Introduction To Java

LESSON 2

Learning Outcomes : Lesson 2

- Random
- String
- Numeric Type Conversion
- Local Variables
- Static Variables
- Static Methods
- Constant

Math.Random

- The **java.lang.Math.random()** is used to return a pseudorandom double type number.
- The double number is greater than 0 and less than 1.
- If you want to specific range of values, you have to multiply the returned value with the magnitude of the range.
  - (Math.random() * ((max-min)+1))+min

String Object

- A String variable contains a collection of characters surrounded by double quotes.
- A String in Java is actually an object, which contain methods that can perform certain operations on strings.
- The java.lang.String class is used to create a Java string object.

String Methods

- **s1.length()** -> returns a number of characters in a string
- **s1.concat(s2)** -> join two strings
- **s1.toLowerCase()** -> returns all characters of string to lower case
- **s1.toUpperCase()** -> returns all characters of string to upper case
- **s1.trim()** -> returns a string without any whitespace on both sides

String Methods

- **s1.compareTo(s2)** -> returns 0 if two strings are equal
- **s1.compareToIgnoreCase(s2)** -> Same as compareTo except that the comparison is case insensitive.
- **s1.equals(s2)** -> returns true if s1 is equal to string s2.
- **s1.equalsIgnoreCase(s2)** -> returns true if s1 is equal to string s2; it is case insensitive.
- **s1.isEmpty()** -> returns true if the length of s1 is 0.

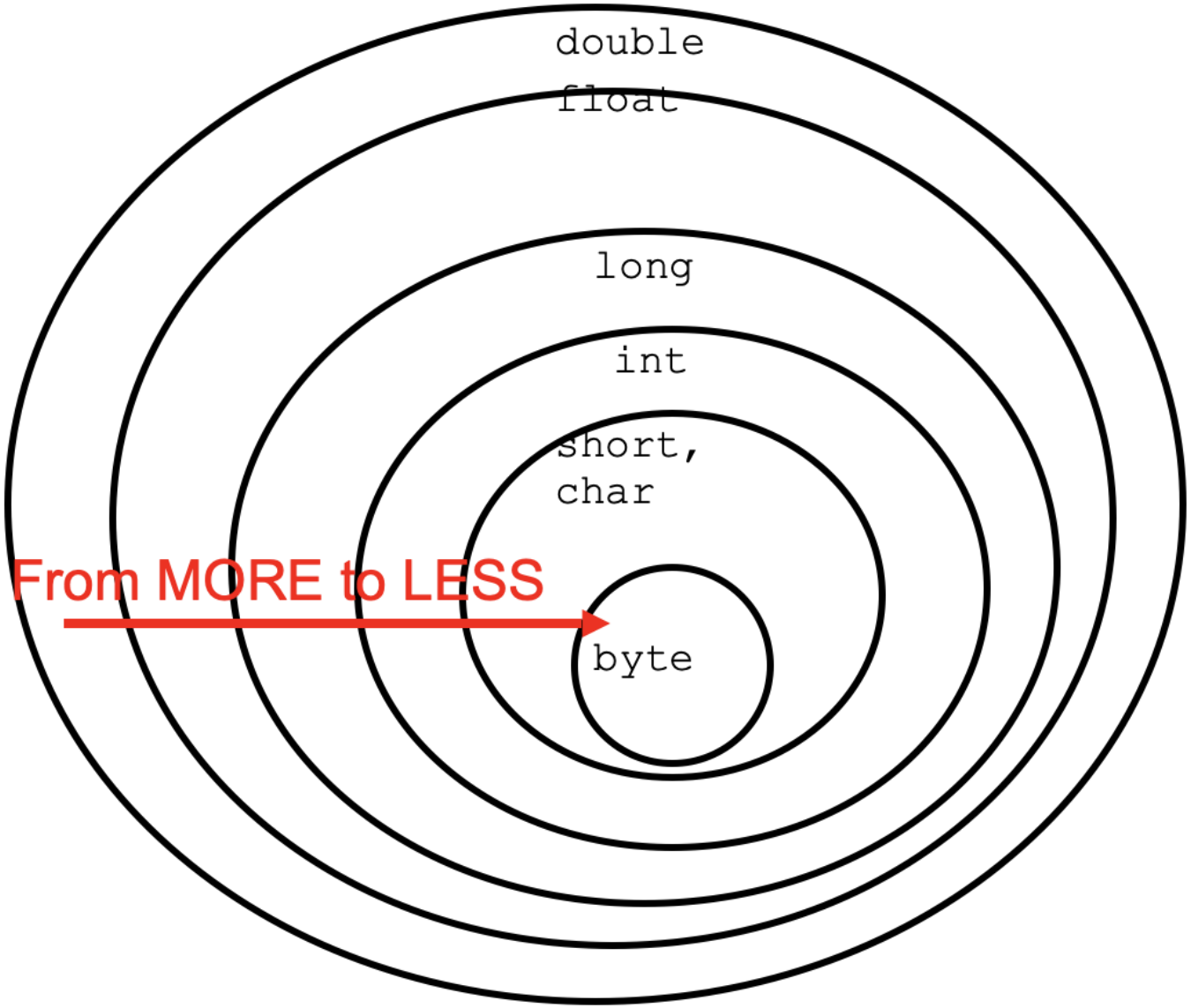
String Methods

- **s1.indexOf(char)** -> returns the index of the first occurrence of char in the string
- **s1.indexOf(char , fromIndex)** -> returns the index of the first occurrence of char after fromIndex in the string.
- **s1.indexOf(s2)** -> returns the index of the first occurrence of string s2 in s1.
- **s1.indexOf(s2, fromIndex)** -> returns the index of the first occurrence of string s2 in s1 after fromIndex.

Numeric Type Conversion

- Outer ring is most inclusive data type while inner ring is the least inclusive.
- In expressions variables and sub expressions of less inclusive data types are automatically cast to more inclusive.
- If trying to place expression that is more inclusive into variable that is less inclusive, explicit cast must be performed.
- You can always assign a value to a numeric variable whose type supports a larger range of values
- You cannot, however, assign a value to a variable of a type with a smaller range unless you use type casting

Primitive Casting



Casting

- Casting is the temporary conversion of a variable from its original data type to some other data type.
- With primitive data types if a cast is necessary from a less inclusive data type to a more inclusive data type it is done automatically.
- If a cast is necessary from a more inclusive to a less inclusive data type the class must be done explicitly by the programmer

Local Variables

- Local variables are defined inside a method
- They are short lived
- They are only accessible from within the method
  - The local scope
  - There is no scope modifier

Static Variables

- Static variables belong to the Class
- Because of this, the values are shared across instances variables
- You can access static variables without Class instantiation
- Declared with the keyword **static**

Static Methods

- Static methods, like static variables, belong to the Class
- Cannot use for Class variables or instance variables
- Can be called without Class instantiation
- Also declared with **static** keyword

Constants

- Constants are like variables but the value does not change
  - The value cannot be reassigned
- Declared with the **final** keyword at the Class level
- Named in all CAPITALS with _ to separate words
- Used to represent things that do not change
  - The value of pi
  - Bound checking values