
Computer Science 1

Lecture 4
Loop de loop

Practical Session 3

Still easy?

- check syntax (See examples in slides or book)
- read the errors given by the compiler
- don't forget to think



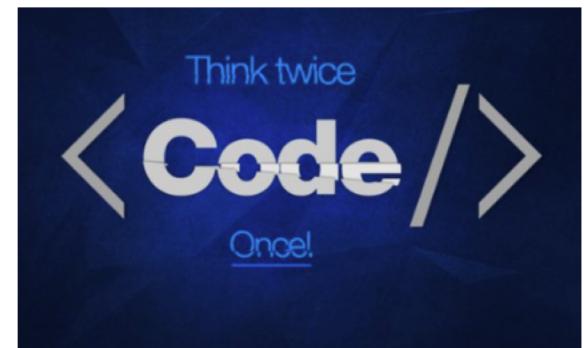
Chase

If you had one word of advice for me,
what would it be?



God

Think.



Exercise Sessions

- It is **not a good idea** to start reading the slides at the start of the exercise session.
- It is **a good idea** to finish all exercises of the previous session as preparation.

Exercise Sessions

- Process:
 1. Read the description of the task
 2. Find your inputs & outputs
 3. Find your code “building blocks”
 4. Design your solution (on paper)
 5. Start coding
 6. Compile (and pray) & check results
 7. Debug if necessary
 - Trace by hand
 - Use print() here and there
 8. Repeat steps 6-7 if necessary



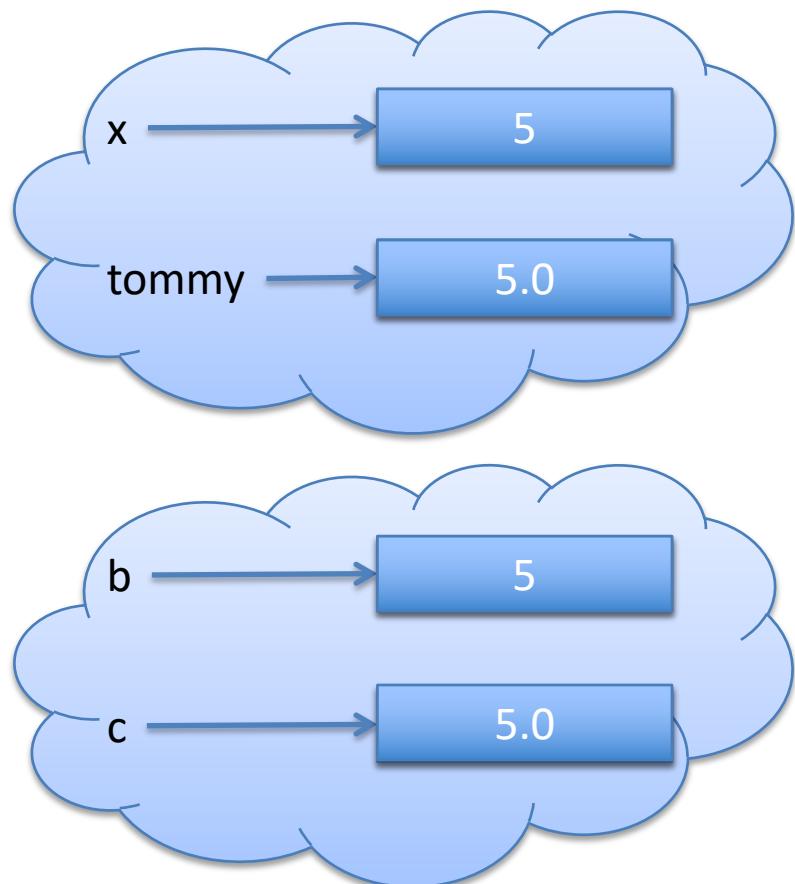
On Methods

A method that returns a Boolean and has an integer and a double as parameters.

```
public static boolean method(int number, double other) {  
    ...  
}
```

Confused about variables?

```
public static void main(String[] args) {  
    int x;  
  
    double tommy = 7.13;  
  
    x = 5;  
  
    tommy = x;  
  
    add(x,tommy);  
}  
  
public static double add(int b, double c) {  
    return b+c;  
}
```



