

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

«Казанский национальный исследовательский технический университет
им. А.Н. Туполева – КАИ»

Институт компьютерных технологий и защиты информации
Отделение СПО ИКТЗИ (Колледж информационных технологий)

ЛАБОРАТОРНАЯ РАБОТА №6
по дисциплине

Основы алгоритмизации и программирования

Тема: «Программирование простейших классов»

Работу выполнил
Студент гр.4238
Бусов В.Р.

Принял
Преподаватель Шмидт. И.Р.

Казань 2023

ВАРИАНТ 4

Цель работы

Приобрести умения и практические навыки для разработки простейших классов с использованием объектно-ориентированных технологий.

Задание на лабораторную работу

Задание 1. Поле `first` – целое положительное число, номинал купюры; номинал может принимать значения 1, 2, 5, 10, 50, 100, 500, 1000, 5000. Поле `second` – целое положительное число, количество купюр данного достоинства. Реализовать метод `summa()` –вычисление денежной суммы

Задание 2. Реализовать класс `Account`, представляющий собой банковский счет. В классе должны быть реализованы 4 поля: фамилия владельца, номер счета, процент начисления и сумма в рублях. Необходимо выполнять следующие операции: сменить владельца счета, снять некоторую сумму со счета, положить деньги на счет, начислить проценты, перевести сумму в доллары, перевести сумму в евро.

Результат выполнения работы

Задание 1

Описание: Вводится номинал купюры, затем вводится количество купюр этого номинала, после чего высчитывается общая сумма.

```
Введите номинал купюры: 200
Вы ввели неподдерживаемый номинал. Поддерживаемые номиналы:
1, 2, 5, 10, 50, 100, 500, 1000, 5000,
Введите номинал купюры: 50
Введите количество купюр номиналом 50: 12
Общая сумма: 600

D:\Documents\KIT\2 курс\лабы\ОАИП\6 лаба\ConsoleApp2\ConsoleApp2\bin\Debug\net7.0\ConsoleApp2.exe (процесс 6756) заверши
л работу с кодом 0.
Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Ав
томатически закрыть консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно: _
```

Рисунок 1 – Результат выполнения задания 1.

Задание 2

Описание: Программа представляет собой класс, позволяющий взаимодействовать с вашим счетом. Сначала вводится фамилия владельца счета. Далее идет процесс снятия средств, начисления, начисления процентов, перевод в доллары и евро. При каждом действии отображается баланс счета.

```
Введите фамилию владельца счёта: Генадьев
Настоящий владелец счёта: Генадьев
Сейчас на вашем счете: 86451
Введите сумму, которую вы хотите снять со своего счета: 86450
Сейчас на вашем счете: 1
Введите сумму, которая хотите добавить к своему счету: 99
Сейчас на вашем счете: 100
На ваш счет начислены проценты!
Сейчас на вашем счете: 100,91666668653488
Ваш счет в долларах: 1,0969202900710313
Ваш счет в евро: 1,0193602695609585

D:\Documents\KIT\2 курс\лабы\ОАИП\6 лаба\ConsoleApp1\ConsoleApp1\bin\Debug\net7.0\ConsoleApp1.exe (процесс 1704) заверши
л работу с кодом 0.
Чтобы автоматически закрывать консоль при остановке отладки, включите параметр "Сервис" ->"Параметры" ->"Отладка" -> "Ав
томатически закрыть консоль при остановке отладки".
Нажмите любую клавишу, чтобы закрыть это окно: _
```

