

Fondamenti di Informatica (117IN)

A.A. 2022 / 2023

Lezione 5 - Input e Oggetti

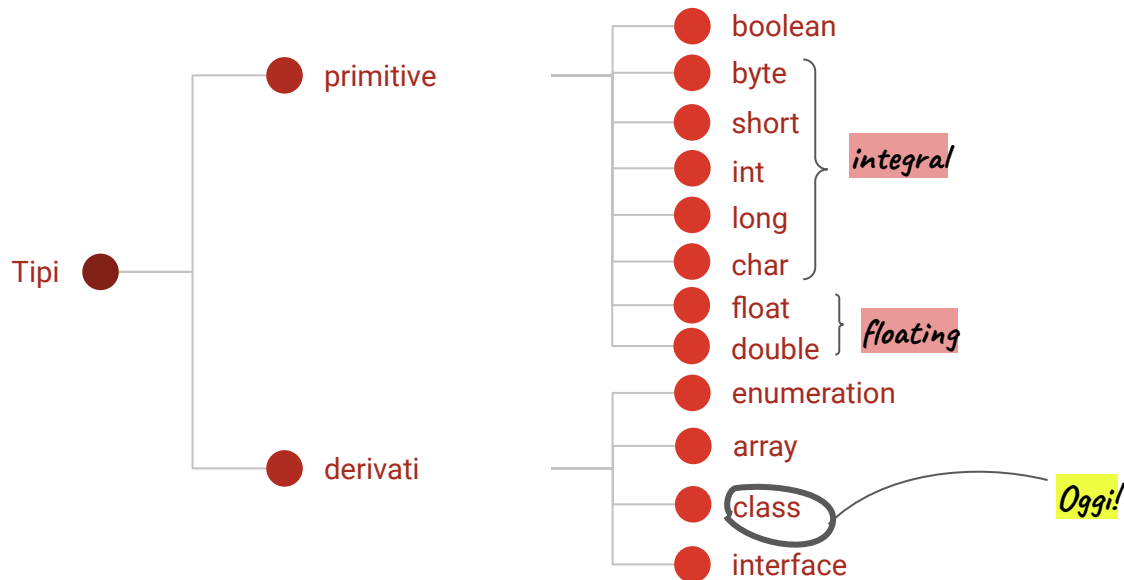
Sylvio Barbon Junior
sylvio.barbonjunior@units.it



Sommario:

- 1) Dati
- 2) Struttura di un programma
- 3) Classi String, Integer, Console e Scanner.
- 4) Classi e Oggetti
- 5) Input in Java

1) Dati



2) Struttura di un programma

Classe principale (o interfacce)

funzione "main" (eseguita per prima)

Commenti

```
Programma.java x
1 // file Programma.java
2 public class Programma
3     { public static void main(String[] args){
4         Commenti //definizioni e istruzioni
5     }
6 }
```

3) Classe String

- Una **stringa** è una sequenza di **caratteri**, che può anche non avere alcun carattere (stringa vuota).
- Una variabile appartenente a questo tipo, detta variabile stringa, rappresenta il riferimento di un oggetto stringa che memorizza una stringa costante.

```
ClassString.java x
1 public class ClasseString{
2     public static void main(String args[]){
3         String name = new String("My name");
4     }
5 }
```

Dichiarazione della Classe *Metodo Costruttore* *parametro*

variabile che fa riferimento all oggetto

3) Classe String

<https://docs.oracle.com/javase/7/docs/api/java/lang/String.html>

documentazione ufficiale.

