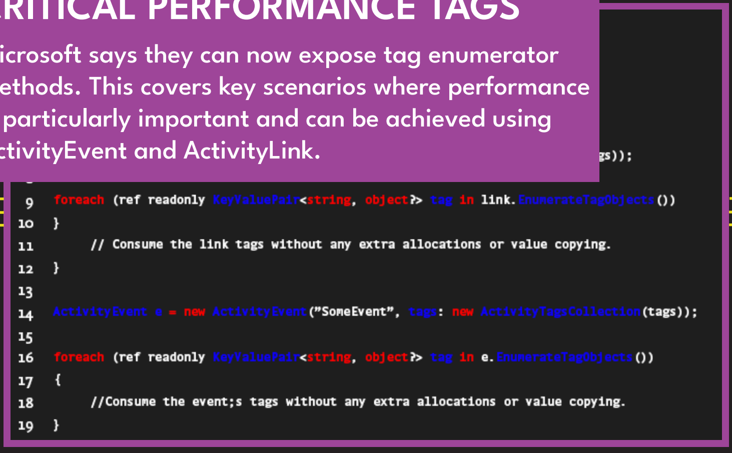
.NET 7
WHAT'S NEW?

If you work with the .NET Framework it's no secret that Microsoft has been working on .NET 7 for quite some time. The code has been tested, and polished and is available for users to enjoy. But what are some of the new features and why is it better than ever?

CRITICAL PERFORMANCE TAGS

Microsoft says they can now expose tag enumerator methods. This covers key scenarios where performance is particularly important and can be achieved using ActivityEvent and ActivityLink.

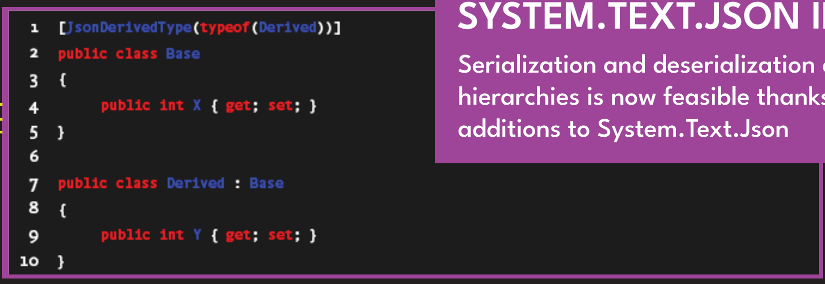


```
gs));
foreach (ref readonly KeyValuePair<string, object?> tag in link.EnumerateTagObjects())
}
    // Consume the link tags without any extra allocations or value copying.
}

ActivityEvent e = new ActivityEvent("SomeEvent", tags: new ActivityTagsCollection(tags));

foreach (ref readonly KeyValuePair<string, object?> tag in e.EnumerateTagObjects())
{
    //Consume the event;s tags without any extra allocations or value copying.
}
```

GAMES

SYSTEM.TEXT.JSON IMPROVEMENTS

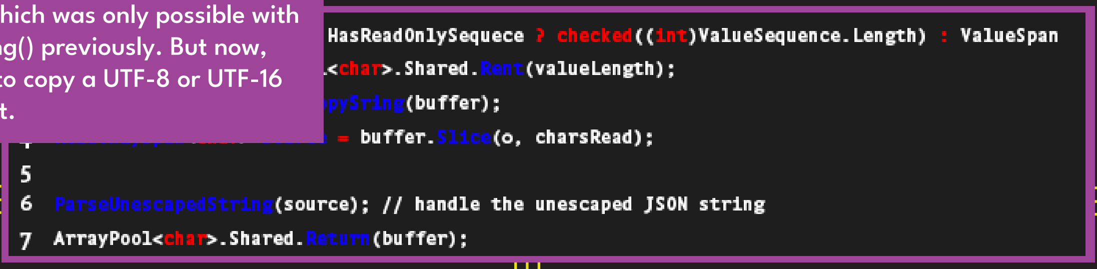
Serialization and deserialization of polymorphic type hierarchies is now feasible thanks to the new additions to System.Text.Json



```
[JsonDerivedType(typeof(Derived))]
public class Base
{
    public int X { get; set; }
}

public class Derived : Base
{
    public int Y { get; set; }
}
```

COPYSTRING UTF-8 & UTF-16

.NET 7 now includes new methods for working with decoded JSON strings, which was only possible with Utf8JsonReader.GetString() previously. But now, CopyString can be used to copy a UTF-8 or UTF-16 string without decoding it.

