Writing a Simple Application

In this part of the lab we will learn how to develop an application that uses already-defined classes. We will see that we do not need to know how the already-defined classes get their work done. All we need to know is how to use them. This primarily means what methods do the classes offer for public use and what do these methods require for correct use.

Let's say that we wanted to develop a small console application that asks the user for a location to place one Fish object, create that object, place it at the desired location, report that we did place the Fish object, move the Fish object, and then report its current location. We would look at information concerning the Fish class and all supporting classes like Environment and Location and see what methods (and public attributes) they have and how they interact with each other. Then we would design our application class, work out the algorithm (sequence of logical steps) for the "main" method and proceed to write Java statements.

How would we "look at the information" for each class that we need? Well, we don't need to look inside each class and understand each statement for that. (We may not even have the source code to do that, e.g. the Location class supplied inside one of the .jar files.) All we need to do is to look at the Java documentation that comes with these files.

Let's begin. For the sake of time we'll skip the design phase (next time we'll do this) and go straight to the code.

**Fiddling with the Location:**

1. Start a new Java file from whatever editor that you are using and save that file as MyFishApp.java in the "Code" subdirectory of JavaMBS. Please make sure that your editor is creating a Java file and not a project. Also make sure that you told the editor the correct location to save your file or you will have to go look for it and move it.
2. Type the following statements into the editor and then click on the "save" icon (please save frequently.):  
   public class MyFishApp  
   {  
    public static void main(String[ ] args)  
    {  
    }  
   }
3. We need to get a location from the user. For us to do this we need to know what a location is. Using Windows Explorer locate the JavaMBS directory and double-click on a file in it called "index.html". A browser (probably Internet Explorer) will start up and load the index.html page. Move down the page and click on the "Documentation" link. You will be placed into the Java online document for the MBSCS. The left frame of the window shows the names of all of the classes developed for the case study. Clicking on any of those links causes documentation to show up in the frame on the right.  
   Click on Location and scroll down on the right frame until the Constructor and Method Summary tables are in sight.
4. How would you recognize a constructor if you were looking through a class' description?  
   \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
5. What is **a** constructor a special type of?  
   \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
6. What is the "signature" of a method, in general?  
   \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
7. What is the "signature" of the Location constructor?  
   \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
8. How many parameters does the Location constructor have and what types are they?  
   \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
9. What is a "default" constructor?  
   \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
10. Does the Location class have a "default" constructor?  
    \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
11. When the Location class is compiled, will the compiler create a "default" constructor for it? Explain.  
    \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
12. What is the return type of the method col()?  
    \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
13. What is the return type of the method toString()?  
    \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
14. Do you see the modifier "static" in front of any of the return types of any of the methods of Location?  
    \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
15. Do you need a Location object to directly access the method row() of the Location class? Explain.  
    \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
16. Can I use the name of the class to access **any** of the methods in the Location class? Explain.  
    \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
17. Will the following line define a reference variable of type Location, create an object of type Location, or define **and** create an object of type Location?  
     Location loc;  
    \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
18. What will the following statement do?  
     Location loc = new Location(10, 20);  
    \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
    \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
19. Why not test this out, for understanding, by typing in the following statements INSIDE the body of the "main" method in your editor. Save the file, then compile and run it (like you did to MBSGUI) at the prompt. You won't see any graphics stuff this time though. Record what the screen shows when you run the code.  
     Location loc = new Location(10, 20);  
     System.out.println("Our row position is " + loc.row( ) );  
     System.out.println("Our column position is " + loc.col( ) );  
    \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
    \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
20. What does the System.out.println( ) method do?  
    \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
21. Why did we need to write loc.row( )? Explain.  
    \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* **Checkpoint 2: Ask your instructor to review what you have done and initial here please.**

**Getting the Location from the user:**

1. OK. We know how to make a location object IF we had the location coordinates like 10 and 20. Since we are going to get this from the user at the keyboard we will need statements that will get the coordinates as input BEFORE we create the location object. Modify the "main" method of the MyFishApp so that it looks like the following (The modifications are underlined.):  
     
   public static void main(String [ ] args) throws IOException  
   {  
    BufferedReader keyboard  
    = new BufferedReader(new InputStreamReader(System.in));  
     
    System.out.println("Please enter a row position on the grid >= 0 ");  
    String rowStr = keyboard.readLine();  
    System.out.println("Please enter a column position on the grid >= 0 ");  
    String colStr = keyboard.readLine();  
     
    int row = Integer.parseInt(rowStr);  
    int col = Integer.parseInt(colStr);  
     
    Location loc = new Location(row, col);  
    System.out.println("Our row position is " + loc.row( ) );  
    System.out.println("Our column position is " + loc.col( ) );  
   }
2. Compile MyFishApp.java as before. (Remember to use classpath because you need the .jar files.) Record what happens below.  
   \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
3. The problem is now that you have specified the use of classes within MyFishApp that were neither defined by you nor were supplied in any of the case study files. In particular, you referred to BufferedReader and InputStreamReader both of which are in the package java.io which is **not** automatically imported by the compiler.
   1. Which package is automatically imported for all compilations?  
       \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
   2. What is the command to import **all** classes in the java.io package?  
       \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
   3. Modify MyFishApp.java so that you place the appropriate import statement at the head of the file and then recompile. What happened?  
      \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
   4. Run the MyFishApp application and when prompted for a row value, enter 10. When prompted for a column value, enter 50. What is the result on the screen?  
      \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
      \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
   5. Run the application again and give -5 for a row and 15 for a column. What is the result?  
      \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
      \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Run the application again and this time when prompted for row and column values enter the words ten and fifty, respectively. What is the result?  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_