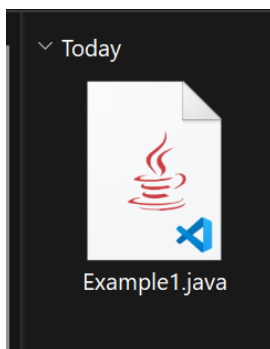# First java lab

Lab Exercises

lab1:
Create a simple application that performs the following actions:
Display a message to the command prompt.

▶ Create simple application that performs the following actions:

　　▶ Display a message to the command prompt.

Open notepad write this:

```
class Example1 {
        public static void main(String args[]){
                System.out.println("Hello Java");
        }
}
```


Make the file name :
public class name.java
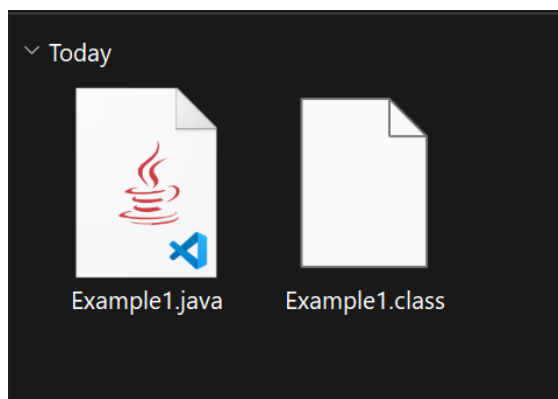
Then on CMD :
Compile the file :
Javac Example1.java
→ the byte code (intermediate code) appears Example1.class (we will
have number of intermediate code file depend on the number of classes)

Run the file :
Java Example1
→ intermediate code file which contain the main function

Lab2:

Create a simple application that performs the following actions: Receives an input (as main arguments) and checks for its value and prints it back.

▶ Create simple application that performs the following actions:

　　▶ Receives an input (as main arguments) and checks for its value and prints it back.

```
C:\>java Example2 CoreJava
C:\> true value CoreJava
```

Same steps

```java
public class Example2{
        public static void main(String args[]){
                // String literal
                //String targetValue = "CoreJava";

                // Using new Keyword
                String targetValue = new String("CoreJava");

                if( args.length != 0 && targetValue.equals(args[0]))
                {
                        System.out.println("true value "+args[0]);
                }
                else{
                        System.out.println("false value");

                }
        }
}
```

Notes :
In Java, the <u>equals()</u> method and the <u>== operator</u> are used to compare objects.

- The main difference is that the string equals() method compares the ***content equality of two strings***

- while the == operator compares the *reference or memory location* **of objects in a heap**, whether they point to the same location or not.
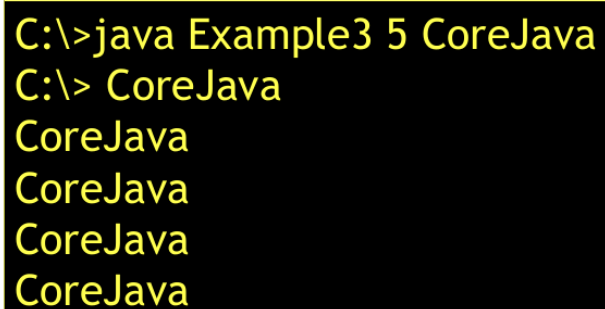
```
C:\Users\nadam\Downloads\open source\Java\day1\lab\lab1.2>javac Example2.java

C:\Users\nadam\Downloads\open source\Java\day1\lab\lab1.2>java Example2
false value
```

```
C:\Users\nadam\Downloads\open source\Java\day1\lab\lab1.2>java Example2 CoreJav
false value

C:\Users\nadam\Downloads\open source\Java\day1\lab\lab1.2>java Example2 CoreJava
true value CoreJava
```

Lab3 :

**Create a simple application that performs the following actions:
Receives two inputs (as main arguments) a number and a string
and prints the string on different lines according to the given
number.**

► Create simple application that performs the following actions:

  ► Receives two inputs (as main arguments) a number and a string and prints the string on
    different lines according to the given number.

```
C:\>java Example3 5 CoreJava
C:\> CoreJava
CoreJava
CoreJava
CoreJava
CoreJava
```

```java
public class Example3{
      public static void main(String args[]){
            try{
                  if(args.length != 0){
                        for(int i = 0 ; i < Integer.parseInt(args[0]) ; i++)
                        {
                              System.out.println(args[1]);
                        }
                  }
            }
            catch(Exception e)
            {
                  System.out.println("can't convert string "+ args[0] +" to
int");
            }
      }
}
```

Note :

Args[] is a string array so we need to convert the first argument to integer

```
C:\Users\nadam\Downloads\open source\Java\day1\lab\lab1.3>javac Example3.java

C:\Users\nadam\Downloads\open source\Java\day1\lab\lab1.3>java Example3 1 CoreJava
CoreJava

C:\Users\nadam\Downloads\open source\Java\day1\lab\lab1.3>java Example3 5 CoreJava
CoreJava
CoreJava
CoreJava
CoreJava
CoreJava

C:\Users\nadam\Downloads\open source\Java\day1\lab\lab1.3>java Example3 k CoreJava
can't convert string k to int

C:\Users\nadam\Downloads\open source\Java\day1\lab\lab1.3>
```