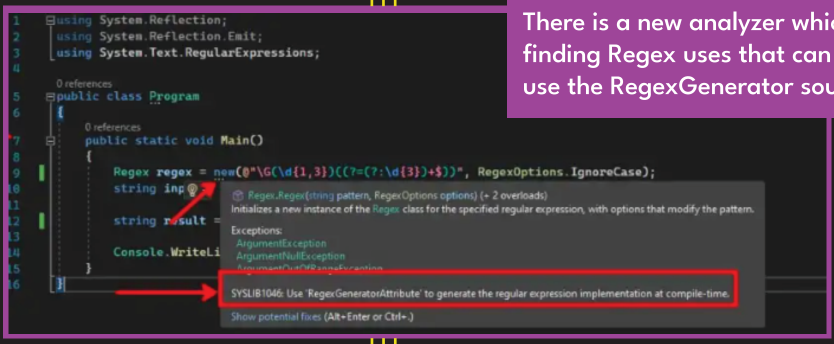


```
HasReadOnlySequece ? checked((int)ValueSequence.Length) : ValueSpan
.<char>.Shared.Rent(valueLength);
pySring(buffer);
 = buffer.Slice(o, charsRead);

ParseUnescapedString(source); // handle the unescaped JSON string
ArrayPool<char>.Shared.Return(buffer);
```

IOT

ANALYZER

There is a new analyzer which is responsible for finding Regex uses that can be transformed to use the RegexGenerator source generator.



```
using System.Reflection;
using System.Reflection.Emit;
using System.Text.RegularExpressions;

0 references
public class Program
{
    0 references
    public static void Main()
    {
        Regex regex = new(@"\G(\d{1,3})((?=(?:\d{3})+$))", RegexOptions.IgnoreCase);
        string inp
        
        string result =
        
        Console.WriteLi
    }
}
```

Regex.Regex(string pattern, RegexOptions options) (+ 2 overloads)
Initializes a new instance of the Regex class for the specified regular expression, with options that modify the pattern.

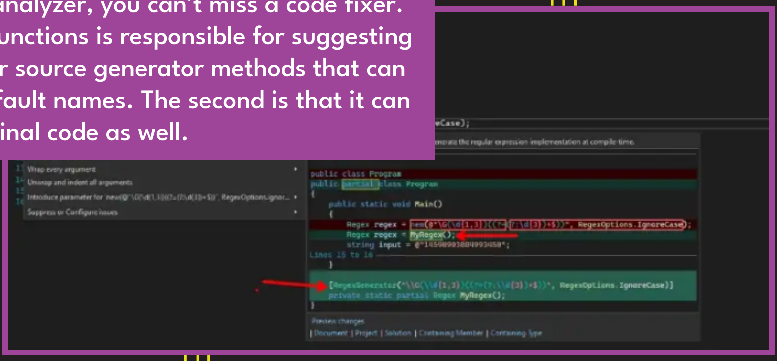
Exceptions:
ArgumentException
ArgumentNullException
ArgumentOutOfRangeException

SYSLIB1046: Use 'RegexGeneratorAttribute' to generate the regular expression implementation at compile-time.

Show potential fixes (Alt+Enter or Ctrl+.)

CODE FIXER

If you have an analyzer, you can't miss a code fixer. One of its two functions is responsible for suggesting RegexGenerator source generator methods that can override the default names. The second is that it can replace the original code as well.



```
eCase);
```

enerate the regular expression implementation at compile-time.

Wrap every argument
Unwrap and indent all arguments
Introduce parameter for 'new(@"\G(\d{1,3})((?=(?:\d{3})+$))", RegexOptions.Ignor...
Suppress or Configure issues

```
public class Program
public partial class Program
{
    public static void Main()
    {
        Regex regex = new(@"\G(\d{1,3})((?=(?:\d{3})+$))", RegexOptions.IgnoreCase);
        Regex regex = MyRegex();
        string input = @"14590903884993450";
Lines 15 to 16
    }

    [RegexGenerator("\\G(\\d{1,3})((?=(?:\\d{3})+$))", RegexOptions.IgnoreCase)]
    private static partial Regex MyRegex();
}
```

Preview changes
| Document | Project | Solution | Containing Member | Containing Type

DESKTOP

develop.

marketing@developrec.net
0207 733 0430

Are you looking for a new software engineering role?
We place the top technical talent across the globe, and we'd love to help you find your next dream role.