

The contents of this document represent and go over the topics covered in lecture on 1/31 and 2/2 Use this to review the content as you please, also be sure not to use these in place of lecture as these documents will only cover lecture highlights and important details and do not represent the full scope of what you may be tested on. As always if you have any questions ask on ED!

Lecture 5 - 1/31/2023

In Java we have to denote the type of our variables when we declare them, because Java is ***Strongly-Typed*** we can separate these data types into primitive and nonprimitive data types.

You are familiar with the primitive types from the past couple of lectures.

1. int
2. double
3. boolean

To declare and initialize a variable in Java - in the generic sense - below is the following format:

datatype variableName = valueOfVariable;

The data type of the variable and values matter, in order to perform floating point division, one of the operands needs to be a floating point value, even if the data type of the variable you are assigning the result to is already a floating point number.

While Java has a lot of functionality natively built into it from the get go, there is a lot that we need to use that isn't in the native library. To bring in additional libraries into Java we can use what is called an import statement. Import statements come before the class declaration in a Java file, an example that you have seen in lecture is the following statement:

import java.util.Scanner;

This statement imports the Scanner class into the current file. If instead of importing a single class from a package but you want to import multiple classes from the same package you can replace the class name with a wildcard:

import java.util.*;

Going back to the Scanner class from earlier, it is essential to collecting user input in Java. In order to use a Scanner we need to instantiate it. The following is how you instantiate a Scanner in Java:

Scanner sc = new Scanner(System.in);

The above line is called Object instantiation, Objects are instances of classes, and you can either use predefined classes to make objects or you can make objects from your own class! The following is a generic version of the previous line which you can use in order to instantiate an object:

ClassName variableName = new ClassName(argument1, ... , argumentN)

The 'new' keyword is integral to creating Objects, without it you cannot create objects in Java, it is best to get used to this syntax now so you will be more comfortable down the road. Once we have instantiated a Scanner object we can print out a prompt to the user and then take in input using a method that is a part of the Scanner class. The following examples are ways to do this:

1. `int x = sc.nextInt();`
2. `double x = sc.nextDouble();`
3. `String x = sc.nextLine();`
4. `String x = sc.next();`

The last two of the methods vary slightly please reference the Scanner class documentation for more information: ([Scanner Java 8](#))

Lecture 6 - 2/2/2023

I see a lot of you typing your notes in the Java files and while you may be used to writing comments like this:

// this is a single lined comment

You can actually do multilined comments as well, the following is an example of a multilined comment in Java:

**/* This comment
* spans over Multiple lines
*/**

From the Lecture you saw the rock paper scissors java file and there were three new concepts we saw. The first was the concept of constants. In Java you can denote constants with the final keyword prior to the data type, the convention is to type the variable name in all caps with underscores separating individual words. The following is an example of this:

final int MY_FIRST_CONSTANT = 42;

The next concept you saw was the use of the Math class's random method. The Math class is native to Java and so you do not need to import it like you would do for a language like python. The Math class has tons of different useful constants and methods you can use and I encourage you to read the documentation here: [Math Class Java 8](#)

Math.random(), returns a double that can hold any value across the following range [0,1) that is to say we can get values starting from 0 up to but not including 1. This allows us to generate random numbers and perform actions based on chance and probability rather than discretely defined values. We can manipulate this range via multiplication and addition/subtraction. Try it out! Using the range I gave you earlier, apply the example from the lecture and see what new range you get.

Finally you were introduced to Type Casting, this is different from Type Promotion which happens naturally by the JVM, casting is a user's choice and when desired, can yield necessary and proper results but also could lead to errors if you are not careful. To cast to a different type you simply put (**datatype**) in front of an expression, be careful for the order of precedence as well, as you may get undesirable output, when in doubt use parentheses to denote intention.

At this point you can complete your homework due 2/6. I will put the Object Oriented concepts in next week's document. Best of luck on your homework!