Books, other learning resources, strategies

Other books you would recommend that would cover material you go over in class, but also that is a reasonable amount of reading. The Nino and H book is awful!

There are lots of Java textbooks, and yet there doesn't seem to be any real winner. Part of the problem is that there is an enormous amount of information that comes under the heading of "Java". Try to focus on finding books that cover the material you are specifically interested in.   The Core Java books by Horstmann and Cornell are the best books I know. Other books I've used include Thinking in Java by Bruce Eckel and Beginning Java by Ivor Horton.

If I get another book, like from the ePost that you put up, how should I read through the various sections? Should I just hunt and skim through the chapters that contain the section that we talked about?

The books I like generally are oriented towards the working programmer. Usually these books start out with a few chapters that everyone interested in the application area should know, and then proceed to give information about every last detail. I read the first couple of chapters closely, skim the rest to get an overall picture, and then read those parts in detail when I need them for specific issues.

How to find definitions of words I don't understand quickly and easily

Words that are Java classes can be looked up in the JavaAPI documentation with a button click
A good programmer's reference to Java can be a big help. The Core Java books by Horstman and Cornell are an example. They are written for programmers so they are concise, but they have a lot of information about the practical uses of Java statements and classes.
Surprisingly, sometimes doing a Google search is a good way to find something like this quickly. If you are looking for information, probably thousands of other people have had the same question, and so there may well be a relevant answer available from the first page of search results.
Words that are Java keywords and statements can be looked up in the Java Language Reference, although that tends to be more technical than you usually want

I have been completing the hw and assigned reading but I still feel unclear on what I need to be getting out of each assignment. The more I learn about what's going on, the more expansive it seems, and therefore I feel kind of lost.

Believe me, I know the feeling! An important skill to learn in order to be an effective programmer is to be able to recognize what you need to know, and what you don't need to know. There is too much information out there to be able to learn it all. I try to get a complete understanding of the basic material in an area

and then when I decide that I need to go deeper in a particular area for a particular application I go back and take another pass at it.

I can read the lines of coding, but unable to write them without having references and I forget or don't know where to put them.

None of us remember every detail all the time and it's okay to make mistakes as you go along. I think the key thing is to just take your time and give yourself a chance to use the knowledge that you do have. It's surprising how much you actually do know if you don't panic about every possible error.

When given a problem in which you need to write the code for it, what steps / guidelines should you take? Where do you start? What questions (general) should you ask yourself? <this is starting from a blank screen> Whenever I'm given this problem, I'm not organized in my thinking. I need a general guideline to follow so I can be organized.

I usually start with "where is the data coming from, what am I supposed to do to it, and where are the results going?" The source and destination of the data tells you a lot about the environment that your program has to fit into, and the processing requirements tell you the kinds of things that you need to actually be able to do. Once you have a context and a reason for being, then you can think about the general structure of the code and the major blocks that you have to do to get the data, manipulate it, and produce results. You can continue to refine the design until you have something you can implement.

Overall format of the source code

The structure of format of writing codes.

The syntax flow. Can you clearly outline the order of declarations from class definition to method to return. There are a lot of variables associated with methods ... I'm pretty confused.

the way to construct the readable program including syntax and format

How to write the code (syntax), the order of the code.

More examples of writing code.

Is there any format to write the program? For example: [constructor]…[method]…

The main thing is to get the idea that the curly braces define blocks of code and that blocks can be nested. Once you start to see the structure in general terms, you can see it as a set of a few areas instead of a mass of obscure detail. The order of some of the elements below can be changed, but this is a general guideline.

In general, the outermost block is the class block. This is something like "public class MyClass { .. body of the class …}. Everything else that follows is part of the "body of the class" and is located between the outermost curly braces.

The first inside block is usually a constructor.   It might be something like "public MyClass() { … body of the constructor …}. A constructor is just like a method, except that it is called only when a new object is being created, and it doesn't explicitly return a value. It is the interior decorator that fills in all the details before handing over the newly created object to whatever piece of code used the "new" operator to request the new object.

There may be zero, one, or more constructors. Each one is a block. The order of the constructors doesn't matter.

After the constructors there are usually some methods. They will be something like "public int getMealsize() { … body of method …}.

I usually put the public methods (which other classes might want to call) ahead of the private methods (which are just for use in this class). It doesn't matter to the compiler, but another programmer is likely to be more interested in the public methods than the private methods.

After all the methods and just before the end of the class, I put the instance variables. Some people put them up at the front of the class, before the constructors. Either way works. I put them at the end because I think they are details that the casual reader does not need to see right away. The instance variable declarations are not inside any constructor or method.

Constructors and methods

The role of the constructor and where it begins and ends in the code.

