|  |  |  |  |
| --- | --- | --- | --- |
|  | | | |
| Федеральное государственное бюджетное  образовательное учреждение высшего образования «Новосибирский государственный технический университет» | | | |
|  | | | |
| Кафедра прикладной математики | | | |
|  | | | |
| Практическое задание № 1 | | | |
| по дисциплине «Разработка объектно-ориентированных программ с использованием C#/C++» | | | |
| **Класс Person** | | | |
|  | | | |
|  | Бригада №10 | ПМ-92 Богданов антон |
|  | ПМ-92 старцев максим |
| Преподаватель | Ступаков Илья Михайлович |
| Дата | 15.12.2020 |
|  |  |
|  |  |
|  | | | |
| Новосибирск | | | |

**Вариант №10**

В качестве обрабатываемых данных на вход программе подаётся файл persons.txt.

### persons.txt

Брандо Дио Иванович Male 100 12.09.1867

Джизус Владислав Дмитриевич Male 50,5 12.06.1997

Пугачева Алла Борисовна Female 101,422 15.04.1949

Пугачев Емельян Иванович Male 111,11111 16.09.1742

Поттер Гарри Джеймс Male 2,432 31.07.1980

Тимошенко Кирилл Бледный Male 44 31.03.1997

Кандрашина Наталья Сергеевна Female 1,9 5.11.1998

Петров Андрей Петрович Female 999,9412 1.09.1996

Горин Геннадий Александрович Male 100 11.12.1972

**Код программы**

using System;

using System.IO;

using System.Linq;

namespace pz1\_sharp

{

class Program

{

static void Main(string[] args)

{

Console.Write("По алфавитному порядку? (да, нет) > ");

bool isReverse = false; bool isAnswered = false;

while (!isAnswered)

switch (Console.ReadLine())

{

case "нет":

isReverse = true;

isAnswered = true;

break;

case "да":

isAnswered = true;

break;

default:

Console.Write("Да или нет? (да, нет) > ");

break;

}

Person[] persons = File.ReadAllLines("persons.txt").

Select(person\_string => Person.Parse(person\_string)).ToArray();

string[] person\_strings = persons.OrderBy(person => person.Surname)

.Select(person => person.ToString()).ToArray();

if (isReverse) Array.Reverse(person\_strings);

File.WriteAllLines("out.txt", person\_strings);

}

}

class Person

{

private bool sex;

public Person(string surname, string first\_name, string second\_name,

bool sex, float weig, DateTime birth\_date)

{

Surname = surname;

FirstName = first\_name;

SecondName = second\_name;

this.sex = sex;

Weight = weig;

BirthDate = birth\_date;

}

public static Person Parse(string person\_string)

{

var data = person\_string.Split();

Person person = new Person(

data[0], data[1], data[2],

data[3] == "Male" ? true : false,

float.Parse(data[4]),

DateTime.Parse(data[5])

);

return person;

}

public override string ToString()

{

string result = $"{Surname}, {FirstName}, {SecondName}, {(sex ? "Male" : "Female")}, {Weight:F2}, {BirthDate.ToString(@"dd MMM yyyy")} ({YearsOld})";

return result;

}

public string Surname { private set; get; }

public string FirstName { private set; get; }

public string SecondName { private set; get; }

public bool Sex { get { return sex; } }

public float Weight { private set; get; }

public DateTime BirthDate { private set; get; }

public int YearsOld

{

get

{

DateTime today = DateTime.Now;

return DateTime.Now.Year - BirthDate.Year +

((BirthDate.Month >= today.Month && BirthDate.Day >= today.Day) ? -1 : 0);

}

}

}

}