Odd or Even

```
public static void main(String[] args) {  
    Scanner input = new Scanner(System.in);  
    // Get number from user  
    System.out.println("Type whole number:");  
    int number = input.nextInt();  
    // Determine even or not  
    if (number % 2 == 0)  
        System.out.println("even");  
    else  
        System.out.println("odd");  
}
```

Prime Number checking

```
if (num == 2 || num == 3 || num == 5 || num == 7 ||  
    num == 11 || num == 13 || num == 17 || num == 19) {  
    System.out.println(num + " is prime.");  
} else {  
    System.out.println("Number is not a prime.");  
}
```

```
if (num!=1 && num%2!=0 && num%3!=0) {  
    System.out.println(num + " is prime.");  
} else if (num==2 || num==3) {  
    System.out.println(num + " is prime.");  
} else {  
    System.out.println("Number is not a prime.");  
}
```

```
if ( number < 20 ) {
    if (number == 2)
        System.out.println("your number is prime");
    else if (number == 3)
        System.out.println("your number is prime");
    else if (number == 5)
        System.out.println("your number is prime");
    else if (number == 7)
        System.out.println("your number is prime");
    else if (number == 11)
        System.out.println("your number is prime");
    else if (number == 13)
        System.out.println("your number is prime");
    else if (number == 17)
        System.out.println("your number is prime");
    else if (number == 19)
        System.out.println("your number is prime");
} else {
    if (number < 0){
        System.out.println("Wrong number");
    }else if (number > 20){
        System.out.println("Wrong number");
    }else{
        System.out.println("your number is not prime");
    }
}
```

Rolling Dice

`Math.random()`



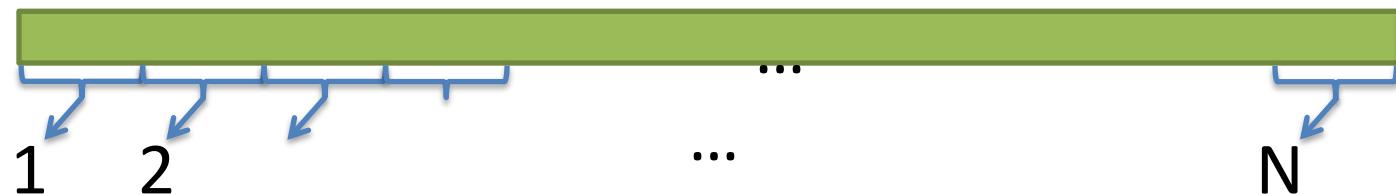
`Math.random() * N`



`Math.random() * N + 1`



`(int) (Math.random() * N + 1)`

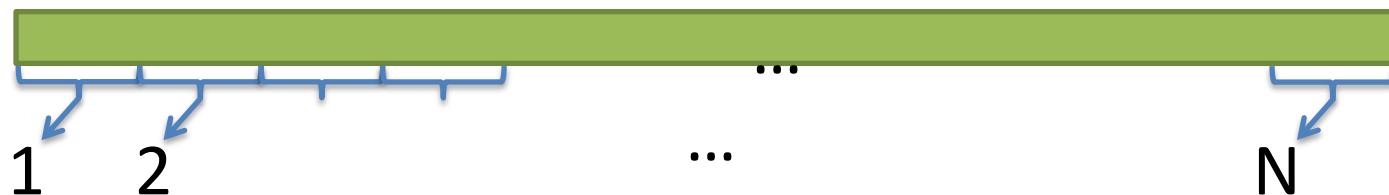


Dice Roller

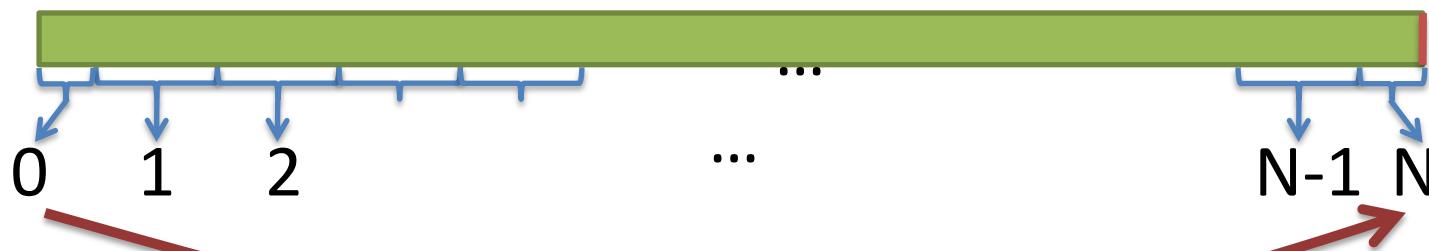
```
public static void main(String[ ] args) {  
    Scanner input = new Scanner(System.in);  
  
    System.out.println("Number of sides?");  
    int number = input.nextInt();  
  
    System.out.println((int) (Math.random()*number+1));  
}
```

Using Math.round() ?

