Java

Pink Programming | GameCamp

Java

What is Java?

- **Programming language**
- **One of the most popular programming languages**
- **Object oriented**
  - **Objects = classes**
- **Strong type language**
  - Variables need to be defined before used

Java - Introduction

Today's goal

- Get familiar with...
  - Variables
  - Scanner
  - If-else statement

Java - Introduction

“Hello world” program

```
public class Example {

    public static void main(String[] args){
        System.out.println("Hello world!");
    }
}
```

Java - Introduction

Getting familiar with the code

- `public class Example {`
  - Defines (declares) a class named Example
    - Every line of code that can run needs to be inside a class
  - Public - any other classes can access it
  - Not important yet

Java - Introduction

Getting familiar with the code

- `public static void main(String[] args){`
  - Entry point of our Java program
  - Main method
  - Has to have the **exact** signature
    - **public** - everyone can access it
    - **static** - able to run this method without creating an *instance* of **Example**
    - **void** - this method doesn't return any value
    - **main** - name of the method
    - **(String[] args)** - arguments we will get when running the program with parameters
    - No need to understand it all yet

Java - Introduction

Getting familiar with the code

- `System.out.println("Hello world!");`

  For you who are curious:

  - **System** - pre-defined class that Java provides. Holds some useful methods and variables
  - **out** - static varable within System, represents the output of you program
  - **println** - method that can be used to print a line

- What's important to know about this method
  - Output: Text inside ()
  - Important to end with **semicolon**

— `break;`

Variables and types

- **Java - strong typed language**
  - variables need to be declared before we use them (end with semicolon!)
    - variable - place where we store data
  - Example of common types
    - int
    - double
    - char
    - Boolean

Integer

Declaring a variable

- Different ways of doing it

```
int myNumber;
myNumber = 5;
```

```
int myNumber = 5;
```

(Do not forget the semicolon!)

Integer

Example

- What value will each variable have after each line has executed?

```
int nbrA = 2;
// 2
int nbrB = nbrA + 3;
// 5
int nbrC = 2 * (nbrA + nbrB) + 1;
// 15
nbrA = nbrC / 3;
// 5
nbrC = 0;
// 0
```

Double

Declaring a variable

- Different ways of doing it

```
double myNumber;
myNumber = 5.0;
```

```
double myNumber = 5.0;
```

Character

Declaring a variable

- Character is its own type, simply not a number (but stored as a number)

```
char myChar;
myChar = 'a';
```

```
char myChar = 'a';
```

String

Declaring a type

- Different ways of declaring a **type**.
- Examples

```
String s1 = new String("Programming is fun!");
//Creates a string with a constructor
String s2 = "Look what I have just created!";
//Just using "" creates a string
String s3 = s1 + s2;
// Java defined the operator "+" on strings to
concatenate
```

Concatenating

Possible to combine String and previous variables

- Example

```
int num = 5;
String s = "I have " + num + " cookies hidden somewhere.";
// What will happen if we put "+ num +" inside the whole sentence
with ""?
```

Try it out!

```
System.out.println(s);
```

Boolean

A type accepting only two values

- true or false
- Example:

```
boolean value = true;
if (value) {
     System.out.println("Pink Programming!");
}
```

Boolean

A type accepting only two values

- Example 2

```
int a = 4;
int b = 5;
boolean result;
result = a < b; //true
result = a >= b; // a larger than or equal to b - false
result = a == b; // a equal to b - false
```

Comparing

Examples of comparison operators

- ==, &&, ||

- Example:

```
int a = 2;
int b = 5;
if (a == 2 && b != 2) {
     System.out.println("This is correct!");
}
if (a == 2 || b != 2) {
     System.out.println("This is correct!");
}
```

If-else statement

Write...

```
Scanner scan = new Scanner(System.in);
double a = scan.nextDouble();
double b = scan.nextDouble();
if (a < b) {
     System.out.println("a is less than b");
} else {
     System.out.println("a is bigger than or equal to b");
}
```