

# Namespace SnowflakeID

## Classes

### [Snowflake](#)

This class represents the Snowflake object. [Wikipedia article about SnowflakeID](#) 

### [SnowflakeIDGenerator](#)

Generator class for [Snowflake](#).

This keeps track of time, machine number and sequence.

### [SnowflakeIDGeneratorOptions](#)

Option object for [SnowflakeIDGenerator](#).

### [SnowflakeIDGeneratorServiceCollectionExtensions](#)

Extension methods to register [SnowflakeIDGenerator](#) services.

## Interfaces

### [ISnowflakeIDGenerator](#)

Interface for the generator class for [Snowflake](#).

This keeps track of time, machine number, and sequence.

### [ISnowflakeIDGeneratorClsCompliant](#)

Interface for the generator class for [Snowflake](#).

This keeps track of time, machine number and sequence.

# Interface ISnowflakeIDGenerator

Namespace: [SnowflakeID](#)

Assembly: SnowflakeIDGenerator.dll

Interface for the generator class for [Snowflake](#).

This keeps track of time, machine number, and sequence.

```
[CLSCompliant(false)]  
public interface ISnowflakeIDGenerator
```

## Remarks

[NuGet](#) 

[Source](#) 

[API](#) 

[Site](#) 

## Properties


### ConfiguredEpoch

Gets the date configured as the epoch for the generator.

```
DateTime ConfiguredEpoch { get; }
```

#### Property Value

[DateTime](#) 

The [DateTime](#)  value representing the custom epoch date.

#### Remarks

The epoch date is used as the starting point for generating unique IDs.

# ConfiguredMachineId

Gets the configured machine ID for the generator.

```
int ConfiguredMachineId { get; }
```

## Property Value

[int](#)

The [int](#) value representing the machine ID.

## Remarks

The machine ID is used to ensure uniqueness across different instances of the generator.

# Methods

## GetCode()

Gets the next Snowflake ID as a number.

```
ulong GetCode()
```

## Returns

[ulong](#)

A [ulong](#) representing the next Snowflake ID.

## Remarks

This method generates a new Snowflake ID and returns it as a numeric value.

## GetCodeString()

Gets the next Snowflake ID as a string.

```
string GetCodeString()
```

## Returns

[string](#)

A [string](#) representing the next Snowflake ID.

## Remarks

This method generates a new Snowflake ID and returns it as a string value.

## GetSnowflake()

Generates the next Snowflake ID.

```
Snowflake GetSnowflake()
```

## Returns

[Snowflake](#)

A [Snowflake](#) object containing the generated ID.

## Remarks

This method generates a new Snowflake ID and returns it as a [Snowflake](#) object.

# Interface ISnowflakeIDGeneratorClsCompliant

Namespace: [SnowflakeID](#)

Assembly: SnowflakeIDGenerator.dll

Interface for the generator class for [Snowflake](#).

This keeps track of time, machine number and sequence.

```
public interface ISnowflakeIDGeneratorClsCompliant
```

## Remarks

[NuGet](#) 

[Source](#) 

[API](#) 

[Site](#) 

## Properties


### ConfiguredEpoch

Gets the date configured as the epoch for the generator.

```
DateTime ConfiguredEpoch { get; }
```

### Property Value

[DateTime](#) 

The [DateTime](#)  value representing the custom epoch date.

### Remarks

The epoch date is used as the starting point for generating unique IDs.

# ConfiguredMachineId

Gets the configured machine ID for the generator.

```
int ConfiguredMachineId { get; }
```

## Property Value

[int](#)

The [int](#) value representing the machine ID.

## Remarks

The machine ID is used to ensure uniqueness across different instances of the generator.

# Methods

## GetCodeString()

Gets the next Snowflake ID as a string.

```
string GetCodeString()
```

## Returns

[string](#)

A [string](#) representing the next Snowflake ID.

## Remarks

This method generates a new Snowflake ID and returns it as a string value.

## GetSnowflake()

Generates the next Snowflake ID.

```
Snowflake GetSnowflake()
```

## Returns

### [Snowflake](#)

A [Snowflake](#) object containing the generated ID.

## Remarks

This method generates a new Snowflake ID and returns it as a [Snowflake](#) object.

# Class Snowflake

Namespace: [SnowflakeID](#)

Assembly: SnowflakeIDGenerator.dll

This class represents the Snowflake object. [Wikipedia article about SnowflakeID](#)

```
public class Snowflake : IEquatable<Snowflake>, IComparable<Snowflake>, IComparable
```

## Inheritance

[object](#) ← Snowflake

## Implements

[IEquatable](#) <[Snowflake](#)>, [IComparable](#) <[Snowflake](#)>, [IComparable](#)

## Inherited Members

[object.Equals\(object, object\)](#), [object.ReferenceEquals\(object, object\)](#), [object.GetType\(\)](#), [object.MemberwiseClone\(\)](#)

## Remarks

[NuGet](#)

[Source](#)

[API](#)

[Site](#)

## Constructors

### Snowflake()

Initializes a new instance of the [Snowflake](#) class using the default epoch (UNIX time 1-1-1970).

```
public Snowflake()
```

### Snowflake(DateTime)



Initializes a new instance of the [Snowflake](#) class using a custom date as epoch.

```
public Snowflake(DateTime epoch)
```

## Parameters

epoch [DateTime](#)

The date to use as the epoch.

