

ממשק interface המשר - מיין

- תכונות מכון עצמים

השיטה Array.sort

2

- השיטה Array.sort מקבלת מערך וממיינת את איבריו
- עבור טיפוסים בסיסיים השיטה יודעת לבצע את העבודה
אחר והיחס >, =, < מוגדרים היטב עבור טיפוסים אלו

```
static void Main(string[] args)
{
    int[] arr = { 4, 6, 2, 8, 1 };

    Array.Sort(arr);

    for (int i=0 ; i < arr.Length ; i++)
        Console.Write("{0} ", arr[i]);
    Console.WriteLine();
}
```

```
1 2 4 6 8
Press any key to continue . . .
```

השיטה

עבור מערך אובייקטים

3

```
static void Main(string[] args)
{
    Person[] arr = new Person[3];
    arr[0] = new Person(222, "momo");
    arr[1] = new Person(333, "gogo");
    arr[2] = new Person(111, "koko");
}
```

Array.Sort(arr);

הקומpileר לא יודע למיין

```
for (int i = 0; i < arr.Length; i++)
    Console.Write("{0} ", arr[i].ToString());
Console.WriteLine();
}
```

יש תעופה בניסוי המיוון כי השיטה
Comparable מtabסת על מימוש הממשק

```
class Person
{
    private int id;
    private string name;

    public Person(int id, string name)
    {
        this.id = id;
        this.name = name;
    }

    public override string ToString()
    {
        return "Id: " + id + ", Name: " + name;
    }
}
```

Unhandled Exception: System.InvalidOperationException: Failed to compare two elements in the array. ---> System.ArgumentException: At least one object must implement IComparable.

```
at System.Collections.Comparer.Compare(Object a, Object b)
at System.Collections.Generic.ObjectComparer`1.Compare(T x, T y)
at System.Collections.Generic.ArraySortHelper`1.SwapIfGreaterWithItems(T[] keys, IComparer`1 comparer, Int32 a, Int32 b)
at System.Collections.Generic.ArraySortHelper`1.QuickSort(T[] keys, Int32 left, Int32 right)
```

השיטה sort מtabססת על הממשק IComparable

4

- לאחר ו- sort רוצה שאיברים למיון ידעו לחת תוצאה
יחס בין שני איברים (גדול/קטן/שווה), היא מtabססת על
כך שהם ימשו את ההתנהגות המוגדרת במבנה הקים
CompareTo IComparable

הממשק Comparable

5

- זהו ממשק המסופק עם השפה, ו כדי למשוך אותו צריך למשוך את השיטה:

public int compareTo(Object obj)

- השיטה תחזיר מין בסדר עולה:

0 אם this == obj זהים

-1 אם this < obj

1 אם this > obj

- ממשק זה ממומש כבר במחלקה String

- נרצה למשוך אותו כאשר נרצה לאפשר השוואת בין אובייקטים

- נעדייף להשתמש בממשק קיים זה ולא להגדיר אותו מחדש כדי לספק שפה אחידה בין המתכנתים

התיקון: מימוש IComparable

6

יש לציין את שם המחלקה,
אחרת תבוצע השוואת מול
casting, Object

השוויה לפי שדה ה- id

```
class Person : IComparable<Person>
{
    private int id;
    private string name;

    public Person(int id, string name)
    {
        this.id = id;
        this.name = name;
    }

    public override string ToString()
    {
        return "Id: " + id + ", Name: " + name;
    }

    #region IComparable<Person> Members

    public int CompareTo(Person other)
    {
        if (id < other.id) return -1;
        else if (id > other.id) return 1;
        else return 0;
    }

    #endregion
} // class Person
```

הтиיךון: שימוש ב-IComparable

7

```
static void Main(string[] args)
{
    Person[] arr = new Person[3];
    arr[0] = new Person(222, "momo");
    arr[1] = new Person(333, "gogo");
    arr[2] = new Person(111, "koko");

    Array.Sort(arr);

    for (int i = 0; i < arr.Length; i++)
        Console.WriteLine($"{i} {arr[i].ToString()}");
    Console.WriteLine();
}
```

```
Id: 111, Name: koko
Id: 222, Name: momo
Id: 333, Name: gogo
```

השוויה לפי שדה ה-name

8

```
class Person : IComparable<Person>
{
    private int id;
    private string name;

    public Person(int id, string name)
    {
        this.id = id;
        this.name = name;
    }

    public override string ToString()
    {
        return "Id: " + id + ", Name: " + name;
    }

    #region IComparable<Person> Members

    public int CompareTo(Person other)
    {
        return name.CompareTo(other.name);
    }

    #endregion
} // class Person
```

```
Id: 333, Name: gogo
Id: 111, Name: koko
Id: 222, Name: momo
```

תרגיל – הרחבות המחלקה Person

9

- הרחבת המחלקה Person שמכילה ת.ז מסוג מספר שלם ושם מסוג מחורצת□
- בצע מיוון ראשי ומיוון משנה בסדר עולה□
- המיוון הראשי יהיה לפי שם האדם (בסדר עולה)□
- במידה והשמות זהים יבוצע מיוון משנה לפי תעודת הזהות (בסדר עולה)□
- דוגמה:□

Id: 333, **Name:** gogo

Id: 111, **Name:** koko

Id: 222, Name: momo

Id: 444, Name: momo

- ממש פונקציה חיצונית חדשה בטור המחלקה Program בשם (...)
שמחליפה את (...).Array.Sort().□