`(int) (Math.random()*number +1)`



`Math.round(Math.random()*number)`



Dice Roller alt.

```
public static void main(String[] args) {
    Scanner input = new Scanner(System.in);

    System.out.println("Number of sides?");
    int number = input.nextInt();

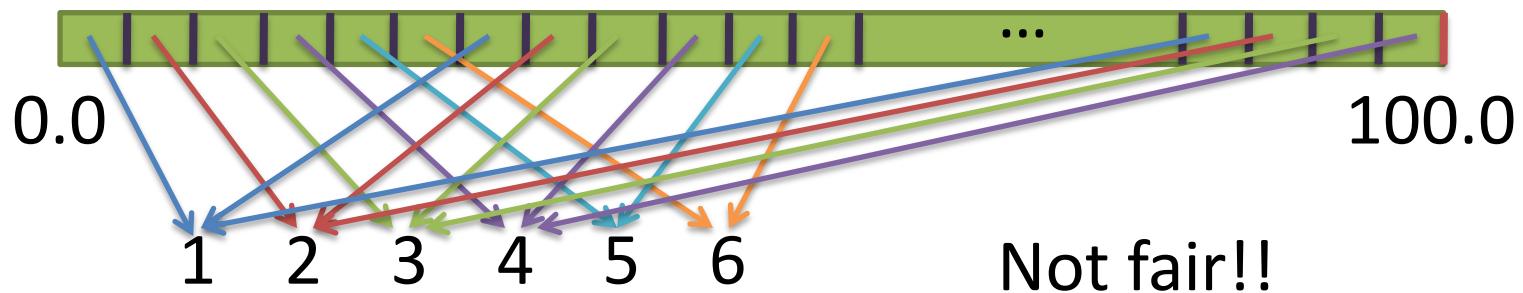
    int rolled = (int) Math.round(Math.random()*number);
    if (rolled==0)
        System.out.println(number);
    else
        System.out.println(rolled);
}
```

Returns long



Dice Roller alt. 2

```
public static void RandomNumberDice (int x) {  
    int tmp = (int) (Math.random () * 100);  
  
    if (x < 0) {  
        System.out.println ("The number is not correct.");  
    }  
    else  
        System.out.println ("The number is " + (tmp % x + 1) + ".");  
}
```



Redundant code

```
public static boolean equalStrings(String a, String b) {  
    return a.equals(b);  
}
```

Overview

Loops

- while
- do – while
- for

Loops

Repetitive task

- computers are ideal for this
- don't get tired or bored
- don't make mistakes because of that

They are fast at doing this.

Real life loops

- ...
4. Pour the milk and cream into a pan and bring just to the boil. Remove from the heat. Add the chocolate and **beat until it is melted and smooth with no lumps.**
 5. Gradually stir hot chocolate mix into paste. Return to pan. Cook, stirring, over a medium-low heat for 5 mins to a smooth, thick paste. Remove from the heat. **Leave until cold, beating occasionally with a wire whisk.**
- ...

