Final specifications for "Book it" program
Software Verification and Validation
Group 8 - MUSS

Name:

Book it - Book rooms in a hotel and identify those staying.

Usage:

Run the script on cmd.

bookIt <input-file>

Description:

Book it is a program that permit to the users request the reservation of a room in a hotel and the administration review all this requests. This program accepts a list of command previously defined in input file where the keys are the name of the command, the definition and the list of parameters whom have different types and lengths depending on its function. These commands must be read and processed in order to achieves the system running correctly.

It is important to mention that the input file is required.

The hotel that is managed using this system, has 4 types of rooms:

- Luxury rooms
- Gold rooms
- Silver rooms
- Platinum rooms

Above classification is based on the cost of the reservation for each room. The luxury is the most expensive and platinum the cheapest one. For each room, there are 2 types of accommodations:

- Simple bed
- Double bed

There are 8 rooms of each type. By last, there are 2 possible more characteristics to select among:

- Rooms with balcony
- Rooms without balcony

The next table show the amount of rooms for each type of accommodation.

	Simple bed with balcony	Simple bed without balcony	Double bed with balcony	Double bed without balcony
Luxury room	1	1	4	2
Gold room	1	1	3	3
Silver room	2	2	2	2
Platinum room	2	2	2	2

The clients can request booking a room depending on the hotel availability and then define all the above required specifications. The booking only lasts one day. So first at all, the user must to define the date for his/her reservation. At which time, he/she receives a list with the type of room and how many of its are available. After that, the user can select one type of room and accommodation. Then, the list with the rooms with balcony or not is shown and finally the user can select one and done her/his booking. A message of success is shown.

The hotel manager is the only one who can see a complete list of all reservations and the information of the clients. Also, the manager can cancel a specific reservation or all of its based on their criterion. Despite of the application does not use a login, each client that do a reservation have to provide some personal information and the manager have to use a specific secret key for identify herself in the program.

Options:

- **guide**
  No argument needed. Just to show a brief description of the system functionality and how to use the commands.

- **description <argument>**
  It is a helpful text for the client about all the type of rooms or one in specific.

  The *argument* is l for luxury, g for gold, s for silver, p for platinum or g for general (all of the above).

- **availability <argument1>**
  The clients can check the availability of rooms in the hotel for a specific month by running this command.
  The *argument1* is the month of his/her interest identified by the name: January, …, December.
  The program has to respond with a user friendly content (the form that you decided, an array matriz, json file, object, etc) that shows the days with how may rooms are available. Important: Only show the remaining days to end the month, for example: if you consult October on day 28, the program will show you the information from that day to the last one. But if you consult December, the program will show you all the days.

- **check <argument1> <argument2>**
  The client can check the availability of a room for a specific date.

  The *argument1* is the date in the form dd-mm-aaaa in numerical values:
  dd: 00 to 31
  mm: 01 to 12
  aaaa: from 2020
  The *argument2* is the type of room to book, again l for luxury, g for gold, s for silver, p for platinum or g for general.
  The command will respond with a list of the available rooms for the type specified in the argument2. If argument1 was g, the list contents all the available rooms regardless its type.

- **book <argument1> <argument2> <argument3> <argument4> <argument5>**
  The client can do the booking by running this command.

  The *argument1* is the date in the same format that above.
  The *argument2* is the type of room: l,g,s or p (for luxury, gold, silver or platinium).
  The *argument3* is the type of accommodation: sm for simple bed or db for double.
  The *argument4* is the name and last name of the person in the format: name_lastname.
  The *argument5* is the ID number without special characters, only numbers are accepted.
  If the user does not provide the argument4, the program will show an error message and will suggest run again the command completing the information. This booking will not consider. If everything is correct, the system will show a success message and print a code for identify the reservation.

- **preference <argument1> <argument2>**
  The client can select if want a room with balcony or not. If the client does not define this, this will assign by default randomly based on available rooms for the selected date.

  The *argument1* is the code of reservation provided in the book command respond.
  The *argument2* is y for room with balcony or n for room without balcony.
  If there is not available the select type, this will display a message informing that the selection is not possible and the available type will be assigned.

- **mybookings <argument1> <argument2>**
  The client can to see a list of all his/her bookings.

  The *argument1* is name_lastname
  The *argument2* is ID number without special characters, only numbers are accepted.
  then the program has to find all the bookings that match with this name and ID number, is there are repeated names, the ID number has the priority. If there are not bookings for that client, a message has to be shown.
  The program will respond with a list of the date and code for each booking.

- **cancelmybooking <argument1> <argument2> <argument3>**
  The client can to cancel any of his/her bookings.

  The *argument1* is name_lastname.
  The *argument2* is ID number without special characters, only numbers are accepted.
  The *argument3* is the code of the booking that will be canceled.
  If there are not bookings for that client, a message has to be shown. If the booking code does not match with any of the booking for this client, an error message has to be shown.

- **list <argument>**
  This command is for the hotel administrator. It will show a list of the bookings in the hotel. The list has to show the code reservation, date, type of room, accommodation, balcony or not and the name of the client for each booking.
  The *argument* is the private key previous provide by the programmer only to the administrator.

- **cancel <argument>**
  The administrator can to cancel any book based on his own criterion.
  The *argument* is the code reservation that he wants to cancel or the word all for cancel all the reservations listed.

Input Data:

For run each command, all arguments are separate one from other and from the command name by one single space. Everything have to be written in lower case letters.

The only input data variable is the argument4 for the book command. For this the lower and upper case is not relevant, the important is that the user do not use a space for separate the name from the last name but only an _ symbol.

Limitations:

1. The program must to show an error message always that a command not been used in the correct way.
2. If there are not rooms available, a new book cannot be able to be done.
3. The secret key is only one and it is previously provided by the programmer, it can change only by programmer decision.
4. Once a client run the book command, he/she will not be able to cancel it.
5. Nobody neither the hotel manager can edit the information provided in the book command.
6. The hotel manager only can see the information and cancel bookings.
7. The commands have to be used in the form here explained or the program will not function correctly.