T5 - Java Seminar

T-JAV-500

Day 01

Hello World

3.0

{EPITECH.}

Day 01

language: Java

- The totality of your source files, except all useless files (binary, temp files, obj files,...), must be included in your delivery.

To run this pool, you need Java to be installed on your computer.
We are here talking about **JSE**, that you could easily find on Internet.

In Java, every line of code must be included into a class.
There must also be one file by class (and one class by file) named the same way.
Here is a very first example of Java code (necessary in a file name "HelloWorld.java"):

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

To compile it:

```
javac HelloWorld.java
```

and to execute it:

```
java HelloWorld
```

Make sure this code snippet displays "Hello World!" when you execute it.

Have you seen the "main" function? Why is it special?

General instructions

For all following exercises, your functions must be in a class called like the file to turn-in.
For instance if the file to turn-in is `Ex00.java` then the class will be as follow:

```
class Ex00 {
	public static void myMethod() {
	}
}
```

Exercise 01

File to hand in: ./ex_01/Ex01.java
Restrictions: "System.out.print" is mandatory, and must be used only once.

Create a function named "myConcat" that takes two parameters.
The function must display the first parameter followed by a space followed by the second parameter.

Prototype

```
public static void myConcat(String str1, String str2);
```

Example

```
Ex01.myConcat("Hello", "world");    // should display: "Hello world"
```

Exercise 02

File to hand in: ./ex_02/Ex02.java
Restrictions: the "for" keyword is mandatory.

Write a function that returns a string composed of as many "woof" as the value of the variable passed as parameter.

Prototype

```
public static String getAngryDog(int nbr);
```

Example

```
Ex02.getAngryDog(3);      // returns "woofwoofwoof"
```

EXERCISE 03

File to hand in: ./ex_03/Ex03.java
Restrictions: the “for” keyword is mandatory.

Write a function that displays the values of the array passed as parameter.

Each value must be followed by a new line.

Prototype

```
public static void printArray(ArrayList<String> myArray);
```

EXERCISE 04

File to hand in: ./ex_04/Ex04.java
Restrictions: the “switch” keyword is mandatory.

Write a function that takes an integer as parameter.
If the value is 3, it displays “The Three Brothers”.
If the value is 6, it displays “The Sixth Sense”.
For 23, it displays “The Number 23”.
And for 28, it displays “28 Days Later”.
For other values, it displays “I don’t know.”.

A new line will be called after each display.

Prototype

```
public static void printMovieFromNbr(int nbr);
```

Example

```
Ex04.printMovieFromNbr(23); // Should display "The Number 23"
```

EXERCISE 05

File to hand in: ./ex_05/Ex05.java

Write a function that takes as parameter a variable number of arguments and returns these arguments in an array.

Prototype

```
public static ArrayList<String> myGetArgs(String... var);
```

EXERCISE 06

File to hand in ./ex_06/Ex06.java

Write a function that outputs the following sequence to the n^{th} iteration.
Your function should not print anything if you pass it a negative number.

Here is the beginning of the sequence:

```
1
11
21
1211
111221
312211
```

The first iteration is 0.

Prototype

```
public static void sequence(int nbr);
```

Example

```
sequence(0);    // should output 1
```