

# **Разработка кода** **информационных систем**

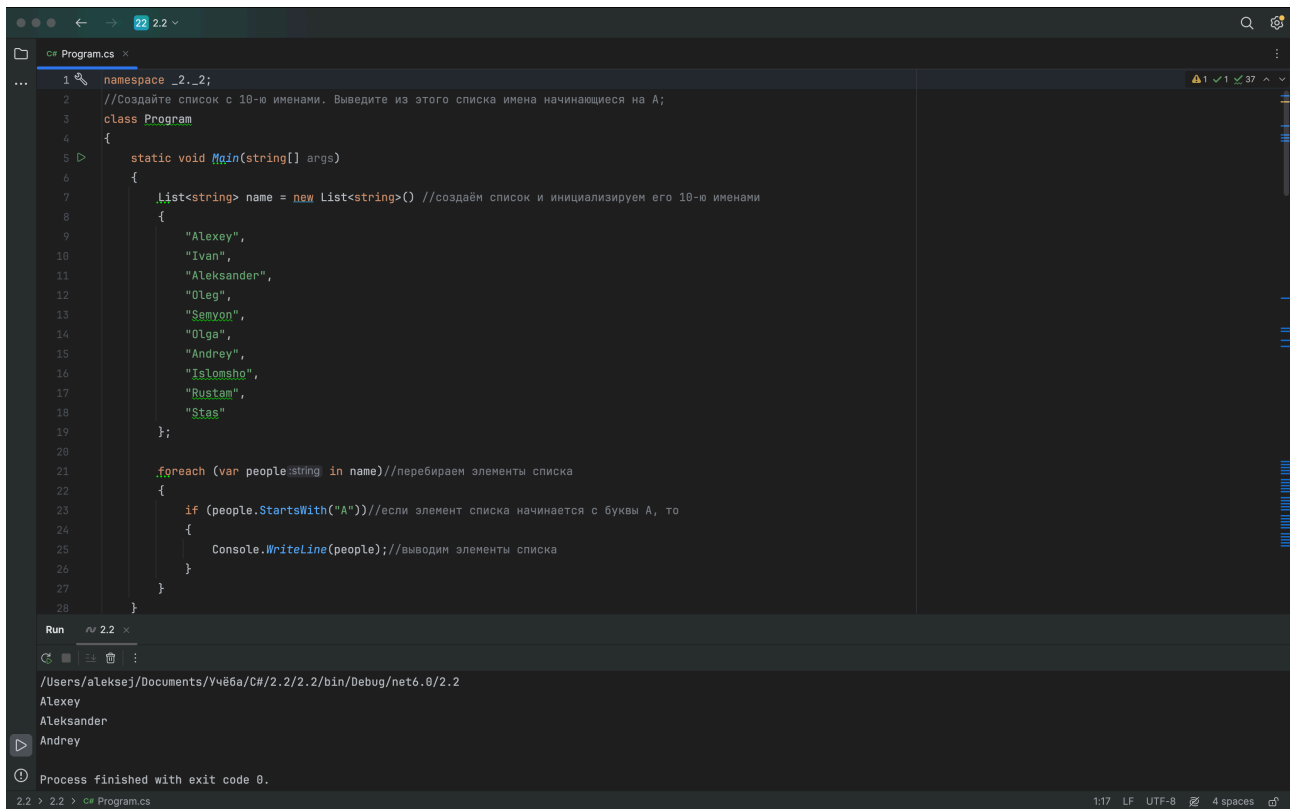
## **Практическая работа 2.2**

**ВЫПОЛНИЛ: ВОЛКОВ АЛЕКСЕЙ, 631 ГРУППА**

# Задание 1

Создайте список с 10-ю именами. Выведите из этого списка имена начинающиеся на А.