## Fields

### MaxMachineId

Max number of machines / servers allowed. Range from 0 to MaxMachineId-1

```
public const long MaxMachineId = 1024
```

## Field Value

[long](#)

### MaxSequence

The maximum sequence number that can be generated per millisecond. When this value is reached, the sequence is reset to 0.

```
public const long MaxSequence = 4096
```

## Field Value

[long](#)

### MaxTimestamp

Max number of milliseconds since epoch. Range from 0 to MaxTimestamp-1

```
public const long MaxTimestamp = 4398046511104
```

Field Value

[long](#)

## NumberOfDigits

Total number of digits for the generated code.

```
public static readonly int NumberOfDigits
```

Field Value

[int](#)

An [int](#) representing the number of digits.

## Properties

### Code

Gets the Snowflake ID as a string.

```
public string Code { get; }
```

Property Value

[string](#)

A [string](#) representing the Snowflake ID.

### Epoch

Gets the current epoch being used.

```
public DateTime Epoch { get; }
```

## Property Value

[DateTime](#) 

A [DateTime](#)  representing the current epoch.

## Id

Gets the Snowflake ID.

```
[CLSCompliant(false)]  
public virtual ulong Id { get; }
```

## Property Value

[ulong](#) 

A [ulong](#)  representing the Snowflake ID.

## MachineId

Gets or sets the machine/server number.

```
[CLSCompliant(false)]  
public ulong MachineId { get; set; }
```

## Property Value

[ulong](#) 

## Exceptions

[ArgumentOutOfRangeException](#) 

Thrown when the value is greater than or equal to [MaxMachineId](#).

# MachineIdInt32

Gets or sets the machine/server number as an integer.

```
public int MachineIdInt32 { get; set; }
```

## Property Value

[int](#)

## Exceptions

[ArgumentOutOfRangeException](#)

Thrown when the value is negative or greater than or equal to [MaxMachineId](#).

# Sequence

Gets or sets the sequence number.

```
[CLSCompliant(false)]  
public ulong Sequence { get; set; }
```

## Property Value

[ulong](#)

## Exceptions

[ArgumentOutOfRangeException](#)

Thrown when the value is greater than or equal to [MaxSequence](#).

# SequenceInt32

Gets or sets the sequence number as an integer.

```
public int SequenceInt32 { get; set; }
```

Property Value

[int](#)

Exceptions

[ArgumentOutOfRangeException](#)

Thrown when the value is negative or greater than or equal to [MaxSequence](#).

## Timestamp

Gets or sets the timestamp as the number of milliseconds since the selected epoch.

```
[CLSCompliant(false)]  
public ulong Timestamp { get; set; }
```

Property Value

[ulong](#)

Exceptions

[ArgumentOutOfRangeException](#)

Thrown when the value is greater than or equal to [MaxTimestamp](#).

## TimestampInt64

Gets or sets the timestamp as the number of milliseconds since the selected epoch.

```
public long TimestampInt64 { get; set; }
```

Property Value

[long](#)

Exceptions

Thrown when the value is less than 0 or greater than or equal to [MaxTimestamp](#).

## UtcDateTime

Sets the timeStamp portion of the snowflake based on current time and selected epoch. Gets real time of the snowflake based on selected epoch.

```
public DateTime UtcDateTime { get; set; }
```

Property Value

[DateTime](#) ↗

## Methods

### ChangeEpoch(DateTime)

Changes the snowflake's epoch keeping the code intact. This will adjust the represented [UtcDateTime](#) to match the new epoch.

```
public void ChangeEpoch(DateTime newEpoch)
```

Parameters

**newEpoch** [DateTime](#) ↗

The new epoch to set.

### CompareTo(Snowflake)

Compares the current Snowflake object with another Snowflake object.

```
public int CompareTo(Snowflake other)
```

## Parameters

**other** [Snowflake](#)

The Snowflake object to compare with the current Snowflake object.

## Returns

[int](#)

A value that indicates the relative order of the objects being compared. The return value has these meanings:

- Less than zero: This object is less than the **other** parameter.
- Zero: This object is equal to **other**.
- Greater than zero: This object is greater than **other**.

## Exceptions

[SnowflakesUsingDifferentEpochsException](#)

Thrown when comparing Snowflake objects generated using different epochs.

## CompareTo(object)

Compares the current Snowflake object with another object.

```
public int CompareTo(object obj)
```

## Parameters

**obj** [object](#)

The object to compare with the current Snowflake object.

## Returns

[int](#)

A value that indicates the relative order of the objects being compared. The return value has these meanings:

- Less than zero: This object is less than the `obj` parameter.
- Zero: This object is equal to `obj`.
- Greater than zero: This object is greater than `obj`.

