UNIVERSITI MALAYSIA PAHANG AL-SULTAN ABDULLAH

BCI1093 DATA STRUCTURE & ALGORITHMS

TITLE: AUGMENTED REALITY (AR) SHOPPING EXPERIENCE PLATFORM

LECTURER’S NAME: DR. SITI SALWANI

SEMESTER 1 2023/2024

MATRIC NO	NAME	SECTION
SD22019	NUR SABIHAH BINTI ANUAR	01G
SD22035	NURUL NAJWA BINTI NORHISHAM	01G
SD22012	PUTRI BALQIS BATRISYIA BINTI MOHD RIZAL	02G
SD22017	BATRISYIA BINTI ISMAIL	01G
SD22007	MIZA SYAZWANA BINTI MOHD SAFIAN	01G

TABLE OF CONTENTS

CASE STUDY 1 : WARBY PARKER

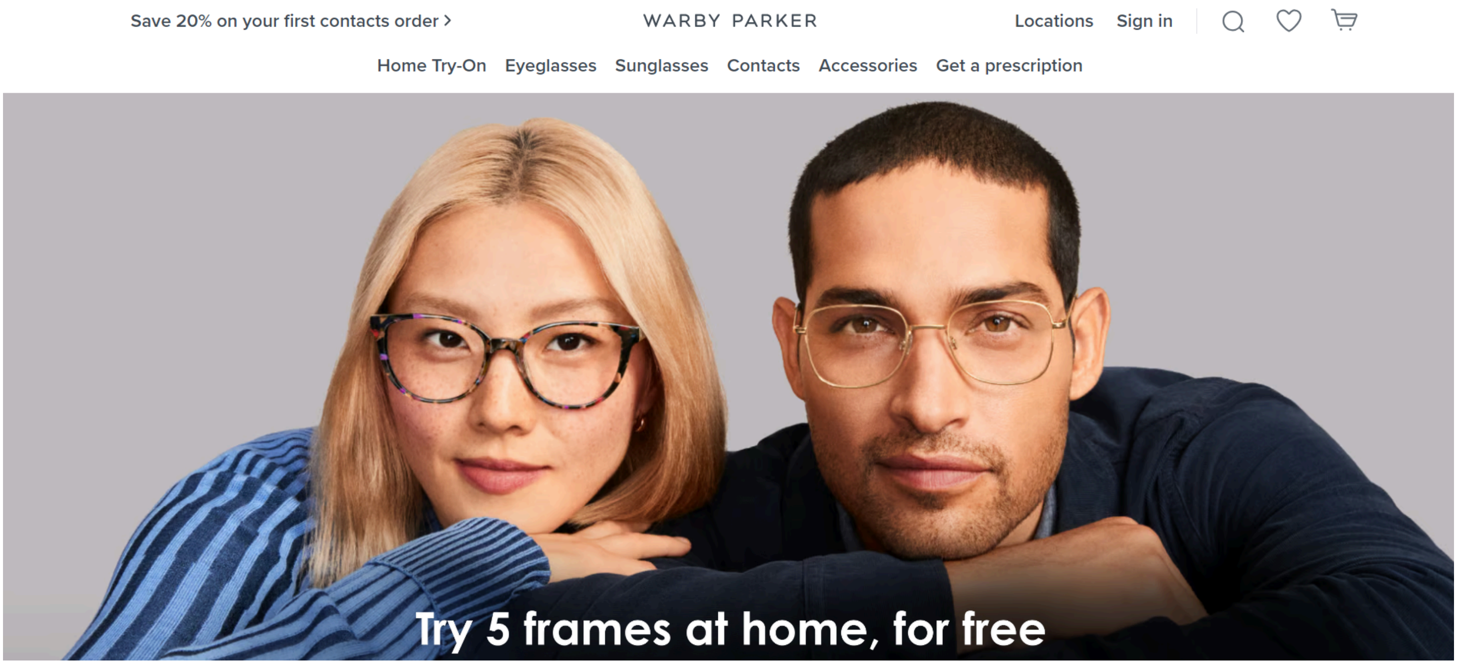


Figure 1: Warby Parker Platform

Warby Parker is an American eyeglasses and sunglasses company that was operating online exclusively in 2010 and opened its first store in 2013. This company has several retail outlets located in cities in different states in the United States and Canada.Warby Parker's company offers blue light lenses, eye drops, sunglasses, eyeglasses, and other associated accessories for both men and women. It also sells frames and prescription glasses. This company was founded with the goal of offering stylish and affordable eyewear while maintaining a customer-friendly business model.

Warby Parker's website provides many options to help the customer to find the perfect frame online. It allows the customers to virtually try on eyeglasses or sunglasses through a smartphone or computer camera anytime and anywhere. This helps the customer in making decisions by enabling them to see how different frames will appear on their face. In addition, customers also can explore details of frames, such as colour options, materials, and design features, in a three-dimensional space before making a purchase. The capability enhances customer confidence and their likelihood of purchasing a specific frame. With their at-home try-on programme, the customer may select up to five frames and have five days to determine whether they like them or not before sending back all five pairs. This approach makes the eyewear purchasing process more convenient.

Besides, not only was Warby Parker founded by its founders in an attempt to address a real-world issue, but its unique setup also makes it impossible to overlook. It is among the first direct-to-consumer companies that is vertically integrated and aims to make a good impression on consumers. (Daniel Pereira, 2023, April 14)

The advantages of Warby Parker :

1. Affordable price on all eyewear.
2. Over 100 locations are available for eye tests.

The disadvantages of Warby Parker :

1. Warby Parker does provide a virtual try-on option, but it only functions on iPhone X and higher and also it needs to download the app first.
2. No discounts for students, military members, or healthcare professionals.
3. All high index lenses are not created equal.
4. The quality of Warby Parker's high index lenses might not be sufficient for those with strong prescriptions.

https://www.warbyparker.com/

CASE STUDY 2 : IKEA

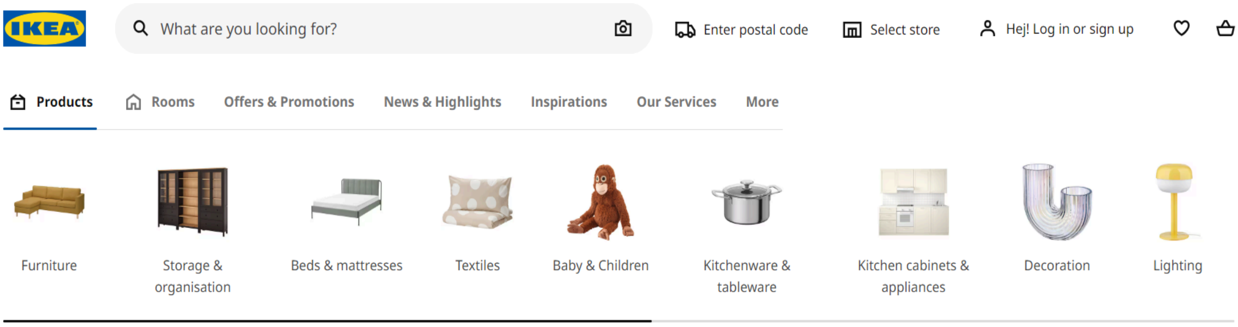


Figure 2: Ikea Platform

The popular Swedish furniture company IKEA is well-known for producing reasonably priced and fashionable home furnishings. IKEA has embraced the digital age with a dedication to innovation, launching an intuitive app that improves its customers' overall shopping experiences. With the aid of augmented reality technology, users can easily explore and visualize furniture and decor items in their own living spaces through the app, which acts as a virtual gateway to the extensive IKEA catalogue. This advanced feature facilitates customers' decision-making and adds a level of convenience to the furniture-buying experience.

Beyond merely providing virtual visualization, the IKEA app offers useful features like customized shopping lists, real-time stock availability, and integrated checkout options. Through the app, users can schedule their orders, keep track of their purchases, and take advantage of special offers. IKEA's ability to effortlessly incorporate technology into the conventional in-store shopping experience is evidence of their flexibility in responding to shifting consumer demands. IKEA remains a leader in the furniture retail sector, meeting the varied needs of consumers worldwide by fusing affordability, creative design, and an easy-to-use app.

The advantage of Ikea:

1. Offers a user-friendly interface.

2. Provides real-time information on product availability.

3. Allows customers to monitor their orders, giving them status updates on delivery and guaranteeing transparency all the way through the purchasing process.

The disadvantages of Ikea:

1. Inability to examine furniture in person before buying.

2. Affecting the general user experience through bugs, malfunctions, or incompatibility issues with devices

CASE STUDY 3 : CONVERSE AR TRY-ON

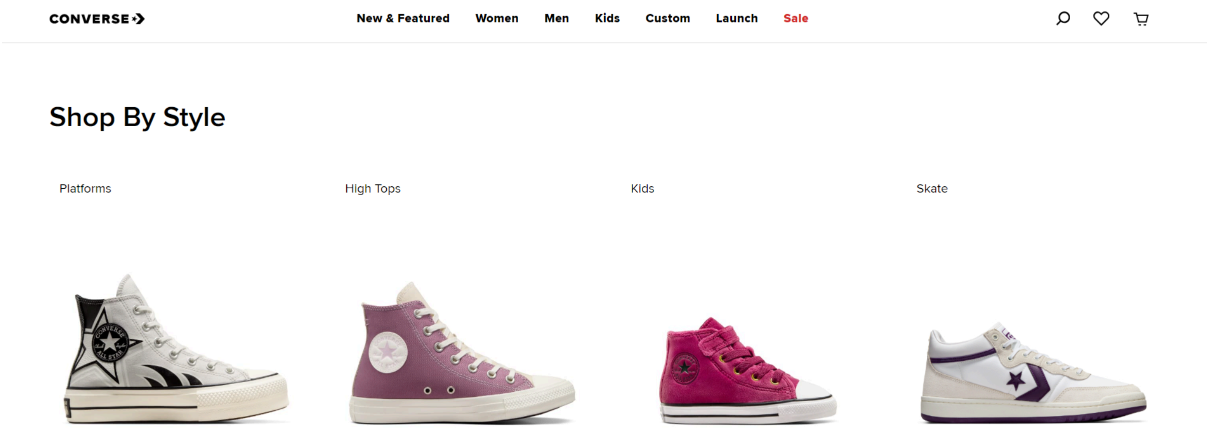


Figure 3: Converse Platform

The Converse AR Try-On app revolutionizes the online shopping experience by integrating augmented reality (AR) technology to allow users to virtually try on various styles of Converse shoes. Customers can use this advanced app to view Converse sneakers on their feet in various models and colours before making a purchase. The application creates a realistic and interactive preview by using the device's camera to superimpose the chosen footwear onto the user's feet in real-time. This adds a playful and interactive element to the online shopping experience, while also boosting the user's confidence in their selection.

In addition to being a useful way to get around the difficulties associated with online shoe shopping, the app's virtual try-on feature also fits with Converse's brand image of individuality and creativity. Users are able to try on various styles until they discover the ideal pair that suits their own sense of style and taste. Converse AR Try-On also satisfies the modern consumer's need for a smooth and seamless integration of technology and retail by developing an engaging and user-friendly platform that connects online and offline shopping. By doing this, Converse shows that it is dedicated to meeting the changing needs and expectations of its customer base while remaining at the forefront of technological developments in the retail sector.

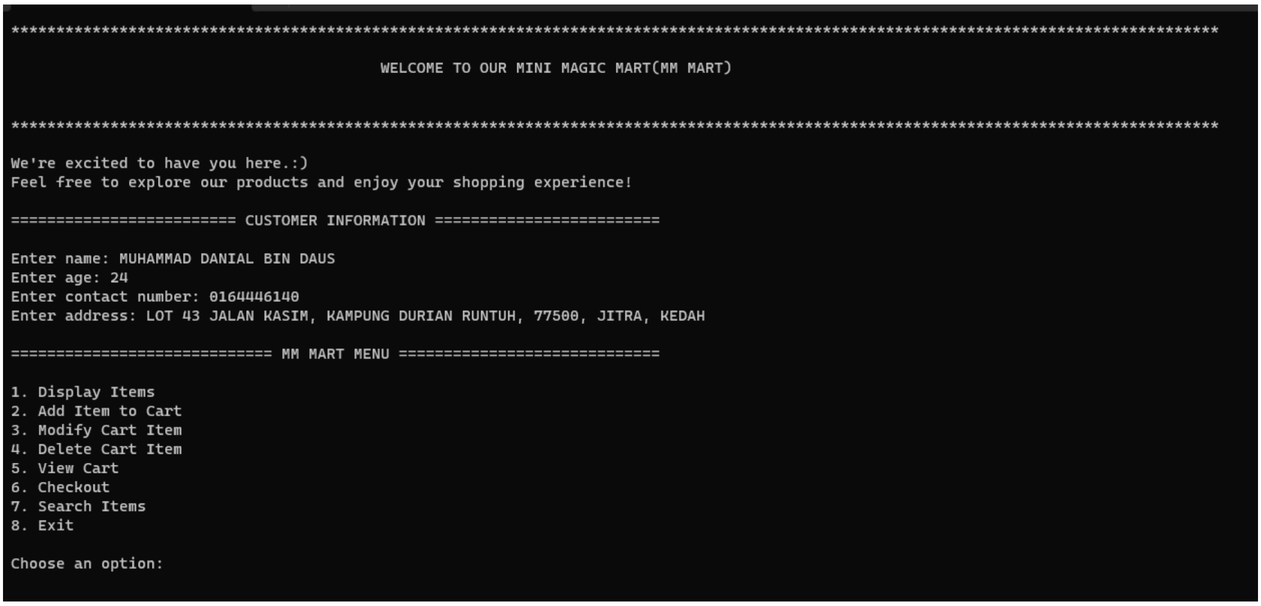
The advantage of Converse AR Try-On:

1. Makes use of the device's camera to produce a realistic and interactive preview.
2. User-friendly platform.

The disadvantage of Converse AR Try-On:

1. Users might not receive an entirely realistic representation of the shoes' appearance in real life.
2. Customers might receive shoes that don't fit their preferred size or level of comfort.

