

Ereditarietà, toString, equals, hashCode; javap, javadoc

In aggiunta a quanto specificato nella esercitazione su *Classi ed oggetti*:

a) La banca gestisce quattro diversi tipi di conto bancario

- Il *conto bancario* (`Conto`)
- Il *libretto di risparmio* (`LibrettoDiRisparmio`), un *conto bancario* che frutta interessi mensili.
- Il *conto corrente* (`ContoCorrente`), un *conto bancario* che non ha interessi, offre un numero limitato di operazioni mensili gratuite ma addebita una commissione per ciascun movimento aggiuntivo.
- Il *conto vincolato* (`ContoVincolato`) che è un *libretto di risparmio* che impegna a lasciare il denaro nel conto per certo numero di mesi ed prevede una penale per il ritiro anticipato.

b) Sul *conto bancario* possono compiersi anche le seguenti operazioni:

- `public boolean equals(Object obj)` che sovrascrive l'omonimo metodo predefinito per la classe `Object`; restituendo `true` se i due conti hanno saldo identico, `false` altrimenti;
- `public int hashCode()` che sovrascrive l'omonimo metodo predefinito per la classe `Object`, restituendo un intero sulla base del valore del campo `saldo`.

c) Definire analogamente i metodi `equals` e `hashCode` sul *cliente*, basandosi esclusivamente sui risultati dei metodi definiti al punto b)

d) Sulla *banca* possono compiersi anche le seguenti operazioni:

- `public void aggiornamenti()` che esegue le operazioni periodiche previste per ciascun tipo conto:
 - aggiunge gli interessi mensili ai *libretti di risparmio*;
 - aggiunge gli interessi ai *conti vincolati*;
 - preleva le commissioni dai *conti correnti*.
- `public String toString()` che restituisce una rappresentazione testuale (intelligibile) dello stato interno dell'oggetto come stringa, sovrascrivendo il metodo predefinito per la classe `Object`, con il seguente formato:

```
nomeClasse [  
  nomeCampo1 = valoreCampo1;  
  ...  
  nomeCampoN = valoreCampoN;  
]
```

dove `nomeCampo` può essere un attributo *proprio* della classe, ovvero una *superclasse* o una *classe contenuta*.

e) Si adoperino i seguenti strumenti a riga di comando della directory *bin* (digitare `-help` dopo il comando per i dettagli):

- `javac` per compilare assieme diversi files di cui sia noto il nome.
- `javadoc` per generare automaticamente la documentazione html
- `javap` per visualizzare i metodi, i campi, o il bytecode di un file `.class`