java.lang

Class String

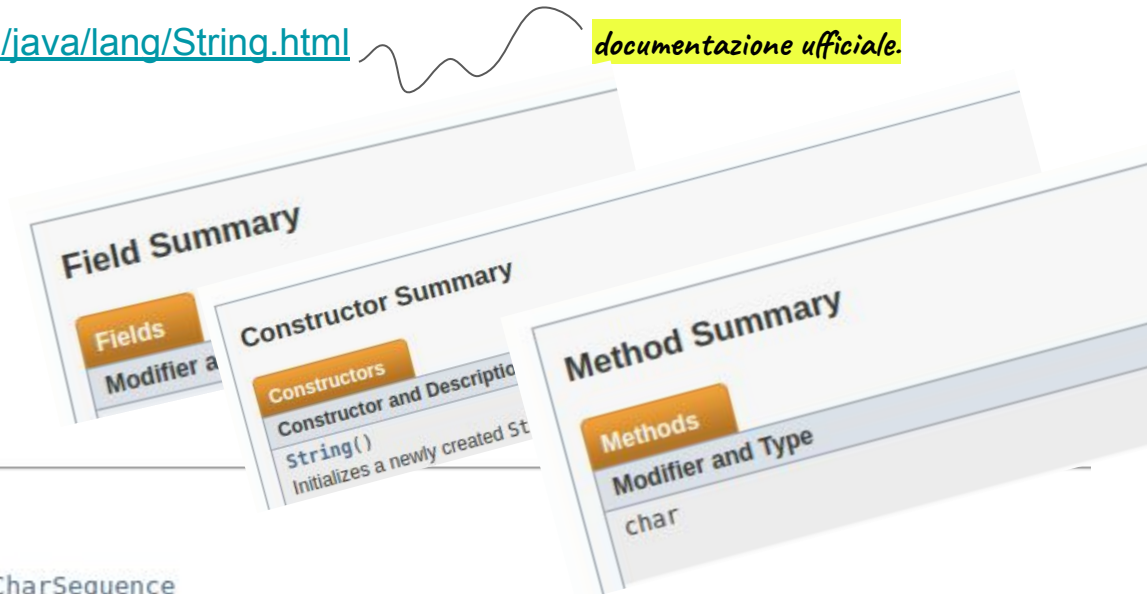
java.lang.Object
java.lang.String

All Implemented Interfaces:

Serializable, CharSequence, Comparable<String>

```
public final class String
extends Object
implements Serializable, Comparable<String>, CharSequence
```

The String class represents character strings. All string literals in Java programs, such as "abc", are implemented as instances of this class.



3) Classe Integer

- Classe **Integer** consente di creare un oggetto classe a partire da un valore del tipo primitivo **int**, è un wrapper!
- Le classi **wrapper** sono predefinite nel linguaggio e servono a racchiudere in oggetti i valori di un tipo primitivo

1. Boolean
2. Byte
3. Character
4. Integer
5. Long
6. Float
7. Double

```
ClasseInteger.java x
1 public class ClasseInteger{
2     public static void main(String args[]){
3         Integer numero = Integer.parseInt("300");
4         System.out.println(numero*2);
5     }
6 }
```

<https://docs.oracle.com/javase/7/docs/api/java/lang/Integer.html>

documentazione ufficiale.

3) Classe Console

← → ↻ <https://docs.oracle.com/javase/7/docs/api/java/io/Console.html>

Classe **Console** c'è la funzioni di effettuare **letture da tastiera e scritture su video** di valori appartenenti ai principali tipi primitivi (tipi boolean, char, int, double) e al tipo stringa (String).

Methods

Modifier and Type	Method and Description
void	flush() Flushes the console and forces any buffered output to be written immediately .
Console	format(String fmt, Object... args) Writes a formatted string to this console's output stream using the specified format string and arguments.
Console	printf(String format, Object... args) A convenience method to write a formatted string to this console's output stream using the specified format string and arguments.
Reader	reader() Retrieves the unique Reader object associated with this console.
String	readLine() Reads a single line of text from the console.
String	readLine(String fmt, Object... args) Provides a formatted prompt, then reads a single line of text from the console.
char[]	readPassword() Reads a password or passphrase from the console with echoing disabled
char[]	readPassword(String fmt, Object... args) Provides a formatted prompt, then reads a password or passphrase from the console with echoing disabled.
PrintWriter	writer() Retrieves the unique PrintWriter object associated with this console.

