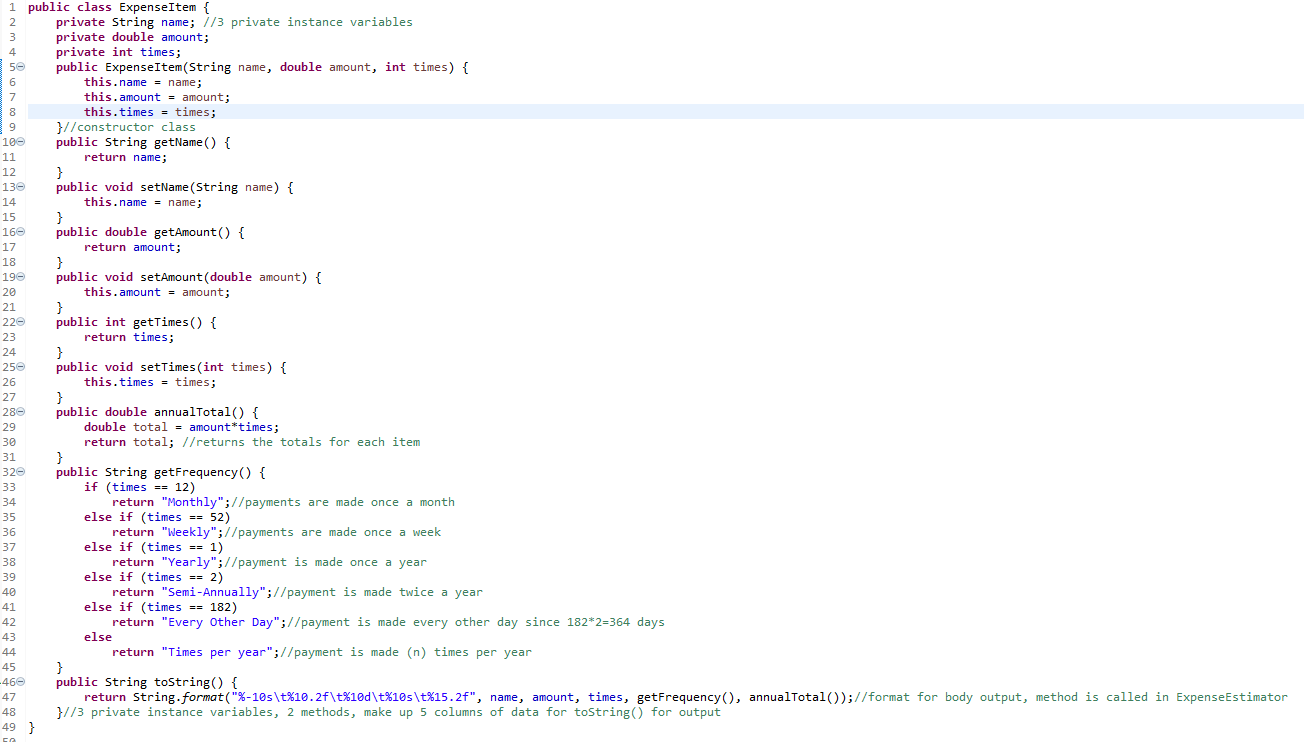
Nick Iudiciani

Prof. Harry Zhu

MIST3050 – Business Applications Development

1. ExpenseItem class that is called from a separate class in order to show a purchased item’s name, amount, and purchase frequency based on number of days per year. There are also 3 additional methods, annualTotal which shows the yearly expense for each item, getFrequency that will return a label for each item’s purchase basis, and toString which will be called in a separate class and returns the 3 variables along with the annualTotal and getFrequncy methods.

Read and observe the above code. Note that it is designed to reference another class ExpenseEstimator, but the following questions pertain only to the code mentioned here.

1. How many variables are defined in this class? Please state their respective data types, instances, and variable names as well.

3 variables, private String name, private double amount, private int times.