## Exceptions

### [SnowflakesUsingDifferentEpochsException](#)

Thrown when comparing Snowflake objects generated using different epochs.

## Equals(Snowflake)

Checks equality between this Snowflake object and another Snowflake object.

```
public virtual bool Equals(Snowflake other)
```

## Parameters

`other` [Snowflake](#)

The Snowflake object to compare with the current Snowflake object.

## Returns

[bool](#)

`true` if the specified Snowflake object is equal to the current Snowflake object; otherwise, `false`.

## Equals(object)

Checks equality between this Snowflake object and another object.

```
public override bool Equals(object obj)
```

## Parameters

`obj` [object](#)

The object to compare with the current Snowflake object.



Returns

[bool](#)

`true` if the specified object is equal to the current Snowflake object; otherwise, `false`.

## FromString(string)

Creates a [Snowflake](#) instance from the specified string.

```
public static Snowflake FromString(string s)
```

Parameters

`s` [string](#)

The string representation of the [Snowflake](#).

Returns

[Snowflake](#)

A [Snowflake](#) instance that corresponds to the specified string.

## FromUInt64(ulong)

Creates a [Snowflake](#) instance from the specified unsigned long integer.

```
[CLSCompliant(false)]  
public static Snowflake FromUInt64(ulong s)
```

Parameters

`s` [ulong](#)

The unsigned long integer representation of the [Snowflake](#).

Returns

## [Snowflake](#)

A [Snowflake](#) instance that corresponds to the specified unsigned long integer.

## GetHashCode()

Serves as the default hash function. Override of [GetHashCode\(\)](#).

```
public override int GetHashCode()
```

Returns

[int](#)

A hash code for the current Snowflake object.

## Parse(string)

Creates a SnowflakeId object from a SnowflakeId code.

```
public static Snowflake Parse(string s)
```

Parameters

s [string](#)

The SnowflakeId code as a string.

Returns

[Snowflake](#)

A new instance of the [Snowflake](#) class.

## Parse(string, DateTime)

Creates a SnowflakeId object from a SnowflakeId code using a custom epoch.

```
public static Snowflake Parse(string s, DateTime customEpoch)
```

## Parameters

**s** [string](#)

The SnowflakeId code as a string.

**customEpoch** [DateTime](#)

The custom date to use as the epoch.

## Returns

[Snowflake](#)

A new instance of the [Snowflake](#) class.

## Parse(ulong)

Creates a SnowflakeId object from a SnowflakeId code.

```
[CLSCompliant(false)]  
public static Snowflake Parse(ulong b)
```

## Parameters

**b** [ulong](#)

The SnowflakeId code as a ulong.

## Returns

[Snowflake](#)

A new instance of the [Snowflake](#) class.

## Parse(ulong, DateTime)

Creates a SnowflakeId object from a SnowflakeId code using a custom epoch.

```
[CLSCompliant(false)]  
public static Snowflake Parse(ulong b, DateTime customEpoch)
```

## Parameters

**b** [ulong](#)

The SnowflakeId code as a ulong.

**customEpoch** [DateTime](#)

The custom date to use as the epoch.

## Returns

[Snowflake](#)

A new instance of the [Snowflake](#) class.

## RebaseEpoch(DateTime)

Rebase the Snowflake to a new epoch CHANGING THE GENERATED CODE but keeping the same date and time.

```
public void RebaseEpoch(DateTime newEpoch)
```

## Parameters

**newEpoch** [DateTime](#)

The new epoch to set.

## ToString()

Gets the Snowflake ID as a string.

```
public override string ToString()
```

Returns

[string](#)

A [string](#) representing the Snowflake ID.

## ToUInt64()

Converts the current [Snowflake](#) instance to its unsigned long integer representation.

```
[CLSCompliant(false)]  
public ulong ToUInt64()
```

Returns

[ulong](#)

An unsigned long integer representation of the current [Snowflake](#) instance.

Remarks

If a loosely typed language (or a language that doesn't differentiate between number types, i.e.: Typescript) is part of your workflow, use [ToString\(\)](#) to avoid issues regarding floating-point underflow and rounding.

## Operators

### operator ==(Snowflake, Snowflake)

Determines whether two specified instances of [Snowflake](#) are equal.

```
public static bool operator ==(Snowflake s1, Snowflake s2)
```

Parameters

**s1** [Snowflake](#)

The first [Snowflake](#) to compare.

**s2** [Snowflake](#)

The second [Snowflake](#) to compare.

Returns

[bool](#)

`true` if the two [Snowflake](#) instances are equal; otherwise, `false`.

## explicit operator Snowflake(string)

Converts the specified string to a [Snowflake](#) instance.

```
public static explicit operator Snowflake(string s)
```

Parameters

`s` [string](#)

The string to convert.

Returns

[Snowflake](#)

A [Snowflake](#) instance that is equivalent to the specified string.

## explicit operator Snowflake(ulong)

Converts the specified unsigned long integer to a [Snowflake](#) instance.

```
[CLSCompliant(false)]  
public static explicit operator Snowflake(ulong s)
```

Parameters

`s` [ulong](#)

The unsigned long integer to convert.

