НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ

«ХАРКІВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

Кафедра «ОБЧИСЛЮВАЛЬНОЇ ТЕХНІКИ ТА ПРОГРАМУВАННЯ»

«DotNet»

*Звiт з лабораторної роботи №2*

*Тема: «Колекції об'єктів С#. Форматований висновок»*

Виконав:

ст. гр. КІТ-119а

Капелька Я. І.

Перевірив:

Бартош М. В.

Харків – 2021

**Тема:** Колекції об'єктів С#. Форматоване виведення.

**Мета:** Оволодіння навичками застосування колекції об’єктів та форматованого виведення на мові програмування C#.

**Розробник**: Капелька Ярослав, КІТ-119а, варіант №11.

**Опис програми:**

**Засоби ООП**: клас, метод класу, поле класу.

**Ієрархія та структура класів:** один публічний клас Main, публічний клас Student, у якого є поля: Firstname, Surname, GroupIndex, Faculty, Specialization, AcademicPerformance, DateOFBirth, DateOfEnter, Printer; гетери, сетери, конструктор класу та метод виведення даних класу. Інтерфейс IPrinter. Класс Collection для створення колекції управління студентами.

**Важливі фрагменти програми:**

using System;

namespace lab2

{

public class Program

{

static void Main(string[] args)

{

var st1 = new Student("Alex", "Ivanov", "Olegovich", 'G', "CIT", "Comp Logic", new DateTime(2000, 11, 19), new DateTime(2017, 7, 23), 88);

var st2 = new Student("Georgiy", "Petrov", "Yaroslavovich", 'A', "CIT", "Comp Logic", new DateTime(2002, 10, 16), new DateTime(2020, 8, 21), 57);

Collection arr = new(st1);

arr.AddStudent(st2);

arr.Print();

Console.WriteLine("\nДоступ к элементу\n");

var st3 = new Student("Oleg", "Sidorov", "Sidorovich", 'D', "CIT", "Comp Logic", new DateTime(2001, 8, 23), new DateTime(2020, 7, 20), 99);

arr.AddStudent(st3);

arr.GetStudent(2);

Console.WriteLine("\nУдаление первого элемента контейнера\n");

arr.RemoveElement(1);

arr.Print();

Console.WriteLine("\nВставка элемента в контейнер\n");

arr.InsertStudent(0, st2);

arr.Print();

Console.WriteLine("\nКоличество элементов в контейнере: " + arr.Size() + "\n");

foreach (Student stud in arr)

{

stud.GetInfo();

}

foreach (Student stud in arr)

{

stud.GetInfo();

}

Console.WriteLine("\nОчистка контейнера\n");

arr.RemoveAll();

if(arr.Size() == 0)

{

Console.WriteLine("Контейнер пуст");

}

Console.ReadLine();

}

}

}

using System;

namespace lab2

{

public class Student

{

//Поля

private string name;

private string surname;

private string patronymic;

DateTime DOB;

DateTime DOA;

private char groupIndex;

private string faculty;

private string speciality;

private byte performance;

//Свойства

public DateTime GetDateOfAdmission()

{

return DOA;

}

public void SetDateOfAdmission(int year, int month, int day)

{

DateCheck(year, month, day);

DOA = new DateTime(year, month, day);

}

public DateTime GetDateOfBrth()

{

return DOB;

}

public void SetDateOfBrth(int year, int month, int day)

{

DateCheck(year, month, day);

DOB = new DateTime(year, month, day);

}

public byte Perf

{

set

{

if (value > 100)

{

Console.WriteLine("Успеваемость не может быть выше 100 и ниже 0 процентов!");

}

else

{

performance = value;

}

}

get { return performance; }

}

public string Spec

{

get

{

return speciality;

}

set

{

speciality = value;

}

}

public string Facul

{

get

{

return faculty;

}

set

{

faculty = value;

}

}

public char GIndex

{

get

{

return groupIndex;

}

set

{

groupIndex = value;

}

}

public string Patronymic

{

get

{

return patronymic;

}

set

{

patronymic = value;

}

}

public string SurName

{

get

{

return surname;

}

set

{

surname = value;

}

}

public string Name

{

get

{

return name;

}

set

{

name = value;

}

}

public void GetInfo()

{

Console.WriteLine("Студент " + Name + " " + SurName + " " + Patronymic + " группы " + GIndex + " факультета " + Facul + " специальности " + Spec + ", " + GetDateOfBrth() + " даты рождения, " + GetDateOfAdmission() + " даты поступления имеет успеваемость " + Perf + "%");

}

public Student(string nm, string srnm, string patr, char Gin, string fcl, string spc, DateTime Birth, DateTime Adm, byte persent)

{

DateCheck(Birth.Year, Birth.Month, Birth.Day);

DateCheck(Adm.Year, Adm.Month, Adm.Day);

name = nm;

surname = srnm;

patronymic = patr;

groupIndex = Gin;

faculty = fcl;

speciality = spc;

DOB = Birth;

DOA = Adm;

performance = persent;

}

public static void DateCheck(int year, int month, int day)

{

var currentYear = DateTime.Now.Year;

if (year > currentYear)

{

Console.WriteLine("Год введен некорректно!");

}

if (month > 12 && month <= 0)

{

Console.WriteLine("Месяц введен некорректно!");

