МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное бюджетное образовательное учреждение

высшего образования

**«УЛЬЯНОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Кафедра «Вычислительная техника»

Дисциплина «Технология программирования»

**Лабораторная работа №4.**

**Стратегия (Strategy)**

Выполнил:

студент группы ИВТАПбд-21

Вершинин Д. В.

Проверил:

Фролов В. А.

Ульяновск, 2018

Паттерн Стратегия (Strategy) представляет шаблон проектирования, который определяет набор алгоритмов, инкапсулирует каждый из них и обеспечивает их взаимозаменяемость. В зависимости от ситуации мы можем легко заменить один используемый алгоритм другим. При этом замена алгоритма происходит независимо от объекта, который использует данный алгоритм.

### Когда использовать стратегию?

* Когда есть несколько родственных классов, которые отличаются поведением. Можно задать один основной класс, а разные варианты поведения вынести в отдельные классы и при необходимости их применять
* Когда необходимо обеспечить выбор из нескольких вариантов алгоритмов, которые можно легко менять в зависимости от условий
* Когда необходимо менять поведение объектов на стадии выполнения программы
* Когда класс, применяющий определенную функциональность, ничего не должен знать о ее реализации

**Техническое задание**

Реализовать паттерн и продемонстрировать его работу с различными типами объектов.

Исходный код

class Program

{

static void Main(string[] args)

{

Student student = new Student("Дмитрий", 19, new StudyTime());

student.Study();

student.studying = new Weekend();

student.Study();

Console.ReadLine();

}

}

interface IStudying

{

void Study();

}

class StudyTime : IStudying

{

public void Study()

{

Console.WriteLine("I'm studying!!!");

}

}

class Weekend : IStudying

{

public void Study()

{

Console.WriteLine("It's not studying time!!!");

}

}

class Student

{

protected int age;

protected string name;

public IStudying studying { private get; set; }

public Student(string name, int age, IStudying study)

{

this.age = age;

this.name = name;

studying = study;

}

public void Study()

{

studying.Study();

}

}