3) Classe Console

I programmi Java possono utilizzare classi facenti parte di un *package*, tipicamente dei package che costituiscono le Java API, attraverso la loro **importazione**.

```
Programma.java
1  import java.io.Console;
2
3  // file Programma.java
4  public class Programma{
5      public static void main(String[] args){
6          Console console = System.console();
7          String input = console.readLine();
8          int pNum = Integer.parseInt(input);
9          System.out.println("Il primo numero è: " + pNum);
10
11         input = console.readLine();
12         int sNum = Integer.parseInt(input);
13         System.out.println("Il secondo numero è: " + sNum);
14
15         System.out.println("La somma è: " + (pNum+sNum));
16     }
17 }
```

Un **commento** può essere costituito da una sequenza di caratteri racchiusa tra `'*'` e `'/'` (commento tradizionale), oppure tra `//` e un terminatore di linea (commento di fine linea):

```
sylvio@ws-piccola: ~/Scaricati/Fi/
sylvio@ws-piccola:~/Scaricati/Fi/Codi
256
Il primo numero è: 256
128
Il primo numero è: 128
La somma è: 384
sylvio@ws-piccola:~/Scaricati/Fi/Codi
```

3) Classe Scanner

La **lettura formattata** di valori di un tipo primitivo (escluso il tipo byte) o di tipo stringa si effettuano comunemente utilizzando stream appartenenti alla classe **Scanner**



User



- Metodi **convenienti** per analizzare le **primitive** (`nextInt()`, `nextFloat()`, ...)
- I metodi di lettura **non sono sincronizzati**

- Lettura della **password** senza fare eco ai caratteri inseriti.
- I metodi di lettura **sono sincronizzati**.
- È possibile utilizzare la **sintassi della stringa di formato**.
- Non funziona in un ambiente non interattivo (come in un **IDE**).



3) Classe Scanner

importazione delle librerie

scopo della classe

dichiarazioni delle obietti:

1. con
2. sc

oggetto con, scrivendo nel console

creando un oggetto dinamico

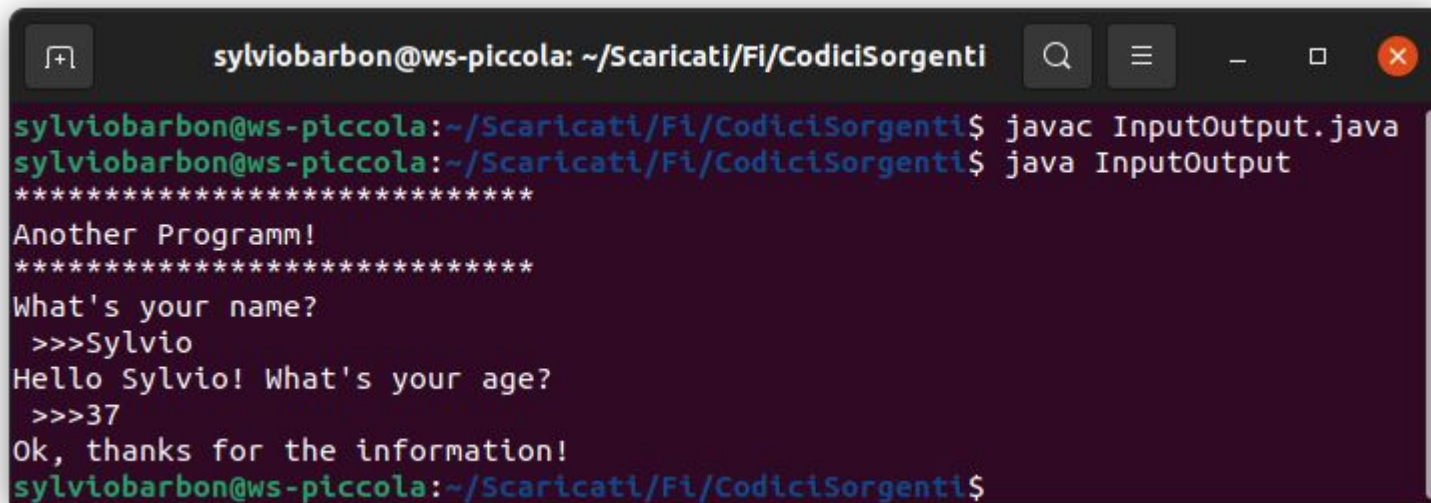
```
InputOutput.java x
1 import java.io.Console;
2 import java.util.Scanner;
3
4 public class InputOutput{
5     public static void main (String args[]){
6         Console con = System.console();
7         Scanner sc = new Scanner(con.reader());
8
9         con.printf("*****\n");
10        con.printf("Another Programm! \n");
11        con.printf("*****\n");
12
13        con.printf("What's your name? \n >>>");
14        String name = sc.nextLine();
15
16        con.printf("Hello "+name+"! What's your age?\n >>>");
17        int age = Integer.parseInt(sc.nextLine());
18
19        con.printf("Ok, thanks for the information!\n");
20    }
21 }
```

