Joshua McCann  
LessonFundamentalsLAB

1. Create a new private static method called LessonFundamentalsLAB().

|  |
| --- |
| private static void LessonFundamentalsLAB(){ } |

1. What is a variable? Write code that prompts the user to enter a value from 1-10, adds 1.75 to the number then outputs it to the console.

A variable is a piece of memory allocated to store data values.

|  |
| --- |
| int qTwoInt; Scanner inputScanner = new Scanner(System.in);  System.out.println("Please enter a number between 1 and 10: "); qTwoInt = inputScanner.nextInt(); System.out.println("1.75 added to " + qTwoInt + " is " + Double.toString(1.75 + qTwoInt)); |

1. Research the eight data types in Java, create variables for these, and output the default, max, and min values (where applicable).

The eight data types are boolean, byte, char, short, int, long, float, and double.

|  |
| --- |
| //declare static variables outside of static class static boolean qThreeBoolean; static byte qThreeByte; static char qThreeChar; static short qThreeShort; static int qThreeInt; static long qThreeLong; static float qThreeFloat; static double qThreeDouble;  //use variables inside of static class for default variables System.out.println("\nBoolean\nDefault: " + qThreeBoolean); System.out.println("\nByte\nDefault: " + Byte.toString(qThreeByte) + "\nMax: " + Byte.toString(Byte.MAX\_VALUE) + "\nMin: " + Byte.toString(Byte.MIN\_VALUE));  System.out.println("\nChar\nDefault: " + Character.toString(qThreeChar) + "\nMax: " + Character.toString(Character.MAX\_VALUE) + "\nMin: " + Character.toString(Character.MIN\_VALUE)); System.out.println("\nShort\nDefault: " + Short.toString(qThreeShort) + "\nMax: " + Short.toString(Short.MAX\_VALUE) + "\nMin: " + Short.toString(Short.MIN\_VALUE));  System.out.println("\nInt\nDefault: " + Integer.toString(qThreeInt) + "\nMax: " + Integer.toString(Integer.MAX\_VALUE) + "\nMin: " + Integer.toString(Integer.MIN\_VALUE));  System.out.println("\nLong\nDefault: " + Long.toString(qThreeLong) + "\nMax: " + Long.toString(Long.MAX\_VALUE) + "\nMin: " + Long.toString(Long.MIN\_VALUE));  System.out.println("\nFloat\nDefault: " + Float.toString(qThreeFloat) + "\nMax: " + Float.toString(Float.MAX\_VALUE) + "\nMin: " + Float.toString(Float.MIN\_VALUE));  System.out.println("\nDouble\nDefault: " + Double.toString(qThreeDouble) + "\nMax: " + Double.toString(Double.MAX\_VALUE) + "\nMin: " + Double.toString(Double.MIN\_VALUE)); |

1. Strings are not part of the eight primitive data types, why not?  Which of the data types is a String actually composed of? Prove this in your code.

Strings are actually character arrays.

Table 1 – Two Tests

|  |
| --- |
| char qFourChar, qFourTestChar; qFourString = "Random Letters";  qFourChar = qFourString.charAt(2); qFourTestChar = 'n';  if(qFourChar == qFourTestChar) System.out.println("The characters are the same"); else System.out.println("The characters are not the same"); |
| char qFourCharArray[];  qFourCharArray = qFourString.toCharArray();  for(int i=0; i < qFourString.length(); i++){  System.out.print("\nString: Character at location " + i + ": " + qFourString.charAt(i));  System.out.print("\nArray: Character at location " + i + ": " + qFourCharArray[i]); } |

1. Create a List, populate it with values and then iterate thru those values outputting them to the console.  Why can you not create a List of 'char' or 'int'?

A list is a sub-interface for the class Collection and receives inheritance from the Collection. Collection is expecting a group of class objects. Int and char are primitive data types and not objects. As a result, a list can be composed of the wrapper for the primitive data types to be treated as an object, but a list cannot be created for primitive data types.

|  |
| --- |
| List<String> qFiveList = new ArrayList<String>();  for(int i = 0; i<10; i++){  qFiveList.add("String for question five in place " + Integer.toString(i + 1));  }  for(String printString : qFiveList){  System.out.println(printString);  } |

1. Create variables of appropriate data types, to store information regarding a lottery ticket (i.e. GameName, WinningNumbers, Jackpot, DrawingDate, etc..). Populate these variables and output them to the console.

|  |
| --- |
| String gameName = new String("Gopher 5"), winningNumbers = new String("09 29 34 38 45"); int costToPlay = 2, jackpot = 510000; DateFormat stringFormater = new SimpleDateFormat("MM/dd/yyyy"); Date drawingDate = new Date();  try {  drawingDate = stringFormater.parse("06/19/2017"); } catch (ParseException e) {  e.printStackTrace(); }  System.out.println("\nGame: " + gameName); System.out.println("Cost to Play: $" + Integer.toString(costToPlay)); System.out.println("Drawing Date: " + stringFormater.format(drawingDate)); System.out.println("Winning Numbers: " + winningNumbers); System.out.println("Prize: $" + Integer.toString(jackpot)); |