



MAGGIO 2024

COS'È?

LA CLASSE OBJECT: LA SUPERCLASSE PIÙ "SUPER" DI TUTTE!

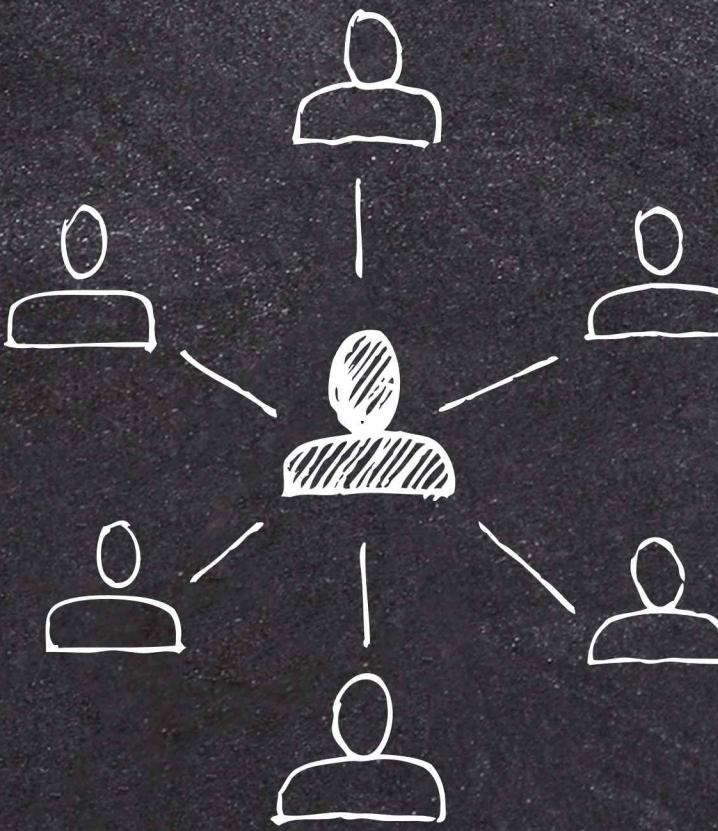
La classe Object in Java è una delle classi da cui tutte le classi derivano secondo il principio di ereditarietà.

Essa non ha bisogno di essere specificata per essere usata poiché definita nell'utility `java.lang` già presente sul compilatore senza ulteriori specificazioni e ci permette di applicare agli oggetti istanziati di qualsiasi classe, metodi specifici (senza il bisogno di crearli da 0).

Insomma ... è l'amica che tutti vorremmo !



I METODI DELLA CLASSE OBJECT



PEZZI DI CODICE , SIMILI A FUNZIONI , CHE SONO ATTIVE IN CONTEMPORANEA CON QUELLA PRINCIPALE ED ALTRI THREAD

...E MOLTI ALTRI, MA MOMENTANEAMENTE NON SONO UTILI , POICHÈ COLLEGATI AI THREADS DI UN CODICE

I METODI DELLA CLASSE OBJECT SONO TANTI E SICURAMENTE SEMPLIFICANO LA VITA DEL PROGRAMMATORE . ESSI SONO:

- EQUALS (OGGETTO)
- HASHCODE()
- TOSTRING()
- GETCLASS()
- CLONE()
- FINALIZE()



IL METODO EQUALS(OGGETTO)



IL METODO EQUALS CONFRONTA DUE OGGETTI ISTANZIATI IN JAVA (QUELLO A CUI IL METODO FA RIFERIMENTO E QUELLO PRESO COME PARAMETRO DEL METODO) E NE DETERMINA L'UGUAGLIANZA.

ESEMPIO SUL METODO EQUALS(OGGETTO)

```
package spiderman;

/**
 * @author Elisa De Salvo
 */
public class Main {

    public static void main(String[] args) {
        Spiderman spiderman1 = new Spiderman("Peter Parker", 25);
        Spiderman spiderman2 = spiderman1; //ho assegnato all'oggetto spiderman2 l'oggetto spiderman1
        Spiderman spiderman3 = new Spiderman("Miles Morales", 18);
        if (spiderman1.equals(spiderman2)==true) { //il metodo equals restituisce un valore booleano
            System.out.println(spiderman1.getNome() + " dice: ehi ma tu " + spiderman2.getNome() + " sei me!");
        } else {
            System.out.println("I due Spiderman sono diversi.");
        }

        if (spiderman1.equals(spiderman3)==true) {
            System.out.println(spiderman1.getNome() + " dice: ehi ma tu " + spiderman2.getNome() + " sei me!");
        } else {
            System.out.println("I due Spiderman sono diversi.");
        }
    }
}
```

```
package spiderman;

/*
 *
 * @author Elisa De Salvo
 */

public class Spiderman {
    private String nome;
    private int eta;

    public Spiderman(String nome, int eta) {
        this.nome = nome;
        this.eta = eta;
    }

    public String getNome(){
        return nome;
    }
}
```

Output - Spiderman (run) X

run:
Peter Parker dice: ehi ma tu Peter Parker sei me!
I due Spiderman sono diversi.
BUILD SUCCESSFUL (total time: 2 seconds)

IL METODO HASHCODE()

HashCode di personal: 849792612

HashCode di persona2: 840755977

IL METODO
RESTITUISCE
IL VALORE
HASH DI UN
OGGETTO

IL METODO TOSTRING()

```
Persona{nome='Mario', eta=30}
```

IL METODO
RESTITUISCE
TUTTI I VALORI
DI UN OGGETTO
COME STRINGA

IL METODO GETCLASS()

```
package tazza;

/**
 * @author Elisa De Salvo
 */

public class Tazza {
    private String colore;

    Tazza(String colore) {
        this.colore=colore;
    }
}
```



run:
class tazza.Tazza

```
package tazza;

/**
 * @author Elisa De Salvo
 */

public class Main {

    public static void main(String[] args) {
        Tazza tazzinaBella = new Tazza ("blu");
        System.out.println(tazzinaBella.getClass());
    }
}
```

IL METODO
RESTITUISCE LA
CLASSE (E IL
RISPETTIVO
PACKAGE)
DELL'OGGETTO A
CUI È ASSOCIAZIO
IL METODO.

