

Introduction to .NET

Florin Olariu

“Alexandru Ioan Cuza”, University of Iași

Department of Computer Science

Agenda

- ▶ Arrays
 - ▶ Demo
- ▶ Generic list
 - ▶ Demo
- ▶ What's Next
- ▶ Interview questions

Arrays

The background of the slide features abstract, overlapping geometric shapes in various shades of blue, ranging from light sky blue to deep navy blue. These shapes are primarily located on the right side and bottom of the frame, creating a modern, dynamic aesthetic.

Arrays

- ▶ Definition
 - ▶ Samples
- ▶ Declaring and populating an array
- ▶ Using collection initializers
- ▶ Retrieving an element from an array
- ▶ Iterating an array
- ▶ Using array methods
- ▶ Best practices
- ▶ Demo

Arrays

- ▶ Definition

Arrays

- ▶ Definition

- ▶ *Is a fixed-size list of elements that can be accessed using a positional index number*

Arrays

- ▶ Definition

- ▶ *Is a fixed-size list of elements that can be accessed using a positional index number*
- ▶ *Sample:*

| |
|-------|
| Red |
| White |
| Green |
| Blue |

| |
|---------------|
| "Salt" 2.21 |
| "Pepper" 3.43 |
| "Onion" 5.01 |
| "Garlic" 3.20 |

Arrays

► Definition

- *Is a fixed-size list of elements that can be accessed using a positional index number*
- *Sample:*

| | |
|---|-------|
| 0 | Red |
| 1 | White |
| 2 | Green |
| 3 | Blue |

| | |
|---|---------------|
| 0 | "Salt" 2.21 |
| 1 | "Pepper" 3.43 |
| 2 | "Onion" 5.01 |
| 3 | "Garlic" 3.20 |

Arrays

- ▶ Definition
 - ▶ Samples
- ▶ Declaring and populating an array

Arrays

- ▶ Definition
 - ▶ Samples
- ▶ Declaring and populating an array
 - ▶ Declaring an array:

| |
|-------|
| Red |
| White |
| Green |
| Blue |

▶ `string[] colors;`

Arrays

- ▶ Definition
 - ▶ Samples
- ▶ Declaring and populating an array
 - ▶ Declaring an array:

| |
|-------|
| Red |
| White |
| Green |
| Blue |

▶ `string[] colors;`

Initializing an array:

```
string[] colors;  
colors = new string[4];  
string[] colors = new string[4];  
var colors = new string[4];
```

Arrays

- ▶ Definition
 - ▶ Samples
- ▶ Declaring and populating an array
 - ▶ Populating an array

Arrays

- ▶ Definition
 - ▶ Samples
- ▶ Declaring and populating an array

- ▶ Populating an array

```
var colors = new string[4];
```

```
colors[0] = "Red";
```

```
colors[1] = "White";
```

```
colors[2] = "Green";
```

```
colors[3] = "Blue";
```

Arrays

► Best practices

| Do | Avoid |
|--|--|
| Use arrays when the size is known at the design time | Do not use arrays when the data comes from a database call |
| For an array name use 'pluralization' => Colors | |

Arrays

- ▶ Demo
 - ▶ Working with arrays
 - ▶ Note: How to create a R# template for tests

Generic List

Generic List

- ▶ Definition
- ▶ Arrays vs Generic List
- ▶ Declaring and populating Generic Lists
- ▶ Using initializers
- ▶ Retrieving elements from Generic lists
- ▶ Iterating through a Generic List
- ▶ Demo

Generic List

- ▶ Definition

- ▶ *It is a strongly typed list of elements that is accessed using a positional index number.*

Generic List

- ▶ Arrays vs Generic List

Generic List

► Arrays vs Generic List

| Arrays | Generic List |
|--------|--------------|
| | |
| | |
| | |
| | |

Generic List

► Arrays vs Generic List

| Arrays | Generic List |
|----------------|----------------|
| Strongly typed | Strongly typed |
| | |
| | |
| | |

Generic List

► Arrays vs Generic List

| Arrays | Generic List |
|----------------|----------------|
| Strongly typed | Strongly typed |
| Fixed length | Expandable |
| | |
| | |

Generic List

► Arrays vs Generic List

| Arrays | Generic List |
|--|------------------------------------|
| Strongly typed | Strongly typed |
| Fixed length | Expandable |
| There is no ability to add/remove elements | Can add, insert or remove elements |
| | |

Generic List

► Arrays vs Generic List

| Arrays | Generic List |
|--|------------------------------------|
| Strongly typed | Strongly typed |
| Fixed length | Expandable |
| There is no ability to add/remove elements | Can add, insert or remove elements |
| Multi-dimensional | One-dimensional |

Generic List

- ▶ Declaring and populating Generic Lists

Generic List

- ▶ Declaring and populating Generic Lists

- ▶ `var cities = new List<string>();`

`cities.Add(" London");`

`cities.Add(" Paris");`

`cities.Add(" Milan");`

Generic List

- ▶ Declaring and populating Generic Lists

```
var cities = new List<string>{"London", "Paris", "Milan"};
```

Generic List

- ▶ Frequently asked questions
- ▶ When is appropriate to use a generic list?

Generic List

- ▶ Frequently asked questions
- ▶ When is appropriate to use a generic list?
 - ▶ Any time the application needs to manage a list of things
- ▶ What are the key differences between an array and a generic list?

Generic List

- ▶ Frequently asked questions
- ▶ When is appropriate to use a generic list?
 - ▶ Any time the application needs to manage a list of things
- ▶ What are the key differences between an array and a generic list?
 - ▶ An array is fixed length and can have multiple dimensions
 - ▶ A generic list can have any length and provides methods to add, insert or remove elements
- ▶ Execution time

Generic List

- ▶ Frequently asked questions
- ▶ When is appropriate to use a generic list?
 - ▶ Any time the application needs to manage a list of things
- ▶ What are the key difference between an array and a generic list?
 - ▶ An array is fixed length and can have multiple dimensions
 - ▶ A generic list can have any length and provides methods to add, insert or remove elements
- ▶ Execution time

```
List/for: 1971ms (589725196)  
Array/for: 1864ms (589725196)  
List/foreach: 3054ms (589725196)  
Array/foreach: 1860ms (589725196)
```

Generic List

- ▶ Demo
 - ▶ Initialization, Add, Insert, Remove, RemoveAt;

What's next ...

- ▶ Generic dictionaries
- ▶ Generic collection interfaces
- ▶ LINQ

Interview questions

One more thing...

- ▶ *"Walking on water and developing software from a specification are easy..."*

One more thing...

- ▶ *"Walking on water and developing software from a specification are easy if both are frozen."*
 - ▶ - Edward V Berard

Questions

- ▶ Do you have any other questions?

Thanks!

See you next time! 😊