**Progetto 4: Gestione Banca**

**Requisiti**:

* Creare un applicativo che, simulando il funzionamento di una banca, permetta di effettuare le seguenti operazioni:

1. Creare nuovi conti.
2. Effettuare depositi sui conti.
3. Effettuare prelievi dai conti.
4. Visualizzare i saldi dei conti.

* Utilizzare le librerie standard di java.
* Utilizzare la riga di comando come input e output.
* Gestire input non validi ed operazioni non valide.
* Produrre documentazione di progetto.

**Possibile soluzione**

Si suggerisce di strutturare il progetto definendo le seguenti classi:

**Conto:** definisce l’anagrafica di un conto corrente costituita da:

* un numero identificativo univoco;
* l’intestatario del conto;
* il saldo;

La classe deve anche esporre metodi che consentono di aggiornare opportunamente il saldo del conto ad ogni operazione di versamento e prelievo.

Ovviamente non può deve essere possibile versare somme negative né prelevare somme superiori al saldo residuo.

**Banca**: è la classe che si deve occupare di gestire l’operatività di ciascun conto a partire dalla loro creazione. La classe deve, pertanto, prevedere una opportuna struttura dati che conterrà tutti i conti aperti. Deve anche occuparsi della generazione dell’identificativo univoco di ciascun conto. Deve inoltre esporre metodi che permettono:

* la creazione di un nuovo conto a partire dal nome dell’intestatario;
* effettuare versamenti e prelievi di somme su un determinato conto;
* visualizzare il saldo di un conto;

**Main**: classe che contiene il main e consente, quindi, l’esecuzione dell’applicativo. Tale classe deve consentire all’utente di effettuare, da riga di comando, le seguenti operazioni:

1. Crea un nuovo conto
2. Effettua un deposito
3. Effettua un prelievo
4. Visualizza saldo
5. Esci (terminare l’applicazione)

Tutte le classi devo appartenere ad un opportuno package.

**Possibili evoluzioni:**

1. **Funzionalità Avanzate**: Aggiungere funzionalità come chiusura conto, trasferimenti tra conti, gestione di più valute, ecc.
2. **Persistenza dei Dati**: Integrare un sistema di persistenza dei dati per salvare i conti su file o database.
3. **Autenticazione**: Implementare un sistema di autenticazione per proteggere l’accesso ai conti.
4. **Interfaccia grafica:** Integrare un’interfaccia grafica (utilizzando le Swing), in alternativa alla riga di comando, attraverso le quali l’utente può effettuare tutte le operazioni previste dai requisiti.