The constructor is the first chance your code gets to do anything when a new object of your class is created. When somebody uses "new MyClass()", the Java system allocates a little memory for your new object, and then transfers control to your constructor. You can do whatever you want at the point, but when you are done, the object should be ready for use. All the instance variables should be in a consistent state.

I know constructor is special type of method. From what I've seen, it initiates instance variables. Is it true? How come it is special?

It's special because Java calls it when a new object of your class is created. When your other methods are called, they usually expect that the constructor has done all required initialization.

Do you need the constructor only for local variables and setting default values for instance variables? Is it possible to use instance variables without using the constructor?

"local variables" are variables that the constructor or method uses and that disappear at the end of that little block of code. Yes, it is possible to use instance variables without using a constructor. You can initialize them right at the point when you declare them: "double mealSize = 0.5;" as long as the initialization is

fairly simple (one statement). Also, you can write your methods in such a way that they recognize that some particular variable has not been initialized, and do it themselves. Both of these ideas are okay for some circumstances, but I tend to put all the initialization in the constructor so that it's easier to find.

Variables and getting values – instance, local, parameter, etc

distinguishing local and instant variables

The local variables are ones that are declared in the body of a constructor or method and are intended for use right there. They disappear after the constructor or method returns to the caller. The instance variables (not instant) are the variables that live as long as the object lives and are declared outside of any of the methods or constructors. They are called "instance variables" because an object is an "instance" of a class.

parameters

Parameters are the way you pass information to your constructors and methods. You declare the parameters you must have by listing them in the parameter list for the method, which is the part in parentheses right after the name of the method. For example, in "public Dog(String theName)" we are telling the compiler that anybody that wants to use this constructor must supply a String value to us, and that we will refer to that String using the variable name "theName".

When we need a value under what conditions can we refer to it by just its name, and when do we have to refer to it by "getDeltaX". I have been figuring out the code by experimenting with what works, but I don't know the logic behind it.

Generally, you use the name of a variable to refer to variables that are part of the class that you are writing, and you use method calls like "myShape.getX()" when you are getting information from objects of other classes. An anology: I can look in my own backpack to see if I have my notebook. (that's like referring to a variable name directly.) If I want to know if you have the notes from last week, I should ask you, and not assume that your pack is organized like mine or that I have permission to go looking in it anyway. (Asking you is like calling a method of yours "isNotebookAvailable()).

Referencing other classes / objects for information or methods.

It is possible to make instance variables visible to other classes, but generally speaking it is best to provide methods for other classes to use when they want to know something from you. This lets you change the implementation of your class without affecting other classes.

General terminology usage

What's the actual definition of object? You've described it as part of classes and other things.

Part of the problem is that there are two ways that "object" is used in Java. The general term "object" (with a lower case o) refers to the thing that is created with the "new" keyword. This is the object that is built from the blueprint (the class file). You store a reference to an object, and then you can ask that object to do things. "Dog rover = new Dog();" creates a new object from the Dog class definition, and then you can say things like rover.sleep().
The specific term "Object" (with an upper case O) refers to a class that is defined in the Java libraries. All of the classes that we define are descendants of the Object class, and inherit some characteristics from it. I haven't talked about inheritance yet, but I will in the next few weeks. It is very similar to the idea of extending an interface in that the child has all the behavior of the parent, plus whatever special things it can do.

Specific Java keywords and access control

Keywords like public/private/static/void/main/etc. What they mean - when you use them?

Clarify when to use public and private and static

public vs private. What's the difference / effect on how something works given these names.

public vs private. What is the difference, why are they there.

"public" and "private" are called access modifiers. They control how visible to other classes your methods and variables are. The methods that you write generally fall into two groups: those that you want other classes to be able to see (public) and those that are just for use in your class internally (private). The instance variables should almost always be private, since they are internal information and you might want to change the design at some future time.

null vs void vs static

null? static?

Keyword "null" is used to indicate "no reference is available". For example, a variable that has been declared to point to a Dog object needs a way to say "I'm not pointing to anything at this time." Ordinarily, you say something like Dog rover = new Dog("Happy"); However, if you later want to say that rover is not pointing any Dog in particular you could say "rover = null;". Similarly, the get() method that gets a String from the pMap in OvalSlider returns "null" if it can't find the keyword that you specified. That is why the code in the constructor is :

```
arg = (String)pMap.get("height");
 if (arg != null) {
     height = Integer.parseInt(arg);
 }
```

If there wasn't an entry for "height" then "get" returns null and we don't try to do anything.

Keyword "static" is used to indicate that the item it applies to is available even before the class is used to create an object, and that there is only one of them per class. So if you apply static to a method (like main) that means that it is available to Java before any objects are created. This is a way to get started – you have to have at least one thing that exists so that the code can start executing somewhere. There are other uses of static as well. One of them is to provide constant values like Math.PI. If you go back and look at the definition of Math.PI you'll see that it is a static field. You don't need to create an object of the Math class to use this field since it already exists. I will talk about "static" more in a later lecture.