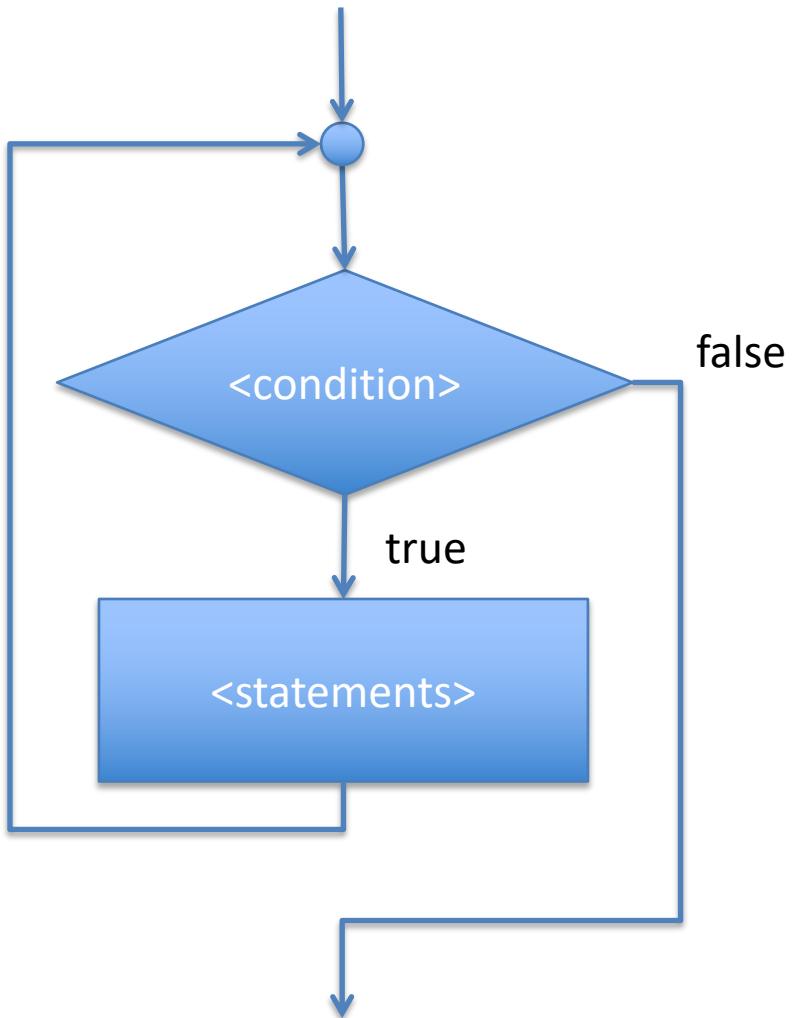
Graduate in 3 (or 5) years

Walking (or traveling in general)

Swimming (running) 30 laps

...

WHILE



```
while (<condition>) {  
    <statements>  
}
```

While example (0)

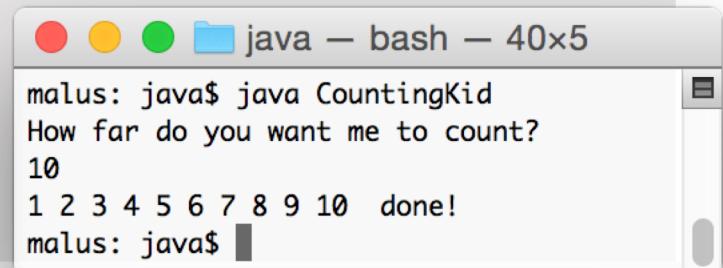
```
import java.util.Scanner;

public class CountingKid {

    public static void main(String[] args){
        Scanner in = new Scanner(System.in);

        System.out.println("How far do you want me to count?");
        int limit = in.nextInt();

        int count = 1;
        while (count <= limit) {
            System.out.print(count + " ");
            count++;
        }
        System.out.println(" done!");
    }
}
```



A terminal window titled "java – bash – 40x5" showing the execution of the Java program. The user types "How far do you want me to count?", followed by "10", and then the program outputs the numbers from 1 to 10 followed by "done!". The terminal window has a standard OS X-style title bar and scroll bars.

```
malus: java$ java CountingKid
How far do you want me to count?
10
1 2 3 4 5 6 7 8 9 10  done!
malus: java$
```

While example (1)

```
import java.util.Scanner;

public class LineReader {

    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);

        System.out.print("Enter a sequence of numbers,");
        System.out.println("end with a letter.");

        int cntr = 0;
        while (input.hasNextDouble()) {
            cntr++;
            System.out.print("Number " + cntr);
            System.out.println(" was: " + input.nextDouble());
        }
    }
}
```

```
wopr: java$ java LineReader
Enter a sequence of numbers,end with a letter.
1 2 3 4 5 6 7 q
Number 1 was: 1.0
Number 2 was: 2.0
Number 3 was: 3.0
Number 4 was: 4.0
Number 5 was: 5.0
Number 6 was: 6.0
Number 7 was: 7.0
wopr: java$ █
```