scopo dello metodo main

metodo costruttore e Scanner per leggere "con" (Console)

oggetto sc leggendo la riga dello console

3) Classe Scanner



```
sylviobarbon@ws-piccola: ~/Scaricati/Fi/CodiciSorgenti
sylviobarbon@ws-piccola:~/Scaricati/Fi/CodiciSorgenti$ javac InputOutput.java
sylviobarbon@ws-piccola:~/Scaricati/Fi/CodiciSorgenti$ java InputOutput
*****
Another Programm!
*****
What's your name?
>>>Sylvio
Hello Sylvio! What's your age?
>>>37
Ok, thanks for the information!
sylviobarbon@ws-piccola:~/Scaricati/Fi/CodiciSorgenti$
```

3) Classe Scanner

creando un oggetto dinamico

importazione delle librerie

scopo della classe

dichiarazioni delle obietti:

1. sc

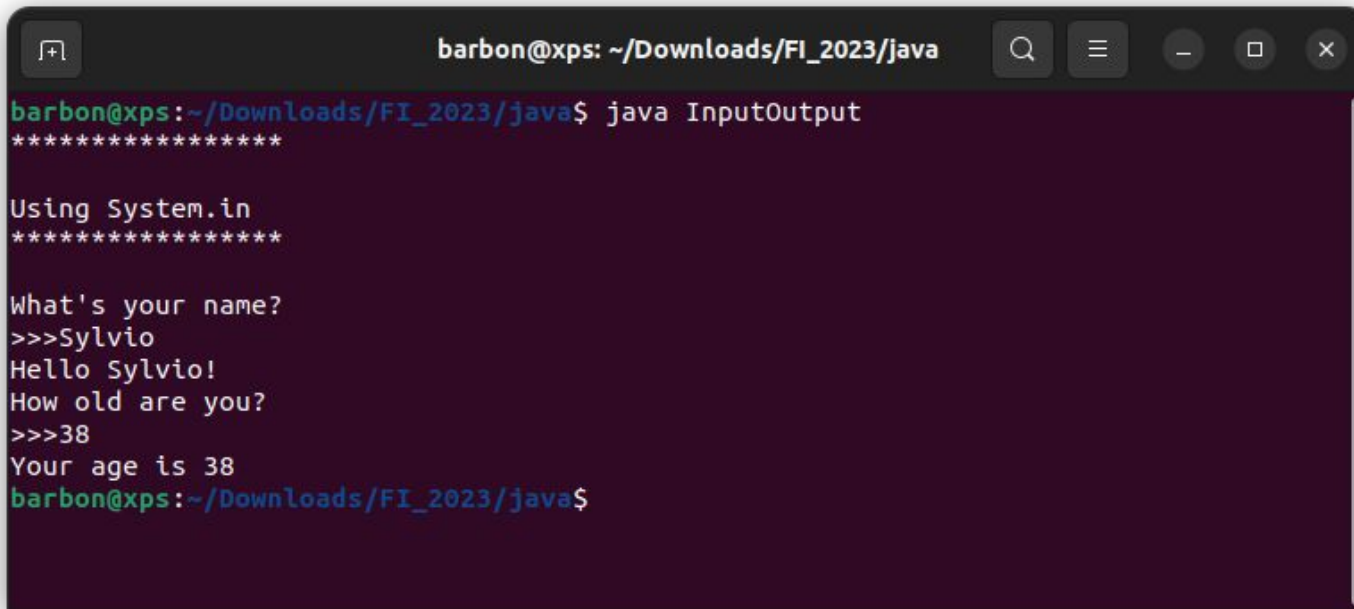
```
1 import java.util.Scanner;
2 public class InputOutput{
3     public static void main(String[] args) {
4         Scanner sc = new Scanner(System.in);
5
6         System.out.println("***** \n");
7         System.out.println("Using System.in");
8         System.out.println("***** \n");
9
10        System.out.print("What's your name? \n>>>");
11        String name = sc.nextLine();
12
13        System.out.println("Hello "+name+"!");
14
15        System.out.print("How old are you?\n>>>");
16        int age = sc.nextInt();
17
18        System.out.println("Your age is "+age+".");
19    }
20 }
```