Решение:



```
1 namespace _2._2;
2 //Создайте список с 10-ю именами. Выведите из этого списка имена начинающиеся на А;
3 class Program
4 {
5     static void Main(string[] args)
6     {
7         List<string> name = new List<string>() //создаём список и инициализируем его 10-ю именами
8         {
9             "Alexey",
10            "Ivan",
11            "Aleksander",
12            "Oleg",
13            "Semyon",
14            "Olga",
15            "Andrey",
16            "Isломsho",
17            "Rustam",
18            "Stas"
19        };
20
21        foreach (var people:string in name)//перебираем элементы списка
22        {
23            if (people.StartsWith("A"))//если элемент списка начинается с буквы А, то
24            {
25                Console.WriteLine(people);//выводим элементы списка
26            }
27        }
28    }
29 }
```

Run 2.2

/Users/aleksej/Documents/Учёба/C#/2.2/2.2/bin/Debug/net6.0/2.2

Alexey  
Aleksander  
Andrey

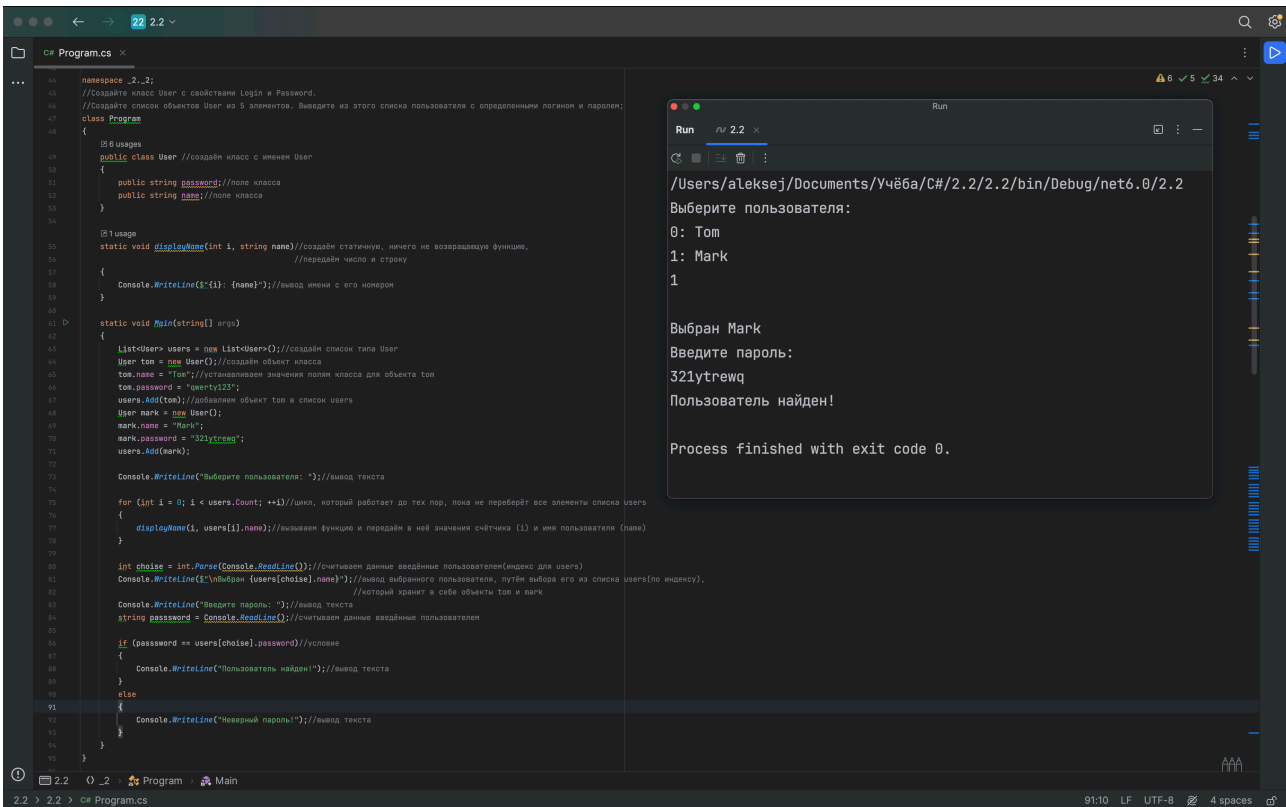
Process finished with exit code 0.

