

Relazione Progetto Programmazione Java

Ludovica Pannitto, mat. 491094

September 23, 2016

Il progetto realizza le funzionalità necessarie per la gestione di una concessionaria di auto. Memorizza l'archivio delle auto e permette semplici operazioni come aggiungere o rimuovere un'auto e modificarne alcuni parametri. Tramite un menu testuale è possibile compiere tali operazioni e visualizzare le auto alle quali si è interessati.

1 Struttura del progetto

Il progetto si divide in 7 packages, come mostrato nella figura allegata. Per la documentazione specifica di ogni classe fare riferimento a quella generata da javadoc.

concessionaria

Il package concessionaria contiene le classi che rappresentano le entità concrete che fanno parte della concessionaria.

compare e filters

I due package implementano funzioni ausiliarie che permettono, rispettivamente, di confrontare le auto in base a diversi parametri e di stabilire un criterio di appartenenza di un'auto a una specifica classe (e filtrare in questo modo una lista di auto).

exceptions

Il package contiene eccezioni definite per rispecchiare problemi specifici della concessionaria.

menu

Il package contiene le classi per gestire l'output e l'input di valori.

main

Il package contiene le classi (in verità, la classe) eseguibili del progetto

2 Scelte implementative

Concessionaria

La concessionaria mantiene un semplice archivio (implementato tramite Vector) di Auto.

Poiché le auto si suddividono in differenti categorie (al momento "nuove" o "usate"), si è scelto di creare una superclasse "Auto" che memorizzi gli attributi comuni all'intera classe, ed estendere la suddetta tramite due sottoclassi specifiche (AutoNuova e AutoUsata). In questo modo è molto semplice l'aggiunta di nuove categorie.

Anche se per l'implementazione attuale potrebbe non essere necessaria una classe Cliente per memorizzare le informazioni relative all'acquirente di una nuova auto, la classe è stata creata per permettere il futuro mantenimento di un archivio anagrafico dei clienti della concessionaria.

Input e memorizzazione dei dati

È stata dedicata attenzione al controllo dei valori inseriti in input. Sono stati posti i seguenti limiti:

- le targhe inserite devono essere nel formato cddddcc (con c carattere alfabetico e d cifra)
- il prezzo di un'auto deve necessariamente essere maggiore di 0,1
- l'anno di immatricolazione di un'auto usata deve essere minore o uguale all'anno corrente, e maggiore di 1900
- la data di consegna di una nuova auto deve essere valida e successiva alla data corrente

Menu

Per il progetto è stata implementata unicamente una interfaccia da linea di comando.

Per rendere tuttavia possibile la successiva implementazione di una interfaccia grafica (o comunque una diversa interfaccia) si è scelto di realizzare un'interfaccia che potesse essere implementata da più classi.

Ai fini dell'implementazione dell'interfaccia da linea di comando, risultava necessario avere un meccanismo che permettesse di associare l'esecuzione di una funzione a ogni scelta del menu.

A questo proposito è stata sviluppata l'interfaccia "MenuItem" che rappresenta ogni voce del menu. L'interfaccia dispone di due metodi: il metodo "getDescription", che restituisce la stringa che descrive la voce del menu e che verrà quindi mostrata all'utente, e il metodo "selected", che contiene la funzione da eseguire al momento della selezione.

L'utilizzo di questa interfaccia permettere di creare facilmente voci del menu che siano versatili, e dispensa il programmatore dall'implementare esplicitamente il codice che gestisce la scelta dell'utente.

3 Funzionalità

Funzionalità richieste

- aggiungere un'auto all'archivio
- rimuovere un'auto dall'archivio
- visualizzare l'elenco di tutte le auto (nuove e usate insieme) in ordine di numero di targa
- visualizzare l'elenco delle auto nuove o di quelle usate
- modificare la data di prevista consegna di un'auto nuova
- ricercare le auto acquistate da un certo cliente sulla base del suo nome o una porzione di esso
- ricercare tutte le auto usate immatricolate dopo un certo anno e con al più un certo numero di chilometri
- ricercare la prossima auto nuova da consegnare
- salvare l'elenco delle auto su file binario
- caricare l'elenco delle auto da un file precedentemente salvato

Altre funzionalità

- Oltre alla data di consegna, è possibile modificare anche il colore o il prezzo di un'auto
- Si può visualizzare la lista di auto ordinata per prezzo, modello o cilindrata
- Sono stati implementati vari filtri aggiuntivi (ad esempio per targa tramite un'espressione regolare), ed è possibile la composizione di più filtri
- I filtri possono essere composti all'ordinamento per ottenere una lista filtrata di auto ordinate secondo uno dei criteri sopra citati

Concessionaria

concessionaria

- Auto.java
- AutoNuova.java
- AutoUsata.java
- Cliente.java
- Concessionaria.java

compare

- CompareByCilindrata.java
- CompareByModello.java
- CompareByPrezzo.java
- CompareByTarga.java
- NoSorting.java

filters

- FilterByCustomer.java
- FilterByKm.java
- FilterByMatriculationYear.java
- FilterByTarga.java
- FilterByType.java
- Filterer.java
- MultipleFiltererAuto.java

exceptions

- AutoException.java
- DateException.java

main

- Main.java

menu

- MenuInterface.java

menuItem

editors

- EditColore.java
- EditDeliveryDate.java
- EditPrezzo.java

filterCreators

- Exit.java
- FilterByCostumerCreator.java
- FilterByKmCreator.java
- FilterByMatriculationYearCreator.java
- FilterByTargaCreator.java
- FilterByTypeCreator.java

sortingCreators

- CompareByCilindrataCreator.java
- CompareByModelloCreator.java
- CompareByPrezzoCreator.java
- CompareByTargaCreator.java
- NoSortingCreator.java
- SortingCreator.java
- SortingSelector.java

- Add.java

- Edit.java

- Exit.java

- Load.java

- MenuItem.java

- NextDelivery.java

- Remove.java

- Save.java

- Show.java

- Menu.java