scopo del metodo main

metodo costruttore e Scanner per leggere "con" System.in

oggetto sc leggendo come int

3) Classe Scanner



```
barbon@xps: ~/Downloads/FI_2023/java
barbon@xps:~/Downloads/FI_2023/java$ java InputOutput
*****

Using System.in
*****

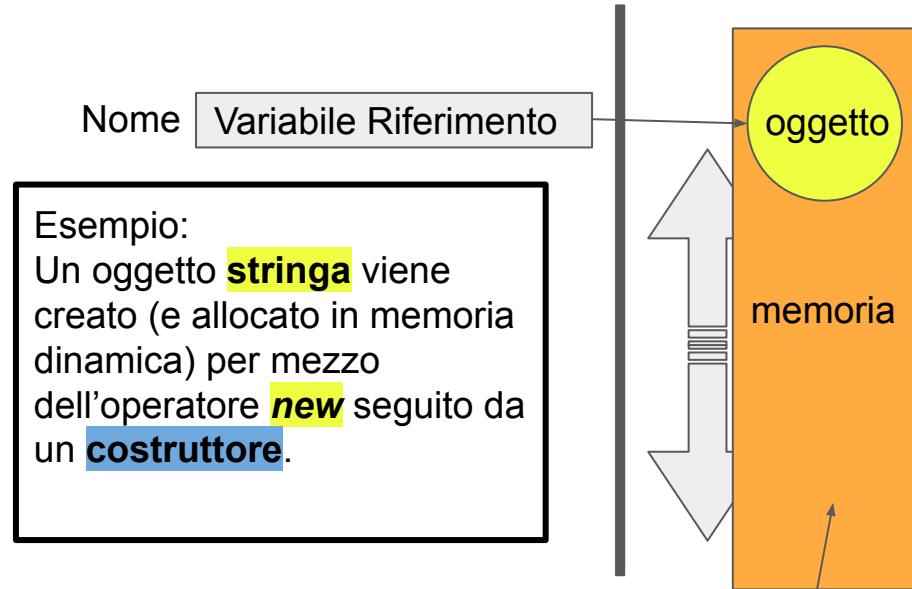
What's your name?
>>>Sylvio
Hello Sylvio!
How old are you?
>>>38
Your age is 38
barbon@xps:~/Downloads/FI_2023/java$
```


4) Classi e Oggetti

- I **tipi primitivi** prevedono la definizione di **variabili** di un dato tipo.
- Una variabile di un tipo primitivo ha un nome e un valore;
- Il **tempo di vita** di una tale **variabile dipende dal luogo in cui è definita** (per esempio, le variabili definite nel corpo di una funzione sono automatiche);

Classe -> La definizione

Oggetto -> Spazio de
memoria allocato seguendo
"La definizione"



Il recupero della **memoria dinamica** non più utilizzata viene effettuato dalla piattaforma **Java** per mezzo di una specifica routine (**Garbage Collector**), che periodicamente libera quelle zone di memoria occupate da oggetti per i quali non esiste più alcuna variabile classe che li riferisce.



