

NOVOSTI V C# 13

Damir Arh, Razum d.o.o.

Microsoft MVP

O MENI

- Razum d.o.o.
- Microsoft MVP
- <https://damirscorner.com>
- @DamirArh@mas.to
- [@DamirArh](#)



PREGLED NOVOSTI

- Razširjena podpora zbirk s params
- Nov tip Lock v .NET 9
- Delne lastnosti in indeksi
- Razširjena podpora tipom ref struct
- Indeks s konca pri inicializaciji instanc
- Ubežna sekvenca \e
- Nastavljiva prioriteta preobloženih metod

TRENTNA RAZPOLOŽLIVOST

- Predogledna različica Visual Studio 2022 v 17.12
- Vklp predogledne različice C# v projektu

```
<PropertyGroup>  
  <LangVersion>preview</LangVersion>  
</PropertyGroup>
```

PARAMS PRED C# 13

- params deluje samo s polji

```
string JoinNames(params string[] names) =>  
    string.Join(", ", names);
```

- Klic s poljubnim številom argumentov

```
var joined = JoinNames("John", "Jane");
```

PARAMS V C# 13

- Podpora istim tipom kot zbirke v izrazih

```
string JoinNames(params string[] names) =>
    string.Join(", ", names);

string JoinNames(params IEnumerable<string> names) =>
    string.Join(", ", names);

string JoinNames(params ReadOnlySpan<string?> names) =>
    string.Join(", ", names);
```

RAZŠIRJENA PODPORA ZBIRK S PARAMS

Demo

KLJUČNA BESEDA LOCK PRED .NET 9

```
public class MyClass
{
    private object myLock = new object();

    public void MyMethod()
    {
        lock (myLock)
        {
            // ...
        }
    }
}
```


KLJUČNA BESEDA LOCK V .NET 9 IN C# 13

```
public class MyClass
{
    private Lock myLock = new Lock();

    public void MyMethod()
    {
        lock (myLock)
        {
            // ...
        }
    }
}
```

NOV TIP LOCK V .NET 9

Demo

DELNE METODE PRED C# 9

```
public partial class Partial
{
    partial void PartialMissing();

    public void PublicPartialMissing()
    {
        PartialMissing();
    }
}
```

DELNE METODE V C# 9

```
public partial class Partial
{
    protected partial bool PartialTry(double input, out int ou
    public bool PublicPartialTry(double input, out int output)
    {
        return PartialTry(input, out output);
    }
}
```

```
public partial class Partial
{
    protected partial bool PartialTry(double input, out int ou
    {
        //
```

DELNE LASTNOSTI IN REGULARNI IZRAZI

- Od .NET 7 dalje

```
[GeneratedRegex(@"\d+")]  
private static partial Regex IsNumeric();
```

- Od .NET 9 dalje

```
[GeneratedRegex(@"\d+")]  
private static partial Regex IsNumeric { get; }
```

DELNE LASTNOSTI V MVVM TOOLKIT

- Pred .NET 9

```
[ObservableProperty]  
private string? name;
```

- Izvedljivo z .NET 9

```
[ObservableProperty]  
public partial string Name { get; private set; }
```

DELNE LASTNOSTI IN INDEKSI

Demo

TIPI REF STRUCT IN VMESNIKI

- Dovoljeno v C# 13

```
ref struct ReadOnlySpanWrapper<T> : IReadOnlyList<T> {  
    // ...  
}
```

- Ni dovoljeno v C# 13

```
var wrapper = new ReadOnlySpanWrapper<char>(span);  
var cast = (IReadOnlyList<char>)wrapper; // error
```


TIPI REF STRUCT V GENERIČNIH METODAH IN TIPIH

```
static int GetCustomHash<T>(T list)
    where T : IReadOnlyList<char>, allows ref struct
{
    // ...
}
```

TIPI REF STRUCT V ASINHRONIH METODAH

```
async Task<int> AsyncMethodWithRefStruct()  
{  
    var span = "NTK2024".AsSpan();  
    var length = span.Length;  
    await Task.Delay(delay);  
    length = span.Length; // ERROR  
    return length;  
}
```

- enako velja za bloke unsafe v kodi

TIPI REF STRUCT V ITERATORJIH

```
IEnumerable<int> IteratorMethodWithRefStruct()  
{  
    var span = "NTK2024".AsSpan();  
    var length = span.Length;  
    yield return length;  
    length = span.Length; // ERROR  
}
```

- enako velja za bloke unsafe v kodi

RAZŠIRJENA PODPORA TIPOM REF STRUCT

Demo

UPORABA INDEKSOV OB INICIALIZACIJI INSTANC

- Podprta sintaksa od C# 3

```
var thing = new IndexersExample
{
    [1] = '1',
    [2] = '4',
    [3] = '9',
}
```

- Tip mora implementirati indeks

```
public char this[int i] { set { /*...*/ } };
```

INDEKSI S KONCA

- Podprta sintaksa od C# 8

```
string[] words = [ // ^n==length-n
    "first",        // index from start      index from end
    "second",      // 0                ^4==4-4
    "third",       // 1                ^3==4-3
    "fourth",      // 2                ^2==4-2
    ];              // 3                ^1==4-1
```

- Od C# 13 dalje tudi ob inicializaciji instanc

INDEKS S KONCA PRI INICIALIZACIJI INSTANC

Demo

UBEŽNE SEKVENCE ANSI

- Pred C# 13

```
Console.WriteLine("This is a \u001b[1mbold\u001b[0m text");
```

- S C# 13

```
Console.WriteLine("This is a \e[1mbold\e[0m text");
```

- `\e` namesto `\u001b`

UBEŽNA SEKVENCA \E

Demo

NASTAVLJANJE PRIORITETE PREOBLOŽENIH METOD

```
public class ModifiedPriority
{
    [OverloadResolutionPriority(1)]
    public string Method(ReadOnlySpan<int> s) => "Span";

    public string Method(int[] a) => "Array";
}

int[] array = [1, 2, 3];
var overload = new ModifiedPriority();
var choice = overload.Method(array);
choice.Should().Be("Span"); // not Array
```

NASTAVLJIVA PRIORITETA PREOBLOŽENIH METOD

Demo

C# 13 JE ŠE V DELU

- Spremembe so še vedno možne
 - Razširitveni tipi so bili prvotno predvideni
 - Dodatne funkcionalnosti so že v delu

VIRI

- damirscorner.com/link/
 - [Cs13GitHub](#)
 - [Cs13Docs](#)
 - [Cs13Status](#)