IL METODO CLONE()



IL METODO CLONE
PERMETTE LA PERFETTA
COPIA DI UN OGGETTO.
ESSA PUÒ ESSERE UNA
"SHALLOW-
COPY"(COPIA
SUPERFICIALE) O UNA
"DEEP-COPY"(COPIA
PROFONDA).

QUALCOSINA IN PIÙ SUL METODO CLONE()

LA SHALLOW-COPY E LA DEEP-COPY SONO MOLTO SIMILI, DIFATTI LA LORO UNICA DIFFERENZA È IL PASSAGGIO DEGLI ATTRIBUTI DI RIFERIMENTO (AD ESEMPIO STRING).

MENTRE LA SHALLOW-COPY LI COPIA IN MANIERA APPROXIMATIVA, LA DEEP-COPY LI RENDE RICORSIVAMENTE UGUALI.

SECONDO UNA SERIE DI REGOLE, CHE NE PERMETTONO LA PRECISIONE

ESEMPIO SUL METODO CLONE()

```
package starwars;

/**
 * 
 * @author Elisa De Salvo
 */
public class Cloni implements Cloneable {
    private String nome;
    private int id;

    Cloni(String nome, int id) {
        this.nome=nome;
        this.id=id;
    }

    @Override
    public Object clone() throws CloneNotSupportedException {
        return super.clone();
    }
}
```

```
package starwars;

/**
 * 
 * @author Elisa De Salvo
 */
public class Produttore {

    public static void main(String[] args) throws CloneNotSupportedException {
        Cloni jangoFett = new Cloni("JF",1);
        Cloni primoClone=(Cloni) jangoFett.clone();
        Cloni clone2=(Cloni) primoClone.clone();
        Cloni clone3=(Cloni) primoClone.clone();
        Cloni clone4=(Cloni) primoClone.clone();
        Cloni clone5=(Cloni) primoClone.clone();
        Cloni clone6=(Cloni) primoClone.clone();
    }
}
```

IL METODO FINALIZE()



IL METODO FINALIZE()
PERMETTE DI "FAR DIRE LE
ULTIME PAROLE" AD
OGGETTO CHE STA PER
ESSERE ELIMINATO (POICHÉ
NON CI SONO PIÙ
RIFERIMENTI AD ESSO NEL
CODICE).

ESEMPIO SUL METODO FINALIZE()

```
package sac;
/*
 * @author Elisa De Salvo
 */
public class Spadaccino {
    private String nome;

    Spadaccino(String nome) {
        this.nome=nome;
    }

    public void attacca() {
        System.out.println(nome + " attacca con la spada!");
    }

    @Override
    protected void finalize() throws Throwable {
        try {
            System.out.println("il giocatore " + nome + " e' morto");
        }
        finally {
            super.finalize();
        }
    }
}
```

```
package sac;
/*
 * @author Elisa De Salvo
 */
public class Main {

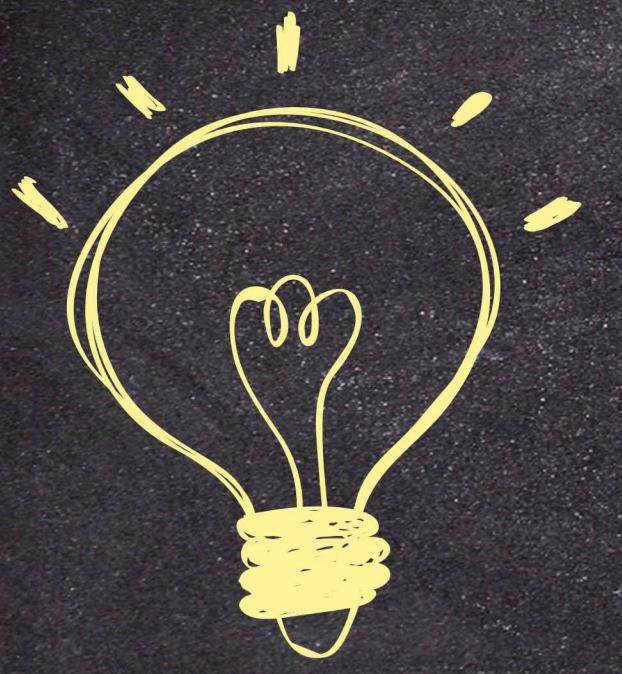
    public static void main(String[] args) {
        Spadaccino assassino = new Spadaccino("Heatclif");
        Spadaccino morto = new Spadaccino("Asuna");
        assassino.attacca();
        morto=null;
        System.gc(); // Richiamo esplicito del Garbage Collector
    }
}
```

Heatclif attacca con la spada!
il giocatore Asuna e' morto

NOTE SUL METODO FINALIZE()



IL METODO FINALIZE() È UN METODO
"DI PULIZIA" PERCIÒ NON È
ASSICURATO CHE ESSO AGISCA NEL
MOMENTO ESATTO IN CUI VENGA
EVOCATO (CON IL COMANDO
SYSTEM.GC()).



FINE DELLA
PRESENTAZIONE
(SPERO SIA
STATA
DIVERTENTE ED
ESAUSTIVA)