

Министерство науки и высшего образования Российской Федерации  
Муромский институт (филиал)  
Федерального государственного бюджетного образовательного учреждения  
высшего образования  
«Владимирский государственный университет  
имени Александра Григорьевича и Николая Григорьевича Столетовых»

Факультет \_\_\_\_\_ ИТР \_\_\_\_\_

Кафедра \_\_\_\_\_ ПИИ \_\_\_\_\_

## **ЛАБОРАТОРНАЯ РАБОТА №7**

По Объектно-ориентированному программированию

Руководитель

Привезенцев Д.Г.

(фамилия, инициалы)

\_\_\_\_\_  
(подпись)

\_\_\_\_\_  
(дата)

Студент ПИИ - 121

(группа)

Ермилов М.В.

(фамилия, инициалы)

\_\_\_\_\_  
(подпись)

\_\_\_\_\_  
(дата)

Муром 2022

## Лабораторная работа №7

**Тема:** Изучение и практическое применение структурного шаблона проектирования Адаптер

Ход работы:

**Задание :**

1. Реализовать примеры из методических указаний и продемонстрировать их работу.
2. Реализовать иерархию классов с применением шаблона Адаптер объекта и Адаптер класса согласно варианту задания.

Таблица 1 - задание

№ Варианта	Требуемый интерфейс	Адаптируемый класс
4, 16	<b>void ModifMass(int dm)</b> – изменить массу маятника на величину dm; <b>double CalculateW()</b> – определить циклическую частоту колебаний; <b>string GetData()</b> – возвращает строку с данными об объекте	Математический маятник <b>double a</b> – амплитуда колебаний; <b>double d</b> – длина нерастяжимой нити; int m – масса маятника; <b>double X(int t)</b> – определить смещение маятника в момент времени t; <b>double CalculateT()</b> – определить период колебаний; <b>string ToString()</b> – возвращает строку с данными об объекте

					МИ ВлГУ 09.03.04		
Изм.	Лист	№ докум.	Подпись	Дата			
Разраб.	Ермилов М.В.				Изучение и практическое применение структурного шаблона проектирования Адаптер	Лит.	Лист
Провер.	Привезенцев Д.Г.						
Реценз.						2	3
Н. Контр.						ПИН-121	
Утверд.	.						

Код по заданию:

Класс MathPendulum:

```
class MathPendulum
{
    public double a, d;
    public int m;
    public MathPendulum(double a, double d, int m)
    {
        this.m = m;
        this.a = a;
        this.d = d;
    }
    public double X(int t) => a * Math.Cos(t * Math.Sqrt(9.81 / d));
    public double CalculateT() => 2 * Math.PI * Math.Sqrt(d / 9.81);
    public override string ToString() => $"a : {a}; b : {d}; m : {m}";
}
```

Интерфейс ITarget:

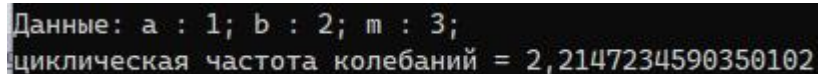
```
interface ITarget
{
    public void ModifMass(int dm);
    public double CalculateW();
    public string GetData();
}
```

Класс Adapter:

```
class Adapter : ITarget
{
    private MathPendulum mathP;
    public Adapter(MathPendulum mathP) => this.mathP = mathP;
    public double CalculateW() => 2 * Math.PI / mathP.CalculateT();
    public void ModifMass(int dm) => mathP.m = dm;
    public string GetData() => mathP.ToString();
}
```

Основной класс:

```
class Program
{
    static void Main(string[] args)
    {
        MathPendulum m = new MathPendulum(1, 2, 3);
        ITarget a = new Adapter(m);
        Console.WriteLine($"Данные: {a.GetData()}");
        Console.WriteLine($"циклическая частота колебаний = { a.CalculateW()}");
    }
}
```



```
Данные: a : 1; b : 2; m : 3;
циклическая частота колебаний = 2,2147234590350102
```

Рис 1 - пример работы программы