C-Sharp Mapping for Identifiers

Slice identifiers map to an identical C# identifier. For example, the Slice identifier `Clock` becomes the C# identifier `Clock`. If a Slice identifier is the same as a C# keyword, the corresponding C# identifier is a *verbatim identifier* (an identifier prefixed with @). For example, the Slice identifier `while` is mapped as `@while`.

> ⓘ You should try to [avoid such identifiers](#) as much as possible.

The Slice-to-C# compiler generates classes that inherit from interfaces or base classes in the .NET framework. These interfaces and classes introduce a number of methods into derived classes. To avoid name clashes between Slice identifiers that happen to be the same as an inherited method, such identifiers are prefixed with `ice_` and suffixed with `_` in the generated code. For example, the Slice identifier `Clone` maps to the C# identifier `ice_Clone_` if it would clash with an inherited `Clone`. The complete list of identifiers that are so changed is:

Clone	Equals	Finalize
GetBaseException	GetHashCode	GetObjectData
GetType	MemberwiseClone	ReferenceEquals
ToString	checkedCast	uncheckedCast

Note that Slice identifiers in this list are translated to the corresponding C# identifier only where necessary. For example, structures do not derive from `ICloneable`, so if a Slice structure contains a member named `Clone`, the corresponding C# structure's member is named `Clone` as well. On the other hand, classes do derive from `ICloneable`, so, if a Slice class contains a member named `Clone`, the corresponding C# class's member is named `ice_Clone_`.

Also note that, for the purpose of prefixing, Slice identifiers are case-insensitive, that is, both `Clone` and `clone` are escaped and map to `ice_Clone_` and `ice_clone_`, respectively.

See Also

- [Identifiers That Are Keywords](#)
- [C-Sharp Mapping for Modules](#)
- [C-Sharp Mapping for Built-In Types](#)
- [C-Sharp Mapping for Enumerations](#)
- [C-Sharp Mapping for Structures](#)
- [C-Sharp Mapping for Sequences](#)
- [C-Sharp Mapping for Dictionaries](#)
- [C-Sharp Collection Comparison](#)
- [C-Sharp Mapping for Constants](#)
- [C-Sharp Mapping for Exceptions](#)