2.2 > 2.2 > Program.cs 1:17 LF UTF-8 4 spaces

## Задание 2

Создайте класс User с свойствами Login и Password. Создайте список объектов User из 5 элементов. Выведите из этого списка пользователя с определенными логином и паролем.

## Решение:



```
44 namespace _2_2;
45 //Создайте класс User с свойствами Login и Password.
46 //Создайте список объектов User из 5 элементов. Выведите из этого списка пользователя с определенными логином и паролем;
47 class Program
48 {
49     [Usage]
50     public class User //создаём класс с именем User
51     {
52         public string password; //поле класса
53         public string name; //поле класса
54     }
55
56     [Usage]
57     static void displayName(int i, string name) //создаём статичную, ничего не возвращающую функцию.
58     { //передаём число и строку
59         Console.WriteLine($"{i}: {name}"); //вывод имени с его номером
60     }
61
62     static void Main(string[] args)
63     {
64         List<User> users = new List<User>(); //создаём список типа User
65         User tom = new User(); //создаём объект класса
66         tom.name = "Tom"; //устанавливаем значения поля класса для объекта tom
67         tom.password = "qwerty123";
68         users.Add(tom); //добавляем объект tom в список users
69         User mark = new User();
70         mark.name = "Mark";
71         mark.password = "321ytrewq";
72         users.Add(mark);
73
74         Console.WriteLine("Выберите пользователя: "); //вывод текста
75
76         for (int i = 0; i < users.Count; ++i) //цикл, который работает до тех пор, пока не переберёт все элементы списка users
77         {
78             displayName(i, users[i].name); //вызываем функцию и передаём в неё значения счётчика (i) и имя пользователя (name)
79         }
80
81         int choice = int.Parse(Console.ReadLine()); //считываем данные введённые пользователем (индекс для users)
82         Console.WriteLine($"{users[choice].name}"); //вывод выбранного пользователя, путём выбора его из списка users (по индексу),
83         //который хранит в себе объекты tom и mark
84
85         Console.WriteLine("Введите пароль: "); //вывод текста
86         string password = Console.ReadLine(); //считываем данные введённые пользователем
87
88         if (password == users[choice].password) //условие
89         {
90             Console.WriteLine("Пользователь найден!"); //вывод текста
91         }
92         else
93         {
94             Console.WriteLine("Неверный пароль!"); //вывод текста
95         }
96     }
97 }
```

Run

/Users/aleksej/Documents/Учёба/C#/2.2/2.2/bin/Debug/net6.0/2.2

Выберите пользователя:

0: Tom

1: Mark

1

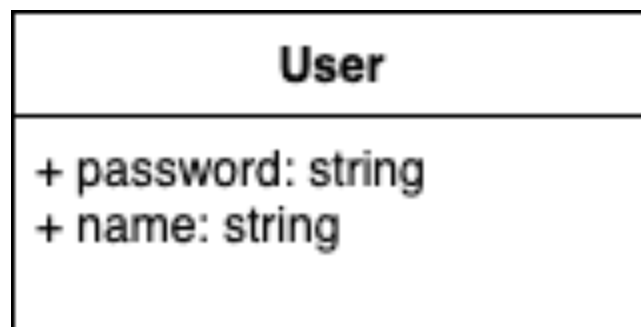
Выбран Mark

Введите пароль:

321ytrewq

Пользователь найден!

