

# Esercizio Canzone-Playlist

Si realizzi una classe JAVA **Canzone** per la rappresentazione di brani musicali. Una **canzone** e' caratterizzata dal titolo (una stringa), dalla durata (un intero) e dall'autore (una stringa). La classe deve avere un metodo costruttore che presi in ingresso due stringhe ed un intero inizializzi opportunamente le variabili di istanza del nuovo oggetto. La classe deve inoltre fornire i seguenti metodi:

- **haAutore(c)** che data una stringa c restituisce TRUE se l'autore della **canzone** e' uguale alla stringa passata come parametro;
- **uguale(s)** che preso in ingresso una **canzone** s restituisce TRUE se la **canzone** in ingresso ha lo stesso titolo e la stessa durata dell'oggetto di invocazione
- **toString()** che restituisce una stringa contenente la rappresentazione della **canzone**

Si realizzi poi una classe **Playlist** per la rappresentazione di elenchi di riproduzione di canzoni. Un'istanza della classe deve contenere, come attributo d'istanza, un array di canzoni e deve fornire i seguenti metodi:

- **aggiungi(u,d,c)** che prese in ingresso due stringhe u,c ed un intero d aggiunge una nuova canzone alla **playlist** avente u come titolo, d come durata e c come autore;
- **eliminaAutore(c)** che presa in ingresso una stringa c elimina dalla **playlist** tutte le canzoni aventi c come autore;
- **haDuplicati()** che restituisce TRUE se all'interno delle **playlist** ci sono due canzoni uguali fra loro (ovvero aventi stesso titolo e durata);
- **quanteAutore(c)** che data una stringa c restituisce un intero corrispondente al numero di canzoni contenute nella **playlist** con autore c;
- **quanteCanzoni()** che restituisce un intero rappresentante il numero totale delle canzoni salvate nella **playlist**;
- **stampaAutore(c)** che data una stringa c stampa tutti le canzoni con autore c;

Scrivere le classi in modo da compilare ed eseguire la seguente classe di prova senza modificarla.

```
public class ProvaEserc1 {
    public static void main(String[] args) {
        Canzone s1 = new Canzone("a",200,"c1");
        Canzone s2 = new Canzone("a",300,"c2");
        Canzone s3 = new Canzone("a3",400,"c3");
        System.out.println(s1);
        System.out.println(s1.haAutore("c1"));
        System.out.println(s1.haAutore("c2"));
        System.out.println(s1.uguale(s2));
        System.out.println(s1.uguale(s3));
        Playlist pref = new Playlist();
        pref.aggiungi("Forever young", 300, "Bob Dylan");
        pref.aggiungi("Tomorrow never comes", 300, "The Beatles");
        pref.aggiungi("To Ramona", 400, "Bob Dylan");
        pref.aggiungi("She's leaving home", 150, "The Beatles");
        pref.aggiungi("Martha my dear", 300, "The Beatles");
        pref.aggiungi("Someday never comes", 430, "Creedence Clearwater
```

```
Revival");
    System.out.println(pref.quanteAutore("Bob Dylan"));
    System.out.println(pref.quanteAutore("The Beatles"));
    System.out.println(pref.quanteAutore("Franco Califano"));
    System.out.println(pref.quanteAutore("Creedence Clearwater
Revival"));
    pref.stampaAutore("The Beatles");
    pref.eliminaAutore("The Beatles");
    pref.stampaAutore("The Beatles");
    System.out.println(pref.quanteAutore("Bob Dylan"));
    System.out.println(pref.quanteAutore("The Beatles"));
    System.out.println(pref.quanteAutore("Creedence Clearwater
Revival"));
    System.out.println(pref.haDuplicati());
    pref.aggiungi("To Ramona", 400, "Bob Dylan");
    System.out.println(pref.haDuplicati());
}
}
```