WSB Merito Chorzów

Programowanie obiektowe. Zadanie 4. E-sklep

[Document subtitle]

Imiona i nazwiska: Zadorozhnyi Yevheni, Yurii Chystiakov
Numery albomów: 143656, 143657

Maily: zhen24320@gmail.com, cistakovu00@gmail.com

Link na git: https://github.com/cistakovu00/E-sklep-ProgramowanieObiektowe.git

Jako projekt do wdrożenia wybraliśmy system zarządzania e-sklepem. Aby zrealizować to zadanie, dodaliśmy następujące klasy i metody do nich:

Product, DiscountProduct, Cart, Customer oraz enum MenuChoices.

```csharp
12 references
public class Product
{
    [JsonProperty]
    2 references
    private int ProductId { get; set; }
    [JsonProperty]
    2 references
    private string Name { get; set; }
    [JsonProperty]
    2 references
    private decimal Price { get; set; }

    2 references
    public Product(int productId, string name, decimal price)...

    1 reference
    public int GetProductId()...

    1 reference
    public string GetName()...

    2 references
    public decimal GetPrice()...
}
```

```csharp
5 references
public class Cart
{
    public List<Product> Products = new List<Product>();

    private static string JsonFileName = "cart.json";

    1 reference
    public void AddProduct(Product product)...

    1 reference
    public void DisplayCart()...

    2 references
    public decimal CalculateTotal()...

    4 references
    public static List<Product> DeserializeFromJson()...

    1 reference
    public void SerializeToJson(List<Product> productList)...
}
```

```csharp
1 reference
public class DiscountProduct : Product
{
    2 references
    private decimal Discount { get; set; }

    0 references
    public DiscountProduct(int productId, string name, decimal price, decimal discount)
        : base(productId, name, price)
    {
        Discount = discount;
    }

    0 references
    public decimal GetDiscount()...
}
```

```csharp
public enum MenuChoices
{
    Exit = 0,
    AddProduct = 1,
    DisplayCart = 2,
    CalculateTotal = 3,
    PlaceOrder = 4,
    SaveCustomer = 5,
}
```
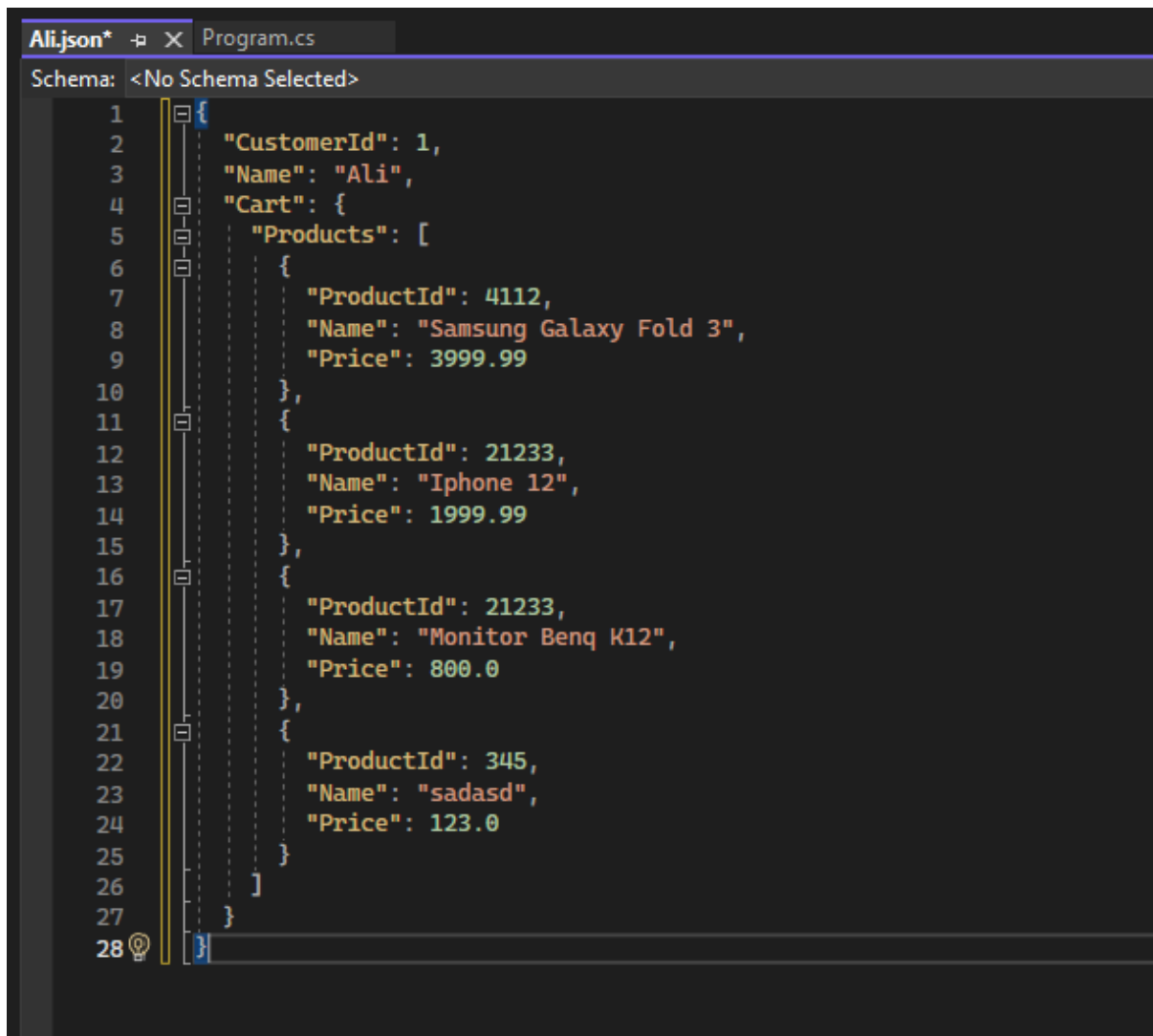
```csharp
namespace Rozetka
{
    0 references
    internal class Program
    {
        private static Cart cart = new Cart();
        0 references
        static void Main(string[] args)
        {
            Console.Write("Enter your name: ");
            string customerName = Console.ReadLine();
            Customer customer = new Customer(1, customerName);

            while (true)