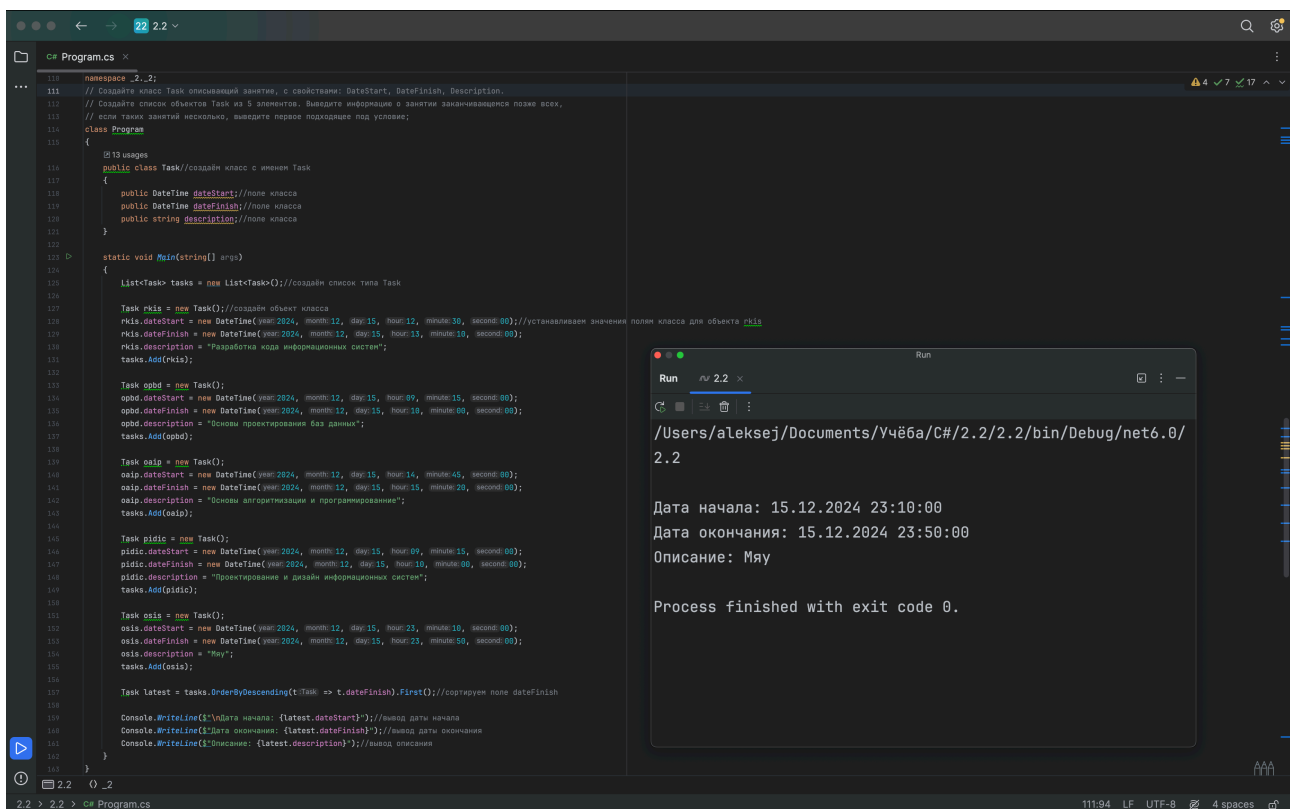
Process finished with exit code 0.



## Задание 3

Создайте класс Task описывающий занятие, с свойствами: DateStart, DateFinish, Description. Создайте список объектов Task из 5 элементов. Выведите информацию о занятии заканчивающемся позже всех, если таких занятий несколько, выведите первое подходящее под условие.