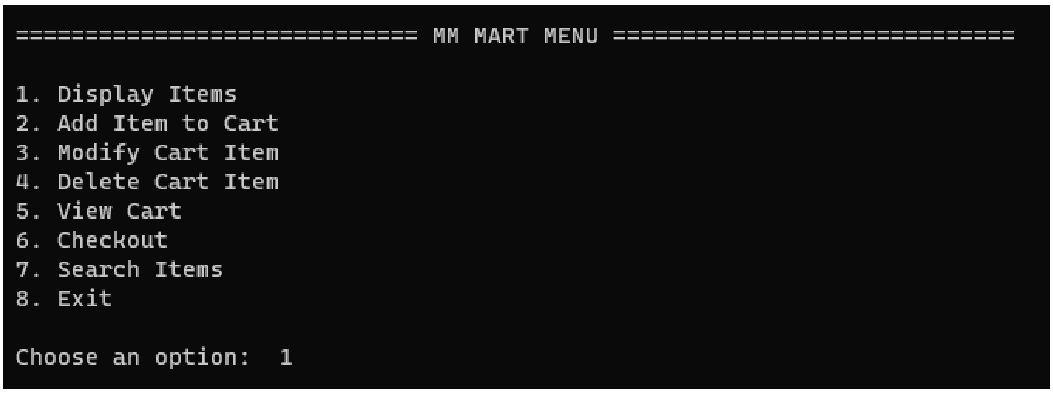
MAIN MENU AND CUSTOMER INFORMATION (AR Shopping Platform)



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

                                         WELCOME TO OUR MINI MAGIC MART(MM MART)


**************************************************************************************************************************************

We're excited to have you here.:)
Feel free to explore our products and enjoy your shopping experience!

========================= CUSTOMER INFORMATION =========================

Enter name: MUHAMMAD DANIAL BIN DAUS
Enter age: 24
Enter contact number: 0164446140
Enter address: LOT 43 JALAN KASIM, KAMPUNG DURIAN RUNTUH, 77500, JITRA, KEDAH

============================= MM MART MENU =============================

1. Display Items
2. Add Item to Cart
3. Modify Cart Item
4. Delete Cart Item
5. View Cart
6. Checkout
7. Search Items
8. Exit

Choose an option:
```

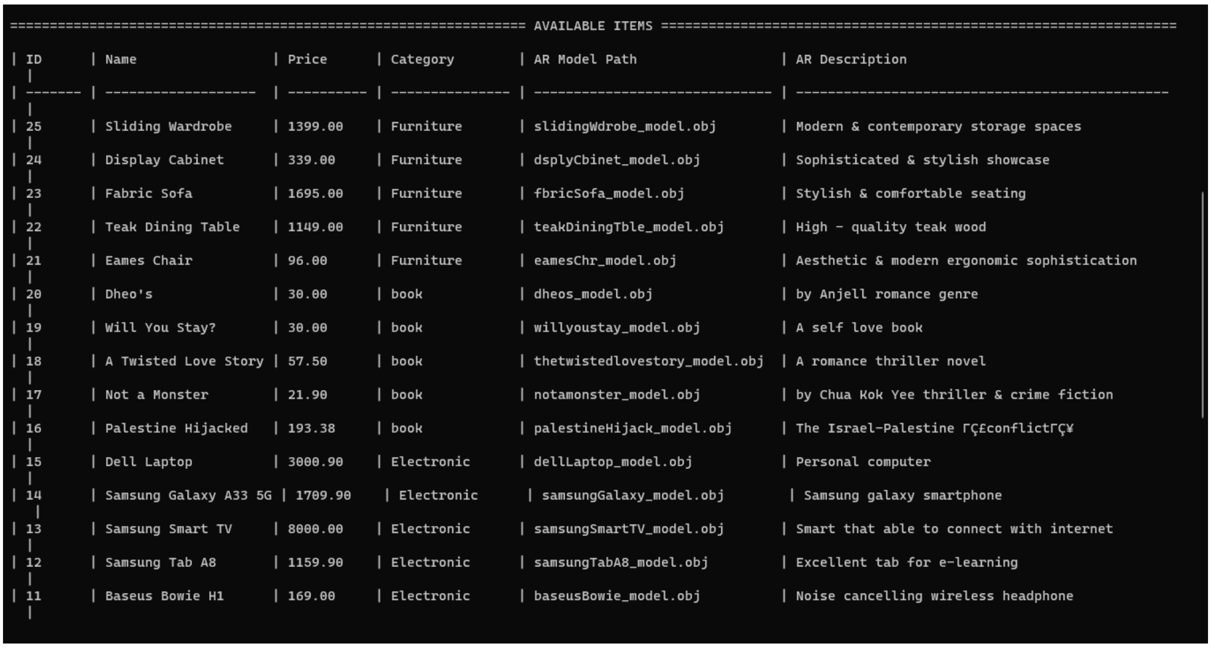
INPUT AND OUTPUT (AR Shopping Platform)

Input (Display Items)

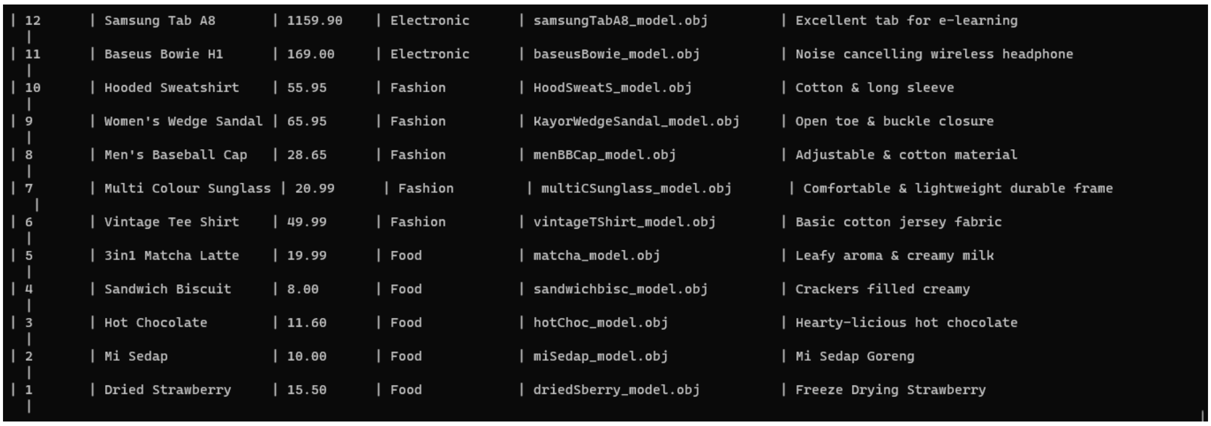


```
============================= MM MART MENU =============================

1. Display Items
2. Add Item to Cart
3. Modify Cart Item
4. Delete Cart Item
5. View Cart
6. Checkout
7. Search Items
8. Exit

Choose an option:  1
```

Output (Display Items)

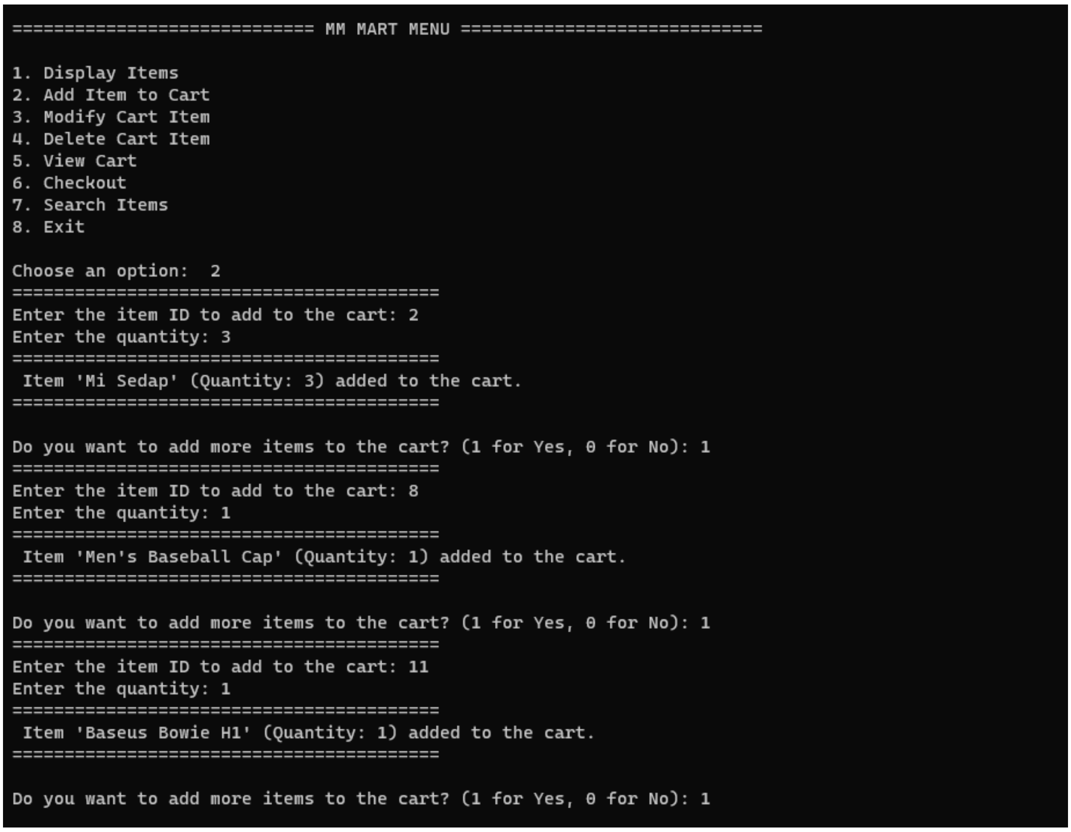


```
================================================================= AVAILABLE ITEMS =================================================================

| ID      | Name               | Price      | Category       | AR Model Path                 | AR Description
  |
| ------- | ------------------ | ---------- | -------------- | ----------------------------- | ----------------------------------------------
  |
| 25      | Sliding Wardrobe   | 1399.00    | Furniture      | slidingWdrobe_model.obj       | Modern & contemporary storage spaces
  |
| 24      | Display Cabinet    | 339.00     | Furniture      | dsplyCbinet_model.obj         | Sophisticated & stylish showcase
  |
| 23      | Fabric Sofa        | 1695.00    | Furniture      | fbricSofa_model.obj           | Stylish & comfortable seating
  |
| 22      | Teak Dining Table  | 1149.00    | Furniture      | teakDiningTble_model.obj      | High - quality teak wood
  |
| 21      | Eames Chair        | 96.00      | Furniture      | eamesChr_model.obj            | Aesthetic & modern ergonomic sophistication
  |
| 20      | Dheo's             | 30.00      | book           | dheos_model.obj               | by Anjell romance genre
  |
| 19      | Will You Stay?     | 30.00      | book           | willyoustay_model.obj         | A self love book
  |
| 18      | A Twisted Love Story | 57.50    | book           | thetwistedlovestory_model.obj | A romance thriller novel
  |
| 17      | Not a Monster      | 21.90      | book           | notamonster_model.obj         | by Chua Kok Yee thriller & crime fiction
  |
| 16      | Palestine Hijacked | 193.38     | book           | palestineHijack_model.obj     | The Israel-Palestine ΓÇ£conflictΓÇ¥
  |
| 15      | Dell Laptop        | 3000.90    | Electronic     | dellLaptop_model.obj          | Personal computer
  |
| 14      | Samsung Galaxy A33 5G | 1709.90    | Electronic     | samsungGalaxy_model.obj       | Samsung galaxy smartphone
   |
| 13      | Samsung Smart TV   | 8000.00    | Electronic     | samsungSmartTV_model.obj      | Smart that able to connect with internet
  |
| 12      | Samsung Tab A8     | 1159.90    | Electronic     | samsungTabA8_model.obj        | Excellent tab for e-learning
  |
| 11      | Baseus Bowie H1    | 169.00     | Electronic     | baseusBowie_model.obj         | Noise cancelling wireless headphone
  |
```

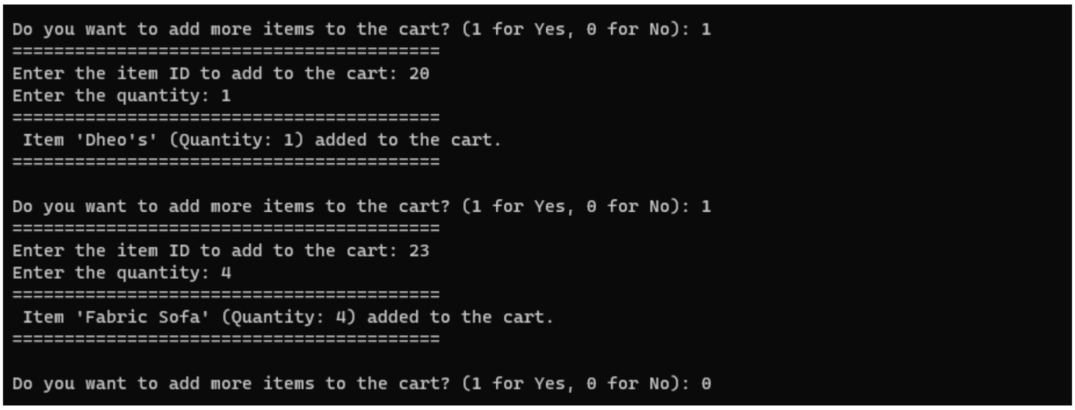


