**Курс**: ПЛАТФОРМА MICROSOFT .NET ТА МОВА ПРОГРАМУВАННЯ C#

**Тема:** Серіалізація об'єктів. Логування

**Модуль 14. Часть 1**

# Завдання 1:

Створіть програму для роботи з масивом цілих чисел з такою функціональністю:

1. Введення масиву цілих чисел з клавіатури.
2. Фільтр масиву. Залежно від вибору користувача, прибираємо прості числа або числа Фібоначчі.
3. Серіалізація масиву.
4. Збереження серіалізованого масиву у файл.
5. Завантаження серіалізованого масиву з файлу. Після завантаження потрібно виконати десеріалізацію.

Вибір певного формату серіалізації потрібно зробити вам. Звертаємо вашу увагу, що вибір має бути обґрунтованим.

# Завдання 2:

Створіть програму для роботи з інформацією про музичний альбом, яка зберігатиме таку інформацію:

1. Назва альбому.
2. Назва виконавця.
3. Рік випуску.
4. Тривалість.
5. Студія звукозапису.

Програма має бути з такою функціональністю:

1. Введення інформації про альбом.
2. Виведення інформації про альбом.
3. Серіалізація альбому.
4. Збереження серіалізованого альбому у файл.
5. Завантаження серіалізованого альбому з файлу. Після завантаження потрібно виконати десеріалізацію альбому.

Вибір певного формату серіалізації потрібно зробити вам. Звертаємо вашу увагу, що вибір має бути обґрунтованим.

# Завдання 3:

Додайте до попереднього завдання список пісень в альбомі. Потрібно зберігати таку інформацію про кожну пісню:

1. Назва пісні.
2. Тривалість пісні.
3. Стиль пісні.

Змініть функціональність з попереднього завдання таким чином, щоб вона враховувала перелік пісень.

Вибір певного формату серіалізації потрібно зробити вам. Звертаємо вашу увагу, що вибір має бути обґрунтованим.

# Завдання 4:

Додайте до попереднього завдання можливість створення масиву альбомів.

Змініть функціональність з другого завдання таким чином, щоб вона враховувала масив альбомів.

Вибір певного формату серіалізації потрібно зробити вам. Звертаємо вашу увагу, що вибір має бути обґрунтованим.

Завдання 1

using System;

using System.Collections.Generic;

using System.Linq;

using System.IO;

using System.Runtime.Serialization;

using System.Runtime.Serialization.Formatters.Binary;

namespace DZ

{

[Serializable]

class MyIntMassive

{

List<int> int\_list;

public List<int> Int\_list

{

set { int\_list = value; }

get { return int\_list; }

}

public MyIntMassive()

{

int\_list = new List<int>();

}

public void PrintAll()

{

for(int i=0;i<int\_list.Count;i++)

{

Console.Write(int\_list[i] + " ");

}

Console.WriteLine();

}

public bool DeleteProsti()

{

int k = 0;

for(int i=0;i<int\_list.Count;i++)

{

for(int j=1;j<=int\_list[i];j++)

{

if (int\_list[i] % j == 0)

k++;

}

if (k == 2)

{

int\_list.RemoveAt(i);

return true;

}

k = 0;

}

return false;

}

public bool DeleteFibonachi()

{

int prev = 0;

int next=1;

int fibo=0;

for(int i=0;i<int\_list.Count;i++)

{

if(int\_list[i]==0)

{

int\_list.RemoveAt(i);

return true;

}

while(fibo<int\_list[i])

{

fibo = prev + next;

prev = next;

next = fibo;

}

if(fibo== int\_list[i])

{

int\_list.RemoveAt(i);

return true;

}

prev = 0;

next = 1;

fibo = 0;

}

return false;

}

}

class Program

{

static void Main(string[] args)

{

System.Console.OutputEncoding = System.Text.Encoding.UTF8;

int a;

FileStream fs;

BinaryFormatter bf=new BinaryFormatter();

