**МІНІСТЕРСТВО**

**ОСВІТИ ТА НАУКИ УКРАЇНИ**

**Національний технічний університет України**

**«Київський політехнічний інститут імені Ігоря Сікорського»**

**Кафедра інформаційних систем та технологій**

**Звіт**

**з лабораторної роботи No 4**

**«**Інтерфейси. Абстрактні класи**»**

**з дисципліни**

**«Алгоритми і структури данних»**

**Варіант No29**

**Перевірив:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**  **Виконав:**

**Студент гр. ІС-12 ФІОТ**

**Хіврич Володимир**

**Київ 2022**

**Лабораторна робота №4**

**Теорія**

Інтерфейси

<https://habr.com/ru/post/343852/>

<https://c-sharp.pro/?p=1119>

В нашій улюбленій книзі є цілий розділ, присвячений інтерфейсам

<https://metanit.com/sharp/tutorial/3.9.php>

Абстрактні класи

<http://bit.ly/2TS9Cig>

<https://metanit.com/sharp/tutorial/3.8.php>

Різниця між інтерфейсом та абстрактним класом — типове запитання на співбесіді будь-якого рівня спеціаліста.

В чому між ними різниця, можна спробувати зрозуміти в цій статті (спойлер: майже ні в чому станом на 2020 рік)

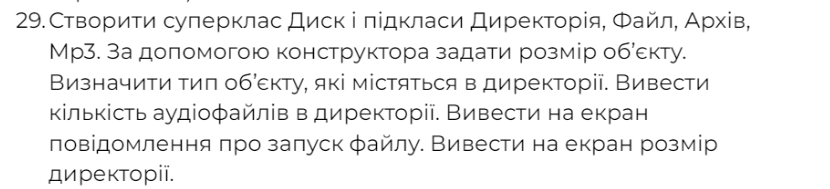
<https://habr.com/ru/post/351384/>

Хороший матеріал про інтерфейси та абстрактні класи (тут про Java, але станом на сьогодні різниця в Java i C# є супермінімальна)

<http://bit.ly/38HhMj0>

**Завдання**

* Проаналізувати завдання, виділити інформаційні об’єкти та дії. Визначити правильну ієрархію об’єктів
* Створити базовий суперклас (абстрактний клас або інтерфейс) і визначити загальні методи для даного класу. Створити підкласи, в які додати специфічні властивості та методи. Частину методів перевизначити.
* Розробити програму з використанням абстрактних класів та інтерфейсів. Чітко розуміти, де доцільно використати суперклас, а де звичайний.
* При розробці використовувати наслідування та поліморфізм
* У всіх класах повинні бути реалізовані доцільні для класу методи, навіть якщо це не вказано у завданні
* Використовувати об’єкти підкласів для моделювання реальних ситуацій на об’єктів

****

**Лістинг програми**

**Class Disk**

using System;

using System.Collections.Generic;

using System.Text;

using System.ComponentModel;

namespace lab4

{

public abstract class Disk

{

private int size;

protected string type;

protected Disk(int size)

{

this.size = size;

}

public int getSize() { return size; }

public string getType() { return type; }

public override string ToString()

{

return "{ " + getType() + ", size -> " + size + " MB}";

}

}

}

**Class Directory**

using System;

using System.Collections.Generic;

using System.Text;

namespace lab4

{

public class Directory : Disk

{

private List<Disk> data;

public Directory(int size, List<Disk> data) : base(size)

{

this.data = data;

this.type = "Directory";

}

public void showDataStoredInDir()

{

Console.WriteLine("Data stored in Dir:");

foreach (Disk item in data)

{

Console.WriteLine(item);

}

}

public void whichTypesOfDataIsStored()

{

HashSet<string> set = new HashSet<string>();

Console.WriteLine("Type of data stored in the dir:");

foreach (Disk item in data)

set.Add(item.getType());

foreach (string item in set)

Console.Write(item + " ");

Console.WriteLine();

}

public string getTotalSizeOfDir()

{

int total = 0;

foreach (Disk item in data)

total += item.getSize();

return total + "MB";

}

public int getAmountOfMP3Files()

{

List<Disk> MP3list = new List<Disk>();

foreach (Disk item in data)

{

if (item.getType().Equals("MP3"))

MP3list.Add(item);