Keyword "void" is used to indicate that a method does not return a value. Every method has to say what it returns, so "void" is a way to say "Here is what I return: Nothing".

please explain "this". For example, public boolean equals(Object obj) { return this == obj;}

Keyword "this" is a reference to the object for which the instance method was invoked or to the object being constructed. In other words "this" is another reference variable like "Dog rover" except that it always points to the object the code is executing for, rather than some other object. It is useful for being clear about which object you are talking about, and it is the only way you can say "me" in a method that needs a reference to itself. In the equals method quoted in the question, we are checking "am I the object that the caller is looking for? Is the reference to me (this) equal to the reference to the other object (obj)?"

What exactly does "this.something" do?

"this.getMealSize()" calls the getMealSize() method of the current object. It is usually not required, since Java looks at the current object for method and fields if you don't specify anything. However it is sometimes useful to be explicit about it. So you could say, for example, this.mealSize = 0.5; in the constructor for Dog and you would be referring to the instance variable for the current object. The effect is the same as if you said "mealSize = 0.5".

Specific Java statements

Basic java syntax

Blocks are defined by curly braces {…} and statements within blocks are terminated with semi-colons ";". A class starts with "public class MyName" and is followed with a body defined inside {…}. The body consists of zero or more constructors and methods, and possibly instance variables.

nesting if statements if (a) {if (b) { c;} else {d;}}

The basic format of an "if" statement is "if (condition) first_statement else other_statement;" Either or both statements can be replaced by a block of statements in curly braces {…}.   The way the question is written is a valid format for a nested if. If the first condition (a) is true, then the nested if statement is executed and it evaluates its conditional expression (b). If this second expression is true, then the first nested block of statements (c) is executed, if it is false then the second block (d) is executed.

return statements

How return statement works.

The return statement causes your method to return immediately to the caller. If your method is supposed to return a value, then the value should be specified in the return statement. If you do not need to return a value, then you can use the return statement by itself, or you can just get to the end of the method (the closing curly brace "}") and Java will automatically return to the caller.

How to make a loop actually loop rather than go through once and stop

This is controlled by the "check condition" part of the "for" or "while" loop.
for (int i=0; i<10; i++) {System.out.println(i);}
This will loop ten times and print out 0 through 9. It sets "i" to 0 initially, then checks that i is less than 10. If it is, it does the loop once, increments "i" and checks again. This continues until the condition check is false.

When and why do we need "interface"

Keywork "interface" is used when you are specifying the definition of an interface. It is like the keyword "class" except that you are not defining an implementation of anything, just what the methods should be if a class wants to implement this interface.

Is something like Car myCar = new Car(); only used in a method like static void main() { ..

No. The "new" operator is used any time you want a new object to be created from some class definition. Your code in PetSet for example used the new operator to create a new instance of each of the animals it was responsible for.

When do we need "package"?

Keyword "package" indicates that the class is part of a group of classes that are related in some way. All classes in a package are stored in the same directory. Classes in the same package sometimes have permission to access methods and fields of other classes in the package.

arg = (String)pMap.get("height"); Please explain "arg"

"arg" is a local String variable that receives a reference to a String returned by the get() method. It might be "null" if the keyword "height" was not in the Map

pMap, or it will point to a String containing whatever was supplied in the original property file that was used to create this Map.

When an object implements an extended interface, does it also have to have the older interface methods?

Yes, all the methods must be implemented one way or another. There are techniques that allow us to inherit implementations from a parent class, so you don't always have to implement the method yourself in your class, but they must be implemented somewhere. I will talk about inheritance in a later lecture.

Don't entirely understand loops. When specifically do you use the "for" loop and when do you use the "while" loop?

Generally the distinction is that you use a "for" loop when you know before you start how many times you want to do it, and you use a "while" loop when you know what the stopping condition should be, but you don't know how many times through the loop it is going to require to achieve that condition.

Can you talk about throwing exceptions?

Throwing an exception is a way to report a problem that the current method cannot deal with. It is basically a way to bail out and let somebody else take care of the problem. For example, the code that opens files on disk uses an exception to report the fact that a requested file does not exist. There is nothing the method can do about the problem, so it throws and exception and the caller gets to deal with the problem. I'll talk about exceptions in a later lecture.

How to apply learned material to problems. Like when to use if, or when to use else if. Also for loops, also the interface and extend are confusing.

Go back over the notes. Read the example pieces of code and trace through what is happening and try to explain to yourself why it is happening. Ask me or the TAs or the consultants for help. Practice writing code.

