**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

**Чорноморський національний університет   
імені Петра Могили**

**Факультет комп'ютерних наук**

**Кафедра інженерії програмного забезпечення**

**ЗВІТ**

з лабораторної роботи № 2

**«Сеттери, геттери, властивості та індексатори.»**

**Варіант 5**

Дисципліна «Об'єктно-орієнтоване програмування»

Спеціальність: **Інженерія програмного забезпечення**

121-ЛР.02-208.22120802

***Студент\_\_Д. М.  Дзундза***

*(підпис)*

*\_\_09.20.2021* \_\_

*(дата)*

***Викладач \_\_\_\_\_\_\_\_С. Ю. Боровльова***

*(підпис)*

*\_\_\_\_\_\_\_\_\_\_\_\_\_*

*(дата)*

Миколаїв – 2021

**Лабораторна робота №2**

|  |  |
| --- | --- |
| Тема роботи: | Сеттери, геттери, властивості та індексатори. |

|  |
| --- |
| Мета роботи: |

Ознайомитись з використанням : Сеттери, геттери, властивості та індексатори. Засвоїти різницю між використанням цих способів. Ознайомитись з виглядом цих методів в коді IL, використавши утіліту IlDasm. **Завдання:**

На базі класів із попередньої лабораторної роботи створіть застосунок для демонстрації наступних можливостей:

1. Різних типів властивостей, а саме:
2. для одного із полів Вашого класу оголосіть геттер та сеттер;
3. для іншого із полів Вашого класу оголосіть геттер та сеттер, але тіло повинно бути записано у вигляді виразу;
4. для другого поля оголосіть звичайну властивість;
5. замість третього поля Вашого класу створіть автоматичну властивість;
6. додайте в клас оголошення довільного масиву та створіть індексатор.
7. Завантажте утіліту IlDasm (дивись відео) та подивіться, як виглядають в коді IL властивості, геттери та сеттери.

**Лістинг коду:**

using System;

namespace Lection2

{

class Worker

{

private int[] array1 = new int[5];

private string \_name;

private int \_age;

public string Name { get => \_name; set => \_name = value; }

public string Surname { get; set; }

public int Age { get { return \_age; } set { \_age = value; } }

public int Lenght { get => array1.Length; }

public int this[int i]

{

get => array1[i]; set => array1[i] = value;

}

public override string ToString()

{

return $"name: {Name}; surname: {Surname}; age: {Age}";

}

public override bool Equals(object obj)

{

if ((obj == null) || !this.GetType().Equals(obj.GetType()))

{

return false;

}

else

{

Worker p = (Worker)obj;

return (Name == p.Name) && (Age == p.Age);

}

}

public Worker() { }

public Worker(string name, int age)

{

Name = name;

Age = age;

}

public Worker(string name, string surname, int age)

{

Name = name;

Age = age;

Surname = surname;

}

}

class Program

{

static void Main(string[] args)

{

var rand = new Random();

int max\_rand\_value = 10;

Worker worker = new Worker();

Worker worker2 = new Worker("Aleksandr","Vasilevich", 52);

worker.Name = "Dima";

worker.Surname = "Dzundza";

worker.Age = 19;

Console.WriteLine(worker.ToString());

Console.WriteLine($"Name: {worker2.Name}");

Console.WriteLine($"Surname: {worker2.Surname}");

Console.WriteLine($"Age: {worker2.Age}");

for (int i = 0; i < worker.Lenght; i++) worker[i] = (int)rand.Next(-max\_rand\_value, max\_rand\_value);

for (int i = 0; i < worker.Lenght; i++) Console.Write(" " + worker[i] + ",");

Console.WriteLine("\b \b");

Console.ReadKey();

}

}

}

**Результат роботи програми:**

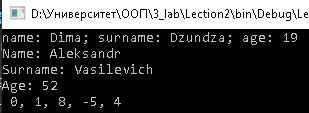


Рисунок 1 – Результат роботи програми з класами

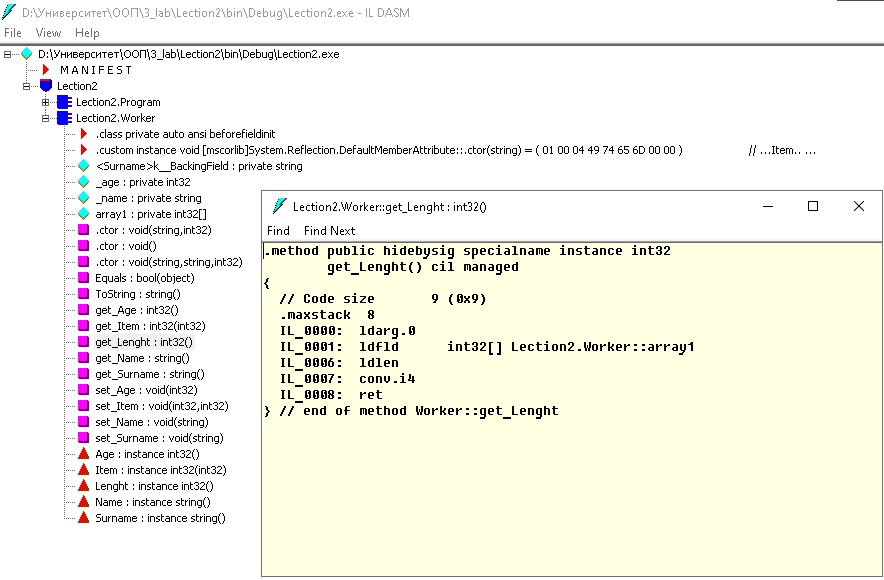
****

Рисунок 2 – Перегляд iLDasm функції get\_Lenght

**Питання до захисту:**

1. <модифікатор доступу><тип><ім’я властивості> <{

get{<тіло>}

set{<тіло>}

}>

Private int a;

Public int A {set => a = value; get => a;}

2.Синтаксис оголошення автоматичних властивостей.

<модифікатор доступу><тип><ім’я властивості> <{get;set;}>

Public int A {set; get;}

3. <тип повертаємого значення><this><[парамет1[,парметр 2 і т.д.]]>{

get{}

set{}

}

4.Так.

static string name;

public static string Name{

set => name = value;

get => name;

}

5. get\_Item – індексатор. get\_Name – властивість.

.method public hidebysig specialname instance int32

get\_Item(int32 i) cil managed

{

// Code size 9 (0x9)

.maxstack 8

IL\_0000: ldarg.0

IL\_0001: ldfld int32[] Lection2.Worker::array1

IL\_0006: ldarg.1

IL\_0007: ldelem.i4

IL\_0008: ret

} // end of method Worker::get\_Item

**Висновок:**

Протягом процесу виконання лабораторної роботи було створено 1 консольний застосунок що використовує сетери, гетери, властивості та індексатори. Було показано різницю в їх використанні. Використано утіліту iLDasm.