МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний аерокосмічний університет ім. М.Є. Жуковського

«Харківський авіаційний інститут»

Факультет радіоелектроніки, комп’ютерних систем та інфокомунікацій

Кафедра комп’ютерних систем, мереж і кібербезпеки (503)

Лабораторна робота № *3*

|  |  |
| --- | --- |
|  | *Сутності та зв’язки між ними. Тип зв’язку «Асоціація». Індексатори* |
|  | (назва лабораторної роботи) |
| з дисципліни | *Технології програмування* |
|  | (шифр)  ХАІ**.**503**.**525a**.**03О**.**123-Комп'ютерна інженерія**,** ПЗ №9629619 |

|  |  |  |
| --- | --- | --- |
| Виконав студент гр. | 525а | *Литвиненко А.В.* |
| 26.10.22 | (№ групи) | (П.І.Б.) |
| (підпис, дата) |  |  |
| Перевірив | канд. техн. наук, доцент | |
|  |  | *Є. В. Бабешенко* |
| (підпис, дата) |  | (П.І.Б.) |

Харків – 2022

**Тема роботи:** Сутності та зв’язки між ними. Тип зв’язку «Асоціація». Індексатори

**Мета роботи**: Ознайомитися з типом зв’язку «асоціація» між сутностями. Вивчити на практиці особливості використання індексаторів.

**Варіант 5**

**Задача 1**

**Частина 1**. Постановка завдання

**Умова:** Створення нових сутностей та розширення існуючих

До бібліотеки класів, розробленої у лабораторній роботі No1 та розширеної у лабораторній роботі No2, додати дві сутності (варіанти дивись у

Додатку А):

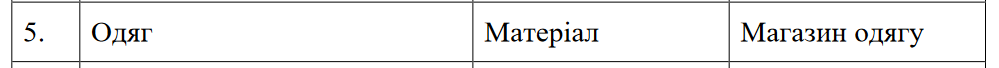
 допоміжний клас;

 клас представлення групи сутностей.

Поле (або властивість) типу допоміжного класу має бути розташоване у класі сутності.

У класі, який представляє групу сутностей, передбачити методи для пошуку даних за двома різними ознаками, а декілька індексаторів з різними типами даних для індексу, методи для додавання / видалення сутностей у групу. Класи повинні містити загальнодоступні властивості.

**Умова з додатка:**



**Частина 2**. Схема

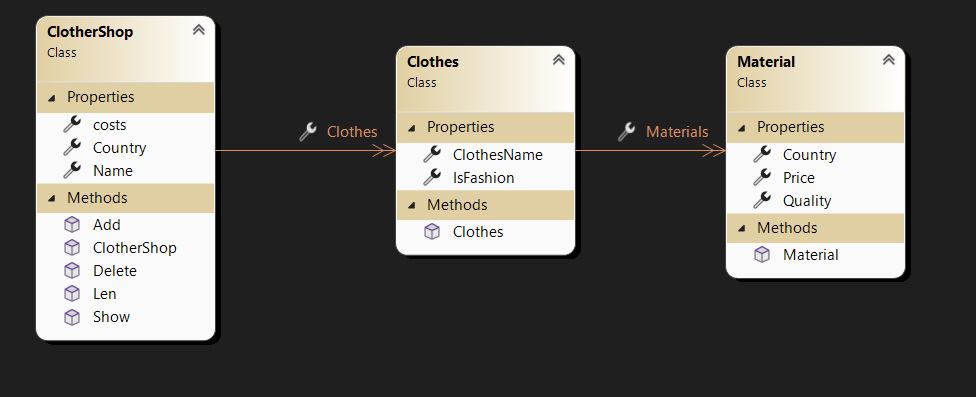


Рисунок 1 - Схема класу

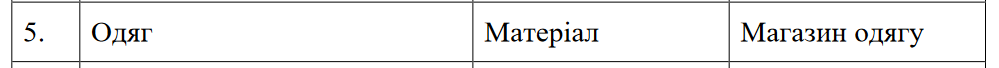
**Задача 2**

**Частина 1**. Постановка завдання

**Умова:** Створення консольного застосунку

Створіть консольний застосунок, що демонструє використання усіх сутностей. Дані, пошук яких потрібно здійснити, необхідно вводити з клавіатури під час виконання. Дані тестів мають бути підготовлені таким чином, щоб пошук давав жодного, один та більше одного результату.

