Sets and Dictionaries Advanced

Sets and Multi-Dictionaries, Nested Dictionaries

SoftUni Team

Technical Trainers

Software University

http://softuni.bg

Table of Contents

Have a Question?

SoftUni Foundation

sli.do

#csharp-advanced

Associative Arrays

Collection of Key and Value Pairs

Associative Arrays (Maps, Dictionaries)

- Associative arrays are arrays indexed by keys
  - Not by the numbers 0, 1, 2, … (like arrays)
- Hold a set of pairs {`key` → `value`}

Key	Value
`John Smith`	`+1-555-8976`
`Lisa Smith`	`+1-555-1234`
`Sam Doe`	`+1-555-5030`

Dictionary

- Dictionary<K, V> - collection of key and value pairs
- Keys are **unique**
- Keeps the keys in their order of addition
- Uses a hash-table + list

```
var fruits = new Dictionary<string,
double>();
fruits["banana"] = 2.20;
fruits["apple"] = 1.40;
fruits["kiwi"] = 3.20;
```

Sorted Dictionary

- SortedDictionary<K, V>
- Keeps its keys always sorted
- Uses a balanced search tree

```
var fruits = new SortedDictionary<string,
double>();
fruits["kiwi"] = 4.50;
fruits["orange"] = 2.50;
fruits["banana"] = 2.20;
```

Built-In Methods

- Add(key, value) method

```
var airplanes = new Dictionary<string,
int>();
airplanes.Add("Boeing 737", 130);
airplanes.Add("Airbus A320", 150);
```

- Remove(key) method

```
var airplanes = new Dictionary<string,
int>();
airplanes.Add("Boeing 737", 130);
airplanes.Remove("Boeing 737");
```

Built-In Methods (2)

- ContainsKey(key)

```
var dictionary = new Dictionary<string, int>();
dictionary.Add("Airbus A320", 150);
if (dictionary.ContainsKey("Airbus A320"))
   Console.WriteLine($"Airbus A320 key exists");
```

- ContainsValue(value)

```
var dictionary = new Dictionary<string, int>();
dictionary.Add("Airbus A320", 150);
Console.WriteLine(airplanes.ContainsValue(150)); //
true
Console.WriteLine(airplanes.ContainsValue(100)); //
false
```

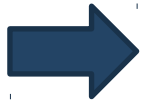
Problem: Count Same Values in Array

- Read a list of real numbers and print them along with their number of occurrences

```
8 2.5 2.5 8 2.5
```

```
8 - 2 times
2.5 - 3 times
```

```
1.5 5 1.5 3
```



```
1.5 - 2 times
5 - 1 times
3 - 1 times
```

Check your solution here: https://judge.softuni.bg/Contests/1465/Sets-and-Dictionaries-Advanced-Lab

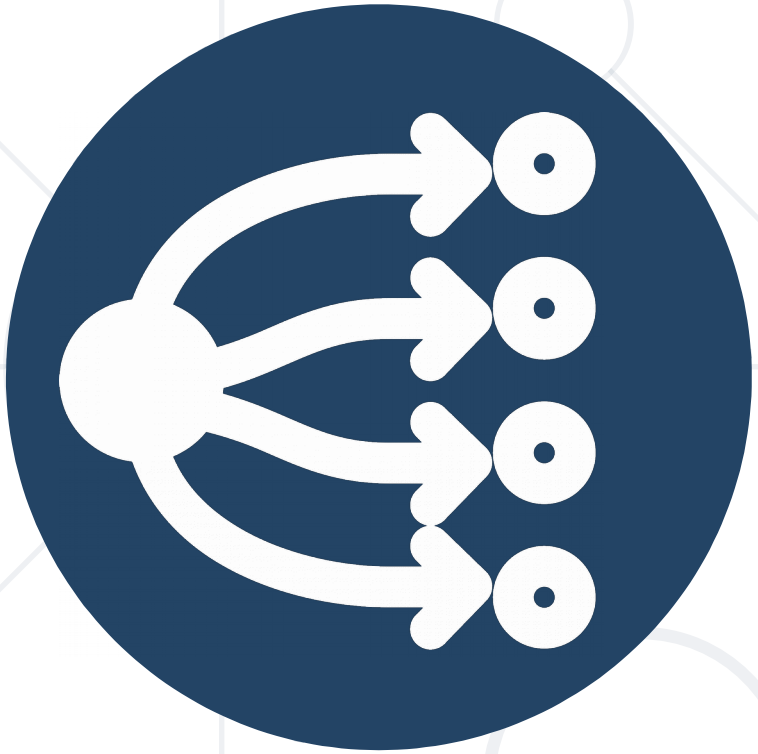
Solution: Count Same Values in Array

SoftUni Foundation

```
double[] nums = Console.ReadLine().Split(' ')
  .Select(double.Parse).ToArray();
var counts = new Dictionary<double, int>();
foreach (var num in nums)
   if (counts.ContainsKey(num))
      counts[num]++;
   else
      counts[num] = 1;
foreach (var num in counts)
    Console.WriteLine($"{num.Key} - {num.Value}
times");
```

counts[num] will hold how many times num occurs in nums

Check your solution here: https://judge.softuni.bg/Contests/1465/Sets-and-Dictionaries-Advanced-Lab



