

Лабораторна робота № 5

ІНТЕРФЕЙСИ В С#.

Мета: Набути умінь і навичок створення та реалізації інтерфейсів С# у середовищі Microsoft Visual Studio 2022.

Варіант 15

Завдання до лабораторної роботи

Реалізувати інтерфейси і наслідування для класів згідно індивідуального завдання:

Варіант 15. interface Лікар → class Хірург → class Нейрохірург.

Створив Program.cs, інтерфейс Doctor.cs, класи Surgeon.cs та NeuroSurgeon.cs.

Код із файлу Program.cs:

```
using System;
```

```
namespace lab5_Novak_prp
```

```
{
```

```
    class Program
```

```
    {
```

```
        static void Main(string[] args)
```

```
        {
```

```
            Surgeon surgeon = new Surgeon();
```

```
            NeuroSurgeon neuroSurgeon = new NeuroSurgeon();
```

```
            surgeon.TreatPatient();
```

```
            neuroSurgeon.TreatPatient();
```

```
        Console.ReadLine();  
    }  
}  
}
```

Код із файлу Doctor.cs:

```
using System;  
using System.Collections.Generic;  
using System.Linq;  
using System.Text;  
using System.Threading.Tasks;
```

```
namespace lab5_Novak_prp  
{  
    interface Doctor  
    {  
        void TreatPatient();  
    }  
}
```

```
using System;  
using System.Collections.Generic;  
using System.Linq;  
using System.Text;  
using System.Threading.Tasks;
```

```
namespace lab5_Novak_prp  
{
```

```
interface Doctor
{
    void TreatPatient();
}
}
```

Код із файлу Surgeon.cs:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace lab5_Novak_prp
{
    class Surgeon : Doctor
    {
        public void TreatPatient()
        {
            Console.WriteLine("Surgeon is treating the patient.");
        }
    }
}
```

Код із файлу NeuroSurgeon.cs:

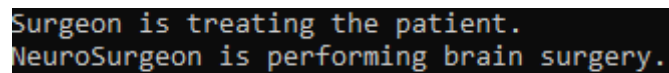
```
using System;
using System.Collections.Generic;
using System.Linq;
```

```

using System.Text;
using System.Threading.Tasks;

namespace lab5_Novak_prp
{
    class NeuroSurgeon : Surgeon
    {
        public void TreatPatient()
        {
            Console.WriteLine("NeuroSurgeon is performing brain surgery.");
        }
    }
}

```



```

Surgeon is treating the patient.
NeuroSurgeon is performing brain surgery.

```

Рис. 1 – Результат реалізації інтерфейсу та класів завдання

Контрольні питання.

1. Що являє собою інтерфейс?

У C# .NET інтерфейс визначає контракт, який клас повинен реалізувати. Він містить лише сигнатури методів, властивостей або подій без їх реалізації. Інтерфейси дозволяють створювати більш гнучкі та розширювані програми шляхом використання поліморфізму.

2. Як реалізується інтерфейс класом?

Для реалізації інтерфейсу клас має використовувати ключове слово `implements`. Наприклад:

```

class MyClass : IMyInterface

```

```
{  
    // Реалізація методів інтерфейсу  
}
```

3. Чому інтерфейс можна назвати заміною абстрактному класу?

Інтерфейс може замінити абстрактний клас, оскільки вони обидва визначають контракт, але інтерфейси не мають реалізації методів, властивостей або подій. Використання інтерфейсів дозволяє класам реалізувати багато контрактів, що робить їх більш гнучкими.

4. Які елементи інтерфейсу можна реалізувати за замовченням?

В C# .NET елементи інтерфейсу за замовчуванням не реалізуються. Інтерфейс може містити лише сигнатури методів, властивостей або подій, але жоден з них не має реалізації за замовчуванням.

5. Як в класі реалізувати декілька інтерфейсів?

Для реалізації декількох інтерфейсів клас повинен реалізувати всі методи, властивості або події, визначені в цих інтерфейсах. Наприклад:

```
class MyClass : IMyInterface1, IMyInterface2  
{  
    // Реалізація методів інтерфейсів IMyInterface1 та IMyInterface2  
}
```

6. В яких випадках потрібно використовувати перетворення типів для створення посилання на інтерфейс?

Перетворення типів для створення посилання на інтерфейс може бути корисним, коли вам потрібно робити змінну або об'єкт загальним для різних класів, які реалізують один і той самий інтерфейс. Це дозволяє вам працювати з об'єктами, використовуючи лише методи, визначені в цьому інтерфейсі.