MyIntMassive myIntMassive = new MyIntMassive();

Console.WriteLine("Додати ціле число->1\nСеріалізувати масив->2\nЗчитати серіалізований масив->3\nВидалити прості числа->4\nВидалити числа Фібоначі->5\nПоказати весь->6\nВихід->0");

Console.Write("->");

try

{

int k = Convert.ToInt32(Console.ReadLine());

Console.Clear();

while (k != 0)

{

switch(k)

{

case 1:

Console.Write("Введіть ціле число->");

a= Convert.ToInt32(Console.ReadLine());

myIntMassive.Int\_list.Add(a);

Console.Clear();

break;

case 2:

fs = new FileStream("MyArr.txt", FileMode.Create, FileAccess.Write, FileShare.Write);

bf.Serialize(fs, myIntMassive);

fs.Close();

Console.WriteLine("Збережено!");

break;

case 3:

fs = new FileStream("MyArr.txt", FileMode.Open, FileAccess.Read, FileShare.Read);

myIntMassive = (MyIntMassive)bf.Deserialize(fs);

fs.Close();

Console.WriteLine("Зчитано!");

break;

case 4:

while (myIntMassive.DeleteProsti());

Console.WriteLine("Видалено!");

break;

case 5:

while (myIntMassive.DeleteFibonachi()) ;

Console.WriteLine("Видалено!");

break;

case 6:

myIntMassive.PrintAll();

break;

}

Console.WriteLine("Додати ціле число->1\nСеріалізувати масив->2\nЗчитати серіалізований масив->3\nВидалити прості числа->4\nВидалити числа Фібоначі->5\nПоказати весь->6\nВихід->0");

Console.Write("->");

k = Convert.ToInt32(Console.ReadLine());

Console.Clear();

}

}

catch(Exception e)

{

Console.WriteLine(e.Message);

}

}

}

}

Завдання 2-3

using System;

using System.Collections.Generic;

using System.Linq;

using System.IO;

using System.Runtime.Serialization;

using System.Runtime.Serialization.Formatters.Binary;

namespace DZ

{

[Serializable]

class Song

{

string name;

int min;

int sec;

string song\_style;

public int Min

{

get { return min; }

}

public int Sec

{

get { return sec; }

}

public Song(string name, int min, int sec, string song\_style)

{

this.name = name;

this.song\_style = song\_style;

this.min = min+sec / 60;

this.sec = sec%60;

}

public override string ToString()

{

return "Пісня: " + name + " Тривалість: " + min + "хв " + sec + "с Стиль пісні: " + song\_style;

}

}

[Serializable]

class MusicAlbum

{

string name;

string artist;

int year;

int min;

int sec;

string recording\_studio;

List<Song> songs;

public MusicAlbum()

{

name = "unknown";

artist = "unknown";

year = 0;

min = 0;

sec = 0;

recording\_studio= "unknown";

songs = new List<Song>();

}

public MusicAlbum(string name, string artist,int year, string recording\_studio)

{

this.name = name;

this.artist = artist;

this.year = year;

this.recording\_studio = recording\_studio;

songs = new List<Song>();

this.min = 0;

this.sec = 0;

}

public void AddSongs(Song song)

{

int m=0, s=0;

songs.Add(song);

for(int i=0;i<songs.Count;i++)

{

m += songs[i].Min;

s += songs[i].Sec;

}

this.min = m + s / 60;

this.sec = s % 60;

}

public void PrintInfo()

{

Console.WriteLine(name);

Console.WriteLine("Виконавець: "+artist);

Console.WriteLine("Назва студії звукозапису: " + recording\_studio);

Console.WriteLine("Рік релізу альбому: "+year);

Console.WriteLine("Тривалість альбому: " + min+"хв "+sec+"с");

for(int i=0;i<songs.Count;i++)

{

Console.WriteLine(songs[i]);

}

}

}

class Program

{

static void Main(string[] args)

{

System.Console.OutputEncoding = System.Text.Encoding.UTF8;

