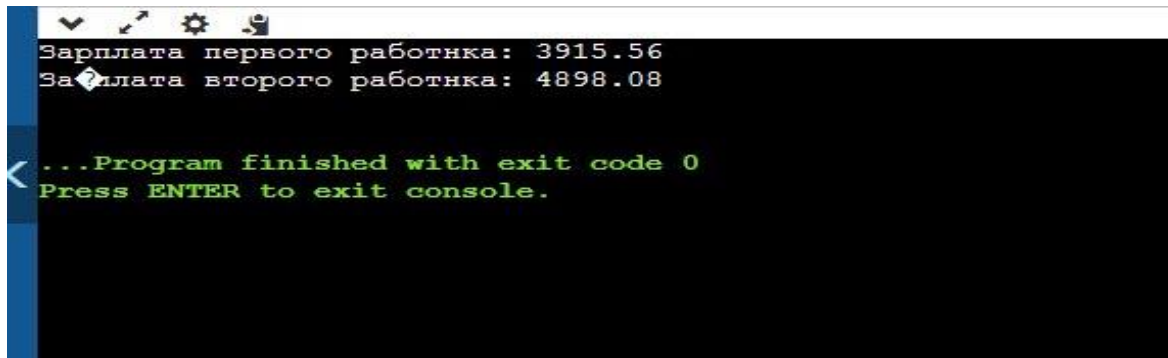


Задание 1. Реализуйте класс `Worker`, который будет иметь следующие свойства: `name`, `surname`, `rate` (ставка за день работы), `days` (количество отработанных дней). Также класс должен иметь метод `GetSalary()`, который будет выводить зарплату работника. Зарплата - это произведение ставки `rate` на количество отработанных дней `days`;

Код программы :

```
1 using System;
2
3 class Worker
4 {
5     public string name;
6     public string surname;
7     public double rate;
8     public int days;
9
10    public Worker(string name, string surname, double rate, int days)
11    {
12        //присваиваю знач аргумента name полю name(this - обращение к объекту)
13        this.name = name;
14        this.surname = surname;
15        this.rate = rate;
16        this.days = days;
17    }
18    //метод возвращает зарплату работника
19    public double GetSalary()
20    {
21        return rate * days; //вычисляю зарплату
22    }
23 }
24
25 class Program
26 {
27     static void Main()
28     {
29         Worker worker1 = new Worker("Полина ", "Злобина", 88.99, 44);
30         Worker worker2 = new Worker("Дмитрий", "Злобин", 55.66, 88);
31
32         Console.WriteLine("Зарплата первого работника: " + worker1.GetSalary());
33         Console.WriteLine("Зарплата второго работника: " + worker2.GetSalary());
34     }
35 }
36
```

Вывод программы:



```
Зарплата первого работника: 3915.56
Зарплата второго работника: 4898.08

...Program finished with exit code 0
Press ENTER to exit console.
```

Задание 2. Модифицируйте класс Worker из предыдущей задачи, сделайте все его свойства приватными, а для их чтения сделайте методы-геттеры;

Код программы:

```
main.cs
1 using System;
2
3 class Worker
4 {
5     private string name;
6     private string surname;
7     private double rate;
8     private int days;
9
10    public Worker(string name, string surname, double rate, int days)
11    {
12        this.name = name;
13        this.surname = surname;
14        this.rate = rate;
15        this.days = days;
16    }
17
18    public string GetName()
19    {
20        return name;
21    }
22
23    public string GetSurname()
24    {
25        return surname;
26    }
27
28    public double GetRate()
29    {
30        return rate;
31    }
32
33    public int GetDays()
34    {
35        return days;
36    }
37
38    public double GetSalary()
39    {
40        return rate * days;
41    }
42 }
43
44 class Program
45 {
46     static void Main()
47     {
48         Worker worker1 = new Worker("Полина ", "Элобина", 88.99, 44);
49         Worker worker2 = new Worker("Дмитрий", "Элобин", 55.66, 80);
50
51         Console.WriteLine("Зарплата первого работника: " + worker1.GetSalary());
52         Console.WriteLine("Зарплата второго работника: " + worker2.GetSalary());
53     }
54 }
55
56
57
58
59
```