## Решение:



```
111 namespace _2_2
112 // Создайте класс Task описывающий занятие, с свойствами: DateStart, DateFinish, Description.
113 // Создайте список объектов Task из 5 элементов. Выведите информацию о занятии заканчивающемся позже всех.
114 // если таких занятий несколько, выведите первое подходящее под условие;
115 class Program
116 {
117     [Usage]
118     public class Task //создаём класс с именем Task
119     {
120         public DateTime dateStart; //поле класса
121         public DateTime dateFinish; //поле класса
122         public string description; //поле класса
123     }
124
125     static void Main(string[] args)
126     {
127         List<Task> tasks = new List<Task>(); //создаём список типа Task
128
129         Task rklis = new Task(); //создаём объект класса
130         rklis.dateStart = new DateTime(year: 2024, month: 12, day: 15, hour: 12, minute: 30, second: 00); //устанавливаем значения полей класса для объекта rklis
131         rklis.dateFinish = new DateTime(year: 2024, month: 12, day: 15, hour: 13, minute: 10, second: 00);
132         rklis.description = "Разработка кода информационных систем";
133         tasks.Add(rklis);
134
135         Task opbd = new Task();
136         opbd.dateStart = new DateTime(year: 2024, month: 12, day: 15, hour: 09, minute: 15, second: 00);
137         opbd.dateFinish = new DateTime(year: 2024, month: 12, day: 15, hour: 18, minute: 04, second: 00);
138         opbd.description = "Основы проектирования баз данных";
139         tasks.Add(opbd);
140
141         Task osip = new Task();
142         osip.dateStart = new DateTime(year: 2024, month: 12, day: 15, hour: 14, minute: 45, second: 00);
143         osip.dateFinish = new DateTime(year: 2024, month: 12, day: 15, hour: 15, minute: 20, second: 00);
144         osip.description = "Основы алгоритмизации и программирования";
145         tasks.Add(osip);
146
147         Task pidic = new Task();
148         pidic.dateStart = new DateTime(year: 2024, month: 12, day: 15, hour: 09, minute: 15, second: 00);
149         pidic.dateFinish = new DateTime(year: 2024, month: 12, day: 15, hour: 10, minute: 00, second: 00);
150         pidic.description = "Проектирование и дизайн информационных систем";
151         tasks.Add(pidic);
152
153         Task osis = new Task();
154         osis.dateStart = new DateTime(year: 2024, month: 12, day: 15, hour: 23, minute: 10, second: 00);
155         osis.dateFinish = new DateTime(year: 2024, month: 12, day: 15, hour: 23, minute: 50, second: 00);
156         osis.description = "May";
157         tasks.Add(osis);
158
159         Task latest = tasks.OrderByDescending(t => t.dateFinish).First(); //копирован none dateFinish
160
161         Console.WriteLine($"Дата начала: {latest.dateStart}"); //вывод даты начала
162         Console.WriteLine($"Дата окончания: {latest.dateFinish}"); //вывод даты окончания
163         Console.WriteLine($"Описание: {latest.description}"); //вывод описания
164     }
165 }
```

Run

/Users/aLeksej/Documents/Учёба/С#/2.2/2.2/bin/Debug/net6.0/2.2

Дата начала: 15.12.2024 23:10:00  
Дата окончания: 15.12.2024 23:50:00  
Описание: May

Process finished with exit code 0.

### Task

- + dateStart: DateTime
- + dateFinish: DateTime
- + description: string

## Задание 4

Дана целочисленная последовательность, содержащая как положительные, так и отрицательные числа. Вывести ее первый положительный элемент и последний отрицательный элемент.

Решение:

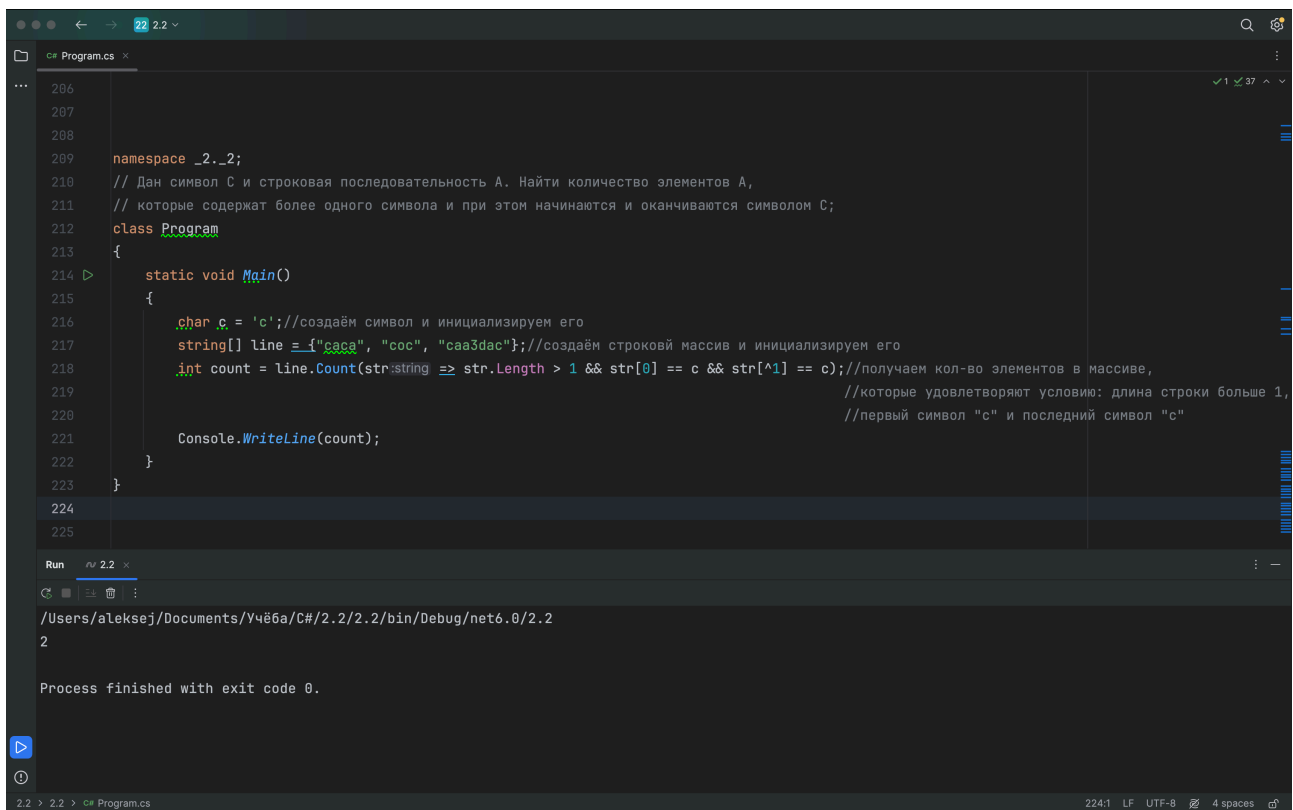
```
177
178 namespace _2._2;
179 // Дана целочисленная последовательность, содержащая как положительные, так и отрицательные числа.
180 // Вывести ее первый положительный элемент и последний отрицательный элемент;
181 class Program
182 {
183     static void Main(string[] args)
184     {
185         int[] nums = { -5, 21, 63, -32, 54, -34, 324, -1313, 4, -1, 43 }; //создаём массив и инициализируем его положительными
186                                     //и отрицательными числами
187         var positive = nums.First(i => i > 0); //возвращает первый элемент массива, который соответствует условию:
188                                     //число должно быть больше 0.
189         Console.WriteLine(positive); //вывод положительного числа
190         var negative = nums.Last(i => i < 0); //возвращает последний элемент массива, который соответствует условию:
191                                     //число должно быть меньше 0.
192         Console.WriteLine(negative); //вывод отрицательного числа
193     }
194 }
195
196
197
```

```
Run 2.2
/Users/aleksej/Documents/Учёба/С#/2.2/2.2/bin/Debug/net6.0/2.2
21
-1
Process finished with exit code 0.
```

## Задание 5

Дан символ С и строковая последовательность А. Найти количество элементов А, которые содержат более одного символа и при этом начинаются и оканчиваются символом С.

Решение:



```
206
207
208
209 namespace _2._2;
210 // Дан символ С и строковая последовательность А. Найти количество элементов А,
211 // которые содержат более одного символа и при этом начинаются и оканчиваются символом С;
212 class Program
213 {
214     static void Main()
215     {
216         char c = 'c'; //создаём символ и инициализируем его
217         string[] line = {"caca", "coc", "caa3dac"}; //создаём строковый массив и инициализируем его
218         int count = line.Count(str => str.Length > 1 && str[0] == c && str[^1] == c); //получаем кол-во элементов в массиве,
219                                                     //которые удовлетворяют условию: длина строки больше 1,
220                                                     //первый символ "с" и последний символ "с"
221         Console.WriteLine(count);
222     }
223 }
224
225
```

Run 2.2

```
/Users/aleksej/Documents/Учёба/C#/2.2/2.2/bin/Debug/net6.0/2.2
2
Process finished with exit code 0.
```

2.2 > 2.2 > Program.cs 224:1 LF UTF-8 4 spaces