## Returns

### [Snowflake](#)

A [Snowflake](#) instance that is equivalent to the specified unsigned long integer.

## operator >(Snowflake, Snowflake)

Determines whether the first specified [Snowflake](#) is greater than the second specified [Snowflake](#).

```
public static bool operator >(Snowflake s1, Snowflake s2)
```

## Parameters

### s1 [Snowflake](#)

The first [Snowflake](#) to compare.

### s2 [Snowflake](#)

The second [Snowflake](#) to compare.

## Returns

### [bool](#)

**true** if the first [Snowflake](#) is greater than the second [Snowflake](#); otherwise, **false**.

## Exceptions

### [SnowflakesUsingDifferentEpochsException](#)

Thrown when comparing [Snowflake](#) objects generated using different epochs.

## operator >=(Snowflake, Snowflake)

Determines whether the first specified [Snowflake](#) is greater than or equal to the second specified [Snowflake](#).

```
public static bool operator >=(Snowflake s1, Snowflake s2)
```

## Parameters

**s1** [Snowflake](#)

The first [Snowflake](#) to compare.

**s2** [Snowflake](#)

The second [Snowflake](#) to compare.

## Returns

[bool](#)

**true** if the first [Snowflake](#) is greater than or equal to the second [Snowflake](#); otherwise, **false**.

## Exceptions

[SnowflakesUsingDifferentEpochsException](#)

Thrown when comparing [Snowflake](#) objects generated using different epochs.

## implicit operator string(Snowflake)

Converts the specified [Snowflake](#) instance to its string representation.

```
public static implicit operator string(Snowflake s)
```

## Parameters

**s** [Snowflake](#)

The [Snowflake](#) instance to convert.

## Returns

[string](#)

A string representation of the specified [Snowflake](#) instance.



# implicit operator ulong(Snowflake)

Converts the specified [Snowflake](#) instance to its unsigned long integer representation.

```
[CLSCompliant(false)]  
public static implicit operator ulong(Snowflake s)
```

## Parameters

**s** [Snowflake](#)

The [Snowflake](#) instance to convert.

## Returns

[ulong](#) 

An unsigned long integer representation of the specified [Snowflake](#) instance.

## Remarks

If a loosely typed language (or a language that doesn't differentiate between number types, i.e.: Typescript) is part of your workflow, use the string representation to avoid issues regarding floating-point underflow and rounding.

# operator !=(Snowflake, Snowflake)

Determines whether two specified instances of [Snowflake](#) are not equal.

```
public static bool operator !=(Snowflake s1, Snowflake s2)
```

## Parameters

**s1** [Snowflake](#)

The first [Snowflake](#) to compare.

**s2** [Snowflake](#)

The second [Snowflake](#) to compare.

## Returns

[bool](#)

`true` if the two [Snowflake](#) instances are not equal; otherwise, `false`.

## operator <(Snowflake, Snowflake)

Determines whether the first specified [Snowflake](#) is less than the second specified [Snowflake](#).

```
public static bool operator <(Snowflake s1, Snowflake s2)
```

## Parameters

`s1` [Snowflake](#)

The first [Snowflake](#) to compare.

`s2` [Snowflake](#)

The second [Snowflake](#) to compare.

## Returns

[bool](#)

`true` if the first [Snowflake](#) is less than the second [Snowflake](#); otherwise, `false`.

## Exceptions

[SnowflakesUsingDifferentEpochsException](#)

Thrown when comparing [Snowflake](#) objects generated using different epochs.

## operator <=(Snowflake, Snowflake)

Determines whether the first specified [Snowflake](#) is less than or equal to the second specified [Snowflake](#).

```
public static bool operator <=(Snowflake s1, Snowflake s2)
```

## Parameters

**s1** [Snowflake](#)

The first [Snowflake](#) to compare.

**s2** [Snowflake](#)

The second [Snowflake](#) to compare.

## Returns

[bool](#)

**true** if the first [Snowflake](#) is less than or equal to the second [Snowflake](#); otherwise, **false**.

## Exceptions

[SnowflakesUsingDifferentEpochsException](#)

Thrown when comparing [Snowflake](#) objects generated using different epochs.

# Class SnowflakeIDGenerator

Namespace: [SnowflakeID](#)

Assembly: SnowflakeIDGenerator.dll

Generator class for [Snowflake](#).

This keeps track of time, machine number and sequence.

```
public class SnowflakeIDGenerator : ISnowflakeIDGenerator, ISnowflakeIDGeneratorClsCompliant
```

## Inheritance

[object](#) ← SnowflakeIDGenerator

## Implements

[ISnowflakeIDGenerator](#), [ISnowflakeIDGeneratorClsCompliant](#)

## Inherited Members

[object.ToString\(\)](#), [object.Equals\(object\)](#), [object.Equals\(object, object\)](#),  
[object.ReferenceEquals\(object, object\)](#), [object.GetHashCode\(\)](#), [object.GetType\(\)](#),  
[object.MemberwiseClone\(\)](#)

## Remarks

[NuGet](#)

[Source](#)

[API](#)

[Site](#)

## Constructors

### SnowflakeIDGenerator(int)

Creates a SnowflakeIDGenerator for a given machine number.

```
public SnowflakeIDGenerator(int machineId)
```

## Parameters

`machineId` [int](#)

Machine number

## Remarks

This constructor initializes the generator with a specific machine ID and uses the default epoch date. It ensures that the machine ID is within the valid range and sets the initial timestamp to prevent overflow issues.