```
| 12      | Samsung Tab A8     | 1159.90    | Electronic     | samsungTabA8_model.obj        | Excellent tab for e-learning
  |
| 11      | Baseus Bowie H1    | 169.00     | Electronic     | baseusBowie_model.obj         | Noise cancelling wireless headphone
  |
| 10      | Hooded Sweatshirt  | 55.95      | Fashion        | HoodSweatS_model.obj          | Cotton & long sleeve
  |
| 9       | Women's Wedge Sandal | 65.95    | Fashion        | KayorWedgeSandal_model.obj    | Open toe & buckle closure
  |
| 8       | Men's Baseball Cap | 28.65      | Fashion        | menBBCap_model.obj            | Adjustable & cotton material
  |
| 7       | Multi Colour Sunglass | 20.99      | Fashion        | multiCSunglass_model.obj      | Comfortable & lightweight durable frame
   |
| 6       | Vintage Tee Shirt  | 49.99      | Fashion        | vintageTShirt_model.obj       | Basic cotton jersey fabric
  |
| 5       | 3in1 Matcha Latte  | 19.99      | Food           | matcha_model.obj              | Leafy aroma & creamy milk
  |
| 4       | Sandwich Biscuit   | 8.00       | Food           | sandwichbisc_model.obj        | Crackers filled creamy
  |
| 3       | Hot Chocolate      | 11.60      | Food           | hotChoc_model.obj             | Hearty-licious hot chocolate
  |
| 2       | Mi Sedap           | 10.00      | Food           | miSedap_model.obj             | Mi Sedap Goreng
  |
| 1       | Dried Strawberry   | 15.50      | Food           | driedSberry_model.obj         | Freeze Drying Strawberry
  |
```

Input (Add Item To Cart)

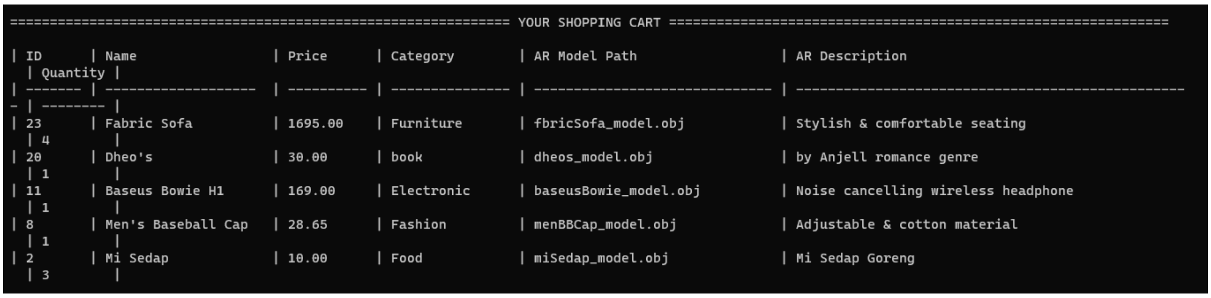


```
============================ MM MART MENU ============================

1. Display Items
2. Add Item to Cart
3. Modify Cart Item
4. Delete Cart Item
5. View Cart
6. Checkout
7. Search Items
8. Exit

Choose an option:  2
=========================================
Enter the item ID to add to the cart: 2
Enter the quantity: 3
=========================================
 Item 'Mi Sedap' (Quantity: 3) added to the cart.
=========================================

Do you want to add more items to the cart? (1 for Yes, 0 for No): 1
=========================================
Enter the item ID to add to the cart: 8
Enter the quantity: 1
=========================================
 Item 'Men's Baseball Cap' (Quantity: 1) added to the cart.
=========================================

Do you want to add more items to the cart? (1 for Yes, 0 for No): 1
=========================================
Enter the item ID to add to the cart: 11
Enter the quantity: 1
=========================================
 Item 'Baseus Bowie H1' (Quantity: 1) added to the cart.
=========================================

Do you want to add more items to the cart? (1 for Yes, 0 for No): 1
```



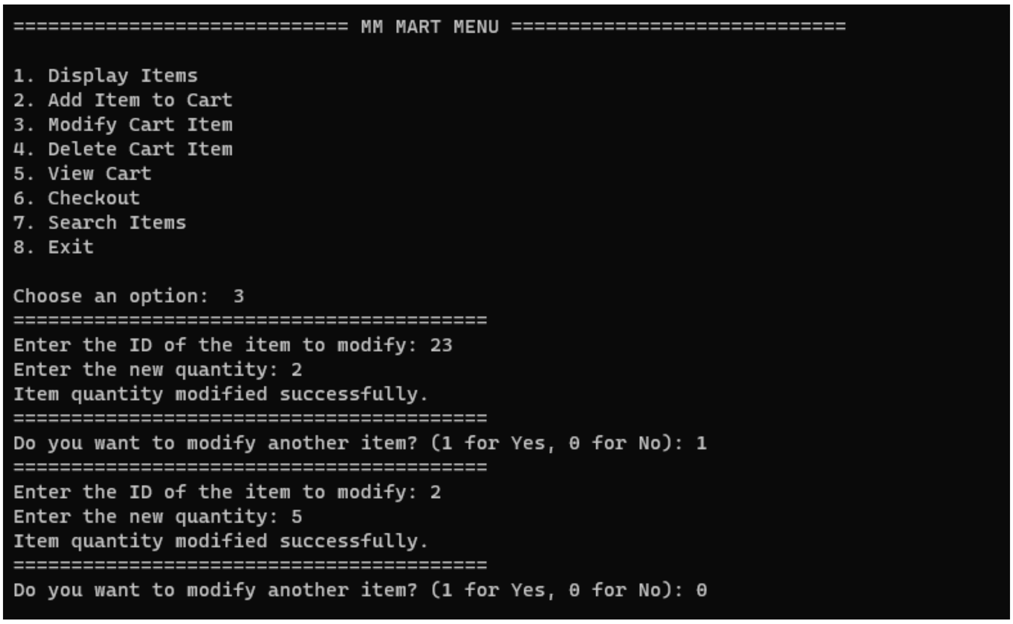
```
Do you want to add more items to the cart? (1 for Yes, 0 for No): 1
=========================================
Enter the item ID to add to the cart: 20
Enter the quantity: 1
=========================================
 Item 'Dheo's' (Quantity: 1) added to the cart.
=========================================

Do you want to add more items to the cart? (1 for Yes, 0 for No): 1
=========================================
Enter the item ID to add to the cart: 23
Enter the quantity: 4
=========================================
 Item 'Fabric Sofa' (Quantity: 4) added to the cart.
=========================================

Do you want to add more items to the cart? (1 for Yes, 0 for No): 0
```

Output (View Cart after add item to cart)



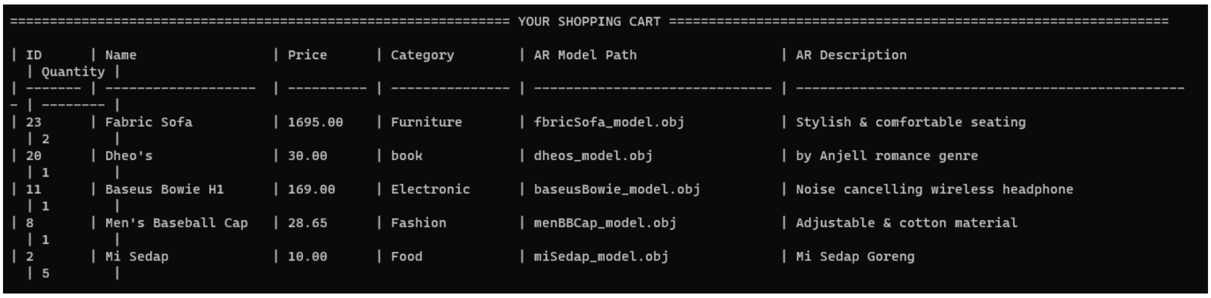
```
=============================================================== YOUR SHOPPING CART ===============================================================

| ID      | Name                  | Price      | Category        | AR Model Path                   | AR Description
  | Quantity |
| ------- | -------------------  | ---------- | --------------- | ------------------------------ | ------------------------------------------------
- | -------- |
| 23      | Fabric Sofa           | 1695.00    | Furniture       | fbricSofa_model.obj             | Stylish & comfortable seating
  | 4        |
| 20      | Dheo's                | 30.00      | book            | dheos_model.obj                 | by Anjell romance genre
  | 1        |
| 11      | Baseus Bowie H1       | 169.00     | Electronic      | baseusBowie_model.obj           | Noise cancelling wireless headphone
  | 1        |
| 8       | Men's Baseball Cap    | 28.65      | Fashion         | menBBCap_model.obj              | Adjustable & cotton material
  | 1        |
| 2       | Mi Sedap              | 10.00      | Food            | miSedap_model.obj               | Mi Sedap Goreng
  | 3        |
```

Input (Modify Cart Item)



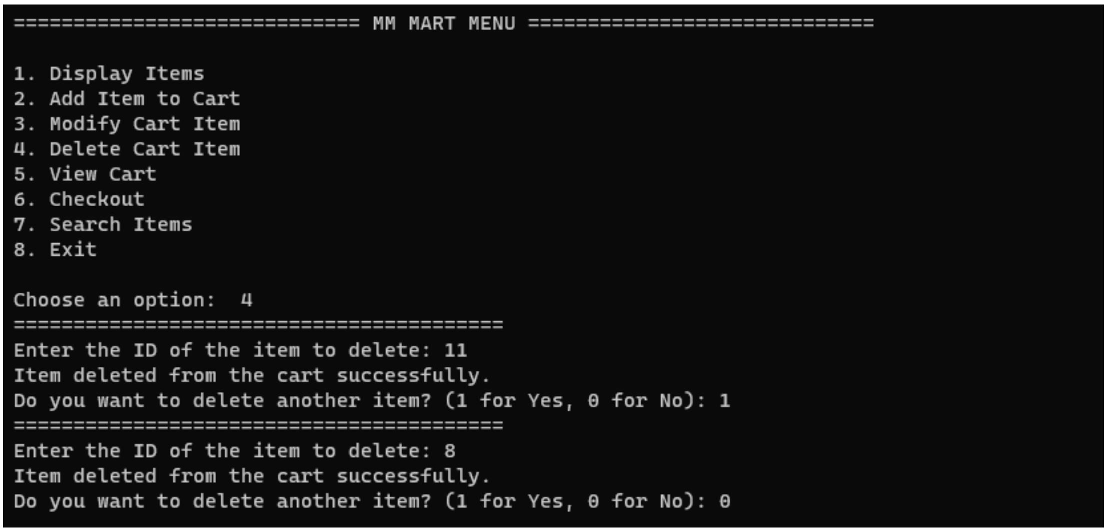
```
============================= MM MART MENU =============================

1. Display Items
2. Add Item to Cart
3. Modify Cart Item
4. Delete Cart Item
5. View Cart
6. Checkout
7. Search Items
8. Exit

