c. (5 pts) Declare `a` as a `double` and assign it the value 5.6 (as a `double`)

   d. (5 pts) Declare `b` as a `float` and assign it the value 7.8 (as a `float`)

3. (5 pts) In C++, the name of a plain old array of `Student` objects is not an object, but is rather a pointer to a `Student` (i.e. it is of type `Student*`. What about in Java—is an array an object, yes or no?

2

h02

CS56 F19

4. Variables that represent a primitive type (e.g. `boolean x;` or `int y;`) and variables containing object references (`String w;` or `Student z;`) have this in common—they are both composed of bits in memory.

   But, as explained in HFJ Chapter 3, they differ in what the bits "actually" represent. You won't get this one by just guessing—you really have to read the book.

   a. (5 pts) What do the bits that represent `int y;` represent?

      Assume that `y` is assigned the value 13

   b. (5 pts) What do the bits that represent `String w;` represent?

      Assume that `w` is assigned the value `"foo"`.

5. Based on your reading in HFJ Chapter 3, p. 59-62 and HFJ Chapter 4 p. 84:

   a. (10 pts) Suppose I have a class called `Course`.
   How do I declare and allocate space for a plain old Java array called `courses` that can hold 5 references to `Course` objects?

   b. (10 pts)

      Java `for` loops look pretty much just like C++ `for` loops (see HFJ page 10 if you really need to check.) Given that, assuming there is a default constructor `Course()` that you can call to create a new `Course` object, write a `for` loop that initializes all of the elements of the array `courses` (from the previous problem) to be instances of the `Course`> class.

6. Consider these questions about memory—answers are in HFJ Chapter 3

   a. (5 pts) Assuming the same JVM, can the amount of memory taken up by an object reference differ for different kinds of objects (say `String` vs. `ArrayList<String>`?)

   b. (5 pts) Assuming the same JVM, can the amount of memory taken up by the object itself differ for different kinds of objects

   c. (5 pts) Can the amount of memory taking up for an object reference for a object particular type (say `String`) differ from one JVM to another?

7. HFJ Chapter 4 discusses the difference between the == operator and the `.equals` method.

   a. (10 pts) Under what circumstances should you use the == operator to compare two variables?

   b. (10 pts) Under what circumstances should you use the `.equals` method to compare two variables?