Рисунок 2 - Результат выполнения задания 2

Листинг кода

Задание 1

```
using program;
using System;
using System.Reflection;

namespace program
{
    class Program
    {
        static void Main(string[] args)
        {
            Account acc = new Account();
            acc.set_owner();
            Console.WriteLine(acc.get_summ());

            acc.take_summ();
            Console.WriteLine(acc.get_summ());

            acc.put_summ();
            Console.WriteLine(acc.get_summ());

            acc.add_interest();
            Console.WriteLine(acc.get_summ());

            Console.WriteLine($"Ваш счет в долларах: {acc.to_dollars()}");

            Console.WriteLine($"Ваш счет в евро: {acc.to_euros()}");
        }
    }

    class Account
    {
        private string owner;
        private double summ;
        private long acc_num;
        private float percent;

        private void generator()
        {
            var rand = new Random();
            this.summ = rand.Next(5000, 1000000);
            this.acc_num = Convert.ToInt64($"408178109991{rand.Next(1000000, 9999999)}");
            this.percent = 11;
        }

        public void set_owner()
        {
            Console.Write("Введите фамилию владельца счёта: ");
            this.owner = Console.ReadLine();

            this.generator();

            Console.WriteLine($"Настоящий владелец счёта: {this.owner}");
        }

        public void take_summ()
        {
            Console.Write("Введите сумму, которую вы хотите снять со своего счета: ");
            double delta = Convert.ToInt32(Console.ReadLine());

            if (delta <= this.summ)
            {
                this.summ -= delta;
            } else if (delta >= this.summ) {
                Console.WriteLine("На вашем счете недостаточно средств, чтобы снять такую сумму.");
            }
        }
    }
}
```

```

    } else if (delta < 0) {
        Console.WriteLine("Введенная сумма должна быть положительной");
    }
}

public void put_summ()
{
    Console.Write("Введите сумму, которая хотите добавить к своему счету: ");
    double delta = Convert.ToInt32(Console.ReadLine());

    if (delta >= 0)
    {
        this.summ += delta;
    } else
    {
        Console.WriteLine("Ошибка. Сумма должна быть положительной");
    }
}

public void add_interest()
{
    this.summ += (this.summ / 100) * (this.percent / 12);
    Console.WriteLine("На ваш счет начислены проценты!");
}

public double get_summ()
{
    return this.summ;
}

public double to_dollars()
{
    return this.summ / 92;
}

public double to_euros()
{
    return this.summ / 99;
}
}
}

```

Задание 2.

```

using program;

namespace program
{
    class Program
    {
        static void Main(string[] args)
        {
            Money bankomat = new Money();
            bool res = false;

            do
            {
                res = bankomat.set_first();
            }
            while (res == false);

            bankomat.set_second();
            bankomat.summa();
        }
    }
}

```

```

class Money
{
    private int first;
    private int second;
    private int[] nominals = { 1, 2, 5, 10, 50, 100, 500, 1000, 5000 };

    public bool set_first()
    {
        Console.Write("Введите номинал купюры: ");
        first = Convert.ToInt32(Console.ReadLine());

        foreach (int i in this.nominals)
        {
            if (first == i)
            {
                this.first = first;

                return true;
            }
        }
        Console.Write("Вы ввели неподдерживаемый номинал. Поддерживаемые номиналы: \n");
        foreach (int i in this.nominals)
        {
            Console.Write(i);
            Console.Write(", ");
        }
        Console.WriteLine();

        return false;
    }

    public void set_second()
    {
        Console.Write($"Введите количество купюр номиналом {this.first}: ");
        this.second = Convert.ToInt32(Console.ReadLine());
    }

    public void summa()
    {
        Console.WriteLine($"Общая сумма: {this.first * this.second}");
    }
}

```

Контрольные вопросы

- 1) *Объект* – сущность, обладающая определенными атрибутами, состояниями, имеющая определенный набор свойств, методов и операций.
- 2) *Класс* – можно сказать, что это пользовательски тип данных.
- 3) Класс – человек, экземпляры класса – Артем, Булат, Данил и тд.
- 4) *Инкапсуляция* - это принцип, согласно которому внутреннее устройство сущностей нужно объединять в специальной «оболочке» и скрывать от вмешательств извне. Доступ к объектам возможен

через специальные открытые методы, а напрямую обратиться к их содержимому нельзя.

- 5) *Абстрагирование* - это процесс выделения общих характеристик и функциональности объектов или системы, игнорируя детали реализации.
- 6) Методы - это действия. Свойства - это данные. Если то, что вы пишете выглядит как кусок данных объекта - делайте его свойством.
- 7) Метод для чтения хранящегося в свойстве значения (часто такие методы называют get-методами). Метод для изменения значения свойства (set-метод)
- 8) Конструктор определяет действия, выполняемые при создании объекта класса, и является важной частью класса. Как правило, программисты стараются явно указать конструктор