Multi-Dictionaries

Dictionaries Holding a List of Values

Multi-Dictionaries

- A dictionary could hold a **set of values** by given key
  - Example: student may have multiple grades:
    - Peter → [5, 5, 6]
    - Kiril → [6, 6, 3, 4, 6]

```
var grades = new Dictionary<string,
List<int>>();
grades["Peter"] = new List<int>();
grades["Peter"].Add(5);
grades["Kiril"] = new List<int>() { 6, 6, 3, 4,
6 };
Console.WriteLine(string.Join(" ",
```

Problem: Average Student Grades

- Write a program to read student names + grades
- Print the grades + average grade for each student

```
6
Ivancho
5.20
Mariika
5.50
Mariika
2.50
Stamat 2.00
Mariika
3.46
```

```
Ivancho -> 5.20 (avg: 5.20)
Mariika -> 5.50 2.50 3.46 (avg: 3.82)
Stamat -> 2.00 3.00 (avg: 2.50)
```

Check your solution here: https://judge.softuni.bg/Contests/1465/Sets-and-Dictionaries-Advanced-Lab

Solution: Average Student Grades

SoftUni Foundation

```
var grades = new Dictionary<string,
List<double>>();
var n = int.Parse(Console.ReadLine());
for (int i = 0; i < n; i++) {
  var tokens = Console.ReadLine().Split();
  var name = tokens[0];
  var grade = double.Parse(tokens[1]);
  if (!grades.ContainsKey(name))
    grades[name] = new List<double>();
  grades[name].Add(grade);
}
// continues on next slide ...
```

Make sure the list is initialized

Add grade into the list

Check your solution here: https://judge.softuni.bg/Contests/1465/Sets-and-Dictionaries-Advanced-Lab

Solution: Average Student Grades (2)

SoftUni Foundation

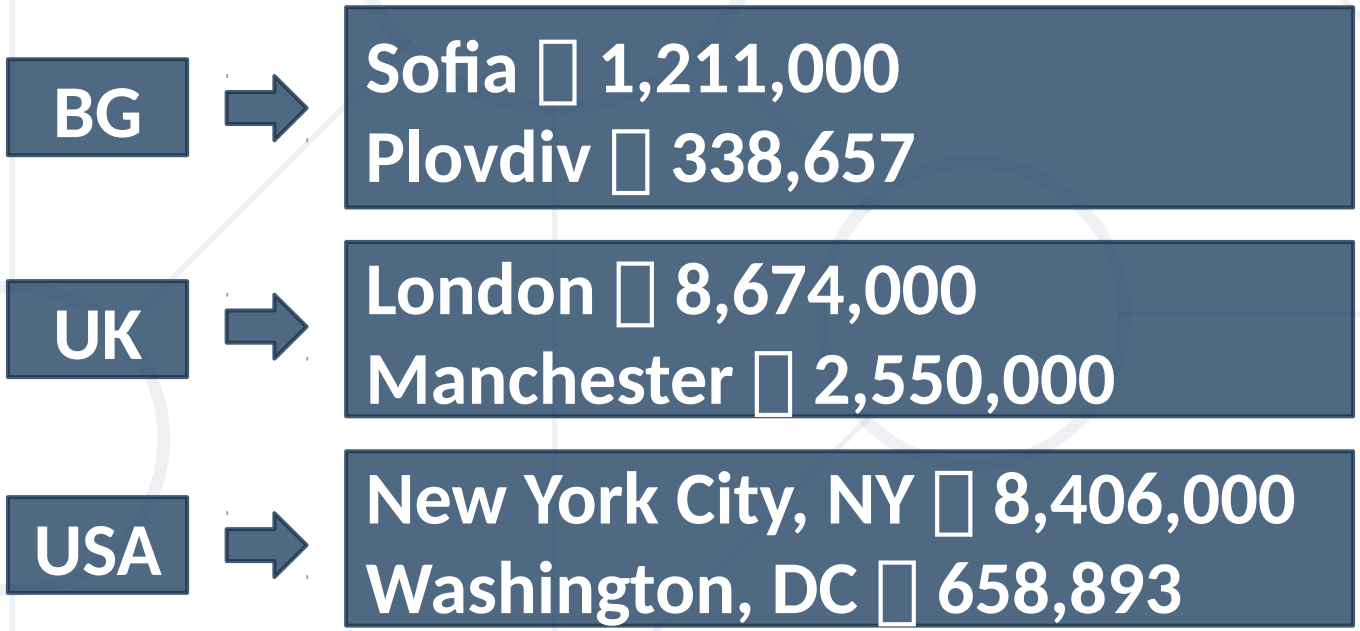
```
foreach (var pair in grades)
{
  var name = pair.Key;
  var studentGrades = pair.Value;
  var average = studentGrades.Average();
  Console.Write($"{name} -> ");
  foreach (var grade in studentGrades)
    Console.Write($"{grade:f2} ");
  Console.WriteLine($"(avg: {average:f2})");
}
```

