

**МІНІСТЕРСТВО**  
**ОСВІТИ ТА НАУКИ УКРАЇНИ**  
**Національний технічний університет України**  
**«Київський політехнічний інститут імені Ігоря Сікорського»**  
**Кафедра інформаційних систем та технологій**

**Звіт**  
**з лабораторної роботи № 2**  
**«Поліморфізм. Наслідування. Перегрузка методів.»**  
**з дисципліни**  
**«Програмування – 2. ООП»**

**Варіант №21**

**Перевірив:**

**доц. Корнага Ярослав**  
**Ігорович**

**Виконала:**

**Студентка ІС-13, ФІОТ**  
**Росновська Ольга**

**Київ 2022**

## Завдання 1

Створити додаток, який задовольняє вимогам, наведеним в завданні. Наслідування застосовувати тільки в тих завданнях, в яких воно логічно обґрунтоване. Аргументувати належність класу кожного створюваного методу і коректно перевизначити для кожного класу методи *Equals*, *GetHashCode*, *ToString*. При виклику будь-якого методу класу, виводити на екран текстове повідомлення.

Створити об'єкт класу Квочка, використовуючи класи Птах, Зозуля, Курка. Методи: літати, співати, нести яйця, висиджувати пташенят.

## Завдання 2

Створити програму, яка задовольняє наступним вимогам:

- Використовувати можливості ООП: класи, наслідування, поліморфізм, інкапсуляція.
- Кожен клас повинен мати змістовну назву та інформативний склад.
- Наслідування має застосовуватися тільки тоді, коли це має сенс.
- Класи повинні бути грамотно розкладені по пакетах.
- Консольне меню повинно бути мінімальним.
- Для зберігання параметрів ініціалізації можна використовувати файли.

Продаж смартфонів. Визначити ієрархію телефонів. Відсортувати телефони по моделі, виробнику, розміру дисплею. Знайти телефон, який відповідає вказаним параметрам. Підрахувати загальну кількість девайсів на складі. Реалізувати пошук телефону по діапазону цін.

## Лістинг програми

### Program.cs

```
using System;
using System.Collections.Generic;

namespace Lab_2
{
    class Program
    {
        static void Main(string[] args)
        {
            void Task1()
            {
                Console.WriteLine("\n" + "Завдання 1: ");
                Bird S = new Sitter("Ряба");
                S.Fly();
                S.Sing();
                S.Eggs();
                S.Child();
                Console.WriteLine(S.GetHashCode());
                Console.WriteLine(S.ToString());
            }
        }
    }
}
```

```

Bird s = new Sitter("\n" + "Ряба");
Console.WriteLine(S.Equals(s));
Console.WriteLine(s.GetHashCode());
Console.WriteLine(s.ToString());

Bird s2 = new Sitter("\n" + "Ряба2");
Console.WriteLine(S.Equals(s2));
Console.WriteLine(s2.GetHashCode());
Console.WriteLine(s2.ToString());

Bird k = new Cuckoo("\n" + "Ряба");
Console.WriteLine(S.Equals(k));
Console.WriteLine(k.GetHashCode());
Console.WriteLine(k.ToString());

int n = 5;
Console.WriteLine(S.Equals(n));
Console.WriteLine(n.GetHashCode());
Console.WriteLine(n.ToString());

Console.WriteLine("\n" + "Список птахів: ");
var B_l = new List<Bird>
    { S, s, s2, k };
Birds B = new Birds(B_l);
B.Print_birds();
}

void Task2()
{
    List<Smartphone> Phones_list = new List<Smartphone>
    {
        new Smartphone("Realme MX6", "Realme", 6.5, 10000),
        new Smartphone("Xiaomi Bv5", "Xiaomi", 6.1, 29999),
        new Smartphone("Redmi Note 11", "Xiaomi", 6.43, 8499),
        new Smartphone("iPhone 11", "Apple", 6.1, 19999),
        new Smartphone("Realme ", "Realme", 6.1, 13999),
        new Smartphone("Redmi 9A", "Xiaomi", 6.53, 4149)
    };

    Shop S = new Shop (Phones_list);
    Console.WriteLine("\n" + "Кількість телефонів на складі: " +
S.Number_of_phones());

    Shop S1 = new Shop(S.Sort_by_model("Xiaomi Bv5"));
    S1.Print_phones("\n"+"Телефони, відсортовані по моделі: ");

    Shop S2 = new Shop(S.Sort_by_producer("Realme"));
    S2.Print_phones("\n" + "Телефони, відсортовані по виробнику: ");

    Shop S3 = new Shop(S.Sort_by_size(6.1));
    S3.Print_phones("\n" + "Телефони, відсортовані по розміру діагоналі:
");

    Shop S4 = new Shop(S.Sort_by_price(5000, 10000));
    S4.Print_phones("\n" + "Телефони у вказаному ціновому діапазоні: ");

    Shop S5 = new Shop(S.Find_by_parameters("Realme r2", "Realme",
6.1));
    S5.Print_phones("\n" + "Телефон, що відповідає вказаним параметрам:
");

