

## Лабораторна робота №2

### Тема: Принципи програмування. SOLID

**Мета роботи:** навчитися дотримуватися принципів SOLID та обґрунтовувати їх.

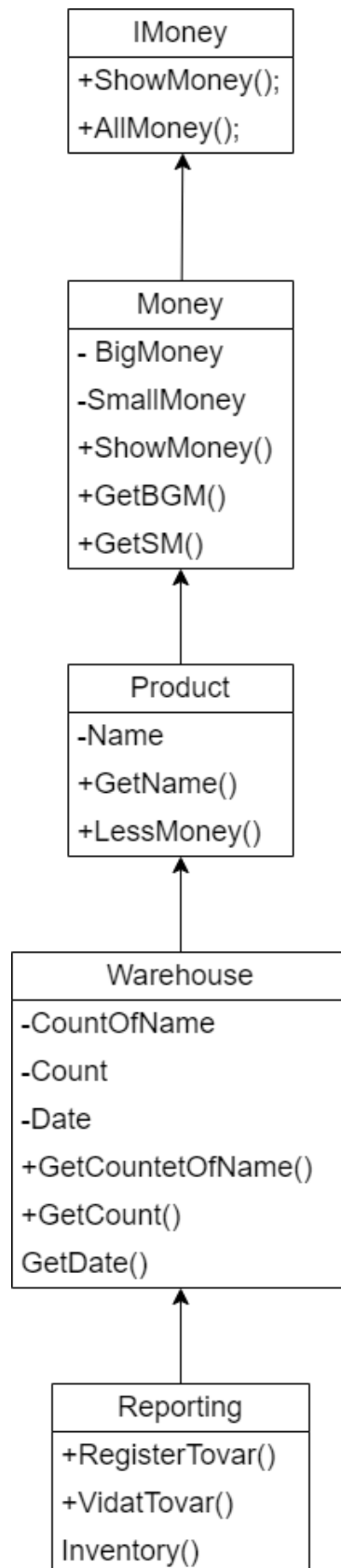
Навчитися описувати дизайн програми за допомогою UML діаграм

Завдання 1: Виконати завдання з дотриманням принципів SOLID.

1. Запрограмуйте клас Money (об'єкт класу оперує однією валютою) для роботи з грошима. У класі мають бути передбачені: поле для зберігання цілої частини грошей (долари, євро, гривні тощо) і поле для зберігання копійок (центи, євроценти, копійки тощо). Реалізувати методи виведення суми на екран, задання значень частин.
2. Створити клас Product для роботи з продуктом або товаром. Реалізувати метод, який дозволяє зменшити ціну на задане число.
3. Реалізувати клас Warehouse, який описує товари, що зберігаються на складі: найменування, одиниця виміру, ціна одиниці, кількість, дата останнього завозу, тощо.
4. Реалізувати клас Reporting для роботи зі звітністю. Реєстрація надходження товару (формування прибуткової накладної) і відвантаження (видаткова накладна). Звіт по інвентаризації (залишки на складі).
5. Для кожного з класів реалізувати необхідні методи і поля. Для класів передбачити реалізацію конструкторів та методів для встановлення та читання значень.
6. Ви також можете додавати власний функціонал для унаочнення принципів SOLID. Приклади додаткового функціоналу:
  - а. категорії для продуктів;
  - б. конкретні дочірні класи валюти
  - с. корзина для замовлень.

					ЖИТОМИРСЬКА ПОЛІТЕХНІКА.23.121.14.000 – Лр.2		
Змн.	Арк.	№ докум.	Підпис	Дата			
Розроб.		Ігліньський І.Ю.			Звіт з лабораторної роботи №2	Літ.	Арк.
Перевір.		Левківський В.Л.					Аркуші
Реценз.							1
Н. Контр.						ФІКТ, гр. ІПЗ-21-4	
Зав.каф.							

