

# CS120: Java Loops

W. Michael Petullo

University of Wisconsin–La Crosse

As of September 28, 2022



# Java Loops: while

$n > 0$ : a **Boolean**  
expression

```
while (n > 0) {  
    System.out.println(n);  
    n = n - 1;  
}  
System.out.println(" Blastoff!");
```

This block  
will repeat as  
long as  $n > 0$   
evaluates to true

Execution con-  
tinues linearly  
once  $n > 0$  eval-  
uates to false

An `while` statement causes a block to execute as long as a condition is met. This allows a program to repeat a portion of its instructions. Java evaluates the Boolean expression before each loop iteration.



# Java Loops: Infinite Loop

Beware ...

```
while (true) {  
    System.out.println("Somebody stop me!");  
}
```

... the infinite ...

```
while (n > 0) {  
    System.out.println("Somebody stop me!");  
    /* Forgot to change n! */  
}
```

... loop!

Press **Control-C** to terminate running program.



# Java Loops: for

Initializer

Condition

Update

```
for (int n = 10; n > 0; n--) {  
    System.out.println(n);  
}  
System.out.println(" Blastoff!");
```

Will repeat as long as  $n > 0$  evaluates to true

Execution continues linearly once condition evaluates to false

Note the scope of  $n$  does not exceed the for block.



# Java Loops: break

```
// Keep trying until the user enters a proper number.
Scanner in = new Scanner(System.in);
while (true) {
    System.out.print("Enter a number: ");
    if (in.hasNextDouble()) {
        break;
    }
    String word = in.next();
    System.err.printf("%s is not a number\n", word);
}
double x = in.nextDouble();
```

The `break` statement immediately and unconditionally terminates a loop, continuing after the loop's block.



# Java Loops: continue

```
// Compute a running total, but skip over negative values.  
// Entering 0 causes loop to terminate.  
Scanner in = new Scanner(System.in)  
int x = -1;  
int sum = 0;  
while (x != 0) {  
    x = in.nextInt();  
    if (x <= 0) {  
        continue;  
    }  
    System.out.println("Adding " + x);  
    sum += x;  
}  
System.out.println(sum);
```

The `continue` statement immediately jumps back to the top of the loop, continuing with the next iteration or terminating.



# Java Loops: End of File

```
1  import java.util.Scanner;

3  class Eof {
4      public static void main(String[] args) {
5          Scanner in = new Scanner(System.in);
6          while (in.hasNextLine()) {
7              String line = in.nextLine();
8              System.out.println(line);
9          }
10         System.out.println("Done!");
11     }
12 }
```

Press **Control-D** to indicate end of file (end of input).



# Java Loops: Pseudo Code

**Pseudo code** is an informal language that is useful for expressing the idea of a program without getting bogged down in the syntactic details.

**Iterative development** captures the idea that you can write a piece of your program, test that piece, and then write another piece.





# Scope: Introduction

- ▶ Scope refers to the visibility of a variable, field, or method name
- ▶ An item is visible by name in the block in which it was defined, as well as any in blocks therein
- ▶ Names in the top-most block as said to have global scope, because they are visible anywhere in the program
- ▶ Java may free the memory associated with an object when the last reference to that object is no longer in scope

Aim for minimal scope necessary.

```
class Max {  
    public static int max(int [] a) {  
        int m = -1;  
        for (int i = 0; i < a.length; i++) {  
            if (a[i] > m) {  
                m = a[i];  
            }  
        }  
        return m;  
    }  
}
```



# Scope: Max.java

```
class Max {  
  a public static int max(int [] a) {  
    int m = -1;  
    for (int i = 0; i < a.length; i++) {  
      if (a[i] > m) {  
        m = a[i];  
      }  
    }  
    return m;  
  }  
}
```



# Scope: Max.java

```
class Max {  
  a  public static int max(int [] a) {  
    m  int m = -1;  
    for (int i = 0; i < a.length; i++) {  
      if (a[i] > m) {  
        m = a[i];  
      }  
    }  
    return m;  
  }  
}
```



# Scope: Max.java

```
class Max {  
  a public static int max(int [] a) {  
    m   int m = -1;  
      i   for (int i = 0; i < a.length; i++) {  
          if (a[i] > m) {  
              m = a[i];  
          }  
      }  
      return m;  
  }  
}
```



