



User Interfaces Design

Definire la classe Prodotto con i seguenti campi:

- Id, che rappresenta un codice univoco per ogni prodotto [Tipo: String]
- Tipologia del prodotto (es. scarpe, abito, ecc.) [Tipo: String]
- Marca [Tipo: String]
- Modello [Tipo: String]
- Anno di produzione [Tipo: int] (compreso tra 1900 e 2050)
- Prezzo [Tipo: double] (≥ 0.0)

Implementare, inoltre, i metodi ritenuti necessari.

Definire una classe Magazzino per la gestione dei prodotti, dove i prodotti sono rappresentati da una lista (List) di Prodotti:

- Aggiunta di un nuovo prodotto (se l'id del prodotto non è presente)
- Rimozione di un prodotto (dato l'id)
- Ricerca di tutti i prodotti di una certa marca
- Ricerca di tutti i prodotti di un certo modello
- Ricerca di tutti i prodotti di un certo anno
- Ordinamento dei prodotti per prezzo crescente
- Ordinamento dei prodotti per prezzo decrescente
- Stampa di tutti i prodotti

Realizzare un main per testare il funzionamento dell'implementazione creando un menù di scelta. Ad esempio:

- 1 – Aggiunta di un prodotto
- 2 – Rimozione di un prodotto
- ...
- 0 – Esci

dove si esegue l'operazione relativa quando l'utente sceglie il corrispondente numero. Ogni operazione (a parte l'uscita) può essere svolta un numero arbitrario di volte.

I dati dei prodotti devono essere inseriti da standard input.

Metodi utili:

- Collections.sort -> [https://docs.oracle.com/en/java/javase/17/docs/api/java.base/java/util/Collections.html#sort\(java.util.List\)](https://docs.oracle.com/en/java/javase/17/docs/api/java.base/java/util/Collections.html#sort(java.util.List))
- sort di List -> [https://docs.oracle.com/en/java/javase/17/docs/api/java.base/java/util/List.html#sort\(java.util.Comparator\)](https://docs.oracle.com/en/java/javase/17/docs/api/java.base/java/util/List.html#sort(java.util.Comparator))