

```
public class Mestro extends Thread {
    private String url;

    public Mestro(String s) {
        url = s;
    }

    public void run(){
        for(int i=0; i<10000; i++) {
            System.out.println(url);
        }
    }
}

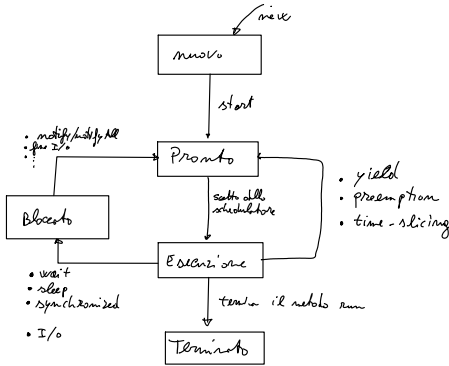
class Main {
    public static void main(String[] args) {
        Mestro uomoLupo = new Mestro("Auuuuu!");
        Mestro frankie = new Mestro("Agggggghaaa!");

        uomoLupo.start();
        frankie.start();
    }
}
```

Specializ. del comportamento
tramite sovraccarico

La JVM rimane in vita fino a che c'è almeno
un thread attivo.

Quali sono gli stati di un thread



Non è possibile chiamare due volte il metodo
start su un thread.

I Thread hanno dei livelli di priorità

Thread.MIN_PRIORITY (1)

Thread.MAX_PRIORITY (10)

NORM_PRIORITY privilegia il main thread

Il livello di priorità viene ereditato dal thread padre

Può essere cambiato con il metodo setPriority()
(c'è anche getPriority)

La politica di scheduling è fixed priority:
viene scelto il thread a priorità più elevata
tra quelli in code pronti

Il metodo yield() può essere usato per cedere volontariamente il
processore (è un suggerimento per lo scheduler).



Se dover estendere la classe Thread può essere scomodo
Metodo alternativo: usare l'interfaccia Runnable

```
public interface Runnable {
    void run();
}
```

- 1) Scrivere una classe che implementa `Runnable` definendo il metodo `run()` (con il comportamento del thread)
- 2) Creare un'istanza della classe
- 3) Creare un oggetto `Thread` passando al costruttore l'oggetto creato al punto 2
- 4) Chiamare il metodo `start`

Esempio

[PA1819_thread2](#)

```
run ▶ open in
```

```
Main.java  History
1 class Main {
2     public static void main(String[] args) {
3
4         ComportamentoMostro c1 = new ComportamentoMostro("Uomo lupo");
5         ComportamentoMostro c2 = new ComportamentoMostro("Zombie");
6
7         Thread t1 = new Thread(c1);
8         Thread t2 = new Thread(c2);
9
10        t1.start();
11        t2.start();
12    }
13 }
```

```
Java version "1.8.0_31"
Java(TM) SE Runtime Environment (Build 1.8.0_31-b13)
Java HotSpot(TM) 64-Bit Server VM (Build 25.31-b07, mixed mode)
```