

Министерство образования Республики Беларусь  
Учреждение образования  
«Брестский Государственный технический университет»  
Кафедра ИИТ

**Лабораторная работа №6**

По дисциплине «Современные платформы программирования»

**Выполнил:**

Студент 3 курса

Группы ПО-8

Лобарев А.М.

**Проверил:**

Крощенко А.А.

Брест 2024

**Цель работы:** приобрести навыки применения паттернов проектирования при решении практических задач.

## Вариант 16

### Задание 1

Музыкальный магазин. Должно обеспечиваться одновременное обслуживание нескольких покупателей. Магазин должен предоставлять широкий выбор товаров различных музыкальных направлений.

Код:

```
class MusicItem
{
    public required string Name { get; set; }
    public decimal Cost { get; set; }

    public override string ToString()
    {
        return $"Item: name - {Name}, cost - {Cost:C}";
    }
}

class MusicItemPool : IEnumerable<MusicItem>
{
    private readonly ConcurrentQueue<MusicItem> items = new();

    // Инициализация пула объектов
    public MusicItemPool(int initialCapacity)
    {
        for (int i = 0; i < initialCapacity; i++)
        {
            items.Enqueue(new MusicItem() { Name = string.Empty });
        }
    }

    public IEnumerator<MusicItem> GetEnumerator()
    {
        return items.GetEnumerator();
    }

    // Аренда объекта из пула
    public MusicItem Rent()
    {
        if (items.TryDequeue(out var item))
        {
            return item;
        }
        else
        {
            // Создаем новый объект, если пул пуст
            return new MusicItem() { Name = string.Empty };
        }
    }

    // Возврат объекта в пул
    public void Return(MusicItem item)
    {
        items.Enqueue(item);
    }
}
```

```

        IEnumerator IEnumerable.GetEnumerator()
        {
            return items.GetEnumerator();
        }
    }

class Program
{
    static void Main()
    {
        CultureInfo.CurrentCulture = CultureInfo.GetCultureInfo("en-US");

        var musicItemPool = new MusicItemPool(initialCapacity: 2);

        var task1 = new Task(() =>
        {
            var item1 = musicItemPool.Rent();
            item1.Name = "Album A";
            item1.Cost = 5.97m;
            musicItemPool.Return(item1);
        });

        var task2 = new Task(() =>
        {
            var item2 = musicItemPool.Rent();
            item2.Name = "Album B";
            item2.Cost = 10.61m;
            musicItemPool.Return(item2);
        });

        task1.Start();
        task2.Start();

        Task.WaitAll(task1, task2);

        foreach (var item in musicItemPool)
        {
            Console.WriteLine(item);
        }
    }
}

```

```

Консоль отладки Microsoft \
Item: name - Album B, cost - $10.61
Item: name - Album A, cost - $5.97

C:\Users\Gygabyte\source\repos\Studies\6 sem\SPP\Lab6\Task1\bin\Debug\net8.0\Task1.exe (процесс 10592) завершил работу с кодом 0.
Нажмите любую клавишу, чтобы закрыть это окно:

```

## Задание 2

Учетная запись покупателя книжного интернет-магазина. Предусмотреть различные уровни учетки в зависимости от активности покупателя. Дополнительные уровни добавляют функциональные возможности и открывают доступ к уникальным предложениям.

```
interface IAccountStrategy
{
    void ShowAccountInfo(string username);
    void ShowUniqueOffers();
}

class BasicAccountStrategy : IAccountStrategy
{
    public void ShowAccountInfo(string username)
    {
        Console.WriteLine($"Basic Account Info for {username}");
    }

    public void ShowUniqueOffers()
    {
        Console.WriteLine("No unique offers for Basic Account");
    }
}

class PremiumAccountStrategy : IAccountStrategy
{
    public void ShowAccountInfo(string username)
    {
        Console.WriteLine($"Premium Account Info for {username}");
    }

    public void ShowUniqueOffers()
    {
        Console.WriteLine("Access to premium book previews and exclusive discounts");
    }
}

// Контекст (класс, использующий стратегии)
class AccountContext
{
    private IAccountStrategy _strategy;

    public void SetStrategy(IAccountStrategy strategy)
    {
        _strategy = strategy;
    }

    public void ShowAccountInfo(string username)
    {
        _strategy.ShowAccountInfo(username);
    }

    public void ShowUniqueOffers()
    {
        _strategy.ShowUniqueOffers();
    }
}
```

```

class Program
{
    static void Main()
    {
        var basicAccount = new BasicAccountStrategy();
        var premiumAccount = new PremiumAccountStrategy();

        var context = new AccountContext();

        context.SetStrategy(basicAccount);
        context.ShowAccountInfo("user123");
        context.ShowUniqueOffers();

        Console.WriteLine();

        context.SetStrategy(premiumAccount);
        context.ShowAccountInfo("premiumUser");
        context.ShowUniqueOffers();
    }
}

```

```

Консоль отладки Microsoft V
Basic Account Info for user123
No unique offers for Basic Account

Premium Account Info for premiumUser
Access to premium book previews and exclusive discounts

C:\Users\Gygabyte\source\repos\Studies\6 sem\SPP\Lab6\Task2\bin\Debug\net8.0\Task2.exe (процесс 26324) завершил работу с кодом 0.
Нажмите любую клавишу, чтобы закрыть это окно:

