



ОТЧЁТ ПО ЛАБРАТОРНОЙ РАБОТЕ №2

Объектно-ориентированное программирование

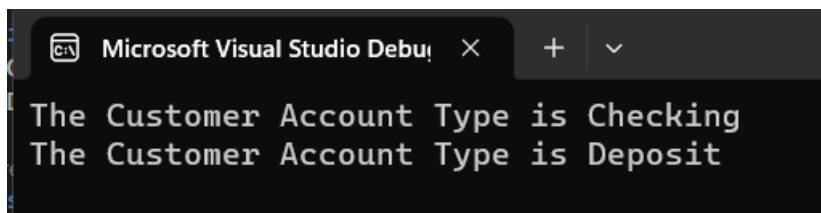
Подготовил Шкода Глеб Ярославович
Студент 2 курса факультета ИКТ Университета ИТМО
Группа K32211
Преподаватель Иванов Сергей Евгеньевич

Упражнение 1

По образцу из текста задания была написана следующая программа.

```
1  using System;
2
3  3 references
4  public enum AccountType {
5      Checking,
6      Deposit
7  }
8  0 references
9  class Program {
10     0 references
11     static void Main() {
12
13         AccountType goldAccount = AccountType.Checking, platinumAccount = AccountType.Deposit;
14         Console.WriteLine("The Customer Account Type is {0}", goldAccount);
15         Console.WriteLine("The Customer Account Type is {0}", platinumAccount);
16     }
17 }
```

При запуске была выведена необходимая информация.



The screenshot shows the Microsoft Visual Studio Debug Console window. It contains two lines of output text: "The Customer Account Type is Checking" and "The Customer Account Type is Deposit".

Упражнение 2

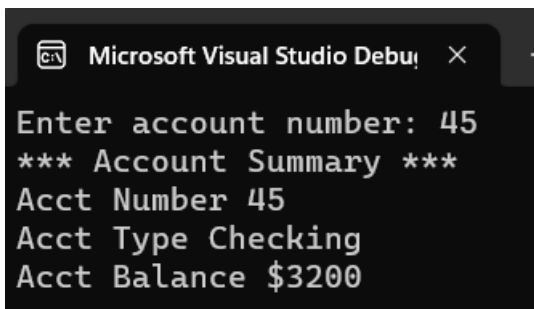
Аналогично предыдущему заданию, следуя образцу, была создана структура BankAccount, которая была заполнена данными и выведена в методе main.

```

1  using System;
2
3  public enum AccountType {
4      Checking,
5      Deposit
6  }
7
8  public struct BankAccount {
9      public long accNo;
10     public decimal accBal;
11     public AccountType accType;
12 }
13
14 class Program {
15     static void Main() {
16         BankAccount goldAccount;
17         goldAccount.accType = AccountType.Checking;
18         goldAccount.accBal = (decimal) 3200.00;
19         Console.Write("Enter account number: ");
20         goldAccount.accNo = long.Parse(Console.ReadLine());
21         Console.WriteLine("*** Account Summary ***");
22         Console.WriteLine("Acct Number {0}", goldAccount.accNo);
23         Console.WriteLine("Acct Type {0}", goldAccount.accType);
24         Console.WriteLine("Acct Balance ${0}", goldAccount.accBal);
25     }
26 }
27

```

В результате, при выполнении программа просит ввести номер счёта, после чего выводится информация о нём.



```

Microsoft Visual Studio Debug Console
Enter account number: 45
*** Account Summary ***
Acct Number 45
Acct Type Checking
Acct Balance $3200

```

Упражнение 3

Для решения задачи была создана структура Distance, состоящая из 2 целочисленных переменных и метода вывода, в соответствии с условием.

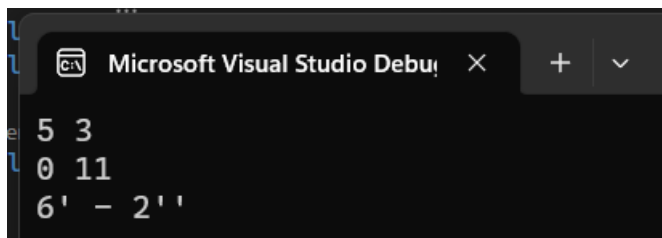
Программе на вход подаются 2 расстояния, каждое в своей строке. После чего считается их сумма: сначала в дюймах, а затем переводится в футы и дюймы. После чего полученный ответ выводится на экран.

```

1  using System;
2
3  1 reference
4  public struct Distance {
5      public int feet;
6      public int inches;
7
8      1 reference
9      public void Print() {
10         Console.WriteLine("{0}' - {1}''", feet, inches);
11     }
12 }
13 0 references
14 class Program {
15     0 references
16     static void Main() {
17
18         Distance d1, d2, d3;
19         var input = Console.ReadLine().Split();
20         d1.feet = int.Parse(input[0]);
21         d1.inches = int.Parse(input[1]);
22         input = Console.ReadLine().Split();
23         d2.feet = int.Parse(input[0]);
24         d2.inches = int.Parse(input[1]);
25         d3.inches = (d1.feet + d2.feet) * 12 + d1.inches + d2.inches;
26         d3.feet = (int) (d3.inches / 12);
27         d3.inches = d3.inches % 12;
28         d3.Print();
29     }
30 }

```

Пример работы написанной программы:



```

Microsoft Visual Studio Debug Console
5 3
0 11
6' - 2''

```