Choose an option:  3
=========================================
Enter the ID of the item to modify: 23
Enter the new quantity: 2
Item quantity modified successfully.
=========================================
Do you want to modify another item? (1 for Yes, 0 for No): 1
=========================================
Enter the ID of the item to modify: 2
Enter the new quantity: 5
Item quantity modified successfully.
=========================================
Do you want to modify another item? (1 for Yes, 0 for No): 0
```

Output (View Cart after modify cart item)



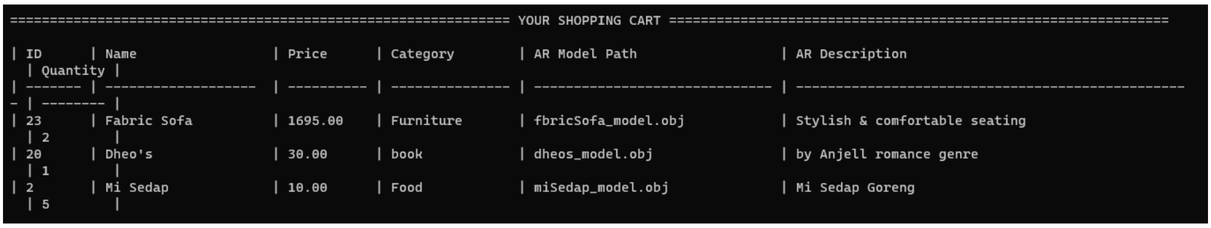
```
============================================================== YOUR SHOPPING CART ==============================================================

| ID      | Name               | Price      | Category       | AR Model Path                | AR Description
  | Quantity |
| ------- | ------------------  | ---------- | -------------- | ---------------------------- | -----------------------------------------------
- | -------- |
| 23      | Fabric Sofa        | 1695.00    | Furniture      | fbricSofa_model.obj          | Stylish & comfortable seating
  | 2        |
| 20      | Dheo's             | 30.00      | book           | dheos_model.obj              | by Anjell romance genre
  | 1        |
| 11      | Baseus Bowie H1    | 169.00     | Electronic     | baseusBowie_model.obj        | Noise cancelling wireless headphone
  | 1        |
| 8       | Men's Baseball Cap | 28.65      | Fashion        | menBBCap_model.obj           | Adjustable & cotton material
  | 1        |
| 2       | Mi Sedap           | 10.00      | Food           | miSedap_model.obj            | Mi Sedap Goreng
  | 5        |
```

Input (Delete Cart Item)



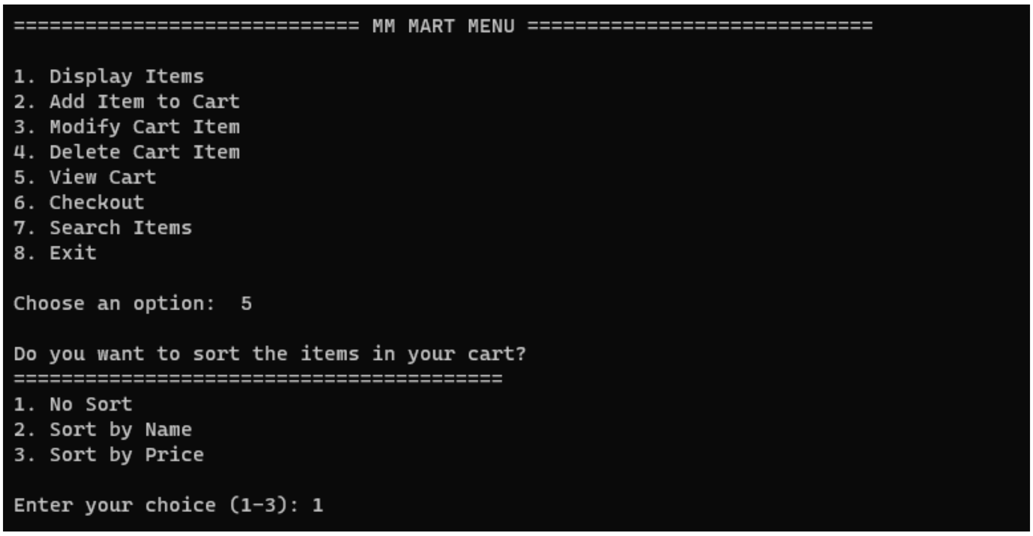
```
============================= MM MART MENU =============================

1. Display Items
2. Add Item to Cart
3. Modify Cart Item
4. Delete Cart Item
5. View Cart
6. Checkout
7. Search Items
8. Exit

Choose an option:  4
=========================================
Enter the ID of the item to delete: 11
Item deleted from the cart successfully.
Do you want to delete another item? (1 for Yes, 0 for No): 1
=========================================
Enter the ID of the item to delete: 8
Item deleted from the cart successfully.
Do you want to delete another item? (1 for Yes, 0 for No): 0
```

Output (View Cart after delete cart item)



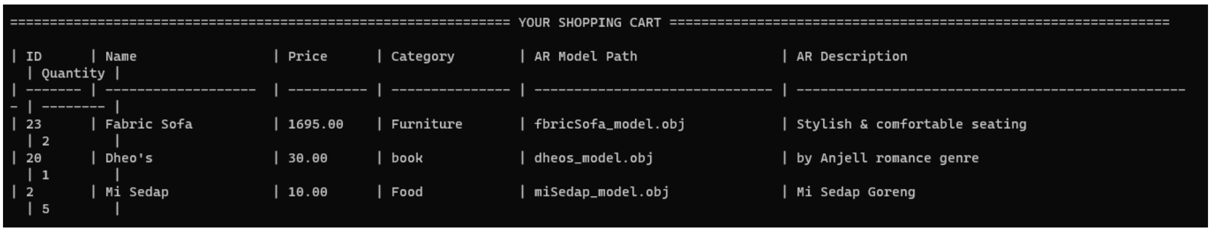
```
============================================================== YOUR SHOPPING CART ==============================================================

| ID      | Name                 | Price      | Category        | AR Model Path                  | AR Description
  | Quantity |
| ------- | -------------------  | ---------- | --------------- | ------------------------------ | -------------------------------------------------
- | -------- |
| 23      | Fabric Sofa          | 1695.00    | Furniture       | fbricSofa_model.obj            | Stylish & comfortable seating
  | 2        |
| 20      | Dheo's               | 30.00      | book            | dheos_model.obj                | by Anjell romance genre
  | 1        |
| 2       | Mi Sedap             | 10.00      | Food            | miSedap_model.obj              | Mi Sedap Goreng
  | 5        |
```

Input (View Cart) - No Sort



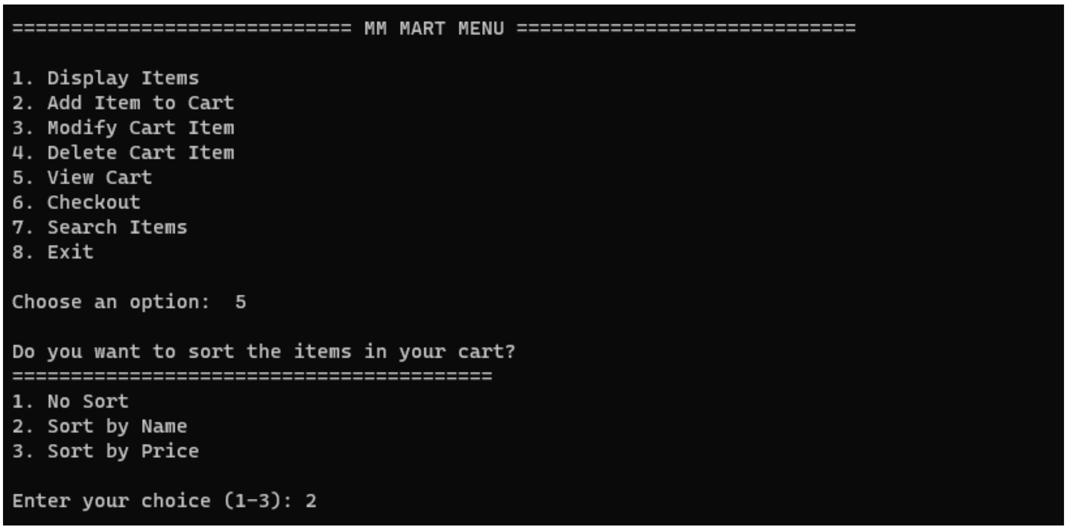
```
============================= MM MART MENU =============================

1. Display Items
2. Add Item to Cart
3. Modify Cart Item
4. Delete Cart Item
5. View Cart
6. Checkout
7. Search Items
8. Exit

Choose an option:  5

Do you want to sort the items in your cart?
=========================================
1. No Sort
2. Sort by Name
3. Sort by Price

Enter your choice (1-3): 1
```

Output (View Chart) - No Sort



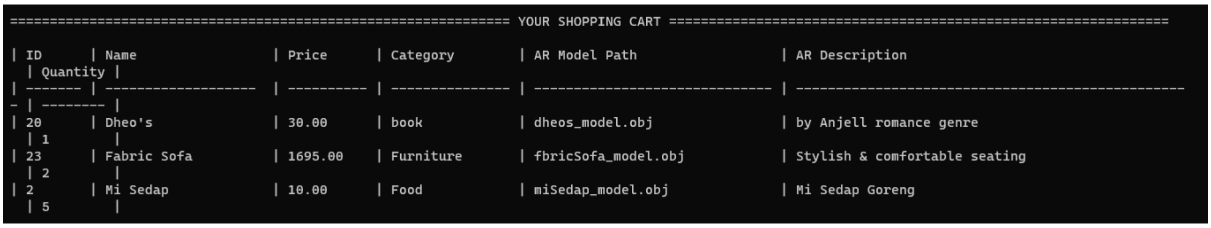
```
============================================================== YOUR SHOPPING CART ==============================================================

| ID      | Name                 | Price      | Category        | AR Model Path                  | AR Description
  | Quantity |
| ------- | -------------------  | ---------- | --------------- | ------------------------------ | -------------------------------------------------
- | -------- |
| 23      | Fabric Sofa          | 1695.00    | Furniture       | fbricSofa_model.obj            | Stylish & comfortable seating
  | 2        |
| 20      | Dheo's               | 30.00      | book            | dheos_model.obj                | by Anjell romance genre
  | 1        |
| 2       | Mi Sedap             | 10.00      | Food            | miSedap_model.obj              | Mi Sedap Goreng
  | 5        |
```

Input (View Cart) - Sort by Name



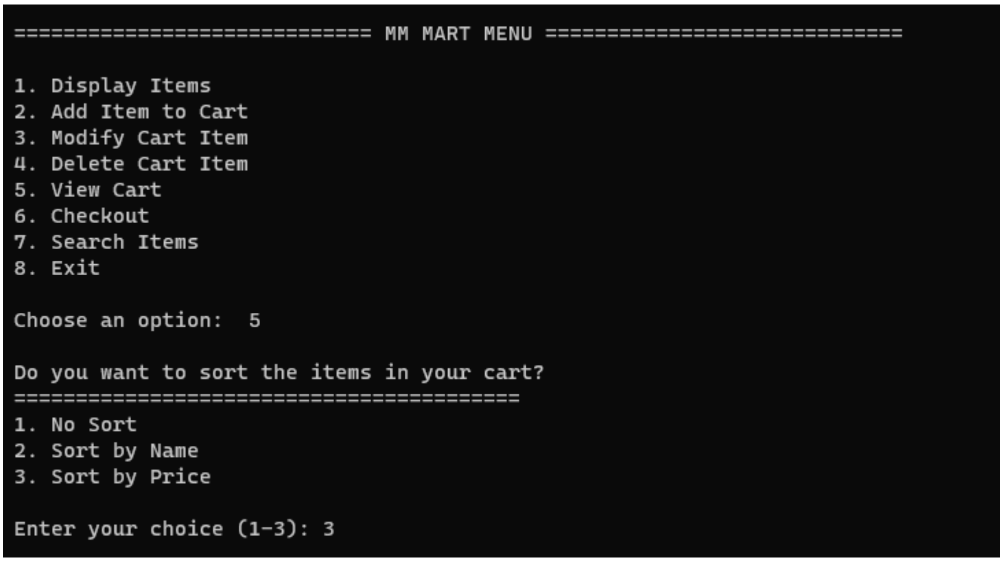
```
============================= MM MART MENU =============================

1. Display Items
2. Add Item to Cart
3. Modify Cart Item
4. Delete Cart Item
5. View Cart
6. Checkout
7. Search Items
8. Exit

Choose an option:  5

Do you want to sort the items in your cart?
=========================================
1. No Sort
2. Sort by Name
3. Sort by Price