## Діаграма класів:



## Лістинг програми:

```
using System.Text;
using ConsoleApp1;
Console.OutputEncoding=Encoding.UTF8;
Money money = new Money(10, 25);
Product product = new Product("Apple",money.GetBGM(),money.GetSM());
product.LessMoney(5);
Reporting reporting = new Reporting(product.GetBGM(), product.GetSM(),
product.GetName(), "Шт.", 10, "15.02.2023");
Console.WriteLine("\nДоступні гроші:\n");
reporting.ShowMoney();
Console.WriteLine("\nРеєстрація надходження товару\n");
reporting.RegisterTovar();
Console.WriteLine("\nВідвантаження\n");
reporting.VidatTovar(1);
Console.WriteLine("\nІнвентаризація\n");
reporting.Inventory(1);
namespace ConsoleApp1
{
    interface IMoney
    {
        void ShowMoney();
        float AllMoney();
    }

    internal class Money :IMoney
    {
        private int BigMoney { get; set; }
        public int GetBGM()
        {
            return BigMoney;
        }
        private int SmallMoney { get; set; }

        public int GetSM()
        {
            return SmallMoney;
        }
        public Money(int bgMoney,int smMoney)
        {
            BigMoney = bgMoney;
            SmallMoney = smMoney;
        }

        public float AllMoney()
        {
            return float.Parse(BigMoney+"."+SmallMoney);
        }
        public void ShowMoney()
        {
            Console.WriteLine("You have {0}.{1}",BigMoney,SmallMoney);
        }
        //usage open-close principle
    }

    internal class Product : Money
    {
        private string Name { get; set; }
    }
}
```

					ЖИТОМИРСЬКА ПОЛІТЕХНІКА.23.121.14.000 – Лр.2	Арк.
						3
Змн.	Арк.	№ докум.	Підпис	Дата		

```

    public string GetName()
    {
        return Name;
    }
    public Product(string name, int bgMoney, int smMoney) : base(bgMoney,
smMoney)
    {
        Name = name;
    }

    public float LessMoney(float price)
    {
        return AllMoney() - price;
    }
    /*A class is used for only one method. (Single responsibility) */
}

internal class Warehouse:Product
{
    private string CountetOfName { get; set; }

    protected string GetCountetOfName()
    {
        return CountetOfName;
    }

    private int Count { get; set; }

    protected int GetCount()
    {
        return Count;
    }

    private string Date { get; set; }

    protected string GetDate()
    {
        return Date;
    }

    protected Warehouse(int bgMoney, int smMoney, string name, string
CountOfName, int count, string date) : base(name, bgMoney, smMoney)
    {
        CountetOfName = CountOfName;
        Count = count;
        Date = date;
    }
    //Open-close principle
}

internal class Reporting :Warehouse
{
    public Reporting(int bgMoney, int smMoney, string name, string
CountOfName, int count, string date) : base(bgMoney, smMoney, name, CountOfName,
count, date)
    {
    }
    //The substitution principle of Barbara Lyskov was used
    public void RegisterTovar()
    {

```

					ЖИТОМИРСЬКА ПОЛІТЕХНІКА.23.121.14.000 – Лр.2	Арк.
						4
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        Console.WriteLine (
            "Найменування:{0}\nОдиниця вимірювання:{1}\nЦіна
одиниці:{2}\nКількість:{3}\n" +
            "Дата
поставки:{4}", GetName (), GetCountetOfName (), AllMoney (), GetCount (), GetDate ()
        );
    }

    public void VidatTovar(int num)
    {
        Console.WriteLine ("№{0}\nТовар:{1}\nКількість:{2}\nОдиниці
вимірювання:{3}\nЦіна:{4}",
            num, GetName (), GetCount (), GetCountetOfName (), AllMoney ()
        );
    }

    public void Inventory(int num)
    {
        Console.WriteLine ("№{0}\nНазва:{1}\nКількість:{2}", num, GetName (), GetCount ());
    }
}

```

**Висновок:** в ході виконання лабораторної роботи було вивчено основні принципи SOLID, а також описувати дизайн програми за допомогою UML діаграм

					ЖИТОМИРСЬКА ПОЛІТЕХНІКА.23.121.14.000 – Лр.2	Арк.
						5
Змн.	Арк.	№ докум.	Підпис	Дата		