1. Briefly explain the role of each of the getters and setters for each variable.

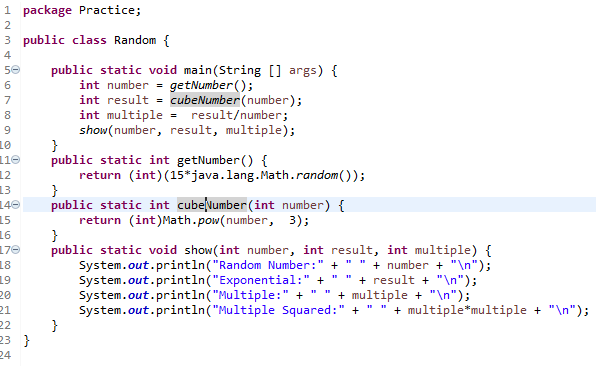
Essentially getters and setters do just what they say, retrieve and set or change the variables that are defined. The reason why they are done in a separate class from where the data is defined is because the variables are defined as private instance variables, meaning that if they were confidential information such as passwords, then they can’t be called as easily as if they were public variables. If you were to set the name of an item to “Coffee” then the getName will access it from the array or data that is defined.

1. Note the order in which the private instance variables are defined (String, double, int). Let’s say I were to create a 1D array for an item in the ExpenseEstimator class to call an output. It goes: {new ExpenseItem(“Coffee”, 182, 2.67)}; This shows the item name, the number of days per year the item is purchased, and the item’s amount. If I were to call the item with this order (String, int, double), will the output show the item correctly? Or will the program throw an error? Explain why.

The program will throw a compilation error. Because the private instance variables and the method where they are called (Line 46 for toString() method) are in the form (String, double, int), and they are called in that order as well, then the ExpenseEstimator array must document that data in this form as well. This will allow the methods to run without error.

1. Please draw the UML diagram for this ExpenseItem class.

|  |
| --- |
| ExpenseItem |
| -name:String  -amount:double  -times:int |
| +<<constructor>> ExpenseItem(name:String, amount:double, times:int)  +getName():String  +setName(name:String):void  +getAmount():double  +setAmount(amount:double):void  +getTimes():int  +setTimes(times:int):void  +annualTotal():double  +getFrequncy():String  +toString():String |

1. Essentially a random number generator (despite following an algorithm) that will find a random number for the specified boundary, raise that number to the 3rd power (or cube the number), divide both of those numbers to find its multiple, and square the multiple (multiply it by itself).
2. How many methods are in the body of this class?

4 methods; the main method, the getnumber() method, the cubeNumber() method, and the show() method.

1. In the getNumber() method, what does 15\*java.lang.Math.*random*() signify? (Hint: there is no multiplication done despite the \* sign).

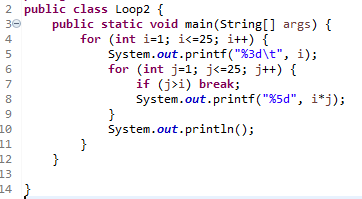
The 15\*java.lang.Math.random() parameter shows that the random number that is generated is on the range of [0,15) meaning that the number can be 0 and any number between 0 and 15.

1. By type casting getNumber() and cubeNumber() to int, does this increase or decrease the chance of the output showing a *greater* variety of numbers?

This will decrease the pool of potential numbers. This is basically how a lottery works. It will pick any whole number between the given boundaries but will not include decimal numbers.

1. Briefly explain how your local lottery system might use this piece of code, like which boundaries do they set, are they int type casted numbers, etc. (not necessarily the multiple or squared multiple part). Bonus: Why does a lottery system reflect the aspect of randomness, when really the number you get from the lottery is never truly random? (Requires higher level thinking than just coding and Java).

For the lottery system such as Mega Millions, the system uses whole numbers between 1 and 70, including 1 and 70 as well. They use a non-repeating system where you can’t draw the same number twice. It does this cycle 5 times and those are what are called the winning numbers. Then a 6th number is chosen from 1 to 25 which is the powerball number, and contestants have to wait for the grand drawing to match up their numbers. When it comes to randomness, there really is no true random number generator. Each number that is drawn follows a set algorithm which isn’t random at all.