**Умова з додатка:**



Частина 2. Тестування

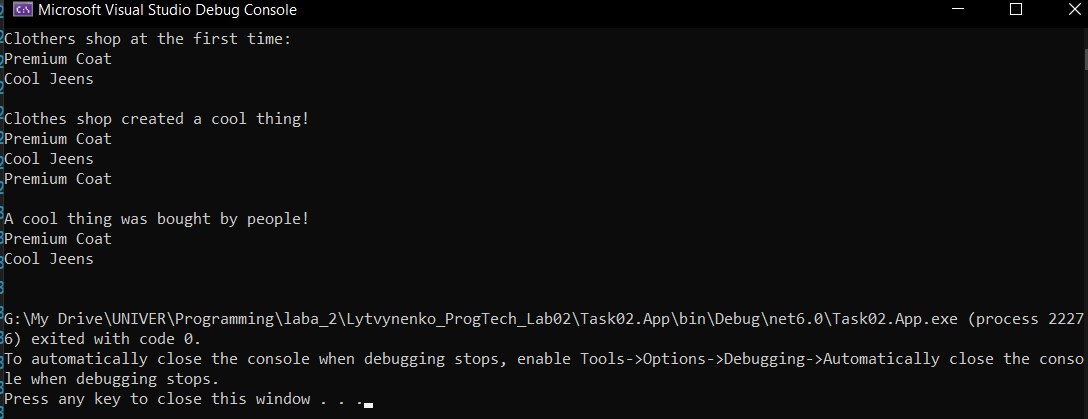


Рисунок 2 – скришот тестування

**Частина 3.** Код

Program.cs

using Task02;

namespace Task02\_App

{

internal class Program

{

static void Main(string[] args)

{

Material Cotton = new Material(50, 100, "Ukraine");

Material Silk = new Material(90, 50, "Poland");

Material Leather = new Material(100, 100, "USA");

Material[] forCoat = new Material[10];

forCoat[0] = Cotton;

forCoat[1] = Leather;

int forCoat\_length = 2;

Material[] forJeens = new Material[10];

forJeens[0] = Silk;

forJeens[1] = Cotton;

int forJeens\_length = 2;

Clothes Coat = new Clothes(forCoat, "Premium Coat", true);

Clothes Jeens = new Clothes(forJeens, "Cool Jeens", false);

Clothes[] goods = new Clothes[10];

goods[0] = Coat;

goods[1] = Jeens;

int goods\_length = 2;

ClotherShop Guchi = new ClotherShop(goods, "Guchi", "Italy", 4234);

Console.WriteLine("Clothers shop at the first time:");

Guchi.Show();

Console.WriteLine("Clothes shop created a cool thing!");

Guchi.Add(Guchi.Clothes[0]);

Guchi.Show();

Console.WriteLine("A cool thing was bought by people!");

Guchi.Delete();

Guchi.Show();

}

}

}

Class1.cs

using System.Data;

namespace Task02

{

public class Clothes

{

public Clothes(Material[] materials, string clothesName, bool isFashion)

{

Materials = materials;

ClothesName = clothesName;

IsFashion = isFashion;

}

public Material[] Materials { get; set; }

public string ClothesName { get; set; }

public bool IsFashion { get; set; }

}

public class Material

{

public Material(int price, int quality, string country)

{

Price = price;

Quality = quality;

Country = country;

}

public int Price { get; set; }

public int Quality{ get; set; }

public string Country { get; set; }

}

public class ClotherShop

{

public ClotherShop(Clothes[] clothes, string name, string country, int costs)

{

Clothes = clothes;

Name = name;

Country = country;

this.costs = costs;

}

public int Len()

{

int length = 0;

for (int i = 0; i < Clothes.Length; i++)

{

if (Clothes[i] == null)

{

break;

}

length++;

}

return length;

}

public void Add(Clothes obj)

{

Clothes[Len()] = obj;

}

public void Delete()

{

Clothes[Len()-1] = null;

}

public void Show()

{

// define length

for(int i = 0; i < Clothes.Length; ++i)

{

if (Clothes[i] == null)

{

break;

}

Console.Write(Clothes[i].ClothesName + "\n");

}

Console.WriteLine();

}

public Clothes[] Clothes { get; set; }

public string Name { get; set; }

public string Country { get; set; }

public int costs { get; set; }

}

}

**Висновки**

Під час цієї лабораторної роботи я ознайомився з типом «асоціація» між сутностями та реалізував її у своєму проекту, використовуючи у подальшому основному коді. Я створив необхідні мені сутності, які були представлені одиницями у великому масиві, туди можна додавати різних одяг, який сам складається з безлічі разніх матеріалів, моя програма дає можливість все це детально описати для гарного розуміння майбутньому розробнику.