

Programmazione a oggetti - Java

Esercitazione 5

Contatti:

Prof. Angelo Gargantini – angelo.gargantini@unibg.it

Dott.ssa Silvia Bonfanti – silvia.bonfanti@unibg.it

Classi e oggetti

Esercizio 28

- Scrivi un programma in grado di gestire le ordinazioni ai tavoli di un ristorante. Nel ristorante sono disposti dei tavoli, ciascuno identificato da un id numerico e da un numero massimo di coperti. Per ciascun **Tavolo** devono essere memorizzati i piatti consumati (utilizza `ArrayList`/`Sequenza`/`Vector` inizialmente vuota). Ogni **Piatto** è identificato da un id, dalla quantità e dal suo prezzo. Inoltre ogni tavolo contiene una lista (inizialmente vuota) di persone sedute al tavolo. Ogni **Persona** è identificata da nome e cognome. Nella classe `Tavolo` implementa i metodi per aggiungere le persone (se ci sono posti disponibili), aggiungere i piatti ordinati, rimuovere i piatti non più desiderati (passa l'id come parametro del metodo), modificare la quantità ordinata di un piatto (passa l'id e la nuova quantità come parametri), visualizzare la lista delle persone e la lista dei piatti ordinati, calcolare il totale del tavolo e calcolare il prezzo medio a persona.
- Scrivere una classe di prova in cui si crea almeno un tavolo, e si testano tutti i metodi della classe tavolo.

Esercizio 29

- Si vuole progettare un sistema per la gestione dei voli.
- Ogni **volo** è identificato da un ID alfanumerico da un aeroporto di destinazione, da un aeroporto di partenza, da un numero di posti disponibili in classe economica, da un numero di posti disponibili in prima classe e dalla lista di passeggeri. Per ogni **passeggero** vengono memorizzati id biglietto, nome, cognome, prezzo del biglietto pagato e inoltre viene memorizzato se viaggia in prima classe o in classe economica ('P': prima classe, 'E': economica).
- L'applicazione deve permettere di aggiungere passeggeri al volo (solo se ci sono posti disponibili nella classe richiesta), rimuovere passeggeri (dato l'id del biglietto), calcolare l'incasso del volo e stampare la lista passeggeri

Esercizio 30

- Scrivi un programma in grado di gestire i negozi. Ogni negozio è identificato da un ID numerico (non inserire id uguali) e da un nome. Ogni negozio ha una serie di prodotti che può vendere (utilizza un ArrayList) ed ogni prodotto è identificato da un id numerico (attenzione che non ci siano duplicati all'interno dello stesso negozio), dalla quantità disponibile e dal prezzo unitario.
- I clienti (identificati da nome, cognome e identificativo numerico) possono fare la spesa in un negozio qualsiasi. Ad ogni cliente è associata una lista della spesa (id prodotto e quantità). Il cliente aggiunge prodotti alla lista della spesa, può eliminarli e può modificare la quantità. Per fare la spesa il cliente necessita anche del negozio (passato come parametro) in cui fare la spesa. Alla fine della spesa viene mostrato il totale. Quando il cliente fa la spesa aggiorna la quantità disponibile in negozio.
- Ricorda che un negozio può anche rifornire un prodotto e rimuoverlo dalla lista dei suoi prodotti.
- Scrivi una classe di prova