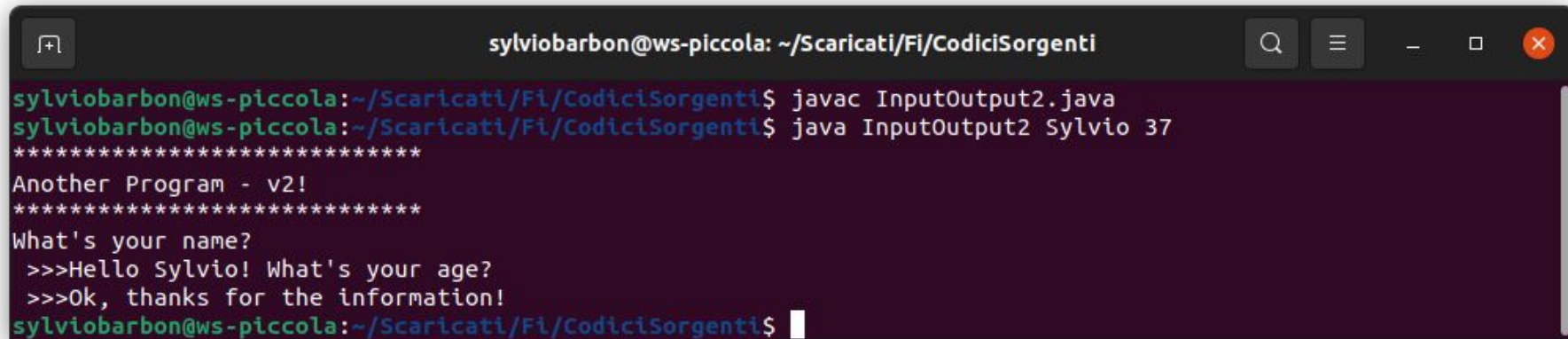
5) Input Java

- È possibile utilizzare vari metodi di input:
 - `System.in` (`java.lang.System`)
 - `Console` (`java.io.Console`)
 - `Console + Scanner` (`java.util.Scanner`)
 - `String args[]` ***** usando il terminale *****

5) Input Java

```
InputOutput.java x InputOutput2.java x
1 import java.io.Console;
2
3 public class InputOutput2{
4     public static void main (String args[]) {
5         Console con = System.console();
6         con.printf("*****\n");
7         con.printf("Another Program - v2! \n");
8         con.printf("*****\n");
9
10        con.printf("What's your name? \n >>>");
11        String name = args[0];
12
13        con.printf("Hello "+name+"! What's your age?\n >>>");
14        int age = Integer.parseInt(args[1]);
15
16        con.printf("Ok, thanks for the information!\n");
17    }
18 }
```

5) Input Java



```
sylviobarbon@ws-piccola: ~/Scaricati/Fi/CodiciSorgenti
sylviobarbon@ws-piccola:~/Scaricati/Fi/CodiciSorgenti$ javac InputOutput2.java
sylviobarbon@ws-piccola:~/Scaricati/Fi/CodiciSorgenti$ java InputOutput2 Sylvio 37
*****
Another Program - v2!
*****
What's your name?
>>>Hello Sylvio! What's your age?
>>>Ok, thanks for the information!
sylviobarbon@ws-piccola:~/Scaricati/Fi/CodiciSorgenti$
```

5) Input Java

```
sylviobarbon@ws-piccola: ~/Scaricati/FI/CodiciSorgenti
sylviobarbon@ws-piccola:~/Scaricati/FI/CodiciSorgenti$ java InputOutput2
*****
Another Program - v2!
*****
What's your name?
>>>Exception in thread "main" java.lang.ArrayIndexOutOfBoundsException: Index 0 out of bounds for length 0
    at InputOutput2.main(InputOutput2.java:11)
sylviobarbon@ws-piccola:~/Scaricati/FI/CodiciSorgenti$
```

5) Input Java

```
IMC.java
1 import java.io.Console;
2 import java.util.Scanner;
3
4 public class IMC{
5     public static void main (String args[]){
6         Console con = System.console();
7         Scanner sc = new Scanner(con.reader());
8
9         con.printf("*****\n");
10        con.printf("**Calcolo - Indice Massa Corpore (IMC) **");
11        con.printf("*****\n");
12
13        con.printf("Altezza: \n >>>");
14        double altezza = sc.nextDouble();
15
16        con.printf("Peso:\n >>>");
17        int peso = sc.nextInt();
18
19        con.printf("IMC = %.2f", (peso/(altezza*altezza)));
20    }
21 }
```

leggendo come double

leggendo come decimale

formatto con due cifre decimali

Grazie!!!

