Progetto Java A.A. 2020/2021

Alice Perin 626005

Silvia Villafranca 628302

1. Struttura codice e informazioni generali

Il programma è pensato per la gestione di un locale per l'organizzazione di compleanni per bambini ed è composto da quattro classi: Prenotazione, Catering, Animazione e quella contenente il Main chiamata GestoreDemo.

Prima del metodo Main troviamo le liste che contengono tutte le prenotazioni possibili da eseguire: affitto, catering, animazione e l'Hashmap per la gestione dell'oggetto data.

Invece la struttura del Main è composta da due menu:

- un primo menu consente di iniziare il programma con l'inserimento di una nuova prenotazione, oppure con il caricamento di prenotazioni esistenti tramite file binari;
- un secondo menu, quello principale, permette all'utente di utilizzare tutte le funzioni del programma.

Dopo aver finito l'interfaccia testuale, la seconda parte del codice è dedicata alla scrittura dei metodi che svolgono le funzioni del programma stesso. Di seguito le funzioni principali che abbiamo ideato sotto forma di metodi: aggiungere nuove prenotazioni, eliminare una o più prenotazioni, visualizzare l'elenco delle prenotazioni in ordine di data dei diversi tipi di lista che abbiamo a disposizione, salvare o caricare un file già esistente grazie alla serializzazione, ricercare delle prenotazioni tramite inserimento data, inserimento del nome del cliente o una porzione di esso e infine decidere di terminare il programma.

2. Classi ed ereditarietà

Come abbiamo detto, le classi che compongono il programma sono quattro. Le tre classi che rappresentano l'oggetto "prenotazione" sono state create con la seguente logica: la superclasse Prenotazione, rappresenta la prenotazione di un semplice affitto e contiene le due variabili *nome* e *data* che vengono ereditate dalle altre due classi. La classe Catering estende la classe Prenotazione, oltre a estendere le variabili e i metodi (*getNome* e *getData*), si aggiungono la variabile privata che andrà a contenere il numero dei bambini che partecipano alla festa e il relativo metodo per accedervi dal main (*getNumeroBambini*). Per ultima, la classe Animazione che rappresenta il tipo di prenotazione più "complesso": necessita di nome, data, numero dei bambini e il tipo di animazione

che si desidera (a scelta tra 3 opzioni). Questa classe estende la classe Catering e ne eredita la variabile *numeroBambini* e il suo metodo get.

3. Gestione menu e Strutture dati

Inizialmente abbiamo creato una singola interfaccia grafica, rappresentata dal menu principale: un totale di sei tasti che permettono di accedere alle funzioni principali del programma. Da qui si può aggiungere o eliminare una prenotazione, visualizzare le prenotazioni per tipo e in ordine di data, cercarle tramite data o nominativo, salvare su file binario gli appuntamenti salvati o caricarli da un file binario (precedentemente scaricato).

Immaginando che potesse essere comodo invece poter caricare un file precedentemente salvato contenente un elenco di prenotazioni, abbiamo creato un primo menu che apparirà solo una volta nell'esecuzione del programma, più piccolo e con sole tre funzioni: aggiungere una prenotazione, caricare un file, uscire dal programma.

Le prenotazioni, una volta create, andranno a salvarsi in base al loro "tipo" in un ArrayList predisposta e inizializzata all'inizio del main. C'è un ArrayList per ognuno dei tipi: uno per gli affitti semplici, uno per l'affitto con catering e uno per l'affitto con animazione (che comprende il catering). Abbiamo deciso di dividere le prenotazioni per tipo per una gestione più semplice delle stesse. Ad esempio, questo è servito nel metodo che visualizza le prenotazioni nel momento in cui bisogna scegliere quali tipi vuoi visualizzare, in modo da iterare una sola delle liste o unirle al bisogno.

Ritenendo che fosse più semplice e efficiente, abbiamo scelto la struttura dati Hashmap per la gestione della data, usando come chiave la data stessa e come valore la prenotazione. La gestione della data è stata usata: per guidare l'utente nell'inserimento della prenotazione visualizzando la prima data disponibile, per consentire la ricerca di prenotazioni risalendo dalla data della prenotazione e per visualizzare l'elenco delle prenotazioni in ordine di data. Per comparare le date e averle in ordine cronologico, è stato necessario fare un override del metodo *compareTo* nella superclasse Prenotazione. Inoltre, per avere una visualizzazione delle date più omogenea e pulita, abbiamo deciso di formattare la data utilizzando la classe SimpleDateFormat (*java.text.SimpleDateFormat*) nel seguente formato dd/mm/yyyy.