Министерство образования Республики Беларусь

Учреждение образования

«Брестский государственный технический университет»

Кафедра ИИТ

**Лабораторная работа №6**

По дисциплине «СПП»

за 5-й семестр

Выполнил:

студент 2 курса

группы ПО-3 (1)

Афанасьев В.В.

Проверил:

Крощенко А.А.

Брест, 2020

**Цель работы:** приобрести навыки использования паттернов проектирования при решении практических задач на языке программирования C#.

**Вариант:** 2

**Задание 1:**

Заводы по производству автомобилей. Реализовать возможность создавать автомобили различных типов на различных заводах.

**Задание 2:**

Проект «Универсальная электронная карта». В проекте должна быть реализована универсальная электронная карта, в которой есть функции паспорта, страхового полиса, банковской картыи т. д.

**Задание 3:**

Проект «Принтеры». В проекте должны быть реализованы разные модели принтеров, которыевыполняют разные виды печати.

**Код программы:**

**1)**

using System;

using System.Dynamic;

namespace task1

{

class Program

{

static void Main(string[] args)

{

Client pclient = new Client(new PetroleFactory());

pclient.TestingRun();

Client dclient = new Client(new DieselFactory());

dclient.TestingRun();

}

}

abstract class AbstractFactory

{

public abstract PassengerCar CreatePassengerCar();

public abstract TruckCar CreateTruckCar();

}

class DieselFactory : AbstractFactory

{

public override PassengerCar CreatePassengerCar()

{

return new DieselPassengerCar();

}

public override TruckCar CreateTruckCar()

{

return new DieselTruckCar();

}

}

class PetroleFactory : AbstractFactory

{

public override PassengerCar CreatePassengerCar()

{

return new PetrolePassengerCar();

}

public override TruckCar CreateTruckCar()

{

return new PetroleTruckCar();

}

}

abstract class PassengerCar

{

public abstract void GetInfo();

}

abstract class TruckCar

{

public abstract void GetInfo();

}

class DieselPassengerCar : PassengerCar

{

public override void GetInfo()

{

Console.WriteLine("Diesel passenger car is working");

}

}

class PetrolePassengerCar : PassengerCar

{

public override void GetInfo()

{

Console.WriteLine("Petrole passenger car is working");

}

}

class DieselTruckCar : TruckCar

{

public override void GetInfo()

{

Console.WriteLine("Diesel truck car is working");

}

}

class PetroleTruckCar : TruckCar

{

public override void GetInfo()

{

Console.WriteLine("Petrole truck car is working");

}

}

class Client

{

private AbstractFactory \_factory;

public Client(AbstractFactory factory)

{

\_factory = factory;

}

public PassengerCar GetPassengerCar()

{

return \_factory.CreatePassengerCar();

}

public TruckCar GetTruckCar()

{

return \_factory.CreateTruckCar();

}

public void TestingRun() {

GetPassengerCar().GetInfo();

GetTruckCar().GetInfo();

}

}

}

**2)**

using System;

namespace task2

{

class Program

{

static void Main(string[] args)

{

ElectronicCard myCard = new ElectronicCard(new Passport(), new InsurancePolicy(), new BankCard());

myCard.ShowInfo();

myCard.ShowInsurance();

myCard.PayFor();

}

}

class Passport

{

public void ShowInfo()

{

Console.WriteLine("You show the information");

}

}

class InsurancePolicy

{

public void ShowInsurance()

{

Console.WriteLine("You show the insurance");

}

}

class BankCard

{

public void PayFor()

{

Console.WriteLine("You have paid for");

}

}

class ElectronicCard

{

Passport passport;

InsurancePolicy insurancePolic;

BankCard bankCard;

public ElectronicCard(Passport pass, InsurancePolicy insur, BankCard bank)

{

passport = pass;

insurancePolic = insur;

bankCard = bank;

}

public void ShowInfo()

{

passport.ShowInfo();

}

public void ShowInsurance()

{

insurancePolic.ShowInsurance();

}

public void PayFor()

{

bankCard.PayFor();

}

}

}

**3)**

using System;

namespace task3

{

class Program

{

static void Main(string[] args)

{

Context matrixPrinter = new Context(new MatrixPrinter());

matrixPrinter.ExecuteAlgorithm();

Context laserPrinter = new Context(new LaserPrinter());

laserPrinter.ExecuteAlgorithm();

Context colorPrinter = new Context(new ColorPrinter());

colorPrinter.ExecuteAlgorithm();

}

}

public interface IPrinter

{

void Print();

}

public class ColorPrinter : IPrinter

{

public void Print()

{

Console.WriteLine("Color print");

}

}

public class MatrixPrinter : IPrinter

{

public void Print()

{

Console.WriteLine("Matrix print");

}

}

public class LaserPrinter : IPrinter

{

public void Print()

{

Console.WriteLine("Laser print");

}

}

public class Context

{

public IPrinter ContextPrinter { get; set; }

public Context(IPrinter type)

{

ContextPrinter = type;

}

public void ExecuteAlgorithm()

{

ContextPrinter.Print();

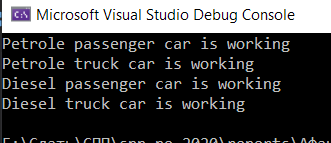
}

}

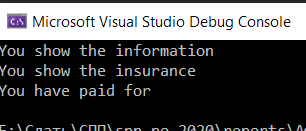
}

**Результаты работы:**

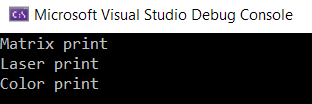
**1)**

****

**2)**

****

**3)**

****

**Выводы:** в ходе выполнения лабораторной работы были получены навыки использования паттернов проектирования при решении практических задач на языке программирования C#.