

LinksPlatform's Platform.Incrementers Class Library

1.1 ./csharp/Platform.Incrementers/IIncrementer.cs

```
1 using System.Runtime.CompilerServices;
2
3 namespace Platform.Incrementers
4 {
5     /// <summary>
6     /// <para>Defines an incrementer that increments any object.</para>
7     /// <para>Определяет инкрементер, который выполняет приращение любого объекта.</para>
8     /// </summary>
9     public interface IIncrementer
10    {
11        /// <summary>
12        /// <para>Increments the object by a specific value.</para>
13        /// <para>Выполняет приращение объекта на определённое значение.</para>
14        /// </summary>
15        [MethodImpl(MethodImplOptions.AggressiveInlining)]
16        void Increment();
17    }
18 }
```

1.2 ./csharp/Platform.Incrementers/IIncrementer[TNumber].cs

```
1 using System.Runtime.CompilerServices;
2
3 namespace Platform.Incrementers
4 {
5     /// <summary>
6     /// <para>Defines an incrementer that increments any number with a specific type.</para>
7     /// <para>Определяет инкрементер, который выполняет приращение любого числа определённого
8     ↪ типа.</para>
9     /// </summary>
10    /// <typeparam name="TNumber"><para>Type of incremented number.</para><para>Тип
11    ↪ увеличиваемого числа.</para></typeparam>
12    public interface IIncrementer<TNumber>
13    {
14        /// <summary>
15        /// <para>Increments the number by a specific value.</para>
16        /// <para>Увеличивает число на определённое значение.</para>
17        /// </summary>
18        /// <param name="number"><para>The number to be incremented.</para><para>Увеличиваемое
19        ↪ число.</para></param>
20        /// <returns><para>The incremented number.</para><para>Увеличенное
21        ↪ число.</para></returns>
22        [MethodImpl(MethodImplOptions.AggressiveInlining)]
23        TNumber Increment(TNumber number);
24    }
25 }
```

1.3 ./csharp/Platform.Incrementers/Incrementer.cs

```
1 using System.Runtime.CompilerServices;
2
3 #pragma warning disable CS1591 // Missing XML comment for publicly visible type or member
4
5 namespace Platform.Incrementers
6 {
7     /// <remarks>
8     /// Must be class, not struct (in order to persist access to Result field value).
9     /// </remarks>
10    public class Incrementer : IIncrementer
11    {
12        /// <summary>
13        /// <para>
14        /// The result.
15        /// </para>
16        /// <para></para>
17        /// </summary>
18        protected ulong _result;
19
20        /// <summary>
21        /// <para>
22        /// Gets the result value.
23        /// </para>
24        /// <para></para>
25        /// </summary>
26        public ulong Result
27        {
28            [MethodImpl(MethodImplOptions.AggressiveInlining)]
29            get => _result;
30        }
31    }
32 }
```

```

30     }
31
32     /// <summary>
33     /// <para>
34     /// Initializes a new <see cref="Incrementer"/> instance.
35     /// </para>
36     /// <para></para>
37     /// </summary>
38     /// <param name="initialValue">
39     /// <para>A initial value.</para>
40     /// <para></para>
41     /// </param>
42     [MethodImpl(MethodImplOptions.AggressiveInlining)]
43     public Incrementer(ulong initialValue) => _result = initialValue;
44
45     /// <summary>
46     /// <para>
47     /// Initializes a new <see cref="Incrementer"/> instance.
48     /// </para>
49     /// <para></para>
50     /// </summary>
51     [MethodImpl(MethodImplOptions.AggressiveInlining)]
52     public Incrementer() { }
53
54     /// <summary>
55     /// <para>
56     /// Increments this instance.
57     /// </para>
58     /// <para></para>
59     /// </summary>
60     [MethodImpl(MethodImplOptions.AggressiveInlining)]
61     public void Increment() => _result++;
62 }
63 }

```

1.4 ./csharp/Platform.Incrementers/Incrementer[TValue, TDecision].cs

```

1 using System.Runtime.CompilerServices;
2
3 #pragma warning disable IDE0060 // Remove unused parameter
4 #pragma warning disable CS1591 // Missing XML comment for publicly visible type or member
5
6 namespace Platform.Incrementers
7 {
8     /// <summary>
9     /// <para>
10    /// Represents the incrementer.
11    /// </para>
12    /// <para></para>
13    /// </summary>
14    /// <seealso cref="Incrementer"/>
15    public class Incrementer<TValue, TDecision> : Incrementer
16    {
17        /// <summary>
18        /// <para>
19        /// The true value.
20        /// </para>
21        /// <para></para>
22        /// </summary>
23        private readonly TDecision _trueValue;
24
25        /// <summary>
26        /// <para>
27        /// Initializes a new <see cref="Incrementer"/> instance.
28        /// </para>
29        /// <para></para>
30        /// </summary>
31        /// <param name="initialValue">
32        /// <para>A initial value.</para>
33        /// <para></para>
34        /// </param>
35        /// <param name="trueValue">
36        /// <para>A true value.</para>
37        /// <para></para>
38        /// </param>
39        [MethodImpl(MethodImplOptions.AggressiveInlining)]
40        public Incrementer(ulong initialValue, TDecision trueValue) : base(initialValue) =>
41            ↪ _trueValue = trueValue;
42
43        /// <summary>