Specific Java library classes

Strings and toString methods

The String class holds a set of characters and lets you do things with them. The toString method is defined for every object, and is called when you do an object to String conversion. This happens for example, when you say "System.out.println(rover)." Java calls the method rover.toString() and prints the result. If you have defined a method called "toString" it uses that, if you have not defined it, then Java uses the simple one that is defined in Object.

public void method(...). void? when to put items in () and when not to.

Keyword "void" is discussed above. Items are put in () depending on the statement you are writing. The control statements of a "for" loop go in (). The control condition for a "while" loop goes in (). The parameter list for a method goes in ().

ArrayList thing

ArrayList

Reread the lecture notes and ask me/TAs/consultants specific questions about what you don't understand.

How would one create an ArrayList containing ints? The ArrayList seems to require objects.

Yes, it does require objects. There are two things you can do. There is a class called Integer that does nothing but hold "int" values. You can do something like "myList.add(new Integer(myValue));" This will create a new object containing the int which is then added to the list. It is also possible to define "int" arrays which contain integer values directly. There are advantages to this (simplicity) and disadvantages (no automatic expansion, no methods to access and manipulate values). I'll talk about arrays in a later lecture.

implementation.

The word implementation just means the code that you write that actually does whatever the method is supposed to do. I use the word to help distinguish the actual code from the interface it is implementing.

Main method and running programs outside of BlueJ

Can a main method be called like an interface? It is a set of methods ... pros and cons? or am I totally off base?

The main method is not usually called by anything except the Java system. It is intended to be the place where all the action begins, and since the action has already started by the time we get to one of your methods, there's no point in calling main at that time. However, you are correct that it is a method just like all other methods. It is possible to define other static methods and use them in your code. Also the Java runtime libraries include numerous static methods. For example, see "static int parseInt(String s)" in class java.lang.Integer.

Starting programs, main method, where does it go?

When you start a program written in Java, you specify the name of the class that Java should start in. It looks in that class to find a method "public static void main(String[] arg)" and starts running there.

Could you compare how C runs a program versus how java does it?

An exe file has a starting address in the code, usually named "main", and the operating system jumps to that location when your program starts running. A java class file has a static method called main and Java jumps to that location when your program starts running.

make a program from scratch dealing with stuff from HW 3

When will we be creating standalone programs (ending in .exe) that we can run outside of BlueJ? I'd like to see how this is done.

I'd like to know a little bit more about compilers, or what BlueJ is doing "for us" when we use it. Thus far I've been able to figure out all of the "setup" provided for the homework as far as the ready-made code is concerned. The roles of the compiler, the virtual machine, machine language, etc remain rather fuzzy concepts for me.

How do I run my programs outside of BlueJ?

Use the "java.exe" program to run your program. I'll explain in class Friday and Monday.

Input/Output

How to use items from outside of Java commands (like sounds and pictures) in Java.

See ShapeSlider in homework 4 for examples of both of these.

How do we input values. eg, for factorial (or any other method) if we want the user to input "n", how is that done?

I'll explain in class Friday and Monday.

Comments and using JavaDoc

The @param stuff.

The comments that you write in a particular format can be read by the Javadoc program to create web pages describing your code. See the lecture from July 1 for information and a pointer to the javadoc documentation.

UWCSE library

Where is the source code for the UWCSE library?

It's in the uwcse.jar file. Use winzip to open the file, then extract the contents to a directory.

Is there a way to write a single pixel (not a rectangle, for example) in the UWCSE graphics library?

I don't think so.

Homework

Hw 3.

HW3.

How all the different files of HW3 were connected and how we would do it on our own.

Ask questions as we go along. I try to explain this when I hand out the assignments, but there are too many details to cover everything in class. Come to office hours.

How can you utilize the prop files to make it work in program?

The contents of the prop files are read in and passed to your constructor when a new object is being created. So just edit the file, and the new values will show up in pMap when your code is running.

Can you show us interesting things people have added to the homework?

Yes, we will do this.

make homework a little more clear. Please don't put other quotes that we don't need to touch to confuse us.

I've separated the code that you don't need to change so that it is not in the way. However, I don't want to hide it.

Midterm

What to expect on midterm?

Questions like those on the homework. The exams are based on the slides and the homework. Some programming, but not a complete program from scratch. I'll talk about this on Wednesday.

General class issues

☺

All the box and arrow diagrams in the book and lecture notes are not always clear.

Ask me to clarify in class or in office hours.

Completely lost. Please slow down the speed of lecture!

We've got a lot to cover. The pace is designed to get you a useful set of knowledge by the end of the quarter. Make sure you're doing the reading and assignments, and ask questions.

I'm confused about a lot and my TA isn't much help. He's really good at saying "yes, that doesn't work" but horrible at saying … and this is why; try doing something like this, and here's what I think you didn't understand.

Keep asking. Make sure you've done what you can to understand the material too. The TAs want to help, but they are not mind readers and they can't do the work for you.