## Exceptions

[ArgumentOutOfRangeException](#)

Thrown when `machineId` is greater than or equal to [MaxMachineId](#).

## SnowflakeIDGenerator(int, DateTime)

Creates a SnowflakeIDGenerator for a given machine number using a custom date as epoch.

```
public SnowflakeIDGenerator(int machineId, DateTime customEpoch)
```

## Parameters

`machineId` [int](#)

Machine number

`customEpoch` [DateTime](#)

Date to use as epoch

## Remarks

This constructor initializes the generator with a specific machine ID and a custom epoch date. It ensures that the machine ID is within the valid range and sets the initial timestamp to prevent overflow issues.

## Exceptions

[ArgumentOutOfRangeException](#)

Thrown when `machineId` is greater than or equal to [MaxMachineId](#).

## SnowflakeIDGenerator(ulong)

Creates a SnowflakeIDGenerator for a given machine number.

```
[CLSCompliant(false)]  
public SnowflakeIDGenerator(ulong machineId)
```

### Parameters

`machineId` [ulong](#) 

Machine number

### Remarks

This constructor initializes the generator with a specific machine ID and uses the default epoch date. It ensures that the machine ID is within the valid range and sets the initial timestamp to prevent overflow issues.

### Exceptions

[ArgumentOutOfRangeException](#) 


Thrown when `machineId` is greater than or equal to [MaxMachineId](#).

## SnowflakeIDGenerator(ulong, DateTime)

Creates a SnowflakeIDGenerator for a given machine number using a custom date as epoch.

```
[CLSCompliant(false)]  
public SnowflakeIDGenerator(ulong machineId, DateTime customEpoch)
```

### Parameters

`machineId` [ulong](#) 

Machine number

`customEpoch` [DateTime](#)

Date to use as epoch

## Remarks

This constructor initializes the generator with a specific machine ID and a custom epoch date. It ensures that the machine ID is within the valid range and sets the initial timestamp to prevent overflow issues.

## Exceptions

[ArgumentOutOfRangeException](#)

Thrown when `machineId` is greater than or equal to [MaxMachineId](#).

# Properties

## ConfiguredEpoch

Gets the date configured as the epoch for the generator.

```
public DateTime ConfiguredEpoch { get; }
```

## Property Value

[DateTime](#)

The [DateTime](#) value representing the custom epoch date.

## Remarks

The epoch date is used as the starting point for generating unique IDs.

## ConfiguredMachineId

Gets the configured machine ID for the generator.

```
public int ConfiguredMachineId { get; }
```

## Property Value

[int](#)

The [int](#) value representing the machine ID.

## Remarks

The machine ID is used to ensure uniqueness across different instances of the generator.

# Methods

## GetCode()

Gets the next Snowflake ID as a number.

```
[CLSCompliant(false)]  
public ulong GetCode()
```

## Returns

[ulong](#)

A [ulong](#) representing the next Snowflake ID.

## Remarks

This method generates a new Snowflake ID and returns it as a numeric value.

If a loosely typed language (or a language that doesn't differentiate between number types, i.e.: Typescript) is part of your workflow, use [GetCodeString\(\)](#) to avoid issues regarding floating-point underflow and rounding.

## GetCode(ulong)

Static method to get the next Snowflake ID as a number for a given machine ID.

```
[CLSCompliant(false)]  
public static ulong GetCode(ulong machineId)
```



## Parameters

**machineId** [ulong](#)

The machine ID as a [ulong](#).

## Returns

[ulong](#)

A [ulong](#) representing the next Snowflake ID.

## Remarks

This method generates a new Snowflake ID and returns it as a numeric value.

If a loosely typed language (or a language that doesn't differentiate between number types, i.e.: Typescript) is part of your workflow, use [GetCodeString\(ulong\)](#) to avoid issues regarding floating-point underflow and rounding.

## GetCode(ulong, DateTime)

Static method to get the next Snowflake ID as a number for a given machine ID using a custom epoch date.

```
[CLSCompliant(false)]  
public static ulong GetCode(ulong machineId, DateTime customEpoch)
```

## Parameters

**machineId** [ulong](#)

The machine ID as a [ulong](#).

**customEpoch** [DateTime](#)

The custom epoch date as a [DateTime](#).

## Returns

[ulong](#)

A [ulong](#) representing the next Snowflake ID.

## Remarks

This method generates a new Snowflake ID and returns it as a numeric value.

If a loosely typed language (or a language that doesn't differentiate between number types, i.e.: Typescript) is part of your workflow, use [GetCodeString\(ulong, DateTime\)](#) to avoid issues regarding floating-point underflow and rounding.

## GetCodeString()

Gets the next Snowflake ID as a string.

```
public string GetCodeString()
```

## Returns

[string](#)

A [string](#) representing the next Snowflake ID.

## Remarks

This method generates a new Snowflake ID and returns it as a string value.

## GetCodeString(int)

Gets the next Snowflake ID as a string for a given machine ID.

```
public static string GetCodeString(int machineId)
```

## Parameters

**machineId** [int](#)

The machine ID as an [int](#).

## Returns

[string](#)

A [string](#) representing the next Snowflake ID.

## GetCodeString(int, DateTime)

Gets the next Snowflake ID as a string for a given machine ID using a custom epoch date.

```
public static string GetCodeString(int machineId, DateTime customEpoch)
```

### Parameters

**machineId** [int](#)

The machine ID as an [int](#).

**customEpoch** [DateTime](#)

The custom epoch date as a [DateTime](#).

### Returns

[string](#)

A [string](#) representing the next Snowflake ID.

## GetCodeString(ulong)

Gets the next Snowflake ID as a string for a given machine ID.

```
[CLSCompliant(false)]  
public static string GetCodeString(ulong machineId)
```

### Parameters

**machineId** [ulong](#)

The machine ID as a [ulong](#).

### Returns

[string](#)

A [string](#) representing the next Snowflake ID.

## GetCodeString(ulong, DateTime)

Gets the next Snowflake ID as a string for a given machine ID using a custom epoch date.

```
[CLSCompliant(false)]  
public static string GetCodeString(ulong machineId, DateTime customEpoch)
```

### Parameters

**machineId** [ulong](#)

The machine ID as a [ulong](#).

**customEpoch** [DateTime](#)

The custom epoch date as a [DateTime](#).

### Returns

[string](#)

A [string](#) representing the next Snowflake ID.

## GetSnowflake()

Generates the next Snowflake ID.

```
public Snowflake GetSnowflake()
```

### Returns

[Snowflake](#)

A [Snowflake](#) object containing the generated ID.

### Exceptions

## [InvalidOperationException](#)

Thrown when the system clock is moved backwards.

## GetSnowflake(int)

Static method to get the next Snowflake ID for a given machine ID.

```
public static Snowflake GetSnowflake(int machineId)
```

### Parameters

**machineId** [int](#)

The machine ID as an [int](#).

### Returns

[Snowflake](#)

A [Snowflake](#) object containing the generated ID.

## GetSnowflake(int, DateTime)

Static method to get the next Snowflake ID for a given machine ID using a custom epoch date.

```
public static Snowflake GetSnowflake(int machineId, DateTime customEpoch)
```

### Parameters

**machineId** [int](#)

The machine ID as an [int](#).

**customEpoch** [DateTime](#)

The custom epoch date as a [DateTime](#).

### Returns

## [Snowflake](#)

A [Snowflake](#) object containing the generated ID.

## GetSnowflake(ulong)

Static method to get the next Snowflake ID for a given machine ID.

```
[CLSCompliant(false)]  
public static Snowflake GetSnowflake(ulong machineId)
```

### Parameters

**machineId** [ulong](#)

The machine ID as a [ulong](#).

### Returns

## [Snowflake](#)

A [Snowflake](#) object containing the generated ID.

### Remarks

This method generates a new Snowflake ID and returns it as a [Snowflake](#) object.

## GetSnowflake(ulong, DateTime)

Static method to get the next Snowflake ID for a given machine ID using a custom epoch date.

```
[CLSCompliant(false)]  
public static Snowflake GetSnowflake(ulong machineId, DateTime customEpoch)
```

### Parameters

**machineId** [ulong](#)

The machine ID as a [ulong](#).

customEpoch [DateTime](#)↗

The custom epoch date as a [DateTime](#)↗.

Returns

[Snowflake](#)

A [Snowflake](#) object containing the generated ID.

# Class SnowflakeIdGeneratorOptions

Namespace: [SnowflakeID](#)

Assembly: SnowflakeIDGenerator.DependencyInjection.dll

Option object for [SnowflakeIDGenerator](#).

```
public class SnowflakeIdGeneratorOptions
```

## Inheritance

[object](#)  ← SnowflakeIdGeneratorOptions

## Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,  
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#) 

## Properties

### Epoch

Gets or sets the epoch date as a string.


```
public string Epoch { get; set; }
```

### Property Value

[string](#) 

The epoch date as a string in ISO 8601 format.

### Remarks

This property allows setting the epoch date using a string representation. The date is parsed and stored internally as a [DateTime](#)  object in UTC.

### MachineId

Gets or sets the machine ID.



```
public int MachineId { get; set; }
```

## Property Value

[int](#)

The machine ID as an [int](#).

# Class

## SnowflakeIdGeneratorServiceCollectionExtensions


Namespace: [SnowflakeId](#)

Assembly: SnowflakeIdGenerator.DependencyInjection.dll








Extension methods to register [SnowflakeIdGenerator](#) services.

```
public static class SnowflakeIdGeneratorServiceCollectionExtensions
```

### Inheritance

[object](#)  ← SnowflakeIdGeneratorServiceCollectionExtensions

### Inherited Members

[object.Equals\(object\)](#)  , [object.Equals\(object, object\)](#)  , [object.GetHashCode\(\)](#)  , [object.GetType\(\)](#)  , [object.MemberwiseClone\(\)](#)  , [object.ReferenceEquals\(object, object\)](#)  , [object.ToString\(\)](#) 

## Remarks

[NuGet](#) 

[Source](#) 

[API](#) 

[Site](#) 

## Methods

### AddSnowflakeIdGeneratorService(IServiceCollection)

Registers a [SnowflakeIdGenerator](#) in [ISnowflakeIdGenerator](#).

```
public static IServiceCollection AddSnowflakeIdGeneratorService(this  
IServiceCollection serviceCollection)
```

### Parameters

**serviceCollection** [IServiceCollection](#)

The [IServiceCollection](#) to add services to.

Returns

[IServiceCollection](#)

The updated [IServiceCollection](#).

## AddSnowflakeIdGeneratorService(IServiceCollection, SnowflakeIdGeneratorOptions)

Registers a [SnowflakeIdGenerator](#) in [ISnowflakeIdGenerator](#).

```
public static IServiceCollection AddSnowflakeIdGeneratorService(this IServiceCollection
serviceCollection, SnowflakeIdGeneratorOptions options)
```

Parameters

**serviceCollection** [IServiceCollection](#)

The [IServiceCollection](#) to add services to.

**options** [SnowflakeIdGeneratorOptions](#)

Option object. Useful when obtaining from IConfigurationSection.

Returns

[IServiceCollection](#)

The updated [IServiceCollection](#).

## AddSnowflakeIdGeneratorService(IServiceCollection, int)

Registers a [SnowflakeIdGenerator](#) in [ISnowflakeIdGenerator](#).

```
public static IServiceCollection AddSnowflakeIdGeneratorService(this IServiceCollection
serviceCollection, int machineId)
```

## Parameters

**serviceCollection** [IServiceCollection](#)

The [IServiceCollection](#) to add services to.

**machineId** [int](#)

Machine number.

## Returns

[IServiceCollection](#)

The updated [IServiceCollection](#).

# AddSnowflakeIdGeneratorService(IServiceCollection, int, DateTime)

Registers a [SnowflakeIdGenerator](#) in [ISnowflakeIdGenerator](#).

```
public static IServiceCollection AddSnowflakeIdGeneratorService(this IServiceCollection  
serviceCollection, int machineId, DateTime customEpoch)
```

## Parameters

**serviceCollection** [IServiceCollection](#)

The [IServiceCollection](#) to add services to.

**machineId** [int](#)

Machine number.

**customEpoch** [DateTime](#)

Date to use as epoch.

## Returns

[IServiceCollection](#)

The updated [IServiceCollection](#).

## AddSnowflakeIdGeneratorService(IServiceCollection, long, DateTime)

Registers a [SnowflakeIdGenerator](#) in [ISnowflakeIdGenerator](#).

```
public static IServiceCollection AddSnowflakeIdGeneratorService(this IServiceCollection  
serviceCollection, long machineId, DateTime customEpoch)
```

### Parameters

**serviceCollection** [IServiceCollection](#)

The [IServiceCollection](#) to add services to.

**machineId** [long](#)

Machine number.

**customEpoch** [DateTime](#)

Date to use as epoch.

### Returns

[IServiceCollection](#)

The updated [IServiceCollection](#).

## AddSnowflakeIdGeneratorService(IServiceCollection, ulong)

Registers a [SnowflakeIdGenerator](#) in [ISnowflakeIdGenerator](#).

```
public static IServiceCollection AddSnowflakeIdGeneratorService(this IServiceCollection  
serviceCollection, ulong machineId)
```

### Parameters

**serviceCollection** [IServiceCollection](#)

The [IServiceCollection](#) to add services to.

`machineId` [ulong](#)

Machine number.

Returns

[IServiceCollection](#)

The updated [IServiceCollection](#).

## AddSnowflakeIdGeneratorService(IServiceCollection, ulong, DateTime)

Registers a [SnowflakeIdGenerator](#) in [ISnowflakeIdGenerator](#).

```
public static IServiceCollection AddSnowflakeIdGeneratorService(this IServiceCollection  
serviceCollection, ulong machineId, DateTime customEpoch)
```

Parameters

`serviceCollection` [IServiceCollection](#)

The [IServiceCollection](#) to add services to.

`machineId` [ulong](#)

Machine number.

`customEpoch` [DateTime](#)

Date to use as epoch.

Returns

[IServiceCollection](#)

The updated [IServiceCollection](#).

# Namespace SnowflakeID.Exceptions

## Classes

### [SnowflakesUsingDifferentEpochsException](#)

Exception thrown when trying to compare IDs using different epochs.