```

```
Shop S6 = new Shop(S.Find_by_parameters(p_to_f: "Apple", s_to_f:
6.1));
S6.Print_phones("\n" + "Телефон, що відповідає вказаним параметрам2:
");

}
Console.WriteLine("Оберіть завдання 1 або 2:");
var answer = Console.ReadLine();
if (answer == "1") Task1();
if (answer == "2") Task2();
}
}
```

## Task1.cs

```
using System;
using System.Collections.Generic;

public class Bird
{
    public Bird(string name)
    {
        Name = name;
    }

    public string Name { get; set; }

    public void Fly()
    {
        Console.WriteLine("Цей птах вміє літати");
    }

    public virtual void Sing()
    {
        Console.WriteLine("Цей птах вміє співати");
    }

    public void Eggs()
    {
        Console.WriteLine("Цей птах несе яйця");
    }

    public virtual void Child()
    {
        Console.WriteLine("Цей птах висиджує пташенят");
    }

    public override int GetHashCode()
    {
        Console.WriteLine("Hash Code: ");
        return (Name + this.GetType()).GetHashCode();
    }

    public override string ToString()
    {
        Console.WriteLine("Ім'я та тип об'єкту:");
        return $"{Name}, {this.GetType()}";
    }

    public override bool Equals(object obj)
    {
        Console.WriteLine("\nЧи однакові об'єкти?");
        if (obj is Bird bird) return Name == bird.Name && obj.GetType() ==
this.GetType();
        return false;
    }
}

public class Cuckoo : Bird
{
    public Cuckoo(string name) : base(name)
    {
    }

    public override void Child()
    {
        Console.WriteLine("Цей птах не висиджує пташенят");
    }
}
```

```

    }
}

public class Chicken : Cuckoo
{
    public Chicken(string name) : base(name)
    {
    }

    public override void Sing()
    {
        Console.WriteLine("Цей птах не вміє співати");
    }
}

public class Sitter : Chicken
{
    public Sitter(string name) : base(name)
    {
    }

    public override void Child()
    {
        Console.WriteLine("Цей птах висиджує пташенят");
    }
}

public class Birds
{
    public List<Bird> List_of_Birds;
    public Birds(List<Bird> birds)
    {
        List_of_Birds = birds;
    }

    public void Print_birds()
    {
        for (int i = 0; i < List_of_Birds.Count; i++)
        {
            Console.WriteLine(List_of_Birds[i].ToString());
            List_of_Birds[i].Fly();
            List_of_Birds[i].Sing();
            List_of_Birds[i].Eggs();
            List_of_Birds[i].Child();
        }
    }
}

