

INFORMATICA GRAFICA - FONDAMENTI DI INFORMATICA
Ing. Civile - Edile/Architettura – Dott. Penzo

SOLUZIONE PROVA SCRITTA DEL 17/06/2002

```
import fiji.io.SimpleReader;

public class Programma {

    public static void main(String[] args) {

        SimpleReader in = new SimpleReader();

        int numClienti;
        String nome, cognome;
        Cliente cliente;

        System.out.println("Inserisci il numero di clienti da gestire: ");
        numClienti = in.readInt();
        int k = 1;
        while (k <= numClienti){
            System.out.println("Inserire il nome del cliente " + k + " : ");
            nome = in.readString();

            System.out.println("Inserire il cognome del cliente " + k + " : ");
            cognome = in.readString();

            cliente = new Cliente(nome, cognome);

            Supermercato.aggiungiCliente(cliente);
            k++;
        }

        System.out.println("Inserire il nome del socio: ");
        nome = in.readString();

        System.out.println("Inserire il cognome del socio: ");
        cognome = in.readString();

        String res = Supermercato.aggiungiSocio(nome, cognome);
        System.out.println(res);

        cliente = Supermercato.getCliente(nome, cognome);
        if (cliente != null) {

            /* qui si procede a riempire il carrello per verificare
            * che il metodo spesaTotale funzioni.
            * Parte non richiesta dal testo */
            compilaCarrello(cliente);

            int spesaTotale = cliente.getCarrello().spesaTotale();
            System.out.println("Spesa totale: " + spesaTotale);
        }

    }

    /* Metodo non richiesto dal testo. Si riporta ai soli fini di test */
    private static void compilaCarrello(Cliente client) {
        Prodotto p1, p2, ...; /* prodotti di esempio..... */

        /* qui fase di inizializzazione dei prodotti da
```

```

        * inserire nel carrello. Non richiesto dal testo */
        .....

        Carrello c = client.getCarrello();
        c.inserisciProdotto(p1);
        c.inserisciProdotto(p2);
        .....
    }
}

```

```

public class Cliente {

    private String nome;
    private String cognome;
    private int numCartaSocio;
    private Carrello carrello;

    /**
     * costruttore nome, cognome
     */
    public Cliente(String nome, String cognome) {
        this.nome = nome;
        this.cognome = cognome;
        this.numCartaSocio = 0;
        this.carrello = new Carrello();
    }

    /**
     * Gets the nome
     */
    public String getNome() {
        return this.nome;
    }

    /**
     * Gets the cognome
     */
    public String getCognome() {
        return this.cognome;
    }

    /**
     * Gets the numCartaSocio
     */
    public int getNumCartaSocio() {
        return this.numCartaSocio;
    }

    /**
     * Gets the carrello
     */
    public Carrello getCarrello() {
        return this.carrello;
    }

    /**
     * Sets the numCartaSocio
     */
    public void assegnaNumeroSocio (int numCarta) {
        this.numCartaSocio = numCarta;
    }
}

```

```

    }

    /**
     * equals
     */
    public boolean equals(Cliente client) {
        return client.getNome().equals(this.nome) &&
            client.getCognome().equals(this.cognome);
    }

    /**
     * toString
     */
    public String toString() {
        String s;
        if (this.numCartaSocio != 0) {
            s = this.nome + ", " + this.cognome + ", " + this.numCartaSocio;
        }
        else
            s = this.nome + ", " + this.cognome;
        return(s);
    }
}

```

```

public class Prodotto {

    private String codice;
    private String descrizione;
    private int costoUnitario;

    /**
     * costruttore
     */
    public Prodotto(String codice,
                    String descrizione,
                    int costoUnitario) {

        this.codice = codice;
        this.descrizione = descrizione;
        this.costoUnitario = costoUnitario;
    }

    /**
     * Gets the costoUnitario
     */
    public int getCostoUnitario() {
        return this.costoUnitario;
    }
}

```

```

public class Carrello {

    private static final int NUM_PRODOTTI = 100;

    private Prodotto[] prodotti;
    private int progrCarrello;

    /**

```

```

    * costruttore
    */
    public Carrello() {
        prodotti = new Prodotto[NUM_PRODOTTI];
        progrCarrello = 0;
    }

    /**
     * inserisciProdotto
     */
    public void inserisciProdotto(Prodotto prodotto) {
        if (progrCarrello < NUM_PRODOTTI){
            prodotti[progrCarrello] = prodotto;
            progrCarrello++;
        }
    }

    /**
     * spesaTotale
     */
    public int spesaTotale() {
        int ret = 0, k;
        for (k=0; k < progrCarrello; k++ ) {
            ret += prodotti[k].getCostoUnitario();
        }
        return ret;
    }
}

```

```

public class Supermercato {

    private static final int NUM_CLIENTI = 1000;

    private static Cliente[] clienti = new Cliente[NUM_CLIENTI];
    private static int progrCliente = 0;
    private static int progrNumCarta = 1;

    /**
     * aggiungiCliente
     */
    public static void aggiungiCliente(Cliente cliente) {
        if (progrCliente < NUM_CLIENTI){
            clienti[progrCliente] = cliente;
            progrCliente++;
        }
    }

    /**
     * aggiungiSocio
     */
    public static String aggiungiSocio(String nome, String cognome) {

        String ret;
        Cliente cl = getCliente(nome, cognome);

        if (cl == null) {
            ret = "Cliente inesistente!";
        } else {
            if (cl.getNumCartaSocio() == 0) {
                cl.assegnaNumeroSocio(progrNumCarta);
            }
        }
    }
}

```

```

        ret = "Nuovo Socio, carta numero: " + progrNumCarta;
        progrNumCarta++;
    } else {
        ret = "Il cliente e' gia' Socio!";
    }
}
return ret;
}

/**
 * getCliente
 */
public static Cliente getCliente(String nome, String cognome) {
    boolean found = false;
    Cliente cl = new Cliente(nome, cognome);
    Cliente retCliente = null;

    for (int k=0; k < progrCliente && !found; k++) {
        if (cl.equals(clienti[k])) {
            found = true;
            retCliente = clienti[k];
        }
    }
    return retCliente;
}
}

```