Restaurant Management System

This is a Python-based Restaurant Management System that helps manage the restaurant's operations such as:

- Menu management (adding items, displaying and sorting the menu)

- Reservation handling (making and processing reservations)

- Order management (placing orders, displaying orders, sorting orders)

- Customer feedback (adding and viewing feedback)

The system allows users to interact with different functionalities through a menu-based interface.

Features

1. Add Menu Item: Allows you to add new items to the restaurant's menu with a name and price.

2. Display Menu: Displays the list of all menu items available.

3. Sort Menu by Price: Sorts the menu items by price using the Insertion Sort algorithm.

4. Make Reservation: Adds a reservation by entering the customer's name.

5. Process Reservation: Processes and removes the first reservation in the queue.

6. Place Order: Allows a customer to place an order from the menu.

7. Display Orders: Displays a list of all customer orders placed.

8. Sort Orders by Customer Name: Sorts orders based on customer name using the Bubble Sort algorithm.

9. Add Feedback: Allows customers to submit feedback.

10. View Feedback: Displays all feedback submitted, with the latest feedback shown first.

How to Use:

1. Run the Python program.

2. From the menu, choose the corresponding number to perform the desired action.

3. Follow the on-screen prompts to provide input (such as adding menu items, making reservations, etc.).

4. The system will give feedback on each action and continue running until you exit.

Prerequisites:

- Python 3.x

- The `collections` module is used for managing the reservations queue (no external libraries are required).

Example Workflow

1. Add a menu item:

- Item name: Pizza

- Price: 12.99

2. Display the menu.

3. Make a reservation:

- Customer name: John Doe

4. Process a reservation (reserves John Doe).

5. Place an order:

- Customer name: John Doe

- Menu item: Pizza

6. View feedback (after adding feedback from customers).

7. Exit the program.

Code Overview

The program uses basic data structures such as:

- \*\*List\*\* (`menu`, `orders`) to store menu items and orders.

- \*\*Deque\*\* (from `collections`) for managing the reservation queue (FIFO).

- \*\*Stack\*\* (feedback\_stack) to store and view customer feedback (LIFO).

Sorting is done using:

- \*\*Insertion Sort\*\* for sorting the menu by price.

- \*\*Bubble Sort\*\* for sorting orders by customer name.

Main Flow

1. The program runs a while loop offering a menu to the user.

2. Based on the user's choice, the relevant function is executed.

3. The system continues until the user chooses to exit.

Exit

To exit the program, select option 11 from the menu.

Enjoy managing your restaurant with this simple system!