Enter your choice (1-3): 2
```

Output (View Chart) - Sort by Name



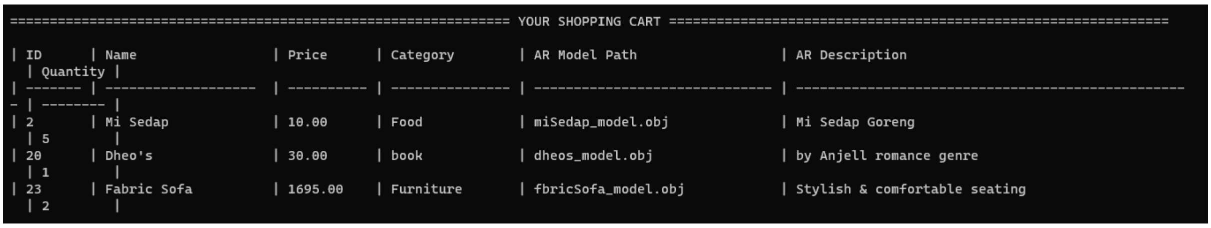
```
=============================================================== YOUR SHOPPING CART ===============================================================

| ID      | Name                | Price      | Category       | AR Model Path                 | AR Description
  | Quantity |
| ------- | ------------------- | ---------- | -------------- | ----------------------------- | -------------------------------------------------
- | -------- |
| 20      | Dheo's              | 30.00      | book           | dheos_model.obj               | by Anjell romance genre
  | 1        |
| 23      | Fabric Sofa         | 1695.00    | Furniture      | fbricSofa_model.obj           | Stylish & comfortable seating
  | 2        |
| 2       | Mi Sedap            | 10.00      | Food           | miSedap_model.obj             | Mi Sedap Goreng
  | 5        |
```

Input (View Cart) - Sort by Price



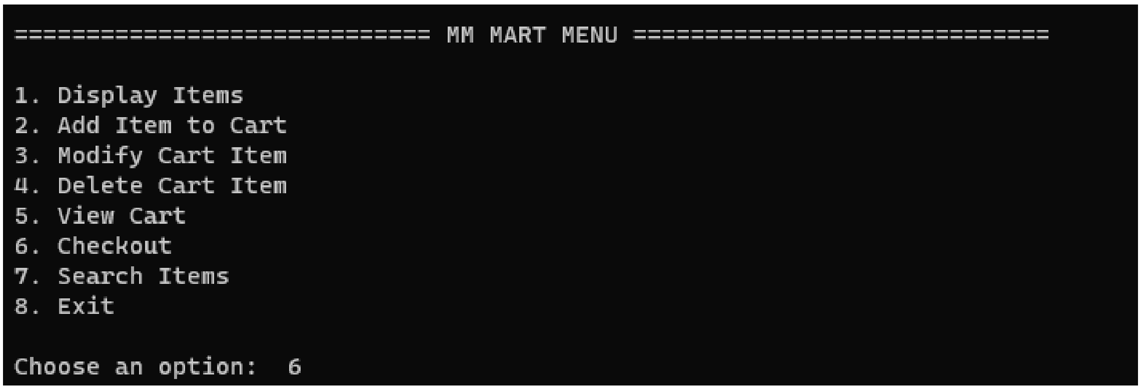
```
============================= MM MART MENU =============================

1. Display Items
2. Add Item to Cart
3. Modify Cart Item
4. Delete Cart Item
5. View Cart
6. Checkout
7. Search Items
8. Exit

Choose an option:  5

Do you want to sort the items in your cart?
=========================================
1. No Sort
2. Sort by Name
3. Sort by Price

Enter your choice (1-3): 3
```

Output (View Chart) - Sort by Price



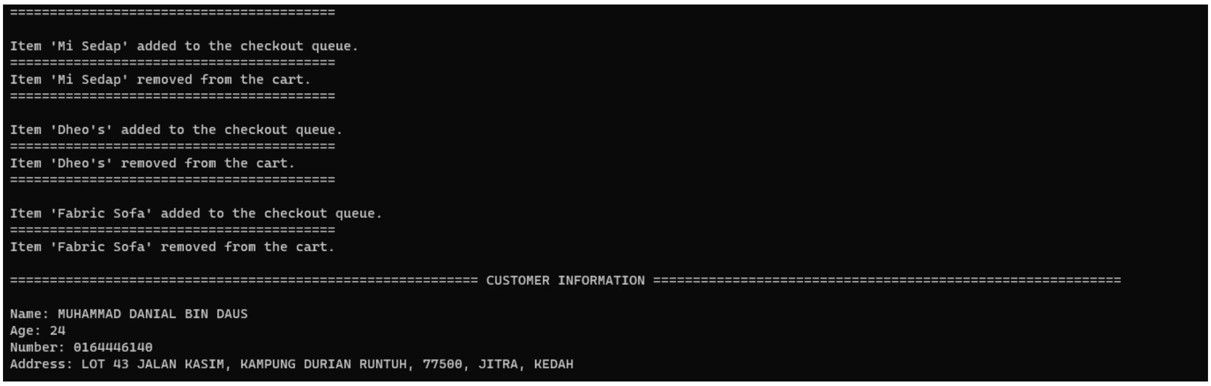
```
============================================================== YOUR SHOPPING CART ==============================================================

| ID      | Name                 | Price      | Category        | AR Model Path                  | AR Description
  | Quantity |
| ------- | -------------------  | ---------- | --------------- | ------------------------------ | -------------------------------------------------
- | -------- |
| 2       | Mi Sedap             | 10.00      | Food            | miSedap_model.obj              | Mi Sedap Goreng
  | 5        |
| 20      | Dheo's               | 30.00      | book            | dheos_model.obj                | by Anjell romance genre
  | 1        |
| 23      | Fabric Sofa          | 1695.00    | Furniture       | fbricSofa_model.obj            | Stylish & comfortable seating
  | 2        |
```

Input (Checkout)

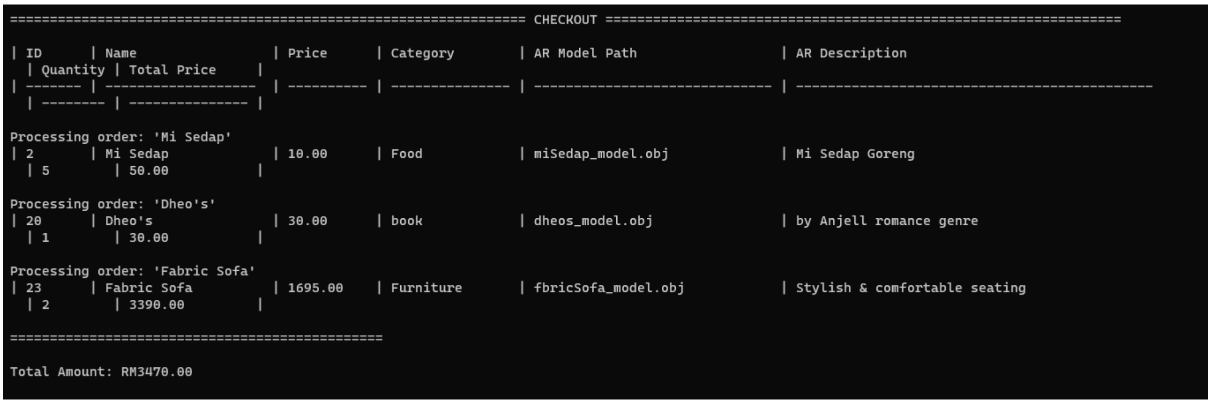


```
============================== MM MART MENU ==============================

1. Display Items
2. Add Item to Cart
3. Modify Cart Item
4. Delete Cart Item
5. View Cart
6. Checkout
7. Search Items
8. Exit

Choose an option:  6
```

Output (Checkout)

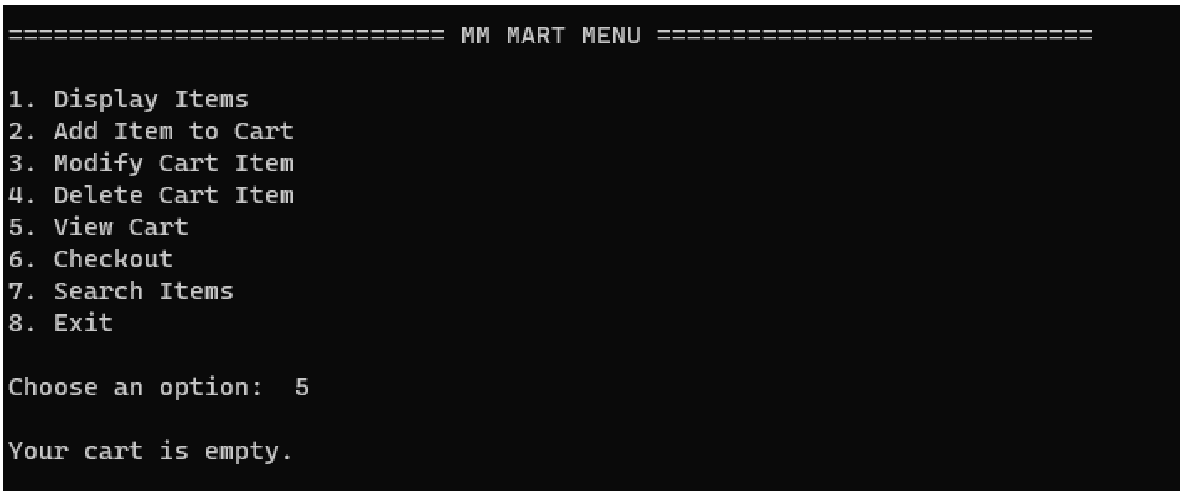


```
=========================================

Item 'Mi Sedap' added to the checkout queue.
=========================================
Item 'Mi Sedap' removed from the cart.
=========================================

Item 'Dheo's' added to the checkout queue.
=========================================
Item 'Dheo's' removed from the cart.
=========================================

Item 'Fabric Sofa' added to the checkout queue.
=========================================
Item 'Fabric Sofa' removed from the cart.

=========================================================== CUSTOMER INFORMATION ===========================================================

Name: MUHAMMAD DANIAL BIN DAUS
Age: 24
Number: 0164446140
Address: LOT 43 JALAN KASIM, KAMPUNG DURIAN RUNTUH, 77500, JITRA, KEDAH
```



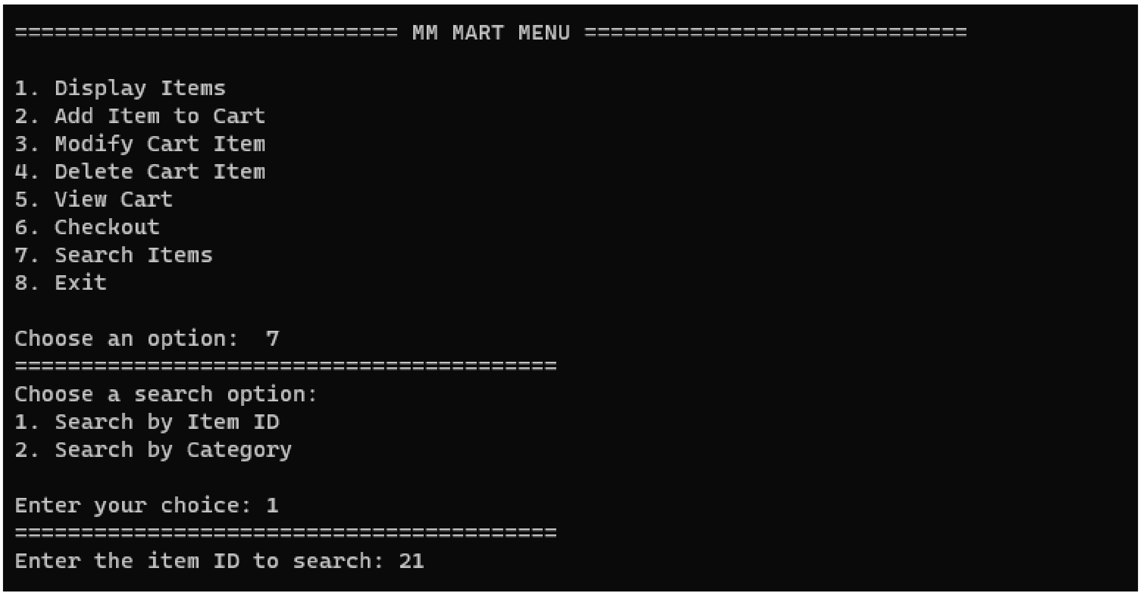
```
================================================================= CHECKOUT =================================================================

| ID      | Name                 | Price      | Category        | AR Model Path                  | AR Description
  | Quantity | Total Price     |
| ------- | -------------------  | ---------- | --------------- | ------------------------------ | ---------------------------------------------
  | -------- | --------------- |

Processing order: 'Mi Sedap'
| 2       | Mi Sedap             | 10.00      | Food            | miSedap_model.obj              | Mi Sedap Goreng
  | 5        | 50.00           |

Processing order: 'Dheo's'
| 20      | Dheo's               | 30.00      | book            | dheos_model.obj                | by Anjell romance genre
  | 1        | 30.00           |

Processing order: 'Fabric Sofa'
| 23      | Fabric Sofa          | 1695.00    | Furniture       | fbricSofa_model.obj            | Stylish & comfortable seating
  | 2        | 3390.00         |

===============================================

Total Amount: RM3470.00
```

Output (View Cart after checkout)

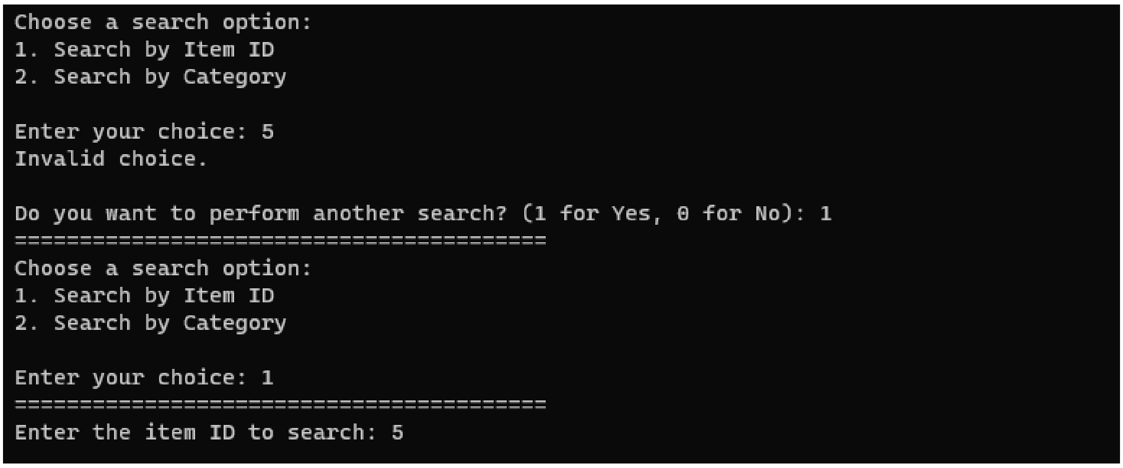


