Scala cheat sheet

## Online

<https://docs.scala-lang.org/>

<https://docs.scala-lang.org/cheatsheets/>

<https://jaxenter.com/cheat-sheet-complete-guide-scala-136558.html>

<http://overapi.com/static/cs/Scala_Cheatsheet.pdf>

## Import

import org.apache.spark.sql.SparkSession

import org.apache.spark.sql.functions.\_

## Commenti

// come in C++, Java, ecc.

/\* commento esteso

su più linee \*/

## Variabili e costanti

va**r** x = "mutabile" // attenzione alla 'r' in var

val y = "immutabile"

val z : BigInt = 43

## Ennuple

val q = (10, 3.4, "ciao!")

println(q.\_1) // 10

## Array

val arr = Array(5,10,20)

val lst = List(5,10,20) // immutabile

println((arr.length, arr.head, arr.last, arr(2)))

println((lst.length, lst.head, lst.last, lst(2)))

arr(0) = 4

lst(0) = 4 // ERRORE (lst è immutabile)

## Stringhe

var s = " Una stringa "

s.length // 18

s.trim // "Una stringa"

val s2 = """

una stringa su più

righe"""

"rosso,nero,blu".split(",") //-> Array(rosso, nero, blu)

"ciao " ++ "mondo!" //-> "ciao mondo!"

"result = %d".format(40+2) //-> "result = 42"

val result = 40+2

s"la risposta è $result" //-> "la risposta è 42"

## Operazioni su vettori

val arr = List(5,10,20)

arr.map(x=>x\*2) //-> List(10, 20, 40)

arr.map(x=>x\*2).reduce((x,y)=>x+y) //-> 70

arr.map(\_\*2).reduce(\_+\_) // notazione abbreviata

arr.filter(\_%2==0) //-> List(10, 20)

## Funzioni

def triplica(x:Int) : Int = x\*3

triplica(6) // -> 18

def norm(x:Double, y:Double) : Double = {

val r2 = x\*x + y\*y

return Math.sqrt(r2)

}

norm(3,4) //-> 5.0

## Case class

case class Persona(name:String, age:Int)

Persona("Silvia", 42) // Persona(Silvia, 42)

## Variabili

var x = 12

val y = 45

## Programma

object {

}