# Scope: Stats.java

```
public static void main(String[] args) {
    double max = 0, total = 0;
    double[] number = new double[args.length];
    for (int i = 0; i < args.length; i++) {
        number[i] = Double.parseDouble(args[i]);
    }
    for (int i = 0; i < number.length; i++) {
        if (number[i] > max) {
            max = number[i];
        }
    }
    for (int i = 0; i < number.length; i++) {
        total += number[i];
    }
    System.out.printf("max: %f\n", max);
    System.out.printf("mean: %f\n", total / number.length);
}
```



# Scope: Stats.java

```
args
public static void main(String[] args) {
    double max = 0, total = 0;
    double[] number = new double[args.length];
    for (int i = 0; i < args.length; i++) {
        number[i] = Double.parseDouble(args[i]);
    }
    for (int i = 0; i < number.length; i++) {
        if (number[i] > max) {
            max = number[i];
        }
    }
    for (int i = 0; i < number.length; i++) {
        total += number[i];
    }
    System.out.printf("max: %f\n", max);
    System.out.printf("mean: %f\n", total / number.length);
}
```



# Scope: Stats.java

```
public static void main(String[] args) {  
    double max = 0, total = 0;  
    double[] number = new double[args.length];  
    for (int i = 0; i < args.length; i++) {  
        number[i] = Double.parseDouble(args[i]);  
    }  
    for (int i = 0; i < number.length; i++) {  
        if (number[i] > max) {  
            max = number[i];  
        }  
    }  
    for (int i = 0; i < number.length; i++) {  
        total += number[i];  
    }  
    System.out.printf("max: %f\n", max);  
    System.out.printf("mean: %f\n", total / number.length);  
}
```

args

max, total, and number



# Scope: Stats.java

```
public static void main(String[] args) {  
    double max = 0, total = 0;  
    double[] number = new double[args.length];  
    for (int i = 0; i < args.length; i++) {  
        number[i] = Double.parseDouble(args[i]);  
    }  
    for (int i = 0; i < number.length; i++) {  
        if (number[i] > max) {  
            max = number[i];  
        }  
    }  
    for (int i = 0; i < number.length; i++) {  
        total += number[i];  
    }  
    System.out.printf("max: %f\n", max);  
    System.out.printf("mean: %f\n", total / number.length);  
}
```

args

max, total, and number

i

# Scope: Stats.java

```
public static void main(String[] args) {  
    double max = 0, total = 0;  
    double[] number = new double[args.length];  
    i1 for (int i = 0; i < args.length; i++) {  
        number[i] = Double.parseDouble(args[i]);  
    }  
    i2 for (int i = 0; i < number.length; i++) {  
        if (number[i] > max) {  
            max = number[i];  
        }  
    }  
    for (int i = 0; i < number.length; i++) {  
        total += number[i];  
    }  
    System.out.printf("max: %f\n", max);  
    System.out.printf("mean: %f\n", total / number.length);  
}
```

args

max, total, and number

# Scope: Stats.java

```
public static void main(String[] args) {  
    double max = 0, total = 0;  
    double[] number = new double[args.length];  
    i1 for (int i = 0; i < args.length; i++) {  
        number[i] = Double.parseDouble(args[i]);  
    }  
    i2 for (int i = 0; i < number.length; i++) {  
        if (number[i] > max) {  
            max = number[i];  
        }  
    }  
    i3 for (int i = 0; i < number.length; i++) {  
        total += number[i];  
    }  
    System.out.printf("max: %f\n", max);  
    System.out.printf("mean: %f\n", total / number.length);  
}
```

args

max, total, and number

# Scope: Stats.java

```
public static void main(String[] args) {  
    double max = 0, total = 0;  
    double[] number = new double[args.length];  
    i1 for (int i = 0; i < args.length; i++) {  
        number[i] = Double.parseDouble(args[i]);  
    }  
    i2 for (int i = 0; i < number.length; i++) {  
        if (number[i] > max) max = number[i];  
    }  
    i3 for (int i = 0; i < number.length; i++) {  
        total += number[i];  
    }  
    System.out.printf("max: %f\n", max);  
    System.out.printf("mean: %f\n", total / number.length);  
}
```

**args**

**max, total, and number**

**Java provides Double;  
its global scope spans  
the entire program**

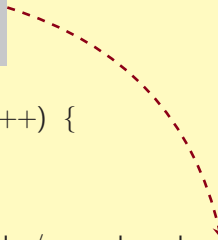
# Scope: Stats.java

```

public static void main(String[] args) {
    double max = 0, total = 0;
    double[] number = new double[args.length];
    i1 for (int i = 0; i < args.length; i++) {
        number[i] = Double.parseDouble(args[i]);
    }
    i2 for (int i = 0; i < number.length; i++) {
        if (
            }
        }
    i3 for (int i = 0; i < number.length; i++) {
        total += number[i];
    }
    System.out.printf("max: %f\n", max);
    System.out.printf("mean: %f\n", total / number.length);
}