}

return MP3list.Count;

}

}

}

**Class MP3**

using System;

using System.Collections.Generic;

using System.Text;

namespace lab4

{

public class MP3 : Disk

{

private string nameOfSong;

public MP3(int size, string nameOfSong) : base(size)

{

this.nameOfSong = nameOfSong;

this.type = "MP3";

}

public void playSong()

{

Console.WriteLine(nameOfSong + " is playing now...");

}

}

}

**Class Archive**

using System;

using System.Collections.Generic;

using System.Text;

namespace lab4

{

public class Archive : Disk

{

public Archive(int size) : base(size)

{

this.type = "Archive";

}

public File unzipArchive()

{

return new File(this.getSize() \* 2);

}

}

}

**Class File**

using System;

using System.Collections.Generic;

using System.Text;

namespace lab4

{

public class File : Disk

{

public File(int size) : base(size)

{

this.type = "File";

}

public void writeDataInFile()

{

Console.WriteLine("Data is writing to the file");

}

}

}

**Class Program**

using System;

using System.Collections.Generic;

namespace lab4

{

class Program

{

static void Main(string[] args)

{

List<Disk> data = new List<Disk>();

data.Add(new Archive(10));

data.Add(new File(20));

data.Add(new MP3(300, "Middle of the night"));

data.Add(new MP3(500, "Ride it"));

data.Add(new MP3(400, "Rolling in the deep"));

data.Add(new MP3(100, "Blinding lights"));

Directory dir = new Directory(0, data);

dir.whichTypesOfDataIsStored();

dir.showDataStoredInDir();

Console.WriteLine("Total size of dir -> " + dir.getTotalSizeOfDir());

Console.WriteLine("Amount of mP3 files in dir -> " + dir.getAmountOfMP3Files());

File txt = new File(123);

txt.writeDataInFile();

MP3 mP3 = new MP3(999, "Umbrella");

mP3.playSong();

Archive archive = new Archive(100);

Console.WriteLine("Size of archive file -> " + archive.getSize());

File unzippedArchive = archive.unzipArchive();

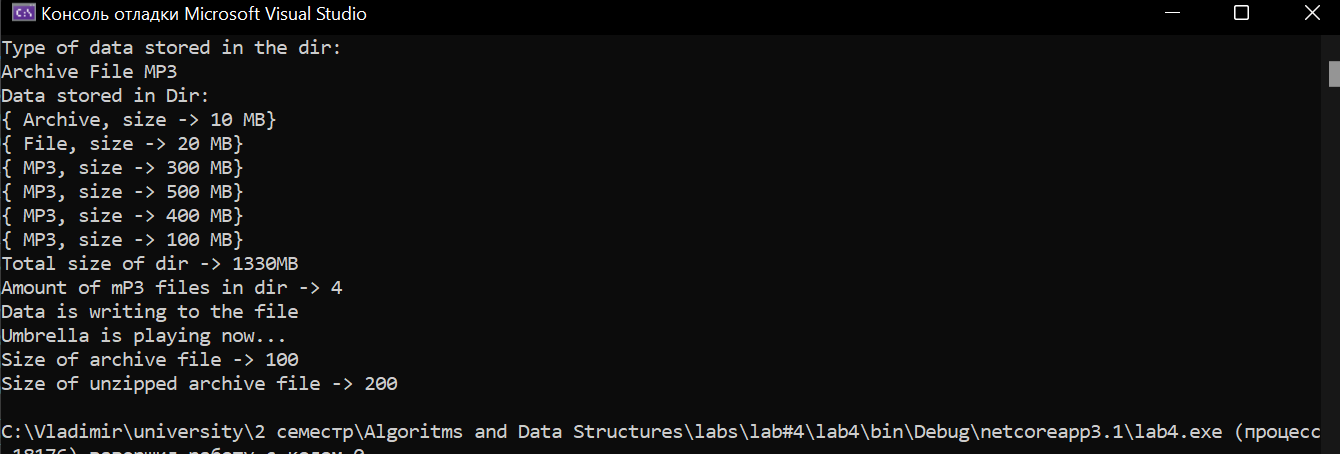
Console.WriteLine("Size of unzipped archive file -> " + unzippedArchive.getSize());

}

}

}

**Результат виконання програми**

****