Mod 10 Homework: Restaurant Reservation Management System

You have been hired by a restaurant to create a program that manages reservations. The program should use a priority queue to keep track of customers in order of their reservation time. Each customer has a unique name. The customer also has reservation time, represented as a string in the format of `HH:MM`. For example, 09:25 has a higher priority than 10:30. The time starts from 00:00 until 23:59. The program should have a menu-based interface that allows the user to choose from the six options listed below:

1. `Add a customer to the waitlist`: This option allows the user to add customers to the waiting list. The user should enter the customer name (assume that all names are unique) and the priority that represents the time of the reservation in the form of `HH:MM`. When adding a customer, the program should validate the input to ensure that the time is in the correct format (HH:MM).

2. `Seat the next customer`: This option allows the user to seat a customer. The program should extract the customer with the highest priority (i.e., the earliest reservation time) from the priority queue.

3. `Change Reservation Time`: This option allows the user to change the reservation time for a specific customer. The user should enter the customer name and the new priority (reservation time). The program should update the priority queue based on the new priority.

4. `Peek at the next customer`: This option allows the user to peek and see the first customer in the waitlist (i.e., the customer with the highest priority). The program should handle cases where there are no customers in the queue when trying to seat or peek at the first customer.

5. `Print the reservation list` print the reservation list in order of priority.

6. The program should be menu-driven, and the user should be prompted to enter their choice of action after each operation is completed. The user should also be able to `exit` the program at any time. This option is already given in the starter code.

Submitting

You should submit the following files:

1. ***`waitlist.py`***: This file should contain the implementation of the `WaitList` class with the five methods: `add_customer()`, `seat_customer()`, `change_reservation_time()`, `peek_first_customer()`, and `print_reservation_list()`. In addition to other methods that you may need.

2. ***`menu.py`***: This file should contain the implementation of the menu-based interface, with the five previous options and sixth option to exit The program when the user chooses to quit.

3. ***`test_waitlist.py`***: This file should contain the unit tests for the `WaitList` class.

Students must submit individually by the due date (typically Tuesday at 11:59 pm EST) to receive credit.

Requirements

- You should implement the menu-based interface to allow the user to choose from the six options.
- You should implement the first five options as separate methods in a WaitList class.
- The program should handle cases where multiple customers have the same reservation time by using their names as a tiebreaker. In other words, if two customers have the same reservation time, the one whose name comes first alphabetically should have a higher priority in the queue. This code is already implemented in the starter code in `Time` and 'Entry` classes `waitlist.py`.
- You should implement unit tests to ensure that the program works as expected.
- You should provide a separate menu file that contains the menu options for the user to choose from. Submission

Examples

Here's an example of how the program should work:

```
Welcome to the Restaurant Reservation System!
==============================================
Please select an option:
1. Add a customer to the waitlist
2. Seat the next customer
3. Change the time of a customer's reservation
4. Peek at the next customer
5. Print the reservation list
6. Quit

Enter your choice (1-5): 1
*************************************************
Enter the customer's name: Ana
Enter the time of the reservation (HH:MM): 12:30

Ana has been added to the waitlist at 12:30

Enter your choice (1-5): 1
*************************************************
Enter the customer's name: Mike
Enter the time of the reservation (HH:MM): 18:45

Mike has been added to the waitlist at 18:45

Enter your choice (1-5): 1
*************************************************
Enter the customer's name: James
Enter the time of the reservation (HH:MM): 02:00

James has been added to the waitlist at 02:00

Enter your choice (1-5): 4
*************************************************

The next customer on the waitlist is: James, reservation time: 02:00

Enter your choice (1-5): 5
```

```
*************************************************

__________________________________________________
The next customer on the waitlist is: James, time: 02:00
The next customer on the waitlist is: Ana, time: 12:30
The next customer on the waitlist is: Mike, time: 18:45


__________________________________________________
Enter your choice (1-5): 2
*************************************************

Seated customer: James, reservation time: 02:00

Enter your choice (1-5): 2
*************************************************

Seated customer: Ana, reservation time: 12:30

Enter your choice (1-5): 3
*************************************************
Enter the customer's name: Mike
Enter the new time of the reservation (HH:MM): 10:23

Mike's reservation time has been changed to 10:23


Enter your choice (1-5): 6
*************************************************
Thank you for using the Restaurant Reservation System!
```