```

**The scope of an object's fields and methods match the containing object**





## Scope: Arrays.java

```
public static void main(String[] args) {
    int[] count = new int[26];
    for (int i = 0; i < args.length; i++) {
        String argument = args[i];
        for (int j = 0; j < argument.length(); j++) {
            if (Character.isLetter(argument.charAt(j))) {
                char c = argument.charAt(j)
                count[Character.toLowerCase(c) - 'a']++;
            }
        }
    }
    for (int i = 0; i < count.length; i++) {
        if (count[i] > 0) {
            System.out.printf("%c: %d\n", 'a' + i, count[i]);
        }
    }
}
```

# Scope: Arrays.java

```
args
public static void main(String[] args) {
    int[] count = new int[26];
    for (int i = 0; i < args.length; i++) {
        String argument = args[i];
        for (int j = 0; j < argument.length(); j++) {
            if (Character.isLetter(argument.charAt(j))) {
                char c = argument.charAt(j);
                count[Character.toLowerCase(c) - 'a']++;
            }
        }
    }
    for (int i = 0; i < count.length; i++) {
        if (count[i] > 0) {
            System.out.printf("%c: %d\n", 'a' + i, count[i]);
        }
    }
}
```

# Scope: Arrays.java

```
public static void main(String[] args) {  
  int[] count = new int[26];  
  for (int i = 0; i < args.length; i++) {  
    String argument = args[i];  
    for (int j = 0; j < argument.length(); j++) {  
      if (Character.isLetter(argument.charAt(j))) {  
        char c = argument.charAt(j);  
        count[Character.toLowerCase(c) - 'a']++;  
      }  
    }  
  }  
  for (int i = 0; i < count.length; i++) {  
    if (count[i] > 0) {  
      System.out.printf("%c: %d\n", 'a' + i, count[i]);  
    }  
  }  
}
```



# Scope: Arrays.java

```
public static void main(String[] args) {  
    int[] count = new int[26];  
    for (int i = 0; i < args.length; i++) {  
        String argument = args[i];  
        for (int j = 0; j < argument.length(); j++) {  
            if (Character.isLetter(argument.charAt(j))) {  
                char c = argument.charAt(j);  
                count[Character.toLowerCase(c) - 'a']++;  
            }  
        }  
    }  
    for (int i = 0; i < count.length; i++) {  
        if (count[i] > 0) {  
            System.out.printf("%c: %d\n", 'a' + i, count[i]);  
        }  
    }  
}
```

# Scope: Arrays.java

```
public static void main(String[] args) {  
    int[] count = new int[26];  
    for (int i = 0; i < args.length; i++) {  
        String argument = args[i];  
        for (int j = 0; j < argument.length(); j++) {  
            if (Character.isLetter(argument.charAt(j))) {  
                char c = argument.charAt(j);  
                count[Character.toLowerCase(c) - 'a']++;  
            }  
        }  
    }  
    for (int i = 0; i < count.length; i++) {  
        if (count[i] > 0) {  
            System.out.printf("%c: %d\n", 'a' + i, count[i]);  
        }  
    }  
}
```

# Scope: Arrays.java

```
public static void main(String[] args) {  
    int[] count = new int[26];  
    for (int i = 0; i < args.length; i++) {  
        String argument = args[i];  
        for (int j = 0; j < argument.length(); j++) {  
            if (Character.isLetter(argument.charAt(j))) {  
                char c = argument.charAt(j);  
                count[Character.toLowerCase(c) - 'a']++;  
            }  
        }  
    }  
    for (int i = 0; i < count.length; i++) {  
        if (count[i] > 0) {  
            System.out.printf("%c: %d\n", 'a' + i, count[i]);  
        }  
    }  
}
```

# Scope: Arrays.java

```
public static void main(String[] args) {  
    int[] count = new int[26];  
    for (int i = 0; i < args.length; i++) {  
        String argument = args[i];  
        for (int j = 0; j < argument.length(); j++) {  
            if (Character.isLetter(argument.charAt(j))) {  
                char c = argument.charAt(j);  
                count[Character.toLowerCase(c) - 'a']++;  
            }  
        }  
    }  
    for (int i = 0; i < count.length; i++) {  
        if (count[i] > 0) {  
            System.out.printf("%c: %d\n", 'a' + i, count[i]);  
        }  
    }  
}
```

Please refer to the course website for reading and homework assignments.

<http://www.flyn.org/courses/cs120-2022-fall/schedule>