# NatsnudaLibrary.ruleset

The following table documents any code analysis rules that have been set to a **lower** severity than their default setting.

|  |  |  |
| --- | --- | --- |
| Rule | Description | Reason |
| CA1006 | Do not nest generic types in member signatures. | This rule is just too restrictive for things like Expression<Func<int, bool>>, Task<IEnumerable<int>>, etc. |
| CA1020 | Avoid namespaces with few types. | This rule may be turned on again further in development. |
| CA1811 | Avoid uncalled private code. | This rule is just annoying for serialization, entity framework, or any object that is consumed by an external source/tool. |
| CA2202 | Do not dispose object multiple times. | This rule seems to be a bit controversial. We ignore it. |
| CA2243 | Attribute string literals should parse correctly. | A long-standing bug causes this rule to trigger when using a non-version string in AssemblyInformationalVersionAttribute. |
| CC0013 | Use ternary operator (return). | I prefer to make the decision myself when a ternary operator is more readable. |
| CC0014 | Use ternary operator (assignment) | <see CC0013>. |
| IDE0003 | Remove ‘this’ qualification. | Conflicts with SA1101 (prefix with ‘this’). |
| MEN008 | File name should match type. | Covered by SA1649. |
| IDE0041 | Use ‘is null’ check. | I prefer ReferenceEquals here. |

The following table documents any code analysis rules that have been set to a **higher** severity than their default setting.

|  |  |  |
| --- | --- | --- |
| Rule | Description | Reason |
| CC0070 | Use ConfigureAwait(false). | This has been set to Info since this is a library project and most cases will not need to resume on the same context. |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# TestExtensions.ruleset

The following table documents any code analysis rules that have been set to a **lower** severity than their default setting.

|  |  |  |
| --- | --- | --- |
| Rule | Description | Reason |
| CA1006 | Do not nest generic types in member signatures. | This rule is just too restrictive for things like Expression<Func<int, bool>>, Task<IEnumerable<int>>, etc. |
| CA1020 | Avoid namespaces with few types. | This rule may be turned on again further in development. |
| CA1811 | Avoid uncalled private code. | This rule is just annoying for serialization, entity framework, or any object that is consumed by an external source/tool. |
| CA2202 | Do not dispose object multiple times. | This rule seems to be a bit controversial. We ignore it. |
| CA2243 | Attribute string literals should parse correctly. | A long-standing bug causes this rule to trigger when using a non-version string in AssemblyInformationalVersionAttribute. |
| CC0013 | Use ternary operator (return). | I prefer to make the decision myself when a ternary operator is more readable. |
| CC0014 | Use ternary operator (assignment) | <see CC0013>. |
| IDE0003 | Remove ‘this’ qualification. | Conflicts with SA1101 (prefix with ‘this’). |
| MEN008 | File name should match type. | Covered by SA1649. |
| IDE0041 | Use ‘is null’ check. | I prefer ReferenceEquals here. |

The following table documents any code analysis rules that have been set to a **higher** severity than their default setting.

|  |  |  |
| --- | --- | --- |
| Rule | Description | Reason |
| CC0070 | Use ConfigureAwait(false). | This has been set to Info since this is a library project and most cases will not need to resume on the same context. |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |