

LinksPlatform's Platform.Random Class Library

./RandomExtensions.cs

```
1 using System.Runtime.CompilerServices;
2 using Platform.Ranges;
3
4 namespace Platform.Random
5 {
6     public static class RandomExtensions
7     {
8         /// <summary>
9         /// Returns a random 64-bit unsigned integer that is greater than or equal to
10         → ulong.MinValue, and less than or equal to ulong.MaxValue.
11         /// Возвращает случайное 64-разрядное целое число без знака, которое больше или равно
12         → ulong.MinValue и меньше или равно ulong.MaxValue.
13         /// </summary>
14         /// <param name="rnd">A pseudo-random number generator. Генератор псевдослучайных
15         → чисел.</param>
16         /// <returns>A 64-bit unsigned integer that is greater than or equal to ulong.MinValue,
17         → and less than or equal to ulong.MaxValue. 64-разрядное целое число без знака,
18         → которое больше или равно ulong.MinValue и меньше или равно ulong.MaxValue.</returns>
19         [MethodImpl(MethodImplOptions.AggressiveInlining)]
20         public static ulong NextUInt64(this System.Random rnd) => rnd.NextUInt64(new
21         → Range<ulong>(ulong.MinValue, ulong.MaxValue));
22
23         /// <summary>
24         /// Returns a random 64-bit unsigned integer that is greater than or equal to minimum of
25         → specified range, and less than or equal to maximum of specified range.
26         /// Возвращает случайное 64-разрядное целое число без знака, которое больше или равно
27         → минимуму указанного диапазона и меньше или равно максимуму указанного диапазона.
28         /// </summary>
29         /// <param name="rnd">A pseudo-random number generator. Генератор псевдослучайных
30         → чисел.</param>
31         /// <param name="range">The range of possible values. Диапазон возможных
32         → значений.</param>
33         /// <returns>A 64-bit unsigned integer that is greater than or equal to the minimum of
34         → specified range, and less than or equal to the maximum of the specified range.
35         → 64-разрядное целое число без знака, которое больше или равно минимуму указанного
36         → диапазона и меньше или равно максимуму указанного диапазона.</returns>
37         [MethodImpl(MethodImplOptions.AggressiveInlining)]
38         public static ulong NextUInt64(this System.Random rnd, Range<ulong> range) =>
39         → (ulong)(rnd.NextDouble() * (range.Maximum - range.Minimum)) + range.Minimum;
40
41         /// <summary>
42         /// Return a random boolean value.
43         /// Возвращает случайное булево значение.
44         /// </summary>
45         /// <param name="rnd">A pseudo-random number generator. Генератор псевдослучайных
46         → чисел.</param>
47         /// <returns>A random boolean value. Случайное булево значение.</returns>
48         [MethodImpl(MethodImplOptions.AggressiveInlining)]
49         public static bool NextBoolean(this System.Random rnd) => rnd.Next(2) == 1;
50     }
51 }
```

./RandomHelpers.cs

```
1 namespace Platform.Random
2 {
3     public static class RandomHelpers
4     {
5         /// <summary>
6         /// Returns the pseudorandom number generator that is using the time of the first access
7         → to this field as seed.
8         /// Возвращает генератор псевдослучайных чисел использующий в качестве seed время
9         → первого обращения к этому полю.
10         /// </summary>
11         public static readonly System.Random Default = new
12         → System.Random(System.DateTime.UtcNow.Ticks.GetHashCode());
13     }
14 }
```

Index

- ./RandomExtensions.cs, 1
- ./RandomHelpers.cs, 1