FileStream fs;

BinaryFormatter bf = new BinaryFormatter();

try

{

Console.Write("Введіть назву альбому->");

string name = Console.ReadLine();

Console.Write("Введіть назву виконавця->");

string artist = Console.ReadLine();

Console.Write("Введіть рік релізу альбому->");

int year = Convert.ToInt32(Console.ReadLine());

Console.Write("Введіть назву студії звукозапису->");

string recording\_studio = Console.ReadLine();

MusicAlbum musicAlbum = new MusicAlbum(name, artist, year, recording\_studio);

Console.Clear();

Console.WriteLine("Додати пісню->1\nСереалізувати альбом->2\nЗчитати сереалізований альбом->3\nПоказати весь альбом->4\nВихід->0");

Console.Write("->");

int k = Convert.ToInt32(Console.ReadLine());

Console.Clear();

int min;

int sec;

string song\_style;

while (k!=0)

{

switch(k)

{

case 1:

Console.Write("Введіть назву пісні->");

name = Console.ReadLine();

Console.Write("Введіть тривалість(хв)->");

min = Convert.ToInt32(Console.ReadLine());

Console.Write("Введіть тривалість(с)->");

sec = Convert.ToInt32(Console.ReadLine());

Console.Write("Введіть стиль пісні->");

song\_style = Console.ReadLine();

musicAlbum.AddSongs(new Song(name, min, sec,song\_style));

Console.Clear();

Console.WriteLine("Додано!");

break;

case 2:

fs = new FileStream("Album.txt", FileMode.Create, FileAccess.Write, FileShare.Write);

bf.Serialize(fs, musicAlbum);

fs.Close();

Console.WriteLine("Збережено!");

break;

case 3:

fs = new FileStream("Album.txt", FileMode.Open, FileAccess.Read, FileShare.Read);

musicAlbum = (MusicAlbum)bf.Deserialize(fs);

fs.Close();

Console.WriteLine("Зчитано!");

break;

case 4:

musicAlbum.PrintInfo();

break;

}

Console.WriteLine("Додати пісню->1\nСереалізувати альбом->2\nЗчитати сереалізований альбом->3\nПоказати весь альбом->4\nВихід->0");

Console.Write("->");

k = Convert.ToInt32(Console.ReadLine());

Console.Clear();

}

}

catch(Exception e)

{

Console.WriteLine(e.Message);

}

}

}

}

Завдання 4

using System;

using System.Collections.Generic;

using System.Linq;

using System.IO;

using System.Runtime.Serialization;

using System.Runtime.Serialization.Formatters.Binary;

namespace DZ

{

[Serializable]

class Song

{

string name;

int min;

int sec;

string song\_style;

public int Min

{

get { return min; }

}

public int Sec

{

get { return sec; }

}

public Song(string name, int min, int sec, string song\_style)

{

this.name = name;

this.song\_style = song\_style;

this.min = min+sec / 60;

this.sec = sec%60;

}

public override string ToString()

{

return "Пісня: " + name + " Тривалість: " + min + "хв " + sec + "с Стиль пісні: " + song\_style;

}

}

[Serializable]

class MusicAlbum

{

string name;

string artist;

int year;

int min;

int sec;

string recording\_studio;

List<Song> songs;

public MusicAlbum()

{

name = "unknown";

artist = "unknown";

year = 0;

min = 0;

sec = 0;

recording\_studio= "unknown";

songs = new List<Song>();

}

public MusicAlbum(string name, string artist,int year, string recording\_studio)

{

this.name = name;

this.artist = artist;

this.year = year;

this.recording\_studio = recording\_studio;

songs = new List<Song>();

this.min = 0;

this.sec = 0;

}

public void AddSongs(Song song)

{

int m=0, s=0;

songs.Add(song);

for(int i=0;i<songs.Count;i++)

{

m += songs[i].Min;

s += songs[i].Sec;

