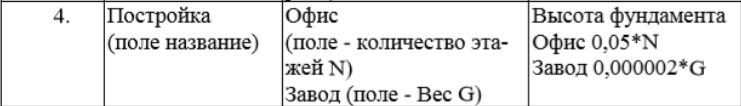
Полиморфизм

Задание №1.



Листинг программы:

using System;

namespace AbstractClassExample

{

// Abstract parent class

abstract class Building

{

public abstract void Foundation();

}

// Descendant class 1

class Office : Building

{

public int NumFloors { get; set; }

public Office(int numFloors)

{

NumFloors = numFloors;

}

public override void Foundation()

{

double foundation = 0.05 \* NumFloors;

Console.WriteLine($"Office foundation: {foundation}");

}

}

// Descendant class 2

class Factory : Building

{

public double Weight { get; set; }

public Factory(double weight)

{

Weight = weight;

}

public override void Foundation()

{

double foundation = 0.000002 \* Weight;

Console.WriteLine($"Factory foundation: {foundation}");

}

}

class Program

{

static void Main(string[] args)

{

// Create an array of Building objects

Building[] buildings = new Building[5];

// Add Office and Factory objects to the array

buildings[0] = new Office(5);

buildings[1] = new Factory(1000);

buildings[2] = new Office(10);

buildings[3] = new Factory(500);

buildings[4] = new Office(3);

// Call the Foundation method for each object in the array

foreach (Building building in buildings)

{

building.Foundation();

}

// Wait for user input before closing the console window

Console.ReadLine();

}

}

}

Анализ программы:

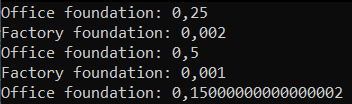


Рисунок 1.1 - Анализ программы

Источник: собственная разработка

Задание №2. Организовать вычисление суммарного расхода ткани.

Листинг программы:

using System;

namespace AbstractClassExample

{

// Abstract parent class

abstract class Fabric

{

protected double costPerMeter;

protected double length;

public Fabric(double costPerMeter, double length)

{

this.costPerMeter = costPerMeter;

this.length = length;

}

// Abstract method to calculate the total cost of the fabric

public abstract double CalculateCost();

// Method to display the fabric details

public virtual void DisplayDetails()

{

Console.WriteLine($"Стоимость за метр: {costPerMeter:C}");

Console.WriteLine($"Длина: {length} метров");

}

}

// First child class

class Cotton : Fabric

{

public Cotton(double costPerMeter, double length) : base(costPerMeter, length)

{

}

// Implementation of the abstract method to calculate the total cost of the fabric

public override double CalculateCost()

{

return costPerMeter \* length;

}

}

// Second child class

class Silk : Fabric

{

public Silk(double costPerMeter, double length) : base(costPerMeter, length)

{

}

// Implementation of the abstract method to calculate the total cost of the fabric

public override double CalculateCost()

{

return costPerMeter \* length \* 1.5;

}

}

class Program

{

static void Main(string[] args)

{

// Create an array of fabric objects

Fabric[] fabrics = new Fabric[5];

// Add cotton and silk fabrics to the array

fabrics[0] = new Cotton(10, 5);

fabrics[1] = new Silk(20, 3);

fabrics[2] = new Cotton(15, 7);

fabrics[3] = new Silk(25, 4);

fabrics[4] = new Cotton(12, 6);

// Display the details of each fabric object and calculate the total cost

double totalCost = 0;

foreach (Fabric fabric in fabrics)

{

fabric.DisplayDetails();

double cost = fabric.CalculateCost();

Console.WriteLine($"Общая стоимость: {cost:C}\n");

totalCost += cost;

}

// Display the total cost of all fabrics

Console.WriteLine($"Общая стоимость всех тканей: {totalCost:C}");

}

}

}

Анализ программы:

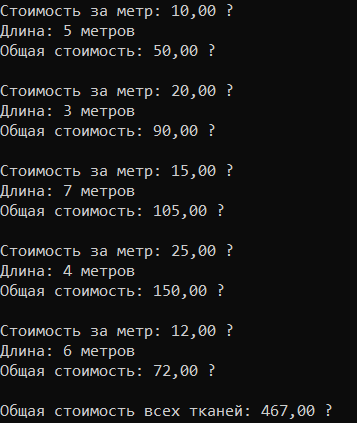


Рисунок 1.2 - Анализ программы

Источник: собственная разработка