Welcome to the Using Objects lab!

Learning Outcomes

By the end of this lab:
1. Be able to define chapter 3 terms.
2. Describe reference variables and compare with primitive data type variables.
3. Draw memory model diagrams contrasting primitive with reference types. Include int, Integer, String, Scanner, etc.
4. Describe char and Character.
5. Describe how calling an instance (non-static) method is different from calling a class (static) method.
6. Use API to lookup Scanner, String, Random, Integer, and Double methods.
7. Trace multiple instance methods calls for Scanner and String.
8. Describe String concatenation and immutability.

Exercise A: Terms

object vs instance vs class, binary, decimal, char, char literal, ASCII, escape sequence (\n, \t, \\, \"), primitive type vs. reference type, a string vs String class, reference, System.in, Integer, Double, wrapper class, Random, Scanner, static method call, non-static method call, API, immutable, package, import, debugging

Exercise B: Trace and Explain

A. double num1 = 9.0;
   Double num2 = 9.0;

   When tracing these, can you draw a picture of memory for each? Check your picture by viewing in Java Visualizer. In Java Visualizer, remember to check: options > Show String/Integer/etc objects, not just values.
   When explaining, describe where the compiler will do an implicit type conversion.

B. int numA;
   numA = 3;
   Integer numB;
   numB = new Integer("3");

   Can you describe in detail what is happening here?
   Sample Answer (select, copy and paste into notepad to read): numA is a variable declared with data type int. The statement numA = 3; puts the int literal 3 into the variable numA. The statement Integer numB; declares numB to be a variable intended to hold a reference to an instance of Integer. The expression new Integer("3") instantiates/creates an instance/object of class Integer with the initial value of 3. The result of the expression is a reference/address/pointer to the location in memory of the newly created instance. The assignment statement puts the reference into the variable numB.

C. String str;
   str = "Your Name";
Is String a class or primitive data type? Are the characters "Your Name" stored in the memory location referred to by str? What is stored in str?

D. String abc = " a b c ";
   System.out.println( abc.trim());

If you don't know what trim() does, where can you find the description the method (or any method)? What are 3 other String methods?

E. String str = " r g h ";
   String.trim();

F. int length = " strange looking but does it work?".length();
   System.out.println( length);

G. String str = "HELLO"
   str.toLowerCase();
   System.out.println( str);

Is the string printed out in all uppercase letters? Why/Why not? What does immutable mean?

E. String str = "hello"
   String str2 = str.toUpperCase();
   System.out.println( str2);

Is the string printed out in uppercase now? Why?

F. String str = "Falling Off a Cliff " + "by Eileen Dover"
   System.out.println( str.substring(3,6));

What exactly will be print out? What is the meaning of each of the parameters to the substring method?

G. String str2 = "The Future of Robotics " + " by Cy Borg and Anne Droid"
   System.out.println( str2.substring( str2.indexOf('e') + 1,
                                      str2.indexOf('o')).trim());

What will be print out? Describe in your own words what the substring, indexOf and trim methods do.

H. char ch = Character.toUpperCase('a');
   System.out.println( ch); 

I. Character ch = new Character('A');
   System.out.println( ch.isLowerCase());

J. int num = '1';
   System.out.println( num);
I. Random rand = new Random();  //remember to import java.util.Random
int num1 = rand.nextInt();
int num2 = rand.nextInt();
int num3 = rand.nextInt();

Consider: will the nextInt() method return the same value each time? What is the range of possible values returned? Each time you run this will the values returned be the same or different?

J. Random numGenerator = new Random();
System.out.println( "number: " + Random.nextInt());

K. Random numGenerator = new Random();
int num1 = numGenerator.nextInt(10);
int num2 = numGenerator.nextInt(10);
int num3 = numGenerator.nextInt(10);

Does the value returned from nextInt method call change each time? What are the possible values returned? If you re-run this same code are the results the same?

L. Random rand = new Random();
rand.setSeed( 365);
int num1 = rand.nextInt(6);
int num2 = rand.nextInt(6);
int num3 = rand.nextInt(6);

How does setting a seed effect the results? What would you change to get the values 1-6, such as for simulating a dice roll?

To learn more about Scanner see https://cs200-www.cs.wisc.edu/wp/resources/learn-to-use-scanner/

M. java.util.Scanner stdin = new java.util.Scanner( System.in);
String str = stdin.nextLine();

N. Scanner stdin = new Scanner( System.in);  //remember to import java.util.Scanner
int num = stdin.nextDouble();

O. Scanner input = new Scanner("This is the input.");
String str = input.next();
str = input.next();
System.out.println( str);

P. Scanner input = new Scanner("one 1 2.0");
System.out.println( input.nextDouble());

Q. Scanner input = new Scanner("one 1 2.0");
String str = input.next();
System.out.println( input.nextDouble());

R. Scanner input = new Scanner("1 2 3\n 4 5 6");
   String str = input.next();
   System.out.println( input.next());

S. Scanner input = new Scanner("1 2 3\n 4 5 6");
   String str = input.nextLine();
   System.out.println( input.nextInt());

T. Scanner input = new Scanner("1 2 3\n 4 5 6");
   System.out.println( input.nextInt());
   System.out.println( input.nextInt());
   input.nextInt();
   System.out.println( input.nextInt());

U. Scanner input = new Scanner("1 2 3\n 4 5 6");
   System.out.println( input.nextInt());
   System.out.println( input.nextLine());
   input.next();
   System.out.println( input.nextInt());

Exercise C: Fix Order and Indenting of Lines

}
import java.util.Scanner;
class Scanner2 {
}
public static void main(String args[]) {
   scnr.close();
   int job = scnr.nextInt();
   double fee = scnr.nextDouble();
   System.out.println("Job:" + job + " name: " + name + " fee: " + fee);
   System.out.println("Enter your name: ");
   System.out.println("Enter your fee: ");
   System.out.println("Enter a 5 digit job number: ");
   String name = scnr.next();
   Scanner scnr = new Scanner("Sara 39.95 43125");

Exercise D: Body Mass Index

**Body Mass Index** (BMI) is used to estimate the amount of muscle, fat and bone in an individual. The BMI is defined as the body mass (kilograms) divided by the square of the body height (meters). Write a program to calculate a BMI based on height in meters and weight in kilograms.

Height (meters): 1.6
Weight (kilograms): 65
BMI: 25.39062499999996

Adapt to take input in inches and pounds. Find the appropriate formulas on the web. There are calculators on the web that you may use to compare results. Express any constant values, such as meters in an inch, as a constant: final double METERS_IN_INCH = 0.0254;

Height (inches): 63
   1.6001999999999998(meters)
Weight (pounds): 143
   64.86370891143507(kilograms)
BMI: 25.33105313448013

Adapt further to take input in feet, inches and pounds.
Height (feet): 5
Height (inches): 3
   1.6001999999999998(meters)
Weight (pounds): 143
   64.86370891143507(kilograms)
BMI: 25.33105313448013

Exercise E: Write Program

Write a program to use a Scanner to read the string, reading each number, "ControlType: 1\nFIRE @ 541.0 387.0\nHERO @ 397.0 287.0\nBUILDING @ 522.0 174.0"

and print out as:
building (522.0, 174.0)
hero (397.0, 287.0)
fire (541.0, 387.0)
using control type: 1

Additional Learning Materials
When you have mastered everything in this lab, then you are welcome to learn from additional learning resources available on the web and beyond this course:
https://cs200-www.cs.wisc.edu/wp/learn-to-program-resources/

Note: Due to programs and zyBooks being individual work, it is Not appropriate to work on them during the Team Lab.

Lab designed by Jim Williams.