While example (2)

```
public static void main(String[] args) {
    Scanner input = new Scanner(System.in);

    printMenu();
    int choice = input.nextInt();

    while (choice != 0) {
        if (choice != 1 &&
            choice != 2 &&
            choice != 3)
            System.out.println("Please select option from menu only");
        else
            System.out.println("You have selected option " + choice);

        printMenu();
        choice = input.nextInt();
    }
}
```

```
public static void printMenu() {
    System.out.println("Type 1 for option 1");
    System.out.println("Type 2 for option 2");
    System.out.println("Type 3 for option 3");
    System.out.println("Type 0 to quit");
}
```

While example (2a)

```
public static void main(String[] args) {
    Scanner input = new Scanner(System.in);

    int choice = doMenuSelection(input);

    while (choice != 0) {
        if (choice != 1 &&
            choice != 2 &&
            choice !=3)
            System.out.println("Please select option from menu only");
        else
            System.out.println("You have selected option " + choice);

        choice = doMenuSelection(input);
    }
}

public static int doMenuSelection(Scanner in) {
    System.out.println("Type 1 for option 1");
    System.out.println("Type 2 for option 2");
    System.out.println("Type 3 for option 3");
    System.out.println("Type 0 to quit");

    return in.nextInt();
}
```

Frequent mistakes

1. Infinite loops

```
while (true) {  
    System.out.println("Wheeee!");  
}
```

```
int cnt = 0;  
while (cnt != 9)  
    cnt +=2;
```

```
int cnt = 100;  
while (cnt > 0)  
    cnt++;
```

can be of use, but not often in this course
(use ctrl-c to put it out of its misery)

What You Need: A gentle, sulfate-free shampoo with the right level of conditioning for hair to move freely and fall naturally in place.

What You Get: Amazingly light and lively, clean hair that's manageable and moves naturally. Pert Plus has clean conditioning that leaves no unwanted residue – rinsing cleanly away without build-up. And because it's pH balanced, it's perfect for everyday use, even on permed or color-treated hair.

Go beyond clean and bring back lively hair.

Visit us at www.pertplus.com

DIRECTIONS: Gently massage into wet hair, lather, rinse, repeat

INGREDIENTS: Water, Ammonium Laureth Sulfate, Ammonium Lauryl Sulfate, Glycol Distearate, Sodium Chloride, Cocamide MEA, Dimethicone, Cetyl Alcohol, Fragrance, Sodium Citrate, Polymethacrylamidopropyl Trimonium Chloride, Sodium Benzoate, PEG 14M, Dihydrogenated Tallowamidoethyl Hydroxyethylmonium Methosulfate, Disodium EDTA, Citric Acid, Ammonium Xylenesulfonate, Methylchloroisothiazolinone, Methylisothiazolinone, Yellow 10, Blue 1.



Questions?

1-800-436-7378



Made in USA of US and imported ingredients./Hecho en EE.UU. con ingredientes de EE.UU. e importados.

Distr. by/por

Procter & Gamble,
Cincinnati, OH 45202.

©2002

US Patent RE 34,584 ano
5,756,436.

95051964

2

1

0

9

0

8

0

7

0

0

0

3

7

0

0

0

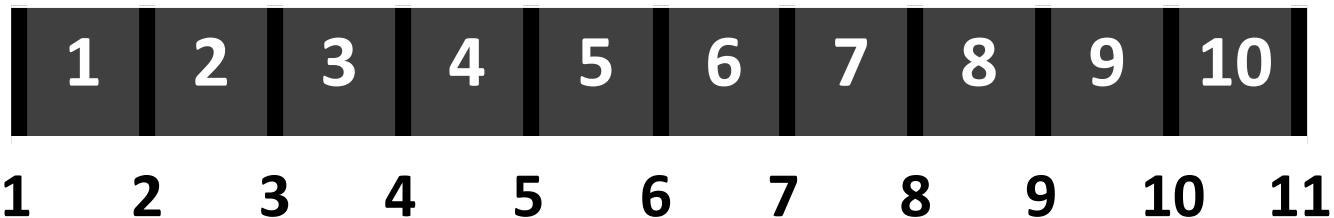
Frequent Mistakes (cont.)

2. Off-by-one

Boundary conditions

- initialization
- stopping condition

If you build a straight fence 100m long with posts 10m apart, how many posts do you need?