```

```

43     /// <para>
44     /// Initializes a new <see cref="Incrementer"/> instance.
45     /// </para>
46     /// <para></para>
47     /// </summary>
48     /// <param name="trueValue">
49     /// <para>A true value.</para>
50     /// <para></para>
51     /// </param>
52     [MethodImpl(MethodImplOptions.AggressiveInlining)]
53     public Incrementer(TDecision trueValue) => _trueValue = trueValue;
54
55     /// <summary>
56     /// <para>
57     /// Initializes a new <see cref="Incrementer"/> instance.
58     /// </para>
59     /// <para></para>
60     /// </summary>
61     [MethodImpl(MethodImplOptions.AggressiveInlining)]
62     public Incrementer() { }
63
64     /// <summary>
65     /// <para>
66     /// Increments the and return true.
67     /// </para>
68     /// <para></para>
69     /// </summary>
70     /// <returns>
71     /// <para>The true value.</para>
72     /// <para></para>
73     /// </returns>
74     [MethodImpl(MethodImplOptions.AggressiveInlining)]
75     public TDecision IncrementAndReturnTrue()
76     {
77         _result++;
78         return _trueValue;
79     }
80
81     /// <summary>
82     /// <para>
83     /// Increments the and return true using the specified value.
84     /// </para>
85     /// <para></para>
86     /// </summary>
87     /// <param name="value">
88     /// <para>The value.</para>
89     /// <para></para>
90     /// </param>
91     /// <returns>
92     /// <para>The true value.</para>
93     /// <para></para>
94     /// </returns>
95     [MethodImpl(MethodImplOptions.AggressiveInlining)]
96     public TDecision IncrementAndReturnTrue(TValue value)
97     {
98         _result++;
99         return _trueValue;
100     }
101 }
102 }

```

1.5 ./csharp/Platform.Incrementers/Incrementer[TValue].cs

```

1 using System.Runtime.CompilerServices;
2
3 #pragma warning disable CS1591 // Missing XML comment for publicly visible type or member
4
5 namespace Platform.Incrementers
6 {
7     /// <summary>
8     /// <para>
9     /// Represents the incrementer.
10    /// </para>
11    /// <para></para>
12    /// </summary>
13    /// <seealso cref="Incrementer{TValue, bool}"/>
14    public class Incrementer<TValue> : Incrementer<TValue, bool>
15    {
16        /// <summary>
17        /// <para>

```

```

18     /// Initializes a new <see cref="Incrementer"/> instance.
19     /// </para>
20     /// <para></para>
21     /// </summary>
22     /// <param name="initialValue">
23     /// <para>A initial value.</para>
24     /// <para></para>
25     /// </param>
26     [MethodImpl(MethodImplOptions.AggressiveInlining)]
27     public Incrementer(ulong initialValue) : base(initialValue, true) { }
28
29     /// <summary>
30     /// <para>
31     /// Initializes a new <see cref="Incrementer"/> instance.
32     /// </para>
33     /// <para></para>
34     /// </summary>
35     [MethodImpl(MethodImplOptions.AggressiveInlining)]
36     public Incrementer() : base(true) { }
37 }
38 }

```

1.6 ./csharp/Platform.Incrementers.Tests/IncrementerTests.cs

```

1 using Xunit;
2
3 namespace Platform.Incrementers.Tests
4 {
5     /// <summary>
6     /// <para>
7     /// Represents the incrementer tests.
8     /// </para>
9     /// <para></para>
10    /// </summary>
11    public class IncrementerTests
12    {
13        /// <summary>
14        /// <para>
15        /// Tests that parameterless constructed setter test.
16        /// </para>
17        /// <para></para>
18        /// </summary>
19        [Fact]
20        public void ParameterlessConstructedSetterTest()
21        {
22            Incrementer<int> incrementer = new Incrementer<int>();
23            Assert.Equal(default, incrementer.Result);
24        }
25
26        /// <summary>
27        /// <para>
28        /// Tests that constructed with default value setter test.
29        /// </para>
30        /// <para></para>
31        /// </summary>
32        [Fact]
33        public void ConstructedWithDefaultValueSetterTest()
34        {
35            Incrementer<int> incrementer = new Incrementer<int>(9UL);
36            Assert.Equal(9UL, incrementer.Result);
37        }
38
39        /// <summary>
40        /// <para>
41        /// Tests that methods with boolean return type test.
42        /// </para>
43        /// <para></para>
44        /// </summary>
45        [Fact]
46        public void MethodsWithBooleanReturnTypeTest()
47        {
48            Incrementer<int> incrementer = new Incrementer<int>();
49            incrementer.Increment();
50            Assert.Equal(1UL, incrementer.Result);
51            Assert.True(incrementer.IncrementAndReturnTrue());
52            Assert.Equal(2UL, incrementer.Result);
53        }
54
55        /// <summary>

```

```
56     /// <para>
57     /// Tests that methods with integer return type test.
58     /// </para>
59     /// <para></para>
60     /// </summary>
61     [Fact]
62     public void MethodsWithIntegerReturnTypeTest()
63     {
64         Incrementer<int, int> incrementer = new Incrementer<int, int>(1);
65         incrementer.Increment();
66         Assert.Equal(1UL, incrementer.Result);
67         Assert.Equal(1, incrementer.IncrementAndReturnTrue());
68         Assert.Equal(2UL, incrementer.Result);
69     }
70 }
71 }
```

Index

./csharp/Platform.Incrementers.Tests/IncrementerTests.cs, 4
./csharp/Platform.Incrementers/IIncrementer.cs, 1
./csharp/Platform.Incrementers/IIncrementer[TNumber].cs, 1
./csharp/Platform.Incrementers/Incrementer.cs, 1
./csharp/Platform.Incrementers/Incrementer[TValue, TDecision].cs, 2
./csharp/Platform.Incrementers/Incrementer[TValue].cs, 3