1. A multiplications table program with a for loop and a nested for loop. Will return an index i number, and the multiplications table for that number that stops at the index number itself. The nested for loop will halt the multiplication for each index when the number of multiples exceeds the index number (the last number it multiplies is the number equal to the index, or when i\*i is the last step). This makes for a cool staggered appearance when you increase the boundaries.
2. What does “%3d\t” mean? Will I break the syntax if I were to write “%3d\t\t”? Why is this format in quotation marks but the variable i is not?

%3d\t in quotation marks is the language for setting the format of a given data type. In this case, 3d shows that the format requests 3 spaces to the left of the data type, with “d” being the notation for an int data type. \t shows that after the index number is shown, there is a space the size of the tab key on the computer followed by the next part of code which would be the multiplication part. This is in quotation marks because printf() method requires the quotation marks to show the format, and the variable to be separated by a comma, in this case the index i number.

1. If j is less than or equal to 20, and i is set to less than or equal to 25, the index number i will still stop at 25, but the multiplication of these numbers when j is equal to 20 will change the staggered look of this output. Explain.

Rather than the continuous staggered look for the output, setting i to 25 and j to 20 will allow the index to go to 25 but the multiplication will stop at the 20th multiple rather than continue to 25, since j<i in this case. So for index i of 21 to 25 will only show the first 20 multiplications for each index.

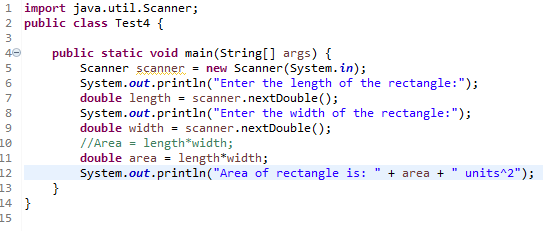
1. If you were to comment out or disable the nested for loop in this program, what will the output of the program look like?

If you were to omit the nested for loop for the variable j, then the output will only show a list of just the index numbers, in this case 1 to 25.

1. Explain what the println function will do in line 10 of this program. What will happen if you were to omit it from the source code and run the program?

The println() in line 10 serves as a cleaner for the organization of the list. After each cycle through the for loops, this println will make a new line for each index and multiplication. If you were to get rid of this println, then the data will all be on one line and will be tough to read and understand. This can also be accomplished by using a printf function and using the \n after each pass to grant a new line for data.

1. Short program that calculates the area of a rectangle. Uses a text scanner class that can parse primitive types and strings and can also freeze the program until further action is implemented. It asks the user to input the dimensions of the rectangle itself. After implementing and running this program, input a number for the width, then hit enter, then input a number for the length, and then hit enter again to calculate the area.



1. Please explain the java.util.Scanner class and its relation to the program.

The java.util.Scanner is a class that will parse primitive and String data types by using regular expressions. It can be used for a class that requires manual input by a user and can even block steps in the code in order to force manual input.

1. Please explain the System.in parameter and its relation to the program.

System.in is a parameter that allows the system to read manual input rather than just file input. This means that the Scanner object is going to read and respond what the manual input is and will react accordingly.

1. Since the parameters are type casted as double, and a multiplication process is performed afterwards, how many decimal places will the program return for the area? Will there be additional decimal places if the user specifies decimal numbers for the length and width?

The double type cast will return each whole number with one decimal place, so if you say the width and length are 5 and 6, then the area will output as 30.0 due to the double cast. But if you were to input 5.5 and 6.5 as the dimensions, then the area will return as 35.75.

1. Explain scanner.nextDouble(). (Hint: notice where in the code it is and what is being done at that step, relative to the actual output and user input).

Because this method occurs right after both println methods, then the scanner will stop the entire program until a number with double data type is manually entered by the user. It requests the length, then the width and then it will run the area method by itself.