# **-FOland Bakery**

## **Background Story**

Nadia has been running FOland Bakery, a beloved neighborhood establishment, for over a decade. Known for their artisanal bread and pastries, FOland Bakery has gained a loyal customer base that grows each year. With the increasing variety of bread products and rising customer demands, Nadia is finding it challenging to manage her inventory using the traditional pen-and-paper method.

The handwritten ledger she currently uses often leads to inventory discrepancies, pricing inconsistencies, and difficulty in tracking which bread varieties are most popular. As FOland Bakery continues to expand its offerings, Nadia realizes she needs a more efficient system to manage her bread products.

Nadia has approached you to develop a computerized Bakery Management System that will help streamline her operations and allow her to focus more on baking and less on administrative tasks.

## **Your Task**

Develop a comprehensive Bakery Management System in C that will help Nadia efficiently organize and manage her bread inventory. The system should implement a doubly linked list data structure to store and manipulate the bread information, maintaining everything in an organized order by unique ID numbers.

## **Functional Requirements**

Nadia needs the system to perform the following operations:

1. **Add New Bread Products**: When FOland introduces a new bread recipe, Nadia should be able to enter its details including a unique ID, name, price, and bread type (e.g., sourdough, ciabatta, baguette).
2. **View All Bread Products**: Nadia should be able to generate a complete list of all bread products in her inventory, displayed in a neat tabular format for easy reference.
3. **Search for Specific Bread**: When customers inquire about a particular bread, Nadia should be able to quickly look it up by ID and access all its details.
4. **Update Bread Information**: When ingredient costs change or recipes are refined, Nadia needs to update the corresponding bread information in the system.
5. **Remove Discontinued Bread**: When FOland stops producing a certain bread variety, Nadia needs to remove it from the system.
6. **Find Bread by Type**: When customers ask about specific types of bread (such as gluten-free or vegan options), Nadia should be able to view all breads of that type.

## **Technical Requirements**

Your solution must include:

1. A Bread structure to store bread information:
   * ID (integer)
   * Name (string, max 50 characters)
   * Price (double)
   * Type (string, max 30 characters)
2. A doubly linked list implementation with nodes that:
   * Store Bread items
   * Maintain sorted order by ID
   * Include both forward and backward links
3. Functions for all the required operations:
   * Insert new bread in sorted order
   * Display all bread products
   * Find bread by ID
   * Update bread information
   * Delete bread from inventory
   * Filter and display bread by type
4. A user-friendly menu interface that:
   * Displays clear options
   * Clears the screen between operations
   * Provides informative feedback messages
   * Pauses for the user to read messages
   * Validates user input
5. Proper memory management to prevent leaks

## **Sample Scenario**

When Nadia creates a new specialty bread:

1. She selects "Add New Bread" from the menu
2. Enters ID: 105
3. Name: "Rosemary Olive Loaf"
4. Price: $6.75
5. Type: "Specialty"

The system adds this bread to the inventory in the correct position based on its ID.

Later, when a customer with dietary restrictions asks about whole grain options, Nadia can select "Display Bread by Type," enter "Whole Grain," and immediately see all whole grain bread options available at FOland Bakery.

Input:

45

One

45

Soft

2

Two

67

Soft

67

Three

67

Hard

3

Four

80

Hard

100

Five

67

Soft