**10 Механизм наследования. Отношения между**

**Классами**

Задание 1. Базовый класс: Товар (поле – количество, цена.) Метод: Стоимость товара. Потомок: Фломастеры (поле – название, сорт – s) Изменения в потомках: Изменить стоимость фломастеров с учетом сорта в 1/√s раз.

Листинг программы:

using System;

class Product

{

private int quantity;

private double price;

public Product(int quantity, double price)

{

this.quantity = quantity;

this.price = price;

}

public int GetQuantity() => quantity;

public double GetPrice() => price;

public void SetQuantity(int quantity) => this.quantity = quantity;

public void SetPrice(double price) => this.price = price;

public virtual double CalculateTotalCost()

{

return quantity \* price;

}

public virtual void PrintInfo()

{

Console.WriteLine($"Товар: Количество = {quantity}, Цена за единицу = {price:F2}, Общая стоимость = {CalculateTotalCost():F2}");

}

}

class Marker : Product

{

private string name;

private int grade;

public Marker(int quantity, double price, string name, int grade) : base(quantity, price)

{

this.name = name;

this.grade = grade;

}

public string GetName() => name;

public int GetGrade() => grade;

public void SetName(string name) => this.name = name;

public void SetGrade(int grade) => this.grade = grade;

public override double CalculateTotalCost()

{

return base.CalculateTotalCost() / Math.Sqrt(grade);

}

public override void PrintInfo()

{

Console.WriteLine($"Фломастер: {name}, Сорт = {grade}, Количество = {GetQuantity()}, Цена за единицу = {GetPrice():F2}, Общая стоимость = {CalculateTotalCost():F2}");

}

}

class Program

{

static void Main()

{

Console.WriteLine("Создание базового товара:");

Product product = new Product(10, 2);

product.PrintInfo();

Console.WriteLine("\nСоздание фломастеров:");

Marker marker = new Marker(10, 2, "Перманентный", 4);

marker.PrintInfo();

}

}

Анализ результатов:

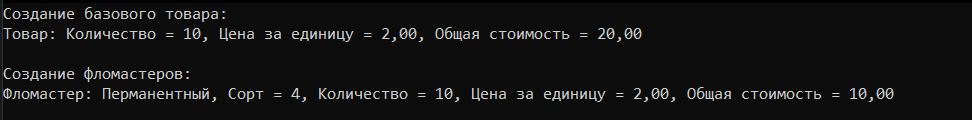


Рисунок 1 – Результат работы программы