```
============================= MM MART MENU =============================

1. Display Items
2. Add Item to Cart
3. Modify Cart Item
4. Delete Cart Item
5. View Cart
6. Checkout
7. Search Items
8. Exit

Choose an option:  5

Your cart is empty.
```

Input (Search Item) - Search by Item ID

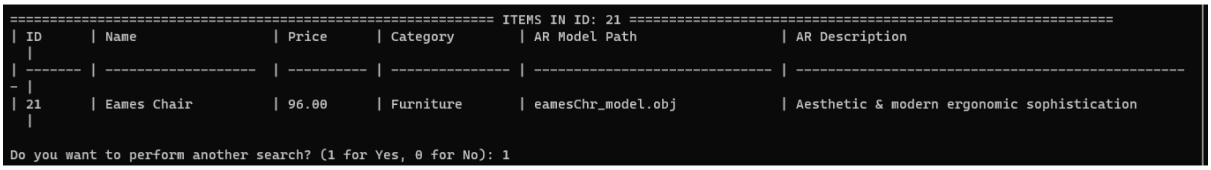


```
============================= MM MART MENU =============================

1. Display Items
2. Add Item to Cart
3. Modify Cart Item
4. Delete Cart Item
5. View Cart
6. Checkout
7. Search Items
8. Exit

Choose an option:  7
=========================================
Choose a search option:
1. Search by Item ID
2. Search by Category

Enter your choice: 1
=========================================
Enter the item ID to search: 21
```

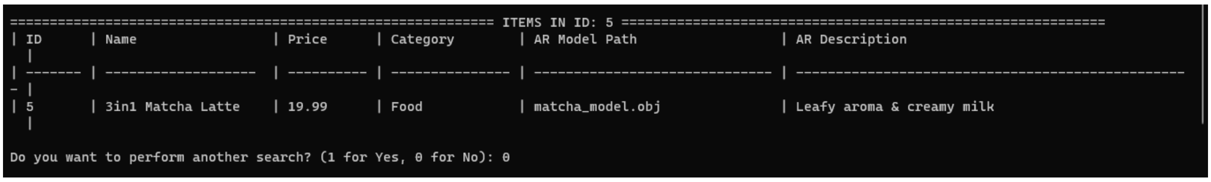


```
Choose a search option:
1. Search by Item ID
2. Search by Category

Enter your choice: 5
Invalid choice.

Do you want to perform another search? (1 for Yes, 0 for No): 1
=========================================
Choose a search option:
1. Search by Item ID
2. Search by Category

Enter your choice: 1
=========================================
Enter the item ID to search: 5
```

Output (Search Item) - Search by Item ID

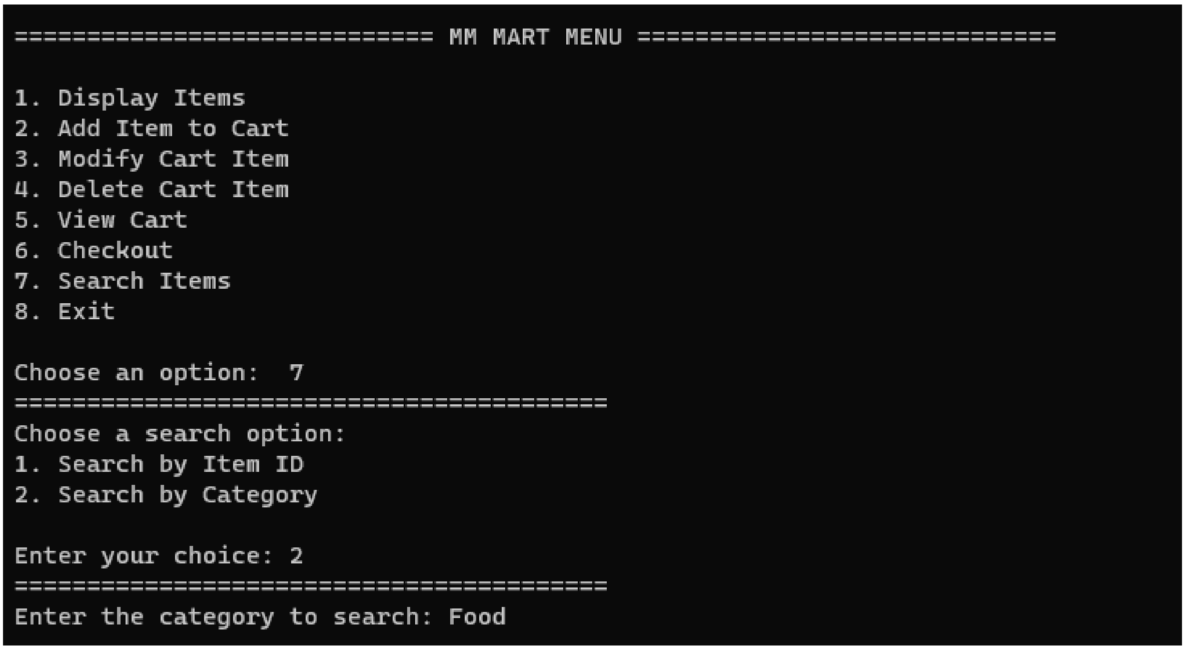


```
================================================================= ITEMS IN ID: 21 =================================================================
| ID      | Name                | Price      | Category        | AR Model Path                  | AR Description
  |
| ------- | ------------------- | ---------- | --------------- | ------------------------------ | --------------------------------------------------
- |
| 21      | Eames Chair         | 96.00      | Furniture       | eamesChr_model.obj             | Aesthetic & modern ergonomic sophistication
  |

Do you want to perform another search? (1 for Yes, 0 for No): 1
```

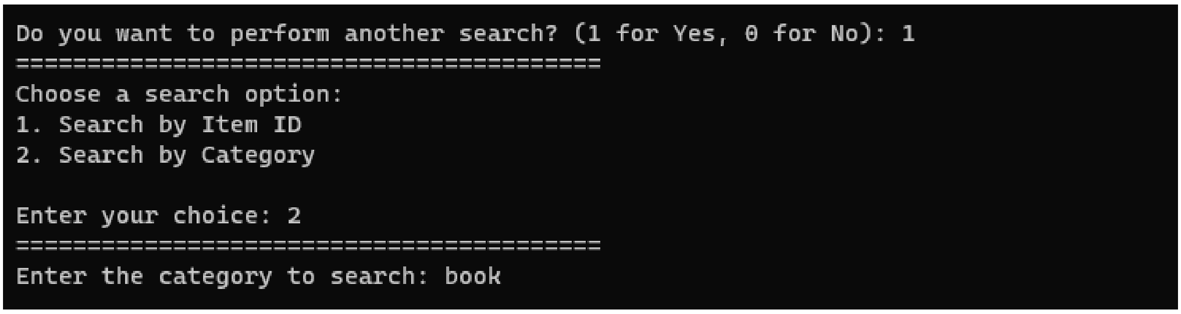


```
================================================================= ITEMS IN ID: 5 =================================================================
| ID      | Name                | Price      | Category        | AR Model Path                  | AR Description
  |
| ------- | ------------------- | ---------- | --------------- | ------------------------------ | --------------------------------------------------
- |
| 5       | 3in1 Matcha Latte   | 19.99      | Food            | matcha_model.obj               | Leafy aroma & creamy milk
  |

Do you want to perform another search? (1 for Yes, 0 for No): 0
```

Input (Search Item) - Search by Category

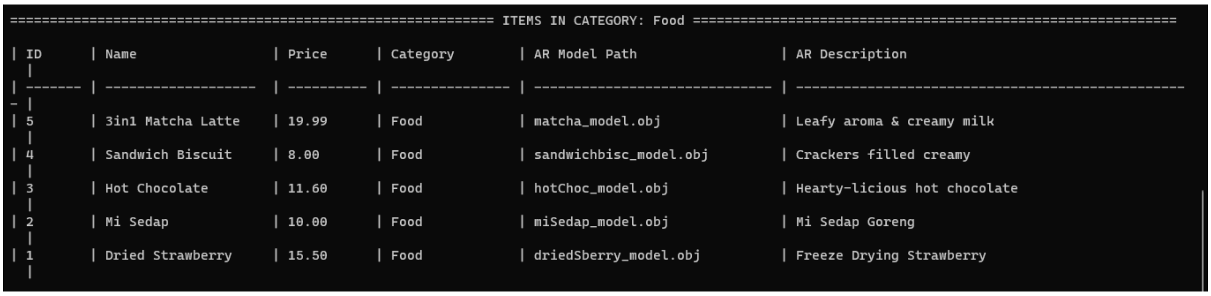


```
============================= MM MART MENU =============================

1. Display Items
2. Add Item to Cart
3. Modify Cart Item
4. Delete Cart Item
5. View Cart
6. Checkout
7. Search Items
8. Exit

Choose an option:  7
=========================================
Choose a search option:
1. Search by Item ID
2. Search by Category

Enter your choice: 2
=========================================
Enter the category to search: Food
```

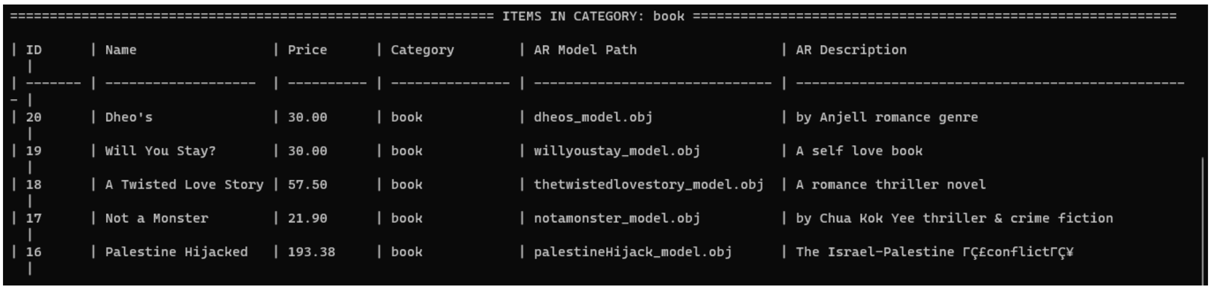


```
Do you want to perform another search? (1 for Yes, 0 for No): 1
=========================================
Choose a search option:
1. Search by Item ID
2. Search by Category

Enter your choice: 2
=========================================
Enter the category to search: book
```

Output (Search Item) - Search by Category

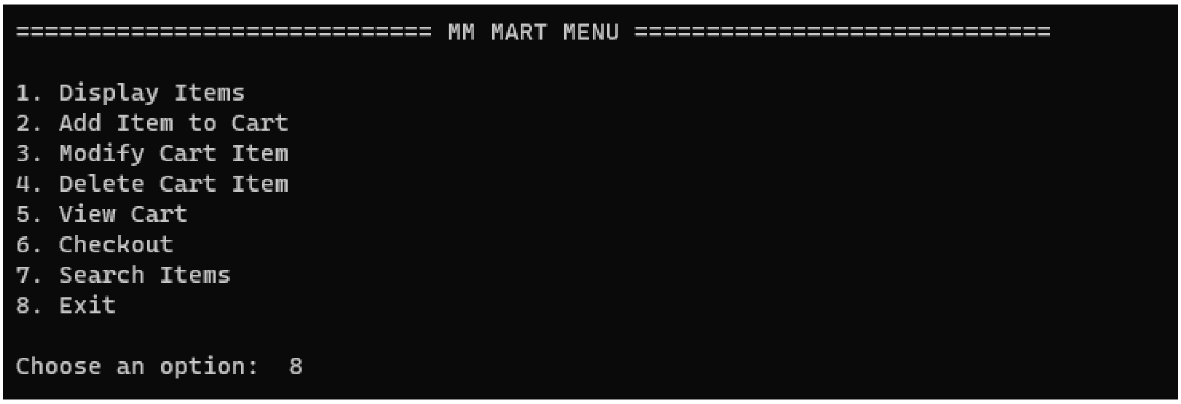