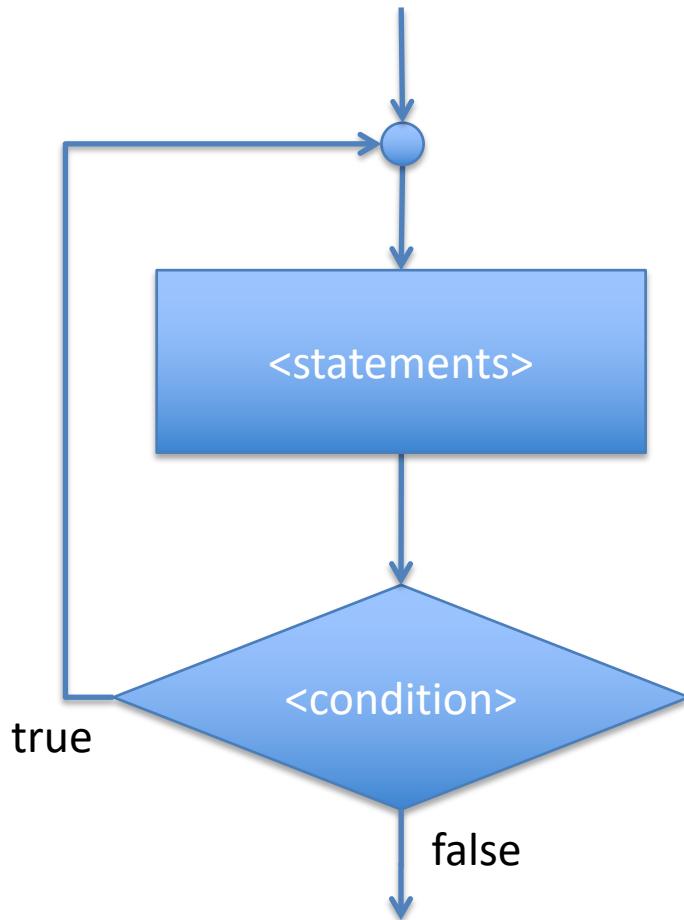


Running a loop at least once

- while checks condition before starting loop
- What if you want to run the loop at least once?

```
boolean firsttime = true;  
  
while (firsttime || <condition>) {  
  
    firsttime = false;  
  
    <other statements>  
  
}
```

DO – WHILE



```
do {  
    <statements>  
} while (<condition>)
```

Frequent loop form

```
public static void main(String[] args) {  
  
    int cntr = 0;  
    int loopcount = ...;  
  
    while (cntr < loopcount) {  
        doSomething();  
        cntr++;  
    }  
}
```

Fixed number of iterations

- cntr is often used in loop for reference

FOR

Syntactic sugar!

```
for (int i = 0; i < loopcount; i++) {  
    doSomething();  
}
```

In general:

```
for (<init> ; <condition> ; <update>) {  
    <statements>  
}
```

For example

```
for (int i = 0; i <= 100; i+=2) {  
    System.out.print(i + " ");  
}
```

```
int n = ...;  
int result = 1;  
  
for (int i = 1; i <= n; i++) {  
    result *= i;  
}
```

Bad form example

Use for-loops to emphasize structure!

```
int choice = 0;  
for (boolean stop = false;  
    !stop;  
    stop = ((choice = int.nextInt()) == 0)) {  
    System.out.println(choice);  
}
```

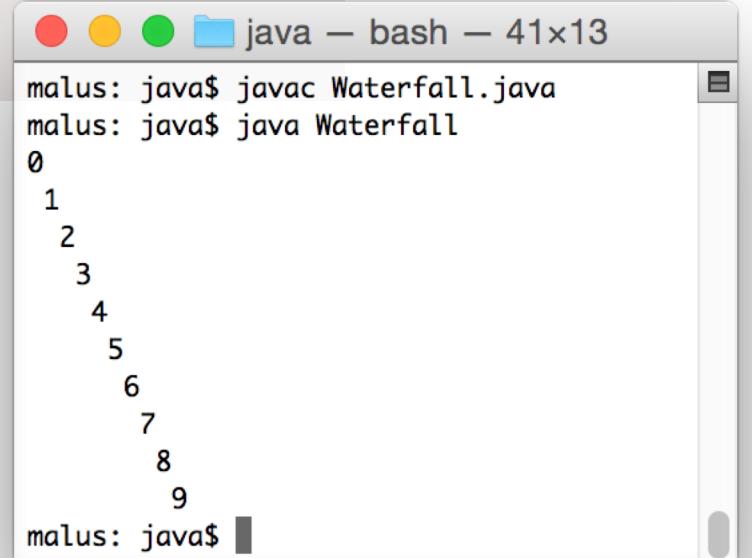
Nested loops

Just like if-statements, loops can be nested:

```
for (int i = 1; i <= 10; i++) {  
    int result = 1;  
  
    for (int j = 1; j <= i; j++)  
        result *= j;  
  
    System.out.println(result);  
}
```