```

### Задание 3

Проект «Принтер». Предусмотреть выполнение операций (печать, загрузка бумаги, извлечение зажатой бумаги, заправка картриджа), режимы – ожидание, печать документа, зажатие бумаги, отказ – при отсутствии бумаги или краски, атрибуты – модель, количество листов в лотке, % краски в картридже, вероятность зажатия.

```

interface IPrinterState
{
    void Print();
    void LoadPaper();
    void ExtractJam();
    void RefillCartridge();
}

```

```
class IdleState : IPrinterState
{
    public void Print()
    {
        Console.WriteLine("Printer is idle. Please select a document to print.");
    }

    public void LoadPaper()
    {
        Console.WriteLine("Loading paper...");
    }

    public void ExtractJam()
    {
        Console.WriteLine("No paper jam detected.");
    }

    public void RefillCartridge()
    {
        Console.WriteLine("Cartridge is already full.");
    }
}
```

```
class PrintingState : IPrinterState
{
    public void Print()
    {
        Console.WriteLine("Printing document...");
    }

    public void LoadPaper()
    {
        Console.WriteLine("Cannot load paper while printing.");
    }

    public void ExtractJam()
    {
        Console.WriteLine("Cannot extract jam while printing.");
    }

    public void RefillCartridge()
    {
        Console.WriteLine("Cannot refill cartridge while printing.");
    }
}
```

```
class JammedState : IPrinterState
{
    public void Print()
    {
        Console.WriteLine("Cannot print due to paper jam.");
    }

    public void LoadPaper()
    {
        Console.WriteLine("Cannot load paper while jammed.");
    }
}
```

```

    public void ExtractJam()
    {
        Console.WriteLine("Extracting paper jam...");
    }

    public void RefillCartridge()
    {
        Console.WriteLine("Cannot refill cartridge while jammed.");
    }
}

```

```

class Printer
{
    private IPrinterState _currentState;

    public Printer()
    {
        _currentState = new IdleState();
    }

    public void SetState(IPrinterState state)
    {
        _currentState = state;
    }

    public void Print()
    {
        _currentState.Print();
    }

    public void LoadPaper()
    {
        _currentState.LoadPaper();
    }

    public void ExtractJam()
    {
        _currentState.ExtractJam();
    }

    public void RefillCartridge()
    {
        _currentState.RefillCartridge();
    }
}

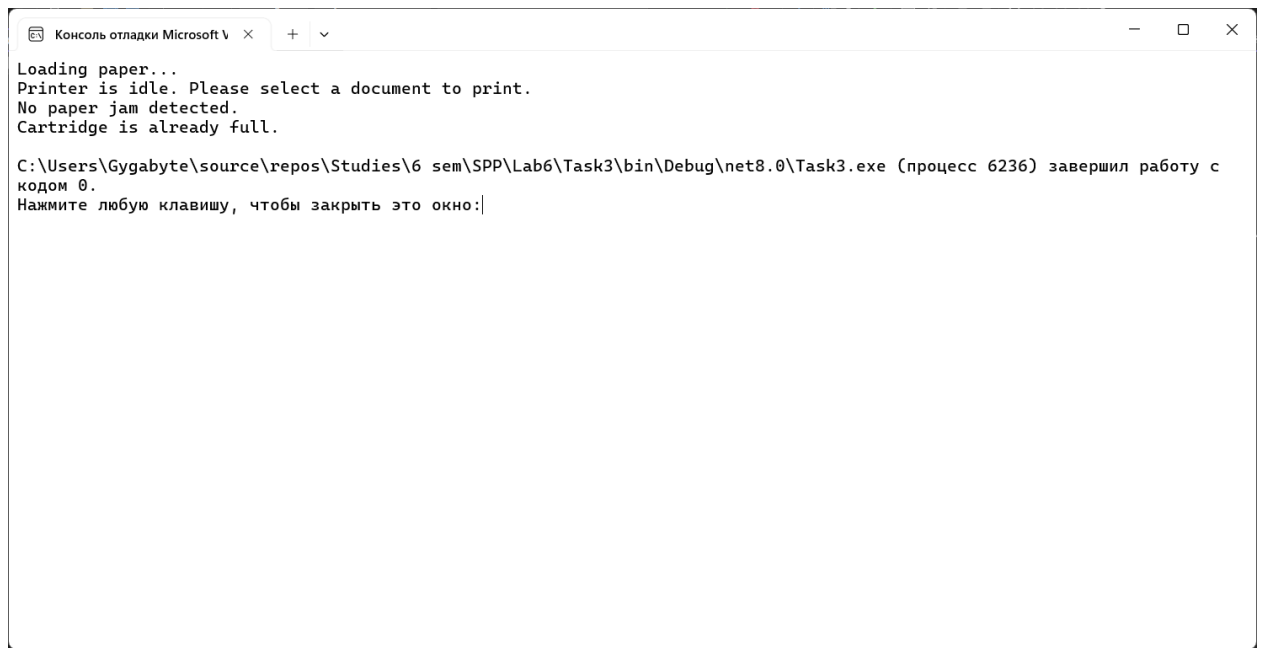
```

```

class Program
{
    static void Main()
    {
        var printer = new Printer();

        printer.LoadPaper();
        printer.Print();
        printer.ExtractJam();
        printer.RefillCartridge();
    }
}

```



```
Консоль отладки Microsoft Visual Studio
Loading paper...
Printer is idle. Please select a document to print.
No paper jam detected.
Cartridge is already full.

C:\Users\Gygabyte\source\repos\Studies\6 sem\SPP\Lab6\Task3\bin\Debug\net8.0\Task3.exe (процесс 6236) завершил работу с кодом 0.
Нажмите любую клавишу, чтобы закрыть это окно:
```

**Вывод:** приобрел навыки применения паттернов проектирования при решении практических задач.