Вывод программы:

```
45 {
  ▼ ↗ ⚙ 📄
Зарплата первого работника: 3915.56
Зарплата второго работника: 4898.08

...Program finished with exit code 0
Press ENTER to exit console. □
```

Задание 3. Создайте класс Calculation , в котором будет одно свойство calculationLine. методы: SetCalculationLine который будет изменять значение свойства, SetLastSymbolCalculationLine который будет в конец строки прибавлять символ, GetCalculationLine который будет выводить значение свойства, GetLastSymbol получение последнего символа, DeleteLastSymbol удаление последнего символа из строки;

Код программы:

```
main.cs
1 using System;
2
3 public class Calculation
4 {
5     //хранения строки с выраж
6     private string calculationLine;
7     // доступ для чтения и записи к переменной(calculationLine)
8     public string CalculationLine
9     {
10         get { return calculationLine; }
11         //при присвоении значения свойству
12         set { calculationLine = value; }
13     }
14     //метод принимает один параметр,явля строкой с вычислением, и устанав значение переменной
15     public void SetCalculation(string recentCalculationLine)
16     {
17         //устанав знач перемен
18         calculationLine = recentCalculationLine;
19     }
20     // добавл символ в конец строки
21     public void SetLastSymbolCalculationLine(char symbol)
22     {
23         //Добавл символа в конец строки
24         calculationLine += symbol;
25     }
26     //возвращ значение переменной
27     public string GetCalculationLine()
28     {
29         return calculationLine;
30     }
31     //возвращ последний символ строки
32     public char GetLastSymbol()
33     {
34         if (calculationLine.Length > 0) //что лин строки бол 0
35         {
36             //Возвр послед сим строки
37             return calculationLine[calculationLine.Length - 1];
38         }
39         else
40         {
41             // возвр нулевого значения
42             return '\0';
43         }
44     }
45     //удал послед символ из строки
46     public void DeleteLastSymbol()
47     {
48         //удал послед символ из строки
49         if (calculationLine.Length > 0)
50         {
51             calculationLine = calculationLine.Substring(0, calculationLine.Length - 1);
52         }
53     }
54 }
55
56 class Program
57 {
58     static void Main()
59     {
60         Calculation calculation = new Calculation();
61         calculation.SetCalculation("7 * 7 * 7");
62         Console.WriteLine("Пример: " + calculation.GetCalculationLine());
63         Console.WriteLine("Последний символ в примере : " + calculation.GetLastSymbol());
64         calculation.SetLastSymbolCalculationLine('-');
65         Console.WriteLine("Пример: " + calculation.GetCalculationLine());
66         calculation.DeleteLastSymbol();
67         Console.WriteLine("пример: " + calculation.GetCalculationLine());
68     }
69 }
70
71
72
```

```
main.cs
28 public string GetCalculationLine()
29 {
30     return calculationLine;
31 }
32 //возвращ последний символ строки
33 public char GetLastSymbol()
34 {
35     if (calculationLine.Length > 0) //что лин строки бол 0
36     {
37         //Возвр послед сим строки
38         return calculationLine[calculationLine.Length - 1];
39     }
40     else
41     {
42         // возвр нулевого значения
43         return '\0';
44     }
45 }
46 //удал послед символ из строки
47 public void DeleteLastSymbol()
48 {
49     //удал послед символ из строки
50     if (calculationLine.Length > 0)
51     {
52         calculationLine = calculationLine.Substring(0, calculationLine.Length - 1);
53     }
54 }
55
56 class Program
57 {
58     static void Main()
59     {
60         Calculation calculation = new Calculation();
61         calculation.SetCalculation("7 * 7 * 7");
62         Console.WriteLine("Пример: " + calculation.GetCalculationLine());
63         Console.WriteLine("Последний символ в примере : " + calculation.GetLastSymbol());
64         calculation.SetLastSymbolCalculationLine('-');
65         Console.WriteLine("Пример: " + calculation.GetCalculationLine());
66         calculation.DeleteLastSymbol();
67         Console.WriteLine("пример: " + calculation.GetCalculationLine());
68     }
69 }
70
71
72
```

Пример: 7 * 7 * 7

Последний символ в примере : 7

Пример: 7 * 7 * 7-

пример: 7 * 7 * 7

...Program finished with exit code 0
Press ENTER to exit console.