```
============================================================= ITEMS IN CATEGORY: Food =============================================================

| ID      | Name               | Price      | Category       | AR Model Path                  | AR Description
  |
| ------- | ------------------ | ---------- | -------------- | ------------------------------ | -------------------------------------------------
- |
| 5       | 3in1 Matcha Latte  | 19.99      | Food           | matcha_model.obj               | Leafy aroma & creamy milk
  |
| 4       | Sandwich Biscuit   | 8.00       | Food           | sandwichbisc_model.obj         | Crackers filled creamy
  |
| 3       | Hot Chocolate      | 11.60      | Food           | hotChoc_model.obj              | Hearty-licious hot chocolate
  |
| 2       | Mi Sedap           | 10.00      | Food           | miSedap_model.obj              | Mi Sedap Goreng
  |
| 1       | Dried Strawberry   | 15.50      | Food           | driedSberry_model.obj          | Freeze Drying Strawberry
  |
```



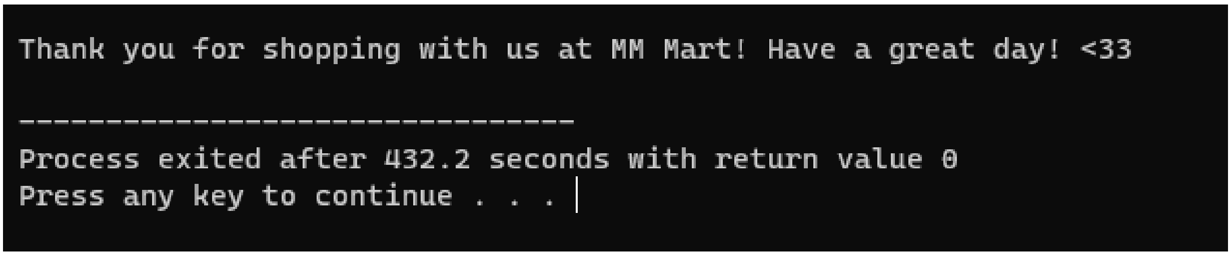
```
============================================================= ITEMS IN CATEGORY: book =============================================================

| ID      | Name               | Price      | Category       | AR Model Path                  | AR Description
  |
| ------- | ------------------ | ---------- | -------------- | ------------------------------ | -------------------------------------------------
- |
| 20      | Dheo's             | 30.00      | book           | dheos_model.obj                | by Anjell romance genre
  |
| 19      | Will You Stay?     | 30.00      | book           | willyoustay_model.obj          | A self love book
  |
| 18      | A Twisted Love Story | 57.50    | book           | thetwistedlovestory_model.obj  | A romance thriller novel
  |
| 17      | Not a Monster      | 21.90      | book           | notamonster_model.obj          | by Chua Kok Yee thriller & crime fiction
  |
| 16      | Palestine Hijacked | 193.38     | book           | palestineHijack_model.obj      | The Israel-Palestine ΓÇ£conflictΓÇ¥
  |
```

Input (Exit)



```
============================ MM MART MENU ============================

1. Display Items
2. Add Item to Cart
3. Modify Cart Item
4. Delete Cart Item
5. View Cart
6. Checkout
7. Search Items
8. Exit

Choose an option:  8
```

Output (Exit)



