Министерство образования Республики Беларусь

Учреждение образования «Белорусский государственный университет

информатики и радиоэлектроники»

Факультет компьютерных систем и сетей

Кафедра Информатики

Дисциплина «Программирование»

**ОТЧЕТ**

к лабораторной работе №8

на тему:

**«Полиморфизм»**

БГУИР 6-05-0612-02 67

|  |
| --- |
| Выполнил студент группы 353503  КОХАН Артём Игоревич |
|  |
| (дата, подпись студента) |
| Проверил ассистент каф. Информатики  РОМАНЮК Максим Валерьевич |
|  |
| (дата, подпись преподавателя) |

Минск 2024

# 1 Индивидуальное задание

**Задание 1. Вариант 7.** Предметная область: Фирма грузоперевозок.

Фирма имеет список тарифов по перевозке грузов. Класс тариф хранит наименование тарифа и цену. На некоторые тарифы предоставлена скидка, заданная в процентах. В классе фирма реализовать методы добавления нового тарифа и метод поиска тарифа с минимальной стоимостью. Составить диаграмму классов проектируемой системы. Запрограммировать классы в соответствии с новой диаграммой. Проиллюстрировать использование интерфейсов. Показать вызов метода интерфейса через интерфейсную ссылку. Применить в программе шаблон проектирования Strategy.

# 2 Выполнение работы

Перед выполнением работы следует разработать диаграмму классов для наглядного выполнения поставленной задачи (см. рисунок 1).

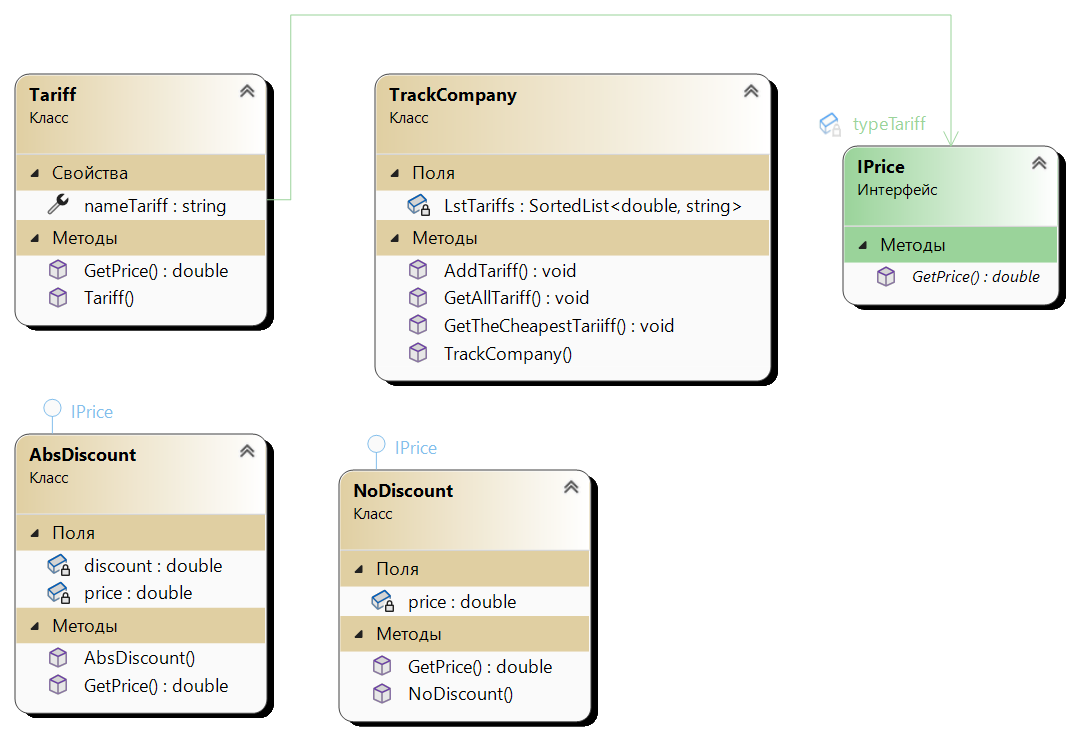


Рисунок 1 – Диаграмма классов

Для выполнения задания был создан интерфейс IPrice, где определён метод GetPrice, который должен будет возвращать стоимость тарифа со скидкой или без неё.

namespace Task1.Classes

{

internal interface IPrice

{

double GetPrice();

}

}

Сперва рассмотрим один из классов, который реализует интерфейс IPrice, AbsDiscount при создании объекта этого класса требуется указать стоимость тарифа и скидку на его, после этого идёт расчёт стоимости в методе GetPrice.