KeyValuePair<string, List<double>

Check your solution here: https://judge.softuni.bg/Contests/1465/Sets-and-Dictionaries-Advanced-Lab

Nested Dictionaries

- Dictionaries may hold another **dictionary** as value
- Example: population by country and city



Problem: Product Shop

- Write a program that stores information about food shops
- If you receive a shop you already have received add the product
- Your output must be ordered by shop name

{shop}, {product}, {price}

```
lidl, juice, 2.30
kaufland, banana, 1.10
lidl, grape, 2.20
Revision
```

End command

```
kaufland->
Product: banana, Price: 1.1
lidl->
Product: juice, Price: 2.3
Product: grape, Price: 2.2
```

Check your solution here: https://judge.softuni.bg/Contests/1465/Sets-and-Dictionaries-Advanced-Lab

Solution: Product Shop

SoftUni Foundation

```
var shops = new Dictionary<string, Dictionary<string, dou-
ble>>();
string line;
while ((line = Console.ReadLine()) != "Revision")
{
  string[] productsInfo = line.Split(", ");
  string shop = productsInfo[0];
  string product = productsInfo[1];
  double price = double.Parse(productsInfo[2]);
  // continues on next slide
```

Check your solution here: https://judge.softuni.bg/Contests/1465/Sets-and-Dictionaries-Advanced-Lab

Solution: Product Shop (2)

SoftUni Foundation

```
    if (!shops.ContainsKey(shop))
    {
        shops.Add(shop, new Dictionary<string, double>());
    }
    shops[shop].Add(product, price);
}
var orderedShops =
shops.OrderBy(s => s.Key).ToDictionary(x => x.Key, x => x.-
Value);
//TODO: Print the ordered dictionary
```

Make sure the inner dictionary is initialized

Check your solution here: https://judge.softuni.bg/Contests/1465/Sets-and-Dictionaries-Advanced-Lab

Problem: Cities by Continent and Country

SoftUni Foundation

- Write a program to read continents, countries and their cities, put them in a nested dictionary and print them

```
6
Europe Bulgaria Sofia
Asia China Beijing
Asia Japan Tokyo
Europe Poland Warsaw
Europe Germany Berlin
Europe Poland Poznan
```

```
Europe:
  Bulgaria -> Sofia
  Poland -> Warsaw, Poznan
  Germany -> Berlin
Asia:
  China -> Beijing
  Japan -> Tokyo
```

Check your solution here: https://judge.softuni.bg/Contests/1465/Sets-and-Dictionaries-Advanced-Lab

Solution: Cities by Continent and Country

SoftUni Foundation

```
var continentsData =
        new Dictionary<string, Dictionary<string,
List<string>>>();
var n = int.Parse(Console.ReadLine());
for (int i = 0; i < n; i++) {
  var tokens = Console.ReadLine().Split();
  var continent = tokens[0];
  var country = tokens[1];
  var city = tokens[2];
  // continues on next slide
```

Check your solution here: https://judge.softuni.bg/Contests/1465/Sets-and-Dictionaries-Advanced-Lab

Solution: Cities by Continent and Country (2)

SoftUni Foundation

```
  if (!continentsData.ContainsKey(continent)) {
    continentsData[continent] = new Dictionary<string,
List<string>>();
  }
  if (!continentsData[continent].ContainsKey(country)) {
    continentsData[continent][country] = new List<string>();
  }
  continentsData[continent][country].Add(city);
}
// continues on next slide...
```

Initialize continet

Initialize cities

Append a city to the country

Check your solution here: https://judge.softuni.bg/Contests/1465/Sets-and-Dictionaries-Advanced-Lab