Nested Loops Example

```
for (int i=0; i<10; i++) {  
  
    for (int j=0; j<i; j++)  
        System.out.print(" ");  
  
    System.out.println(i);  
  
}
```

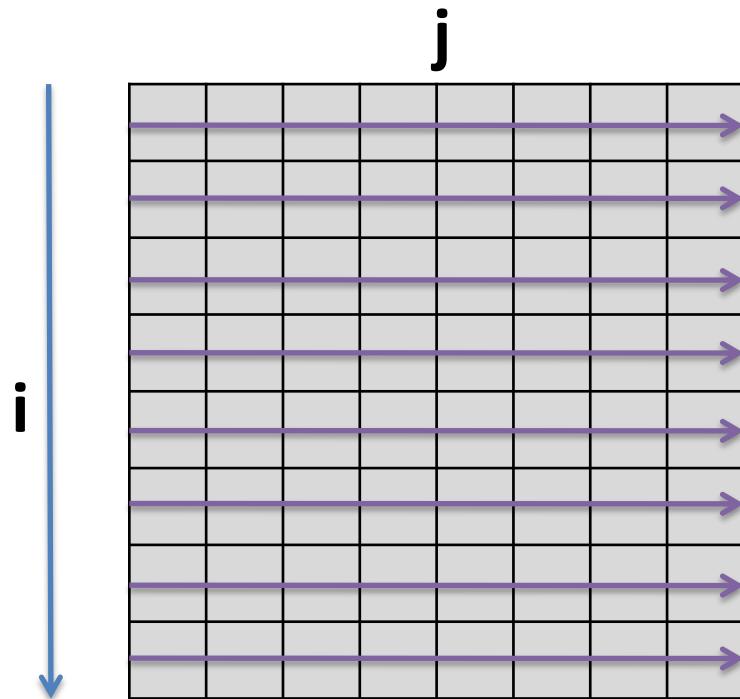


A screenshot of a terminal window titled "java – bash – 41x13". The window shows the following text:
malus: java\$ javac Waterfall.java
malus: java\$ java Waterfall
0
1
2
3
4
5
6
7
8
9
malus: java\$ █

The terminal window has a light gray background and a dark gray border. The title bar at the top is labeled "java – bash – 41x13". The main area of the terminal shows the command "javac Waterfall.java" being run, followed by the output of the program itself, which prints the numbers 0 through 9 each on a new line. The prompt "malus: java\$" appears at the bottom right.

Nested Loops Examples

Matrices!



Matrices?



Summary

Loops

- While
- Do-while
- For

Coming up next

This week:

- Wednesday: Practical 4 (loops)
- Friday: Arrays (with Kurt)

Next week:

- Monday & Tuesday: 2 practicals on arrays
- Wednesday: Recursion (with Jerry)

Example 1

```
public static void main() {  
    double bubble = 867.5309;  
    double x = 10.01;  
    printer(double x, double y);  
    printer(x);  
    printer("barack", "obama");  
    System.out.println("z = " + z);  
}  
  
public static void printer(x, y double) {  
    int z = 5;  
    System.out.println("x = " + double x + " and y = " + y);  
    System.out.println("The value from main is: " + bubble);  
}
```

Example 2

```
int number = 32;

for (int count = 1; count <= number; count++) {
    System.out.println(number);
    number = number / 2;
}
```

Example 3

Write a for loop that produces the song *Bottles of Beer on the Wall*:

10 bottles of beer on the wall, 10 bottles of beer

Take one down, pass it around, 9 bottles of beer on the wall

9 bottles of beer on the wall, 9 bottles of beer

Take one down, pass it around, 8 bottles of beer on the wall

...

...

1 bottles of beer on the wall, 1 bottles of beer

take one down, pass it around, 0 bottles of beer on the wall

Example 4

Can you write a validation loop that will prompt the user to give a number between 0 and 12 (included)? That means that if the user gives a number outside this range, then the loop should keep asking the user for a new number.