# Class

## SnowflakesUsingDifferentEpochsException

Namespace: [SnowflakeID.Exceptions](#)

Assembly: SnowflakeIDGenerator.dll

Exception thrown when trying to compare IDs using different epochs.

```
[Serializable]  
public class SnowflakesUsingDifferentEpochsException : ArgumentException,  
    _Exception, ISerializable
```

### Inheritance



















[object](#)  ← [Exception](#)  ← [SystemException](#)  ← [ArgumentException](#)  ←

SnowflakesUsingDifferentEpochsException

### Implements

[\\_Exception](#) , [ISerializable](#) 

### Inherited Members

[ArgumentException.GetObjectData\(SerializationInfo, StreamingContext\)](#)  ,  
[ArgumentException.Message](#)  , [ArgumentException.ParamName](#)  , [Exception.GetBaseException\(\)](#)  ,  
[Exception.ToString\(\)](#)  , [Exception.GetType\(\)](#)  , [Exception.Data](#)  , [Exception.InnerException](#)  ,  
[Exception.TargetSite](#)  , [Exception.StackTrace](#)  , [Exception.HelpLink](#)  , [Exception.Source](#)  ,  
[Exception.HResult](#)  , [object.Equals\(object\)](#)  , [object.Equals\(object, object\)](#)  ,  
[object.ReferenceEquals\(object, object\)](#)  , [object.GetHashCode\(\)](#)  , [object.MemberwiseClone\(\)](#) 

## Constructors

### SnowflakesUsingDifferentEpochsException()

Initializes a new instance of the [SnowflakesUsingDifferentEpochsException](#) class.

```
public SnowflakesUsingDifferentEpochsException()
```



## SnowflakesUsingDifferentEpochsException(SerializationInfo, StreamingContext)

Initializes a new instance of the [SnowflakesUsingDifferentEpochsException](#) class with serialized data.

```
protected SnowflakesUsingDifferentEpochsException(SerializationInfo serializationInfo,  
StreamingContext streamingContext)
```

### Parameters

**serializationInfo** [SerializationInfo](#) 

The object that holds the serialized object data.

**streamingContext** [StreamingContext](#) 

The contextual information about the source or destination.

## SnowflakesUsingDifferentEpochsException(string)

Initializes a new instance of the [SnowflakesUsingDifferentEpochsException](#) class with a specified error message.

```
public SnowflakesUsingDifferentEpochsException(string message)
```

### Parameters

**message** [string](#) 

The error message that explains the reason for the exception.

## SnowflakesUsingDifferentEpochsException(string, Exception)

Initializes a new instance of the [SnowflakesUsingDifferentEpochsException](#) class with a specified error message and a reference to the inner exception that is the cause of this exception.

```
public SnowflakesUsingDifferentEpochsException(string message, Exception innerException)
```

## Parameters

**message** [string](#)

The error message that explains the reason for the exception.

**innerException** [Exception](#)

The exception that is the cause of the current exception.

## SnowflakesUsingDifferentEpochsException(string, string)

Initializes a new instance of the [SnowflakesUsingDifferentEpochsException](#) class with a specified error message and the name of the parameter that causes this exception.

```
public SnowflakesUsingDifferentEpochsException(string message, string paramName)
```

## Parameters

**message** [string](#)

The error message that explains the reason for the exception.

**paramName** [string](#)

The name of the parameter that caused the current exception.

## SnowflakesUsingDifferentEpochsException(string, string, Exception)

Initializes a new instance of the [SnowflakesUsingDifferentEpochsException](#) class with a specified error message, the parameter name, and a reference to the inner exception that is the cause of this exception.

```
public SnowflakesUsingDifferentEpochsException(string message, string paramName, Exception innerException)
```

## Parameters

**message** [string](#)

The error message that explains the reason for the exception.

paramName [string](#)

The name of the parameter that caused the current exception.

innerException [Exception](#)

The exception that is the cause of the current exception.

## Fields

### DefaultMessage

Default message for the exception.

```
public const string DefaultMessage = "When comparing SnowflakeIds, both should be using the  
same epoch for the comparison to make sense."
```

### Field Value

[string](#)

# Namespace SnowflakeID.Helpers

## Classes

### [GlobalConstants](#)

Global constants used throughout the Snowflake ID generator.

# Class GlobalConstants


Namespace: [SnowflakeID.Helpers](#)

Assembly: SnowflakeIDGenerator.dll








Global constants used throughout the Snowflake ID generator.

```
public static class GlobalConstants
```

## Inheritance

[object](#)  ← GlobalConstants

## Inherited Members

[object.ToString\(\)](#)  , [object.Equals\(object\)](#)  , [object.Equals\(object, object\)](#)  ,  
[object.ReferenceEquals\(object, object\)](#)  , [object.GetHashCode\(\)](#)  , [object.GetType\(\)](#)  ,  
[object.MemberwiseClone\(\)](#) 

## Fields

### DefaultEpoch

The default date used as the epoch if not configured.

```
public static readonly DateTime DefaultEpoch
```

## Field Value

[DateTime](#) 

## Remarks

This is set to Unix Epoch (January 1, 1970).