

**Sinan Dumansız / 150119812**

## **CSE225/CSE2025 Data Structures PROJECT #1 E- COMMERCE SYSTEM REPORT**

In this project I tried to implement an e-commerce system in console with using linked list in C language. First of all, we have a menu which we can select options whatever user wants.

The code goes the functions accordingly user selection, all of the menu options inside of do-while loop and switch statements.

```
=====
E- COMMERCE SYSTEM
=====

1. Add a customer
2. Add basket
3. Remove customer
4. List the customers who bought a specific product
5. List the total shopping amounts of each customer
6. Exit
→ Please enter an option from the main menu 1 to 6:
```

### **Read Customer – Read Basket – Read Products**

This function created for reading the txt file for customers-Basket-Products, it reads line by line and then sends to other functions which is puts that data to linked list.

```
void read_product_txt(){
    char p_name[50], p_category[50];

70
71 void read_basket_txt(){
72     int customer_id, basket_id, productID;
73     FILE *basketText = fopen(_Filename: "basket.txt", _Mode: "r");

55
56 void read_customer_txt(){
57     char name[50], surname[50];
58     int id;
```

## Create Customer – Create Basket – Create Products

In these functions basically we take the data from read functions and then put all of them to linked lists. For Customers it goes one by one. There are two conditions for the customer linked list, first one is if the linked list empty other is if its not empty just iterate and then add elements to end of linked list. For Basket, first we have to find the specific customers to add basket, after finding there are 3 conditions this time. First conditions if the basket list is null other is if we have already created a basket with same id, if we have same basket with same id, we just have to add the products no need to create new one, but if we don't then we have to create a new basket with different id. As we came for the last one which is creating products linked list, its overall same as creating customers. In first we have to check if its null then iterate to last elements and add element to last position.

```
111
112
113 void create_customer( int id, char*name ,char*surname){
114
115
```

```
void create_products(int txt_product_id,char*p_name,char*p_category,int price){
```

```
void create_basket(int customer_id,int basket_id,int product_id){
```

## Add Basket – Add Customer

In this part I tried to add a specific customer and basket to linked lists. First adding to customer. In adding customer first, we have found last person then iterate to add last position, after we find the last position, we have to find the previous customers id and add 1 to it to create our new customers id. For the basket linked list, we just have to call create basket function but as we call this, we don't know the basket id, to find basket id we just have to iterate forward in customers basket linked list then find the last one to put in function.

```
275
276
277 void add_customer(){
278
```

```
void add_basket(){
```

## Remove Customer

Removing something from linked list is always the hardest part because you can remove in the middle or end or the root itself. So there are 3 conditions for the removing. If you want to remove the first one then you have to re-direct to root itself to second item in list, if you want to delete from the middle, you have to create a temporary point to save next value so we don't lose the '.next' pointer and after we show our '.next' pointers to temporary node we can safely delete the item in linked list.

```
13
14 void remove_customer(){
15
```

## Find Product name - Find Price of Item – Find Category

This function for the finding names prices and category for the basket items. Simply it iterates in Products linked lists and then finds the target with product id. then returns the value.

```
157
158
159 const char * find_product_name(int product_id){
160
```

```
174
175 const char * find_product_category(int product_id){
176     Product * p_iter = product_root;
```

```
143
144 int find_price_of_product(int product_id){
145
```

## Printing Specific Products – Print All Amounts

Lastly, we have functions to print out some values to the console. First one is for the printing the list of customers who bought the selected product. First, I took the product id and iterate in customer basket linked list if code finds a match it prints out the values in the console. Second one is the printing all the amount specific customer has made and sum them together and finally print out the sum values in the console.

```
void print_specific_product(){
512 void print_all_amount(){
```

Example Run shots :

## Main menu

```
=====
E- COMMERCE SYSTEM
=====

1. Add a customer
2. Add basket
3. Remove customer
4. List the customers who bought a specific product
5. List the total shopping amounts of each customer
6. Exit
→ Please enter an option from the main menu 1 to 6:
```

## Adding user

```
Enter name:Sinan
Enter surname:Dumansiz
Customer with Sinan Dumansiz name-surname added to list.

ID  Name  Surname
--  ---  -
1  Ayhan  Altan
2  Bora   Cakir
3  Cenker  Saglam
4  Engin   Altan
5  Guler   Sevimli
6  Mustafa Acar
7  Temel   Aktas
8  Yagmur  Ozden
9  Sinan   Dumansiz
```

## Deleting user

```
Enter name:Ayhan
Enter surname:Altan
Customer with Ayhan Altan name-surname deleted from list.

ID  Name  Surname
--  ---  -
2  Bora   Cakir
3  Cenker  Saglam
4  Engin   Altan
5  Guler   Sevimli
6  Mustafa Acar
7  Temel   Aktas
8  Yagmur  Ozden
9  Sinan   Dumansiz
```

## Adding basket

```
25 Soap hygiene 6
26 Shampoo hygiene 15
27 Deodorant hygiene 12

Enter a product id (to complete shopping press '-1'):3
→ Product id with 3 added to basket id: 3 for Bora Cakir.
Enter a product id (to complete shopping press '-1'):4
→ Product id with 4 added to basket id: 3 for Bora Cakir.
Enter a product id (to complete shopping press '-1'):5
→ Product id with 5 added to basket id: 3 for Bora Cakir.
Enter a product id (to complete shopping press '-1'):6
→ Product id with 6 added to basket id: 3 for Bora Cakir.
Enter a product id (to complete shopping press '-1'):-1
→ Shopping completed for Bora Cakir. Have a nice day!
```

## Printing the user who bought the same product

```
26 Shampoo hygiene 15
27 Deodorant hygiene 12

Enter product id to print specific buyer list:27
→ Customer Bora Cakir bought product id: 27 in his basket id: 2
→ Customer Temel Aktas bought product id: 27 in his basket id: 2
```

**Sinan Dumansız / 150119812**