

## ESERCITAZIONE TIGA: File, compareTo, package, jar

1) In aggiunta a quanto specificato nella esercitazione su *ereditarietà*, sulla *banca* possono essere fatte (almeno) le operazioni:

- `public void salva(String fname)` che salva questa banca sul file il cui nome è specificato dal parametro `fname`. Supponendo che l'operazione sia eseguita quotidianamente, il nome del file deve essere strutturato come segue: "banca." + `<data attuale>` + ".dat", dove `<data attuale>` indica la data in cui viene eseguita l'operazione, nel formato giorno-mese-anno<sup>1</sup>.
- `public void ripristina(String fname)` che legge questa banca dal file il cui nome è specificato da `fname`, ricostruendo lo stato dell'oggetto al momento del salvataggio.
- `public void elaboraESalvaStatistiche(String fname)` che *aggiunge* ad un file di testo, il cui nome è specificato dal parametro `fname`, le seguenti informazioni:

```
n. di clienti della banca <tabulazione> importo complessivo dei saldi depositati <invio>
```

- `public void caricaEdElaboraStatistiche(String fname)` che legge dal file di testo, il cui nome è specificato dal parametro `fname`, le varie registrazioni precedentemente eseguite mediante `salvaStatistiche` (oppure inserite manualmente mediante un text editor, o inserite da altre applicazioni che scrivono nel file rispettando il formato suddetto), e visualizza sullo schermo i due valori medi (del n. di clienti e dell'importo complessivo dei saldi) sul totale di registrazioni.
- `public void ordina()` che riordina l'array di clienti adoperando il metodo `sort` della classe `Arrays`. A tale scopo, si implementi il metodo `compareTo` dell'interfaccia `Comparable` sia in `Cliente`<sup>2</sup> che in `Conto`, basando il confronto esclusivamente sul valore del saldo. Si faccia l'ipotesi semplificativa che l'array non contenga riferimenti a `null`<sup>3</sup>.

2) Riorganizzare i file dell'intera applicazione banca come in Fig.1. Creare una cartella *lab4*. Il package *lab4.banca* (cartella *banca*) contiene le classi *Banca* e *Cliente*, ed il package *lab4.banca.contoBancario* (cartella *banca/contoBancario*) contiene i diversi tipi di conto. I file sorgenti sono invece nella cartella *sorgente* con la medesima struttura vista per i corrispondenti package. La cartella *dati* contiene i file prodotti dai metodi `salva` (in una sottocartella *archivio*) ed `elaboraESalvaStatistiche` (in una sottocartella *statistiche*). Infine, la cartella *documentazione* contiene le pagine web prodotte da javadoc.

3) Creare un unico file *lab4.jar*, contenente solo i file *.class*, mediante il Java Archiver (JAR), con il seguente comando (da Shell):

```
jar cmf lab4/mainClass.txt lab4.jar lab4/banca
```

dove *mainClass.txt* è un file di testo che indica quale classe possiede il metodo `main`, nel seguente formato<sup>4</sup>:

```
Main-Class: lab4.banca.Banca ↵
```

4) Portare il file *jar* in un altro path, ed eseguirlo:

```
java -jar lab4.jar <eventuali parametri>
```

Funziona ancora? Adesso, dove vengono salvati e letti i file dati?

5) Inserire tutti i comandi di compilazione ed esecuzione, produzione ed esecuzione del *.jar*, etc., in un file di testo di tipo *batch* "lab4.bat", ossia uno script contenente una sequenza di comandi per la Shell, invocabile da riga di comando come "lab4".

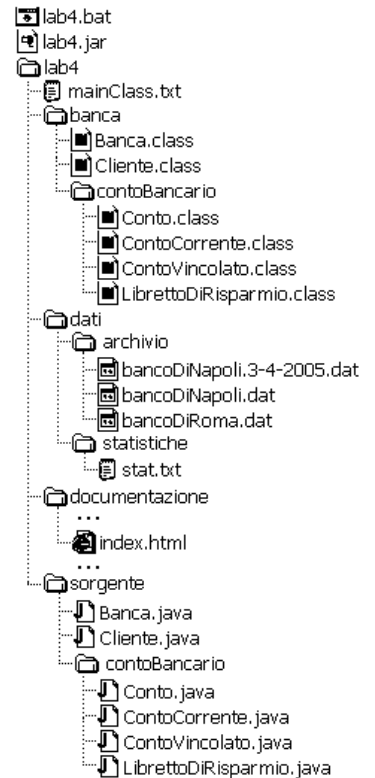


Fig.1 riorganizzazione dei file

<sup>1</sup> Si consigliano le classi `Calendar` e `GregorianCalendar` del pacchetto `java.util`. Si noti che la classe `Calendar` è astratta.

<sup>2</sup> Il metodo `compareTo` di `cliente` si limita ad invocare il metodo `compareTo` di `conto`, dopo aver eseguito un cast.

<sup>3</sup> Altrimenti il metodo `Arrays.sort` si bloccherebbe generando l'eccezione `java.lang.NullPointerException`. Questo problema si potrebbe risolvere, ad esempio, adoperando un oggetto *Cliente dummy* ("fantoccio") al posto del `null`, definito come una costante privata e statica della classe `Banca`:

```
private static final Cliente DUMMY = new Cliente ("", 0, 0, new Conto());
```

e rivedendo il costruttore e tutti i metodi di `Banca` in modo che i riferimenti a `null` nell'array `elencoClienti` siano convertiti in riferimenti all'oggetto `DUMMY`, e cambiando i test `"== null"` in `"== DUMMY"` (non occorre usare `equals`).

<sup>4</sup> Assicurarsi di battere anche il ritorno carrello (↵), in assenza del quale, nei sistemi Windows®, il JAR non inserisce nel file manifesto le informazioni sulla `Main-Class`, per cui l'interprete java produce a run-time l'errore `Failed to load Main-Class manifest attribute`.