```
Thank you for shopping with us at MM Mart! Have a great day! <33

--------------------------------
Process exited after 432.2 seconds with return value 0
Press any key to continue . . .
```

CODING

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#define MAX_ITEMS 100

// Structure for customer information
struct Customer
{
   char cust_name[200];
   int age;
]\   char contNum[20];     char address[200];

};

// Structure for item information
struct Item
{
   int item_id;
   char item_name[100];
   double price;
   char category[50];
   char ar_model_path[100];
   char ar_description[200];
   int quantity; // Added quantity for cart management
};

// Structure for a node in a linked list
struct Node
{
   struct Item data;
   struct Node* next;
};

// Function to print the main menu options
void printMenu()
{
        // Menu options are printed here
   printf("\n================================ MM MART MENU
================================\n");
   printf("\n1. Display Items\n");
   printf("2. Add Item to Cart\n");
   printf("3. Modify Cart Item\n");
   printf("4. Delete Cart Item\n");
   printf("5. View Cart\n");
   printf("6. Checkout\n");
   printf("7. Search Items\n");
   printf("8. Exit\n");
   printf("\nChoose an option: ");
   printf(" ");
}

// Function to display available items
void displayItems(struct Node* head)
{
        // Displaying items in a table

printf("\n============================================================
============ AVAILABLE ITEMS
===================================================================
====\n");
   printf("\n");
   printf("| %-7s | %-20s | %-10s | %-15s | %-30s | %-50s |\n", "ID", "Name", "Price",
"Category", "AR Model Path", "AR Description");
   printf("| %-7s | %-20s | %-10s | %-15s | %-30s | %-50s |\n", "-------",
"-------------------", "---------", "--------------", "----------------------------",
"--------------------------------------------");

   struct Node* current = head;
   while (current != NULL)
        {
      printItem(current->data);
      current = current->next;
   }
}

// Function to modify the quantity of an item in the cart
void modifyCartItem(struct Node* cart)
```

```
{
  int itemId, newQuantity, choice;

      // Allows the user to modify the quantity of an item in the shopping cart
  do
      {
    printf("=============================================\n");
    printf("Enter the ID of the item to modify: ");
    scanf("%d", &itemId);

    struct Node* current = cart;

    while (current != NULL)
           {
      if (current->data.item_id == itemId)
                 {
        printf("Enter the new quantity: ");
        scanf("%d", &newQuantity);
        current->data.quantity = newQuantity;
        printf("Item quantity modified successfully.\n");
        break;
      }
      current = current->next;
    }
    if (current == NULL)
           {
      printf("Item not found in the cart.\n");
    }
    printf("=============================================\n");
    printf("Do you want to modify another item? (1 for Yes, 0 for No): ");
    scanf("%d", &choice);
  }
      while (choice != 0);
}

// Function to delete an item from the cart
void deleteCartItem(struct Node** cart)
{
  int itemId, choice;
```

```
           // Allows the user to delete an item from the shopping cart
    do
           {
        printf("============================================\n");
        printf("Enter the ID of the item to delete: ");
        scanf("%d", &itemId);

        struct Node* current = *cart;
        struct Node* prev = NULL;

        while (current != NULL)
                      {
            if (current->data.item_id == itemId)
                                 {
                if (prev == NULL)
                                            {
                    // If the item to be removed is the first node
                    *cart = current->next;
                } else
                                            {
                    prev->next = current->next;
                }
                free(current);
                printf("Item deleted from the cart successfully.\n");
                break;
            }
            prev = current;
            current = current->next;
        }
        if (current == NULL)
                      {
            printf("Item not found in the cart.\n");
        }

        printf("Do you want to delete another item? (1 for Yes, 0 for No): ");
        scanf("%d", &choice);

    }
           while (choice != 0);
}
```

```
// Function to search for items by ID or category
void searchItem(struct Node* head)
{
        // Allows the user to search for items by ID or category
   int choice;
   printf("========================================\n");
   printf("Choose a search option:\n");
   printf("1. Search by Item ID\n");
   printf("2. Search by Category\n");
   printf("\nEnter your choice: ");
   scanf("%d", &choice);


  if (choice == 1)
        {
      int searchItemId;
      printf("========================================\n");
      printf("Enter the item ID to search: ");
      scanf("%d", &searchItemId);

      struct Node* current = head;
      int found = 0;


printf("\n======================================================
======= ITEMS IN ID: %d
=============================================================\
n", searchItemId);
      printf("| %-7s | %-20s | %-10s | %-15s | %-30s | %-50s |\n", "ID", "Name",
"Price", "Category", "AR Model Path", "AR Description");
      printf("| %-7s | %-20s | %-10s | %-15s | %-30s | %-50s |\n", "-------",
"--------------------", "----------", "---------------", "------------------------------",
"--------------------------------------------------");

      while (current != NULL)
                {
         if (current->data.item_id == searchItemId)
                        {
            printItem(current->data);
```

```
                found = 1;
                break;
            }
            current = current->next;
        }
        if (!found)
                    {
            printf("No item found with the specified ID.\n");
        }

    }
           else if (choice == 2)
           {
        char searchCategory[50];
        printf("=============================================\n");
        printf("Enter the category to search: ");
        scanf("%s", searchCategory);

        struct Node* current = head;
        int found = 0;


printf("\n==============================================================
======= ITEMS IN CATEGORY: %s
===================================================================\
n", searchCategory);
        printf("\n");
        printf("| %-7s | %-20s | %-10s | %-15s | %-30s | %-50s |\n", "ID", "Name",
"Price", "Category", "AR Model Path", "AR Description");
        printf("| %-7s | %-20s | %-10s | %-15s | %-30s | %-50s |\n", "-------",
"--------------------", "----------", "---------------", "------------------------------",
"--------------------------------------------------");

        while (current != NULL)
                    {
            if (strcmp(current->data.category, searchCategory) == 0)
                                {
                printItem(current->data);
                found = 1;
            }
            current = current->next;
        }
        if (!found)
                      {
            printf("No items found in the specified category.\n");
        }

    }
           else
           {
        printf("Invalid choice.\n");
    }
}

// Function to print an item's information
void printItem(struct Item item)
{
           // Prints the details of a single item
    printf("| %-7d | %-20s | %-10.2f | %-15s | %-30s | %-50s |\n",
           item.item_id, item.item_name, item.price, item.category, item.ar_model_path,
item.ar_description);
}

// Function to insert a node at the beginning of a linked list
struct Node* insertNode(struct Node* head, struct Item item)
{
           // Inserts a new node at the beginning of a linked list
    struct Node* newNode = (struct Node*)malloc(sizeof(struct Node));
    if (newNode == NULL)
           {
        printf("Memory allocation error\n");
        return head;
    }

    newNode->data = item;
    newNode->next = head;

    return newNode;
}
```

```
// Function to sort the cart by item name
void sortCartByName(struct Node** cartHead)
{
           // Sorts the shopping cart by item name using a simple algorithm
    if (*cartHead == NULL || (*cartHead)->next == NULL)
            {
        printf("The cart has one or no items, no need to sort.\n");
        return;
    }

    struct Node *i, *j;
    struct Item temp;
    for (i = *cartHead; i != NULL; i = i->next)
            {
        for (j = i->next; j != NULL; j = j->next)
                       {
            if (strcmp(i->data.item_name, j->data.item_name) > 0)
                                  {
                temp = i->data;
                i->data = j->data;
                j->data = temp;
            }
        }
    }
}

// Function to sort the cart by item price
void sortCartByPrice(struct Node** cartHead)
{
           // Sorts the shopping cart by item price using a simple algorithm
    if (*cartHead == NULL || (*cartHead)->next == NULL)
            {
        printf("The cart has one or no items, no need to sort.\n");
        return;
    }

    struct Node *i, *j;
    struct Item temp;
    for (i = *cartHead; i != NULL; i = i->next)
            {
```

```
        for (j = i->next; j != NULL; j = j->next)
                       {
            if (i->data.price > j->data.price)
                                  {
                temp = i->data;
                i->data = j->data;
                j->data = temp;
            }
        }
    }
}

// Function to view the items in the shopping cart
void viewCart(struct Node* cart)
{
           // Displays the items in the shopping cart, with an option to sort
    if (cart == NULL)
            {
        printf("\nYour cart is empty.\n");
        return;
    }

    int sortChoice;
    printf("\nDo you want to sort the items in your cart?\n");
    printf("==============================================\n");
    printf("1. No Sort\n");
    printf("2. Sort by Name\n");
    printf("3. Sort by Price\n");
    printf("\nEnter your choice (1-3): ");
    scanf("%d", &sortChoice);

    switch (sortChoice)
            {
        case 2:
            sortCartByName(&cart);
            break;
        case 3:
            sortCartByPrice(&cart);
            break;
        case 1:
```

```
        default:
            break; // No sort or invalid choice
    }


printf("\n======================================================
========= YOUR SHOPPING CART
=============================================================
==\n");
    printf("\n");
    printf("| %-7s | %-20s | %-10s | %-15s | %-30s | %-50s | %-8s |\n", "ID", "Name",
"Price", "Category", "AR Model Path", "AR Description", "Quantity");
    printf("| %-7s | %-20s | %-10s | %-15s | %-30s | %-50s | %-8s |\n", "-------",
"--------------------", "----------", "---------------", "------------------------------",
"--------------------------------------------------", "--------");

    struct Node* current = cart;
    while (current != NULL)
          {
        printItemWithQuantity(current->data);
        current = current->next;
    }
}

// Function to print an item's information along with quantity
void printItemWithQuantity(struct Item item)
{
         // Prints the details of an item along with its quantity
    printf("| %-7d | %-20s | %-10.2f | %-15s | %-30s | %-50s | %-8d |\n",
         item.item_id, item.item_name, item.price, item.category, item.ar_model_path,
item.ar_description, item.quantity);
}

// Function to process the checkout of items in the cart
void checkout(struct Node** cart, struct Customer customer)
{
         // Processes the checkout, displays customer info, and prints the receipt
    struct Item checkoutQueue[MAX_ITEMS];
    int front = 0, rear = -1;
    double totalAmount = 0.0;
    // Enqueue items from the cart to the checkout queue
    while (*cart != NULL)
        {
      enqueue(checkoutQueue, &rear, (*cart)->data);
      totalAmount += ((*cart)->data.price) * ((*cart)->data.quantity); // Accumulate
total price
      pop(cart);
    }


printf("\n=======================================================
===== CUSTOMER INFORMATION
============================================================\n"
);
    printf("\n");
    printf("Name: %s\n", customer.cust_name);
    printf("Age: %d\n", customer.age);
    printf("Number: %s\n", customer.contNum);
    printf("Address: %s\n", customer.address);

    // Process orders from the checkout queue

printf("\n=======================================================
=========== CHECKOUT
==============================================================
====\n");
    printf("\n");
    printf("| %-7s | %-20s | %-10s | %-15s | %-30s | %-50s | %-8s | %-15s |\n", "ID",
"Name", "Price", "Category", "AR Model Path", "AR Description", "Quantity", "Total
Price");
    printf("| %-7s | %-20s | %-10s | %-15s | %-30s | %-50s | %-8s | %-15s |\n", "-------",
"--------------------", "----------", "---------------", "------------------------------",
"--------------------------------------------------", "--------", "---------------");

    while (front <= rear)
        {
      struct Item currentItem = checkoutQueue[front];
      dequeue(checkoutQueue, &front, rear);
      printf("| %-7d | %-20s | %-10.2f | %-15s | %-30s | %-50s | %-8d | %-15.2f |\n",
               currentItem.item_id, currentItem.item_name, currentItem.price,
currentItem.category,
               currentItem.ar_model_path, currentItem.ar_description, currentItem.quantity,
               currentItem.price * currentItem.quantity);

        // Optionally, you can perform other actions related to processing the order here.
    }

           printf("\n=================================================");

           printf("\n");

    printf("\nTotal Amount: RM%.2f\n", totalAmount);
}

// Function to push an item to the top of a stack (used for cart)
void push(struct Node** top, struct Item item, int quantity)
{
           // Pushes an item onto the top of a stack, used for managing the shopping cart
    struct Node* newNode = (struct Node*)malloc(sizeof(struct Node));
    if (newNode == NULL)
           {
        printf("Memory allocation error\n");
        return;
    }
    newNode->data = item;
    newNode->data.quantity = quantity;
    newNode->next = *top;
    *top = newNode;
    printf("===========================================\n");
    printf(" ");
    printf("Item '%s' (Quantity: %d) added to the cart.\n", item.item_name, quantity);
}

// Function to pop an item from the top of a stack (used for cart)
void pop(struct Node** top)
{
           // Pops/removes an item from the top of a stack, used for managing the
shopping cart
    if (*top == NULL)
```

```
            {
        printf("Cart is empty.\n");
        return;
    }

    struct Node* temp = *top;
    *top = (*top)->next;

            printf("========================================\n");
    printf("Item '%s' removed from the cart.\n", temp->data.item_name);

    free(temp);
}

// Function to enqueue an item into a queue (used for checkout)
void enqueue(struct Item queue[], int* rear, struct Item item)
{

            // Enqueues an item into a queue, used for processing the checkout
    if (*rear == MAX_ITEMS - 1)
            {
        printf("Checkout queue is full.\n");
        return;
    }

    (*rear)++;

    queue[*rear] = item;

            printf("========================================\n");
    printf("\nItem '%s' added to the checkout queue.\n", item.item_name);
}

// Function to dequeue an item from a queue (used for checkout)
void dequeue(struct Item queue[], int* front, int rear)
{
            // Dequeues/removes an item from a queue, used for processing the checkout
    if (*front > rear)
            {
        printf("Checkout queue is empty.\n");
        return;
    }

    printf("\nProcessing order: '%s'\n", queue[*front].item_name);
    (*front)++;
}

// Function to perform linear search for an item by ID
struct Node* linearSearch(struct Node* head, int targetId)
{
           // Performs a linear search to find an item by its ID in the linked list
    struct Node* current = head;

    while (current != NULL)
            {
        if (current->data.item_id == targetId)
                      {
            return current;
        }
        current = current->next;
    }
    return NULL;
}

// Function to perform selection sort on a linked list
void selectionSort(struct Node* head)
{
           // Sorts a linked list using the selection sort algorithm
    struct Node *i, *j;
    struct Item temp;

    for (i = head; i != NULL; i = i->next)
            {
        for (j = i->next; j != NULL; j = j->next)
                      {
            if (strcmp(i->data.item_name, j->data.item_name) > 0)
                                  {
                temp = i->data;
                i->data = j->data;
                j->data = temp;
            }
        }
    }
}

// Function to get customer information
struct Customer getCustomerInfo()
{
// Prompts the user to input customer information and returns a Customer structure
    struct Customer customer;

    while (getchar() != '\n');

    printf("\nEnter name: ");
    scanf("%199[^\n]", customer.cust_name);

    printf("Enter age: ");
    scanf("%d", &customer.age);

    while (getchar() != '\n');

    printf("Enter contact number: ");
    scanf("%19[^\n]", customer.contNum);

    while (getchar() != '\n');

    printf("Enter address: ");
    scanf("%199[^\n]", customer.address);

    return customer;
}

// Function to free the memory occupied by a linked list
void freeLinkedList(struct Node* head)
{
          // Frees the memory occupied by each node in a linked list
    while (head != NULL)
          {
        struct Node* temp = head;
        head = head->next;
        free(temp);
    }
}

// Main function where the program execution begins
int main()
{
            // Initialization of linked lists and customer information
    struct Node* itemList = NULL;
    struct Node* cart = NULL;
    struct Customer customer;


printf("*****************************************************************
***********************************************************************
**\n");
            printf("\n");
            printf("                              WELCOME TO OUR MINI MAGIC
MART(MM MART)                                       \n");
            printf("\n");

printf("*****************************************************************
***********************************************************************
**\n");
            printf("\n");
            printf("We're excited to have you here.:)\n");
            printf("Feel free to explore our products and enjoy your shopping
experience!\n");
            printf("\n");
            printf("========================= CUSTOMER INFORMATION
=========================\n");
            printf("");
    customer = getCustomerInfo();

            // Adding items to the item list
    struct Item item1 = {1, "Dried Strawberry", 15.50, "Food",
"driedSberry_model.obj", "Freeze Drying Strawberry"};
    struct Item item2 = {2, "Mi Sedap", 10.00, "Food", "miSedap_model.obj", "Mi
Sedap Goreng"};
```

```
   struct Item item3 = {3, "Hot Chocolate", 11.60, "Food", "hotChoc_model.obj",
"Hearty-licious hot chocolate"};
   struct Item item4 = {4, "Sandwich Biscuit", 8.00, "Food",
"sandwichbisc_model.obj", "Crackers filled creamy"};
   struct Item item5 = {5, "3in1 Matcha Latte", 19.99, "Food", "matcha_model.obj",
"Leafy aroma & creamy milk"};
   struct Item item6 = {6, "Vintage Tee Shirt", 49.99, "Fashion",
"vintageTShirt_model.obj", "Basic cotton jersey fabric"};
   struct Item item7 = {7, "Multi Colour Sunglass", 20.99, "Fashion",
"multiCSunglass_model.obj", "Comfortable & lightweight durable frame"};
   struct Item item8 = {8, "Men's Baseball Cap", 28.65, "Fashion",
"menBBCap_model.obj", "Adjustable & cotton material"};
   struct Item item9 = {9, "Women's Wedge Sandal", 65.95, "Fashion",
"KayorWedgeSandal_model.obj", "Open toe & buckle closure"};
   struct Item item10 = {10, "Hooded Sweatshirt", 55.95, "Fashion",
"HoodSweatS_model.obj", "Cotton & long sleeve"};
   struct Item item11 = {11, "Baseus Bowie H1", 169.00, "Electronic",
"baseusBowie_model.obj", "Noise cancelling wireless headphone"};
   struct Item item12 = {12, "Samsung Tab A8", 1159.90, "Electronic",
"samsungTabA8_model.obj", "Excellent tab for e-learning"};
   struct Item item13 = {13, "Samsung Smart TV", 8000.00, "Electronic",
"samsungSmartTV_model.obj", "Smart that able to connect with internet"};
   struct Item item14 = {14, "Samsung Galaxy A33 5G", 1709.90, "Electronic",
"samsungGalaxy_model.obj", "Samsung galaxy smartphone"};
   struct Item item15 = {15, "Dell Laptop", 3000.90, "Electronic",
"dellLaptop_model.obj", "Personal computer"};
   struct Item item16 = {16, "Palestine Hijacked", 193.38, "book",
"palestineHijack_model.obj", "The Israel-Palestine “conflict”"};
   struct Item item17 = {17, "Not a Monster", 21.90, "book",
"notamonster_model.obj", "by Chua Kok Yee thriller & crime fiction"};
   struct Item item18 = {18, "A Twisted Love Story", 57.50, "book",
"thetwistedlovestory_model.obj", "A romance thriller novel"};
   struct Item item19 = {19, "Will You Stay?", 30.00, "book",
"willyoustay_model.obj", "A self love book"};
   struct Item item20 = {20, "Dheo's", 30.00, "book", "dheos_model.obj", "by Anjell
romance genre"};
   struct Item item21 = {21, "Eames Chair", 96.00, "Furniture",
"eamesChr_model.obj", "Aesthetic & modern ergonomic sophistication"};
   struct Item item22 = {22, "Teak Dining Table", 1149.00, "Furniture",
"teakDiningTble_model.obj", "High - quality teak wood "};
```

```
    struct Item item23 = {23, "Fabric Sofa", 1695.00, "Furniture",
"fbricSofa_model.obj", "Stylish & comfortable seating"};
    struct Item item24 = {24, "Display Cabinet", 339.00, "Furniture",
"dsplyCbinet_model.obj", "Sophisticated & stylish showcase"};
    struct Item item25 = {25, "Sliding Wardrobe", 1399.00, "Furniture",
"slidingWdrobe_model.obj", "Modern & contemporary storage spaces"};

    itemList = insertNode(itemList, item1);
    itemList = insertNode(itemList, item2);
    itemList = insertNode(itemList, item3);
    itemList = insertNode(itemList, item4);
    itemList = insertNode(itemList, item5);
    itemList = insertNode(itemList, item6);
    itemList = insertNode(itemList, item7);
    itemList = insertNode(itemList, item8);
    itemList = insertNode(itemList, item9);
    itemList = insertNode(itemList, item10);
    itemList = insertNode(itemList, item11);
    itemList = insertNode(itemList, item12);
    itemList = insertNode(itemList, item13);
    itemList = insertNode(itemList, item14);
    itemList = insertNode(itemList, item15);
    itemList = insertNode(itemList, item16);
    itemList = insertNode(itemList, item17);
    itemList = insertNode(itemList, item18);
    itemList = insertNode(itemList, item19);
    itemList = insertNode(itemList, item20);
    itemList = insertNode(itemList, item21);
    itemList = insertNode(itemList, item22);
    itemList = insertNode(itemList, item23);
    itemList = insertNode(itemList, item24);
    itemList = insertNode(itemList, item25);

    int choice;

        // Main program loop
    do{
        // Displaying main menu and getting user choice
        printMenu();
        scanf("%d", &choice);
```

```
                    // Handling user choice
             switch (choice)
                    {
                         // Frees the memory occupied by each node in a linked list
               case 1:
                 displayItems(itemList);
                 break;

               case 2:
                         {
                         int itemId;
                         int quantity;

                         do
                              {

printf("===============================================\n");
                         printf("Enter the item ID to add to the cart: ");
                         scanf("%d", &itemId);

                         struct Node* foundItem = linearSearch(itemList, itemId);

                         if (foundItem != NULL)
                                   {
                              printf("Enter the quantity: ");
                              scanf("%d", &quantity);
                              push(&cart, foundItem->data, quantity);
                         }
                                   else
                                   {
                              printf("Item not found.\n");
                                   }


printf("===============================================\n");
                         printf("\nDo you want to add more items to the cart? (1 for Yes, 0
for No): ");
                         scanf("%d", &choice);
```

```
                        }
                                while (choice != 0);
                        }
                        break;

                        case 3:

                                modifyCartItem(cart);
                break;

        case 4:

                deleteCartItem(&cart);
                break;

         case 5:

            // View Cart
            viewCart(cart);
            break;

         case 6:
            // Checkout
            checkout(&cart, customer);  // Pass the customer information
            break;

         case 7:
            // Search Items
            do
            {
               searchItem(itemList);

               printf("\nDo you want to perform another search? (1 for Yes, 0 for No): 
");
               scanf("%d", &choice);

            } while (choice != 0);
            break;

         case 8:
                printf("\nThank you for shopping with us at MM Mart! Have a great day!
<33\n");
                break;

            default:

                printf("Invalid option. Please try again.\n");
        }

    } // end switch

           while (choice != 8); // Continue until the user chooses to exit

           // Freeing memory occupied by linked lists
    freeLinkedList(itemList);
    freeLinkedList(cart);

    return 0;
}
```

REFERENCES

Warby Parker. (2019). *Warby Parker*. Warby Parker. https://www.warbyparker.com/

Warby Parker Review: Options, Pros & Cons, Is It Worth It? (2021, February 22). Healthline. https://www.healthline.com/health/warby-parker-review#options

Warby Parker Review 2021: Pros, Cons, and How They Compare. (2020, November 25). Eyewear Genius. https://www.eyeweargenius.com/reviews/eyewear/warby-parker/

IKEA Store App | IKEA Malaysia. (n.d.). Www.ikea.com. https://www.ikea.com/my/en/customer-service/mobile-apps/

Converse: an augmented reality app for trying on shoes. (2020, October 15). Разработка дополненной реальности AR. Разработка AR приложений. https://arvar.org/en/cases/converse-an-augmented-reality-app-for-trying-on-shoes/