using System;

using System.Collections.Generic;

using System.Linq;

using System.Security.Cryptography.X509Certificates;

using System.Text;

using System.Threading.Tasks;

using System.Collections;

namespace Classwork\_10.\_10.\_23

{

//1

/\*abstract class Base

{

public abstract void print();

}

class PP : Base

{

public override void print()

{

Console.WriteLine("dddddddd");

}

}

interface IMyInterface

{

double Summ(double x,double b);

}

class MyClass : IMyInterface {

public double Summ(double x,double b)

{

return x + b;

}

}\*/

//2

/\*class Book:IComparable

{

public string title;

public int year;

public double price;

public Book(string title, int year, double price)

{

this.title = title;

this.year = year;

this.price = price;

}

//Объяснение

//obj1<obj2 return -1

// obj1==obj2 return 0

//obj1>obj2 return 1

public int CompareTo(object obj)

{

if(obj is Book)

{

if (year < (obj as Book).year)

return 1;

else if (year == (obj as Book).year)

return 0;

else

return -1;

// return year.CompareTo((obj as Book).year);

}

//сравнение только Book исключение генерируеться если сравниваем Book с фиг чем

throw new NotImplementedException();

}

}

class ArrayBooks

{

public Book[] B;

public ArrayBooks()

{

B = null;

}

public void AddBook(Book book)

{

if(B == null)

{

B = new Book[1];

B[0]= book;

return;

}

Book[]B2 = new Book[B.Length+1];

for(int i =0;i< B.Length;i++)

{

B2[i]= B[i];

}

B2[B.Length] = book;

B = B2;

}

public void Sort()

{

Array.Sort(B);

}

public void Print()

{

Console.WriteLine("List of books:");

if (B == null)

{

Console.WriteLine("List is empty");

return;

}

for(int i =0;i<B.Length;i++)

{

Console.WriteLine("Title = "+ B[i].title);

Console.WriteLine("Year = " + B[i].year);

Console.WriteLine("Price = " + B[i].price);

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

}

}

}\*/

//3

/\*class ArrayDouble : IEnumerator, IEnumerable

{

double[] AD = null;

private int index;

public ArrayDouble(int size)

{

AD = new double[size];

Random rnd = new Random();

for (int i = 0; i < size; i++)

{

AD[i] = rnd.NextDouble() \* 10 - 5;//диапозон от -5 5

}

index = -1;

}

public IEnumerator GetEnumerator()

{

return this;

}

public object Current

{

get { return AD[index]; }

}

public void Reset()

{

index = -1;

}

public bool MoveNext()

{

if (index == AD.Length - 1)

{

Reset();

return false;

}

index++;

return true;

}

public void Print()

{

\*//\*for (int i = 0; i < AD.Length; i++)

{

Console.Write("{0:f2}",AD[i]);

}\*//\*

foreach (double i in AD) { Console.Write("{0:f2} | ", i); }

Console.WriteLine();

}

}\*/

//4

class ABC

{

public override bool Equals(object obj)

{

return this == obj;

}

public override int GetHashCode()

{

return this.GetHashCode();

}

public override string ToString()

{

return "class ABC";

}

}

internal class Program

{

static void Main(string[] args)

{

//2

/\*Book B1=new Book("Book-1",1996,100000);

Book B2 = new Book("Book-2", 2007, 10);

Book B3 = new Book("Book-3", 2006, 1000);

Book B4 = new Book("Book-4", 2001, 2000);

ArrayBooks AB = new ArrayBooks();

AB.AddBook(B1);

AB.AddBook(B2);

AB.AddBook(B3);

AB.AddBook(B4);

AB.Sort();

AB.Print();\*/

//3

/\* ArrayDouble AD=new ArrayDouble(10);

AD.Print();

foreach(double x in AD)

{

Console.Write("{0:f2} | ", x);

}\*/

//4

/\*string s = "hello world";

foreach(char c in s)

Console.Write(c+" ");

IEnumerator enumerator = s.GetEnumerator();//короч как понял на первый элемент адресс

while(enumerator.MoveNext())

{

char c = (char)enumerator.Current;

Console.Write(c +" ");

}

Console.WriteLine();\*/

}

}

}