

Banca

Progettare e realizzare un programma che possa essere utilizzato per la gestione di una banca. Per ogni nuovo conto il sistema memorizza nome e cognome dell'intestatario, nome della banca, capitale depositato e il tasso di interesse annuo applicato su tale conto.

*1) Interagire solo con la classe **Principale** ragionando assumendo di non conoscere ancora il paradigma OOP*

Operando solo sulla classe **Principale** contenente il **main()**, memorizzare le informazioni relative a due nuovi conti bancari.

Primo conto:

nomeIntestatario	Fabio,
cognomeIntestatario	Garcea,
nomeBanca	Unicredit Corso Sebastopoli 280 10136 Torino TO
capitaleDepositato	30000.0
tassoInteresse	1.5

Secondo conto:

nomeIntestatario	Fabrizio
cognomeIntestatario	Lamberti
nomeBanca	Intesa Sanpaolo Corso Alcide De Gasperi 14 10129 Torino TO
capitaleDepositato	100000.0
tassoInteresse	2.5

Stampare nella console le informazioni relative ai conti una per riga, nel formato nome intestatario, cognome intestatario, nome banca, capitale depositato e tasso di interesse separati da una virgola seguita da uno spazio, ad esempio:

```
Fabio, Garcea, Unicredit Corso Sebastopoli 280 10136 Torino TO, 30000.0, 1.5
Fabrizio, Lamberti, Intesa Sanpaolo Corso Alcide De Gasperi 14 10129 Torino TO, 100000.0, 2.5
```

Creare, fuori dal **main()**, una funzione che calcoli l'interesse maturato da un conto bancario in un determinato periodo di tempo. La funzione riceva come parametri il capitale depositato, il tasso di interesse annuo e la durata del periodo espressa in numero di giorni. Sulla base di tali parametri calcolare l'interesse come:

$$I = \frac{(C * i * t)}{36500}$$

dove C rappresenta il capitale depositato, i il tasso di interesse applicato al conto preso in considerazione e t il periodo di tempo espresso come numero di giorni. Si facciano, a titolo di esempio, i calcoli dei tassi di interesse per i conti precedentemente memorizzati su un periodo pari a 5 anni e si stampi il risultato nella console.

Creare una struttura dati di tipo array in grado di memorizzare le informazioni relative a più conti; si inseriscano, a titolo di esempio, le informazioni relative a un numero determinato di conti (ad esempio 5) e si stampino con il suddetto formato. Si assuma che tale struttura possa registrare fino a 100 conti e che quello sia pertanto il numero massimo di conti da memorizzare.

2) Iniziare a ragionare in termini di classi e oggetti

Modificare la classe **Banca**, ed eventualmente aggiungere altre classi qualora ritenuto necessario.

Una banca è caratterizzata da un nome e da un tasso di ricavo annuo. Questo valore rappresenta il ricavo annuo di una banca sui capitali in deposito e deve essere inteso come una semplificazione delle strategie di investimento da essa attuate. È possibile impostare e accedere al nome della banca e al tasso di guadagno annuo rispettivamente tramite i metodi `setNome()`, `getNome()`, `setTasso()` e `getTasso()`.

La registrazione di un nuovo conto viene gestita tramite il metodo `nuovoConto()`, che riceve come parametri nome e cognome dell'intestatario, capitale depositato e tasso di interesse annuo, e si occupa di memorizzare tali informazioni nel sistema. Si assuma che la banca possa registrare fino a 100 conti e che quello sia pertanto il numero massimo di conti da memorizzare.

Per accedere alle informazioni relative all'ultimo conto memorizzato si utilizza il metodo `ultimoConto()`, che restituisce una stringa contenente nome dell'intestatario, cognome dell'intestatario, nome della banca, capitale depositato e tasso di interesse annuo. Tali informazioni devono essere riportate nel formato descritto nella prima parte dell'esercitazione.

Per accedere alle informazioni relative ad uno qualsiasi dei conti memorizzati si utilizza il metodo `conto()`, che riceve come parametri nome e cognome dell'intestatario, e restituisce una stringa con lo stesso formato utilizzato dal metodo precedente.

Infine, il metodo `guadagnoBanca()` riceve come parametro la durata del periodo espressa in numero di giorni e restituisce un valore di tipo `double` relativo al guadagno della banca in tale periodo. Tale valore può essere calcolato come:

$$G = R_{tot} - I_{tot}$$

con I_{tot} pari alla somma degli interessi dovuti ai clienti (si faccia riferimento alla prima parte per il calcolo degli interessi maturati da un singolo conto bancario) e con R_{tot} pari al ricavo sul capitale complessivo depositato in banca, calcolabile come:

$$R_{tot} = \frac{(C_{tot} * r * t)}{36500}$$

Dove C_{tot} rappresenta il capitale complessivo depositato, r il tasso di ricavo annuo della banca e t il periodo di tempo espresso come numero di giorni.

Note

- Una variabile di tipo stringa, in Java rappresentata da una classe ad-hoc, può essere dichiarata come `String nomeStringa;`
- Per stampare nella console, utilizzare la funzione `System.out.println("<testo da stampare>")` eventualmente concatenando stringhe, variabili, ecc. con l'operatore `+`
- La sintassi per la dichiarazione di una variabile di tipo array è: `Tipo[] nomeArray;` oppure `Tipo nomeArray[];`
- La sintassi per allocare la memoria atta a contenere n elementi dell'array è `nomeArray = new Tipo[n];`
- L'accesso agli elementi dell'array avviene per indice; ad esempio, per accedere al quarto elemento si utilizzerà `nomeArray[3];`
- Con un ragionamento orientato agli oggetti, la classe **Principale** riporta tutte le operazioni, effettuate invocando i metodi della classe **Banca** (contenuta nel package **banca**)
- Per risolvere l'esercitazione è necessario definire, tra gli altri, il corpo dei metodi della classe **Banca** e gli eventuali attributi