Solution: Cities by Continent and Country (3)

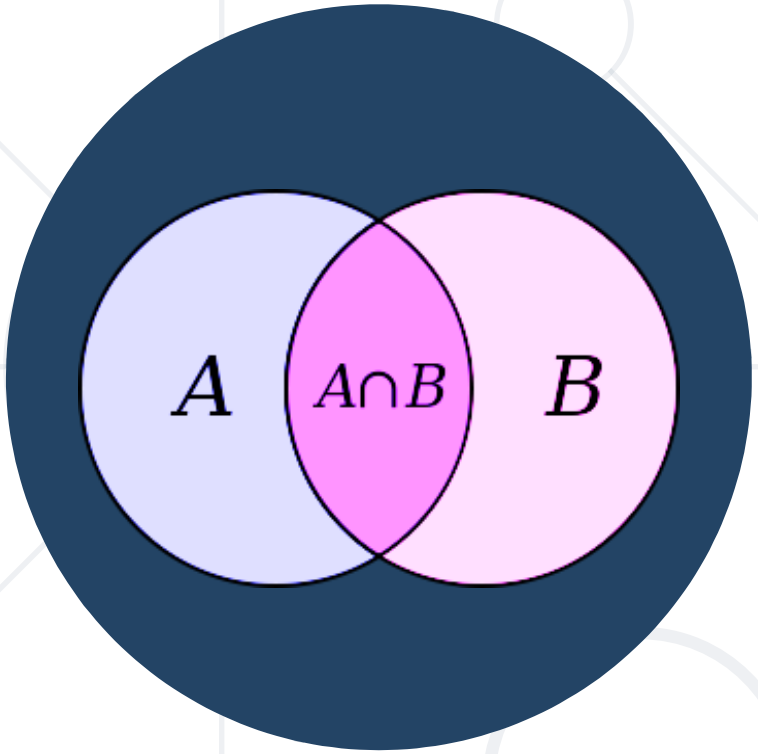
SoftUni Foundation

```
foreach (var continentCountries in continentsData) {
  var continentName = continentCountries.Key;
  Console.WriteLine($"{continentName}:");
  foreach (var countryCities in continentCountries.-
Value) {
    var countryName = countryCities.Key;
    var cities = countryCities.Value;
    // TODO: Print each country with its cities
  }
}
```

Cities in the country

Check your solution here: https://judge.softuni.bg/Contests/1465/Sets-and-Dictionaries-Advanced-Lab

Live Exercises



Sets

HashSet<T> and SortedSet<T>

Sets in C#

- A set keeps **unique elements**
  - Allows **add** / **remove** / **search** elements
  - Very **fast performance**
- HashSet<T>
  - Keeps a set of elements in a **hash-table**
  - Elements are in **no particular order**
  - Similar to List<T>, but a different implementation