namespace Task1.Classes

{

internal class AbsDiscount : IPrice

{

private double price;

private double discount;

public AbsDiscount(double price, double discount)

{

this.price = price;

this.discount = discount;

}

public double GetPrice() =>

price \* (100 - discount) / 100;

}

}

Также имеет класс NoDiscount, который также реализует интерфейс IPrice, но при у него на одно поле меньше, так как класс представляет собой тариф без скидки.

namespace Task1.Classes

{

internal class NoDiscount : IPrice

{

private double price;

public NoDiscount(double price) =>

this.price = price;

public double GetPrice() => price;

}

}

Ниже будет представлен класс, который хранит само наименование тарифа, объект класса IPrice, реализующего интерфейс для получения стоимости тарифа со скидкой, либо без скидки.

namespace Task1.Classes

{

internal class Tariff

{

private IPrice typeTariff;

public string nameTariff { get; set; }

public Tariff(string nameTariff, IPrice typeTariff)

{

this.nameTariff = nameTariff;

this.typeTariff = typeTariff;

}

public double GetPrice()

{

return typeTariff.GetPrice();

}

}

}

Основой программы является класс ТrackCompany, который хранит список тех самых тарифов, эти тарифы автомастически сортируются при помощи функционала коллекции языка С#, а именно SortesList. Добавление тарифа происходит при помощи метода Add. Для общего понимания работы данных классов имеют методы GetAllTariff, данный метод при помощи цикла foreach проходится по SortedList и выводит тарифы в порядке возрастания стоимости, и GetTheCheapestTariff, который просто выводит первый элемент SortedList, так как тарифы в нём уже отсортированы в порядке возрастания.

namespace Task1.Classes

{

internal class TrackCompany

{

private SortedList<double, string> LstTariffs;

public TrackCompany() =>

LstTariffs = new SortedList<double, string>();

public void AddTariff(Tariff tariff) =>

LstTariffs.Add(tariff.GetPrice(), tariff.nameTariff);

public void GetAllTariff()

{

foreach (KeyValuePair<double, string> entry in LstTariffs)

{

Console.WriteLine("Стоимость тарифа: {0}$, Название тарифа: {1}\n", entry.Key, entry.Value);

}

}

public void GetTheCheapestTariiff() =>

Console.WriteLine("Стоимость тарифа: {0}$, Название тарифа: {1}\n", LstTariffs.Keys[0], LstTariffs.Values[0]);

}

}

Покажем принцип работы этой программы в классе Program. Создадим 5 тарифов и зададим им стоимость и скидку, продемонстрируем работу методов GetAllTariff и GetTheCheapestTariff.

namespace Task1

{

file class Program

{

static void Main(string[] args)

{

TrackCompany myCompany = new TrackCompany();

myCompany.AddTariff(new Tariff("Быстрый", new NoDiscount(30)));

myCompany.AddTariff(new Tariff("Ускоренный люкс", new AbsDiscount(200, 35)));

myCompany.AddTariff(new Tariff("Экономный", new AbsDiscount(20, 15)));

myCompany.AddTariff(new Tariff("Безопасный", new AbsDiscount(30, 5)));

myCompany.AddTariff(new Tariff("Люкс", new NoDiscount(100)));

Console.WriteLine("Выведем список всех тарифов:");

myCompany.GetAllTariff();

Console.WriteLine("Выведем самый дешёвый тариф: ");

myCompany.GetTheCheapestTariiff();

}

}

}

Результат работы программы продемонстрирован ниже (см. рисунок 2).

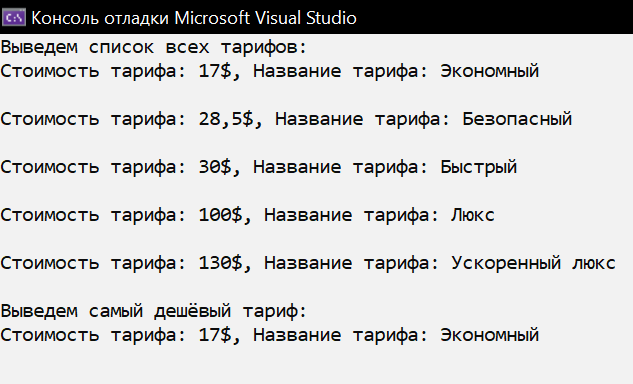


Рисунок 2 – Результат работы программы

# Вывод

В ходе лабораторной работы были изучены базовые принципы работы с интерфейсами и освоен принцип работы с шаблоном проектирования Strategy.