```

## Task2.cs

```
using System;
using System.Collections.Generic;

public class Smartphone
{
    public string Model { get; set; }
    public string Producer { get; set; }
    public double Size { get; set; }
    public int Price { get; set; }

    public Smartphone(string model, string producer, double size, int price)
    {
        Model = model;
        Producer = producer;
        Size = size;
        Price = price;
    }
}

public class Shop
{
    public List<Smartphone> Items;
    public Shop(List<Smartphone> items)
    {
        Items = items;
    }

    public List<Smartphone> Sort_by_model(string m_to_f)
    {
        List<Smartphone> answer = new List<Smartphone> { };
        for (int i = 0; i < Items.Count; i++)
        {
            if (Items[i].Model == m_to_f)
            {
                answer.Add(Items[i]);
            }
        }
        return answer;
    }

    public List<Smartphone> Sort_by_producer(string p_to_f)
    {
        List<Smartphone> answer = new List<Smartphone> { };
        for (int i = 0; i < Items.Count; i++)
        {
            if (Items[i].Producer == p_to_f)
            {
                answer.Add(Items[i]);
            }
        }
        return answer;
    }

    public List<Smartphone> Sort_by_size(double s_to_f)
    {
        List<Smartphone> answer = new List<Smartphone> { };
        for (int i = 0; i < Items.Count; i++)
        {
            if (Items[i].Size == s_to_f)
            {
                answer.Add(Items[i]);
            }
        }
    }
}
```

```

        answer.Add(Items[i]);
    }
}
return answer;
}

public List<Smartphone> Sort_by_price(int minp_to_f = 0, int maxp_to_f =
100000000)
{
    List<Smartphone> answer = new List<Smartphone> { };
    for (int i = 0; i < Items.Count; i++)
    {
        if (Items[i].Price >= minp_to_f && Items[i].Price <= maxp_to_f)
        {
            answer.Add(Items[i]);
        }
    }
    return answer;
}

public List<Smartphone> Find_by_parameters(string m_to_f = null, string p_to_f =
null, double s_to_f = 0)
{
    List<Smartphone> answer = new List<Smartphone> { };
    for (int i = 0; i < Items.Count; i++)
    {
        if ((Items[i].Model == m_to_f || m_to_f == null) && (Items[i].Produser
== p_to_f || p_to_f == null) && (Items[i].Size == s_to_f || s_to_f == 0))
        {
            answer.Add(Items[i]);
        }
    }
    return answer;
}

public int Number_of_phones()
{
    return Items.Count;
}

public void Print_phones(string title = "")
{
    Console.WriteLine(title);
    for (int i = 0; i < Items.Count; i++)
    {
        Console.WriteLine("Модель: " + Items[i].Model + " Виробник: " +
Items[i].Produser + " Розмір дисплею: " + Items[i].Size + " Ціна: " +
Items[i].Price);
    }
}
}

```



## Результат виконання програми

Оберіть завдання 1 або 2:

1

Завдання 1:

Цей птах вміє літати

Цей птах не вміє співати

Цей птах несе яйця

Цей птах висиджує пташенят

Hash Code:

803937537

Ім'я та тип об'єкту:

Ряба, Sitter

Чи однакові об'єкти?

False

Hash Code:

-145590634

Ім'я та тип об'єкту:

Ряба, Sitter

Чи однакові об'єкти?

False

Hash Code:

1791297079

Ім'я та тип об'єкту:

Ряба2, Sitter

```
Чи однакові об'єкти?  
False  
Hash Code:  
-1350141259  
Ім'я та тип об'єкту:  
  
Ряба, Cuckoo  
  
Чи однакові об'єкти?  
False  
5  
5  
  
Список птах?в:  
Ім'я та тип об'єкту:  
Ряба, Sitter  
Цей птах вміє літати  
Цей птах не вміє співати  
Цей птах несе яйця  
Цей птах висиджує пташенят  
Ім'я та тип об'єкту:  
  
Ряба, Sitter  
Цей птах вміє літати  
Цей птах не вміє співати  
Цей птах несе яйця  
Цей птах висиджує пташенят  
Ім'я та тип об'єкту:  
  
Ряба2, Sitter  
Цей птах вміє літати  
Цей птах не вміє співати  
Цей птах несе яйця  
Цей птах висиджує пташенят  
Ім'я та тип об'єкту:  
  
Ряба, Cuckoo  
Цей птах вміє літати  
Цей птах вміє співати  
Цей птах несе яйця  
Цей птах не висиджує пташенят
```

## Висновок

Під час виконання цієї лабораторної роботи ми ознайомились з поліморфізмом, наслідуванням та перегрузкою методів.