}

this.min = m + s / 60;

this.sec = s % 60;

}

public void PrintInfo()

{

Console.WriteLine(name);

Console.WriteLine("Виконавець: "+artist);

Console.WriteLine("Назва студії звукозапису: " + recording\_studio);

Console.WriteLine("Рік релізу альбому: "+year);

Console.WriteLine("Тривалість альбому: " + min+"хв "+sec+"с");

for(int i=0;i<songs.Count;i++)

{

Console.WriteLine(songs[i]);

}

}

}

[Serializable]

class MyAlbums

{

List<MusicAlbum> musicAlbums;

public MyAlbums()

{

musicAlbums = new List<MusicAlbum>();

}

public void AddAlbum(MusicAlbum musicAlbum)

{

musicAlbums.Add(musicAlbum);

}

public void PrintInfo()

{

for(int i=0;i<musicAlbums.Count;i++)

{

musicAlbums[i].PrintInfo();

Console.WriteLine("---------------------------------------");

}

}

}

class Program

{

static void Main(string[] args)

{

System.Console.OutputEncoding = System.Text.Encoding.UTF8;

MyAlbums myAlbums = new MyAlbums();

FileStream fs;

BinaryFormatter bf = new BinaryFormatter();

Console.WriteLine("Додати альбом->1\nСереалізувати альбоми->2\nЗчитати сереалізовані альбоми->3\nПоказати всі альбоми->4\nВихід->0");

Console.Write("->");

try

{

int k = Convert.ToInt32(Console.ReadLine());

Console.Clear();

while (k != 0)

{

switch (k)

{

case 1:

Console.Write("Введіть назву альбому->");

string name = Console.ReadLine();

Console.Write("Введіть назву виконавця->");

string artist = Console.ReadLine();

Console.Write("Введіть рік релізу альбому->");

int year = Convert.ToInt32(Console.ReadLine());

Console.Write("Введіть назву студії звукозапису->");

string recording\_studio = Console.ReadLine();

MusicAlbum musicAlbum = new MusicAlbum(name, artist, year, recording\_studio);

Console.Clear();

Console.WriteLine("Додати пісню->1\nВихід->0");

Console.Write("->");

int k2 = Convert.ToInt32(Console.ReadLine());

Console.Clear();

while (k2 != 0)

{

switch (k2)

{

case 1:

Console.Write("Введіть назву пісні->");

name = Console.ReadLine();

Console.Write("Введіть тривалість(хв)->");

int min = Convert.ToInt32(Console.ReadLine());

Console.Write("Введіть тривалість(с)->");

int sec = Convert.ToInt32(Console.ReadLine());

Console.Write("Введіть стиль пісні->");

string song\_style = Console.ReadLine();

musicAlbum.AddSongs(new Song(name, min, sec, song\_style));

Console.Clear();

break;

}

Console.WriteLine("Додати пісню->1\nВихід->0");

Console.Write("->");

k2 = Convert.ToInt32(Console.ReadLine());

Console.Clear();

}

myAlbums.AddAlbum(musicAlbum);

break;

case 2:

fs = new FileStream("Albums.txt", FileMode.Create, FileAccess.Write, FileShare.Write);

bf.Serialize(fs, myAlbums);

fs.Close();

Console.WriteLine("Збережено!");

break;

case 3:

fs = new FileStream("Albums.txt", FileMode.Open, FileAccess.Read, FileShare.Read);

myAlbums = (MyAlbums)bf.Deserialize(fs);

fs.Close();

Console.WriteLine("Зчитано!");

break;

case 4:

myAlbums.PrintInfo();

break;

}

Console.WriteLine("Додати альбом->1\nСереалізувати альбоми->2\nЗчитати сереалізовані альбоми->3\nПоказати всі альбоми->4\nВихід->0");

Console.Write("->");

k = Convert.ToInt32(Console.ReadLine());

Console.Clear();

}

}

catch (Exception e)

{

Console.WriteLine(e.Message);

}

}

}

}