}

if (day > 31 && day <= 0)

{

Console.WriteLine("День введен некорректно!");

}

}

}

}

using System.Collections;

using System.Collections.Generic;

using System.Linq;

namespace lab2

{

public class Collection : IEnumerator, IEnumerable

{

private List<Student> catalog;

private int position = -1;

public Collection(Student student)

{

catalog = new List<Student>() { student };

}

public void AddStudent(Student student)

{

catalog.Add(student);

}

public void InsertStudent(int index, Student student)

{

catalog.Insert(index, student);

}

public void Print()

{

foreach (var stud in catalog)

{

stud.GetInfo();

}

}

public int Size()

{

return catalog.Count;

}

public void RemoveAll()

{

catalog.Clear();

}

public void RemoveElement(int index)

{

catalog.RemoveAt(index);

}

public void GetStudent(int index)

{

catalog[index].GetInfo();

}

public IEnumerator GetEnumerator()

{

return catalog.GetEnumerator();

}

public bool MoveNext()

{

position++;

return (position < catalog.Count);

}

public void Reset()

{

position = 0;

}

public object Current

{

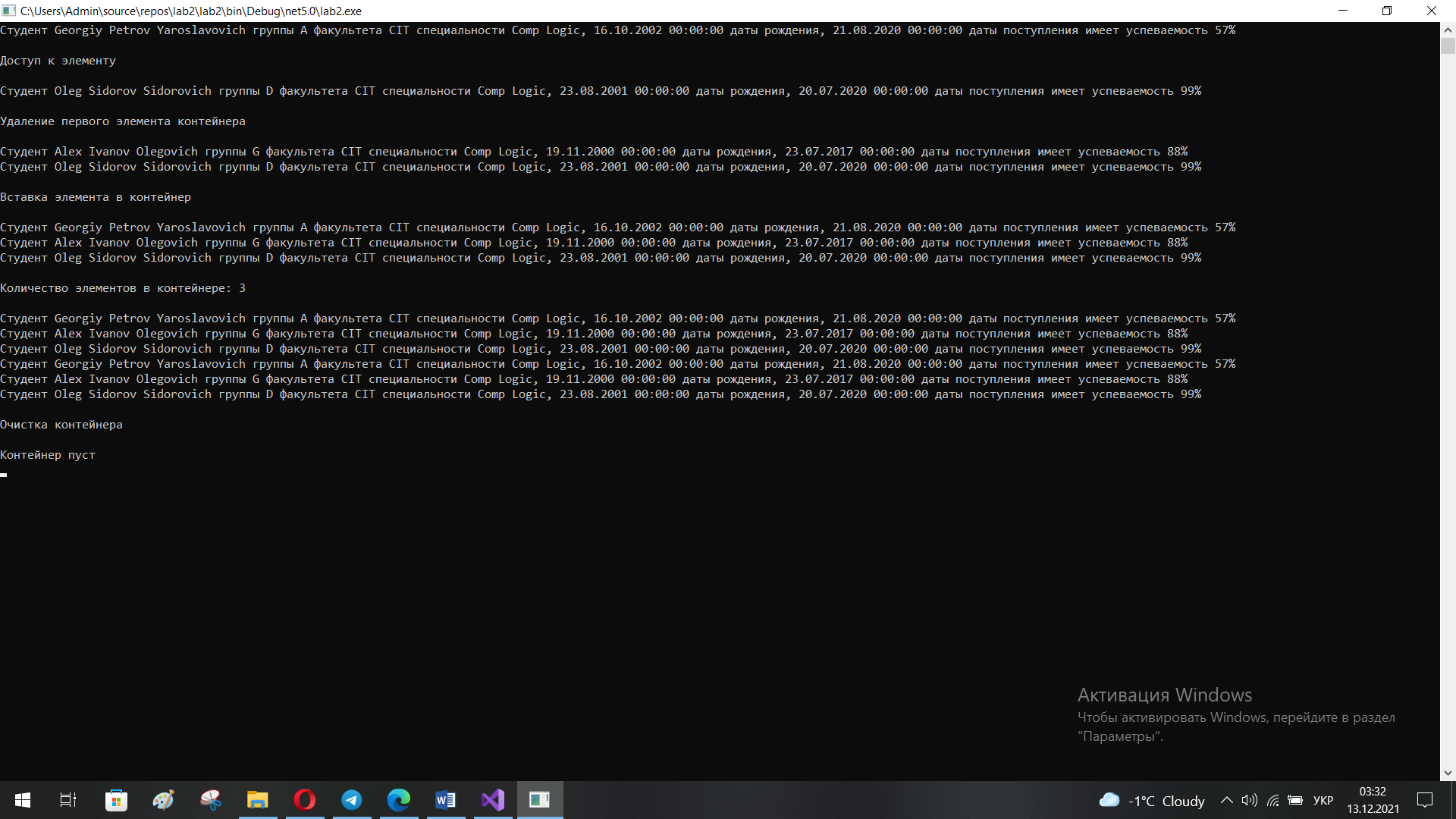
get { return catalog.ElementAt<Student>(position); }

}

}

}

**Результат роботи програми:**



**Висновок:**

При виконанні даної лабораторної роботи було набуто застосування колекції об’єктів та форматованого виведення на мові програмування C#.