            {
                Console.WriteLine("Menu:");
                Console.WriteLine("1. Add Product to Cart");
                Console.WriteLine("2. Display Cart");
                Console.WriteLine("3. Calculate Total");
                Console.WriteLine("4. Place Order");
                Console.WriteLine("5. Save Customer");
                Console.WriteLine("0. Exit");

                string choice = Console.ReadLine();

                if (Enum.TryParse(choice, out MenuChoices menuChoice))
                {
                    switch (menuChoice)
                    {
                        case MenuChoices.AddProduct:
                            AddNewProduct();
                            break;
                        case MenuChoices.DisplayCart:
                            cart.DisplayCart();
                            break;
                        case MenuChoices.CalculateTotal:
                            CalculatingTotal();
                            break;
                        case MenuChoices.PlaceOrder:
                            customer.PlaceOrder();
                            break;
                        case MenuChoices.SaveCustomer:
                            customer.SerializeToJson(customer);
                            break;
                        case MenuChoices.Exit:
                            return;
                    }
                }
                else
                {
                    Console.WriteLine("Invalid choice. Please try again.");
                }
            }
        }

        1 reference
        private static void CalculatingTotal()...

        1 reference
        private static void AddNewProduct()...
    }
}
```

Dodaliśmy również możliwość zapisywania klienta. Gdy użytkownik wybierze opcję Zapisz klienta w menu, klient zostanie zapisany w pliku json o nazwie Customername.json.

Przykład takiego pliku:

```json
{
    "CustomerId": 1,
    "Name": "Ali",
    "Cart": {
      "Products": [
        {
          "ProductId": 4112,
          "Name": "Samsung Galaxy Fold 3",
          "Price": 3999.99
        },
        {
          "ProductId": 21233,
          "Name": "Iphone 12",
          "Price": 1999.99
        },
        {
          "ProductId": 21233,
          "Name": "Monitor Benq K12",
          "Price": 800.0
        },
        {
          "ProductId": 345,
          "Name": "sadasd",
          "Price": 123.0
        }
      ]
    }
}
```