Белорусско-Российский университет

Кафедра ПОИТ

Дисциплина ООПП

Отчет по лабораторной работе №5

«ВИРТУАЛЬНЫЕ МЕТОДЫ И ПОЛИМОРФИЗМ»

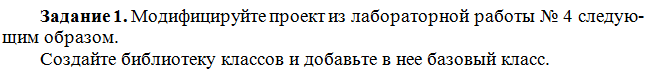
Выполнил студент группы АСОИ-181

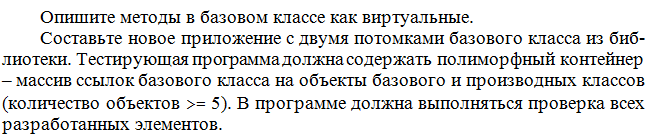
Самусев Д.А.

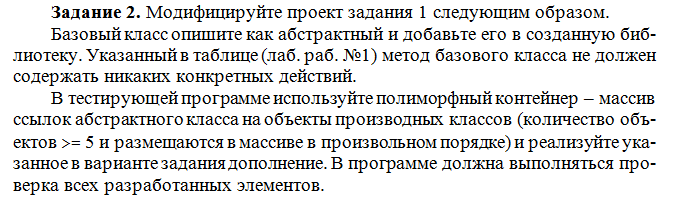
Проверил \_\_\_\_\_\_ Горбатенко Н.Н.

Могилёв 2020г

Цель работы: формирование представления о реализации принципа полиморфизма с помощью механизма позднего связывания.









Код программы :

using System;  
using Lab5.ClassLibrary;  
  
namespace Lab5  
{  
 class Program  
 {  
 public static void Main(string[] args)  
 {  
 var plane = new Plane("Plane", 10, 10, 10, 10);  
 var expensiveShip = new Ship("ExpensiveShip", 15, 15, 5, 3);  
 var ship = new Ship("Ship", 15, 15, 5, 2);  
 var vehicle2 = new Vehicle("Vehicle2", 15, 10);  
 var vehicle1 = new Vehicle("Vehicle1", 15, 15);  
   
 var vehicles = new Vehicle[]  
 {  
 plane, expensiveShip, ship, vehicle2, vehicle1  
 };  
  
 foreach (var vehicle in vehicles)  
 {  
 Console.WriteLine($"{vehicle.Name} total price - {vehicle.CalculateTravelPrice()}");  
 }  
   
 Task2();  
   
 Console.ReadKey();  
 }  
  
 public static void Task2()  
 {  
 var planeTwo1 = new PlaneTwo("Plane1", 10, 10, 11, 10);  
 var shipTwo1 = new ShipTwo("Ship1", 15, 15, 5, 3);  
 var planeTwo2 = new PlaneTwo("Plane2", 10, 10, 12, 10);  
 var shipTwo2 = new ShipTwo("Ship2", 15, 15, 6, 2);  
 var planeTwo3 = new PlaneTwo("Plane3", 10, 10, 14, 10);  
   
 var vehicles = new BaseVehicle[]  
 {  
 planeTwo1, shipTwo1, planeTwo2, shipTwo2, planeTwo3  
 };  
   
 foreach (var vehicle in vehicles)  
 {  
 Console.WriteLine($"{vehicle.Name} total price - {vehicle.CalculateTravelPrice()}");  
 }  
   
 Console.ReadKey();  
 }  
 }  
}

using Lab5.ClassLibrary;  
  
namespace Lab5  
{  
 public class PlaneTwo : BaseVehicle  
 {  
 private double \_height;  
 private double \_speed;  
   
 public PlaneTwo(string name, double distance, double price, double height, double speed)  
 {  
 \_name = name;  
 \_distance = distance;  
 \_price = price;  
 \_height = height;  
 \_speed = speed;  
 }  
  
 public override double CalculateTravelPrice()  
 {  
 \_price \*= \_height \* \_speed;  
   
 return \_distance \* \_price;  
 }  
 }  
}

using System;  
using Lab5.ClassLibrary;  
  
namespace Lab5  
{  
 public class ShipTwo : BaseVehicle  
 {  
 private int \_deckCount;  
 private int \_deckNumber;  
   
 public ShipTwo(string name, double distance, double price, int deckCount, int deckNumber)  
 {  
 \_name = name;  
 \_distance = distance;  
 \_price = price;  
 \_deckCount = deckCount;  
 \_deckNumber = deckNumber;  
 }  
  
 public override double CalculateTravelPrice()  
 {  
 if (\_deckNumber == 3 || \_deckNumber == 4)  
 {  
 \_price += \_price / 100 \* Math.Pow(\_deckCount, 2);  
 }  
   
 return \_distance \* \_price;  
 }  
 }  
}

namespace Lab5.ClassLibrary  
{  
 public abstract class BaseVehicle  
 {  
 protected string \_name;  
 protected double \_distance;  
 protected double \_price;  
   
 public string Name  
 {  
 get => \_name;  
 set => \_name = value;  
 }  
   
 public abstract double CalculateTravelPrice();  
 }  
}

private int \_deckCount;  
private int \_deckNumber;  
  
public int DeckCount  
{  
 get => \_deckCount;  
 set => \_deckCount = value;  
}  
  
public int DeckNumber  
{  
 get => \_deckNumber;  
 set => \_deckNumber = value;  
}  
  
public Ship(string name, double distance,   
 double price, int deckCount, int deckNumber) :  
 base(name, distance, price)  
{  
 \_deckCount = deckCount;  
 \_deckNumber = deckNumber;  
   
 if (deckNumber == 3 || deckNumber == 4)  
 {  
 \_price += price / 100 \* Math.Pow(deckCount, 2);  
 }  
}

namespace Lab5  
{  
 public class Plane : Vehicle  
 {  
 private double \_height;  
 private double \_speed;  
  
 public double Height  
 {  
 get => \_height;  
 set => \_height = value;  
 }  
  
 public double Speed  
 {  
 get => \_speed;  
 set => \_speed = value;  
 }  
  
 public Plane(string name, double distance,   
 double price, double height, double speed) :   
 base(name, distance, price)  
 {  
 \_price \*= height \* speed;  
 \_height = height;  
 \_speed = speed;  
 }  
 }  
}

public class Vehicle  
{  
 protected string \_name;  
 protected double \_distance;  
 protected double \_price;  
  
 public string Name  
 {  
 get => \_name;  
 set => \_name = value;  
 }  
  
 public double Distance  
 {  
 get => \_distance;  
 set => \_distance = value;  
  
 }  
 public double Price  
 {  
 get => \_price;  
 set => \_price = value;  
 }  
  
 public Vehicle()  
 {  
 \_name = "Bus";  
 Distance = 10;  
 Price = 10;  
 }  
  
 public Vehicle(string name, double distance, double price)  
 {  
 \_name = name;  
 Distance = distance;  
 Price = price;  
 }  
  
 public virtual double CalculateTravelPrice()  
 {  
 return \_distance \* \_price;  
 }  
}

