

First name (color-in initial)	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	section (4, 5 or 6)	first name initial	last name initial
Last name (color-in initial)	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z			

# H05: Due Wednesday, 01.14 in Lecture

## Inheritance, Interfaces, Abstract Classes, Polymorphism (HFJ Ch 7,8)

Assigned: Wed 01.07      Total Points: 50

MAY ONLY BE TURNED IN IN THE LECTURE/LAB LISTED ABOVE AS THE DUE DATE, or offered in person, for in person grading, during instructor or TAs office hours. See the course syllabus at <https://foo.cs.ucsb.edu/56wiki/index.php/W15:Syllabus> for more details.

(1) (10 pts) Fill in the information below. Also, fill in the A-Z header by

- **coloring in** the first letter of your first and last name (as it appears in Gauchospace),
- writing **either 4, 5, or 6** to indicate your **discussion section (lab)** meeting time
- writing your **first and last initial** in large capital letters.

All of this helps us to manage the avalanche of paper that results from the daily homework.

name:	
umail address:	@umail.ucsb.edu

If you collaborated with AT MOST one other person on this homework, write his/her name below. She/he should also have your name on his/her paper.

### Reading Assignment:

We are moving right along, reading two more chapters. These two chapters are short, and a good bit of this is basic review of OOP concepts you may have already seen in CS24 (and possibly in CS32 if you took that, which is recommended, though not required, as a pre-req to CS56). The coverage of these chapters will be limited on the first midterm exam, since the programming assignments that reinforce this material aren't due until after the exam. But you need to read this material now and get comfortable with the ideas before you tackle the first lab that uses inheritance. We also need at least a basic understanding of inheritance and polymorphism before we can do assignments involving GUIs and graphics, or tackle the open source projects.

- HFJ, Chapter 7, **165** through 196 and reading notes HFJ:Chapter\_7
- HFJ, Chapter 8, **197** through 235. HFJ:Chapter\_8
- Also: Review Chapter 5 in HFJ, p. **95** -124.Extra Strength Methods (and reading notes: HFJ:Chapter\_5
- Also: Review Chapter 6 in HFJ, p. **125** -164. Using the Java Library and reading notes: HFJ:Chapter\_6
- If there are reading notes on the wiki, consult those too—sometimes they contain helpful hints.

(2) (12 pts) Based on your reading in HFJ Chapter 7:

Complete the following exercise from p. 179, putting a check next to the relationships that make sense.

(3) (7 pts) Based on your reading in HFJ Chapter 7:

What does it mean to have a "polymorphic argument" or a "polymorphic return type" for a method? Explain with an example—but NOT using the example of Vets and Animals used in the book. Substitute your own example. Give a detailed enough description of the class hierarchy you have in mind to make it clear that you get the concept.

	Oven extends Kitchen
	Guitar extends Instrument
	Person extends Employee
	Ferrari extends Engine
	FriedEgg extends Food

	Beagle extends Pet
	Container extends Jar
	Metal extends Titanium
	GratefulDead extends Band
	Blonde extends Smart
	Beverage extends Martini

(4) (7 pts) Based on your reading in HFJ Chapter 8:

Briefly describe the difference between an abstract class and an interface.

(5) (7 pts) What is one advantage of using an ArrayList over a plain old Java array?

(6) (7 pts) Why do some classes in the Java API have package names that start with java.blah while others have package names that start with javax.blah? What does the x mean?