List<T> vs HashSet<T>

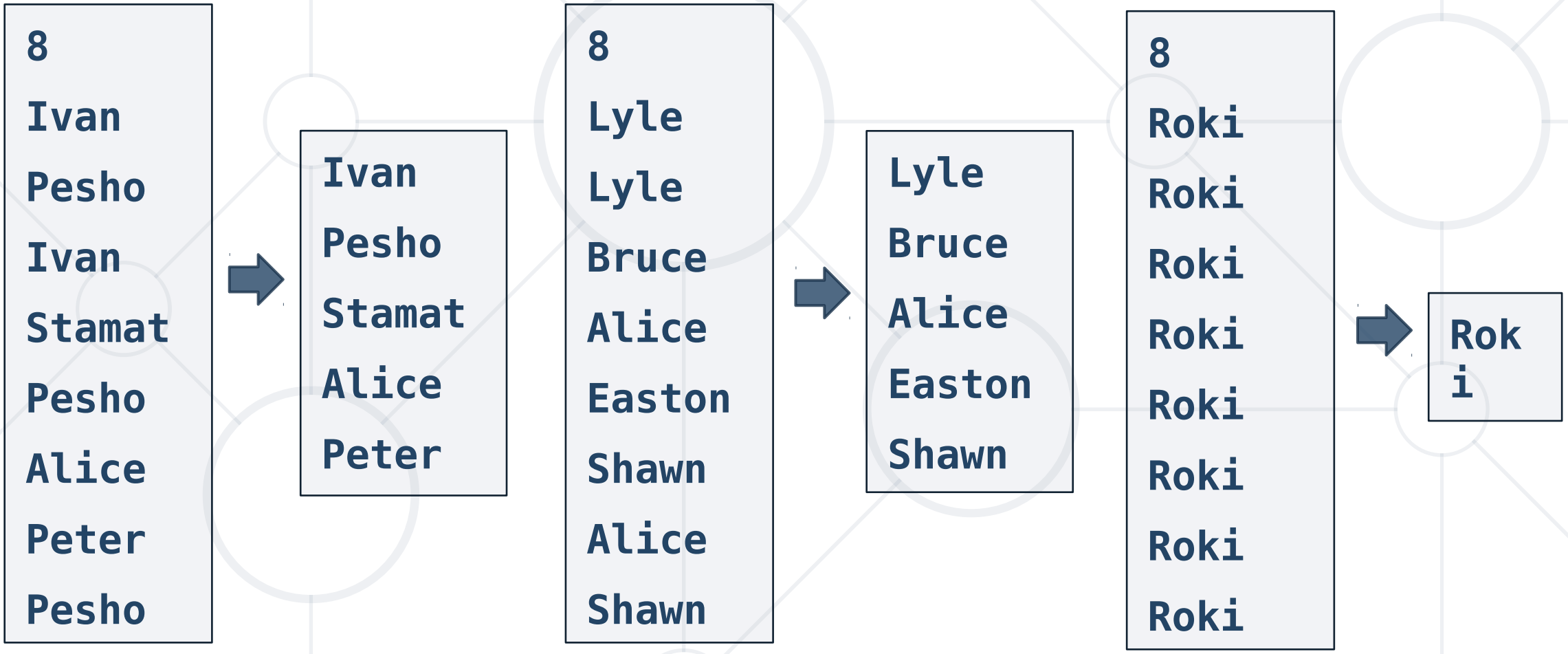
- List<T>
  - Fast "add", slow "search" and "remove" (pass through each element)
  - Duplicates are allowed
  - Insertion order is guaranteed
- HashSet<T>
  - Fast "add", "search" and "remove" thanks to hash-table
  - Does not allow duplicates
  - Does not guarantee the insertion order

HashSet<T> – Example


```
HashSet<string> set = new HashSet<string>();
set.Add("Pesho");
set.Add("Pesho"); // Not added again
set.Add("Gosho");
Console.WriteLine(string.Join(", ", set)); // Pesho, Gosho
Console.WriteLine(set.Contains("Georgi")); // false
Console.WriteLine(set.Contains("Pesho")); // true
set.Remove("Pesho");
Console.WriteLine(set.Count); // 1
```

Problem: Record Unique Names

- Read a sequence of names and print only the unique ones



Check your solution here: https://judge.softuni.bg/Contests/1465/Sets-and-Dictionaries-Advanced-Lab

Solution: Record Unique Names

```
var names = new HashSet<string>();
var n =
int.Parse(Console.ReadLine());
for (int i = 0; i < n; i++)
{
  var name = Console.ReadLine();
  names.Add(name);
}
foreach (var name in names)
  Console.WriteLine(name);
```

HashSet stores unique values

Adds non-existing names only

Check your solution here: https://judge.softuni.bg/Contests/1465/Sets-and-Dictionaries-Advanced-Lab

SortedSet<T>

- SortedSet<T>
  - The elements are **ordered incrementally**

```
var set = new SortedSet<string>();
set.Add("Pesho");
set.Add("Pesho");
set.Add("Gosho");
set.Add("Maria");
set.Add("Alice");
Console.WriteLine(string.Join(", ",
set));
```

Alice, Gosho, Maria, Pesho

Live Exercises

Summary

SoftUni Foundation

- Multi-dictionaries allow **keeping a collection** as a **dictionary value**
- Nested dictionaries **allow keeping a dictionary** as **dictionary value**
- Sets allow keeping **unique values** in **unspecified order**
  - No duplicates
  - Fast add, search & remove

Questions?

SoftUni

Software University

SoftUni Svetlina

SoftUni Creative

SoftUni Digital

SoftUni Foundation

SoftUni Kids

SoftUni Diamond Partners

SoftUni Foundation

XSsoftware

SBTech
we know sports

telenor

SoftwareGroup
doing it right

NETPEAK

SmartIT

SUPER HOSTING .BG

INDEAVR
Serving the high achievers

INFRAGISTICS®

SoftUni Organizational Partners

SoftUni Foundation

Trainings @ Software University (SoftUni)

SoftUni Foundation

- Software University – High-Quality Education and Employment Opportunities
  - softuni.bg
- Software University Foundation
  - http://softuni.foundation/
- Software University @ Facebook
  - facebook.com/SoftwareUniversity
- Software University Forums
  - forum.softuni.bg

License

SoftUni Foundation

- This course (slides, examples, demos, videos, homework, etc.) is licensed under the "Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International" license