

简介

StyleCop.Analyzers是一个Nuget，用于.NET编译器平台的StyleCop分析器，该存储库包含使用.NET编译器平台的StyleCop规则的实现。在可能的情况下，还会提供代码修复以简化纠正违规的过程。

使用分析器的首选方法是将nuget包StyleCop.Analyzers添加到要执行StyleCop规则的项目中。

单个规则的严重性可以使用Visual Studio 2015或更新版本中的规则集文件进行配置。Settings.StyleCop不受支持，但可以使用stylecop.json文件来自定义某些规则的行为。

GitHub上有详细介绍：

StyleCop: <https://github.com/StyleCop/StyleCop>

StyleCop.Analyzers: <https://github.com/DotNetAnalyzers/StyleCopAnalyzers>

安装

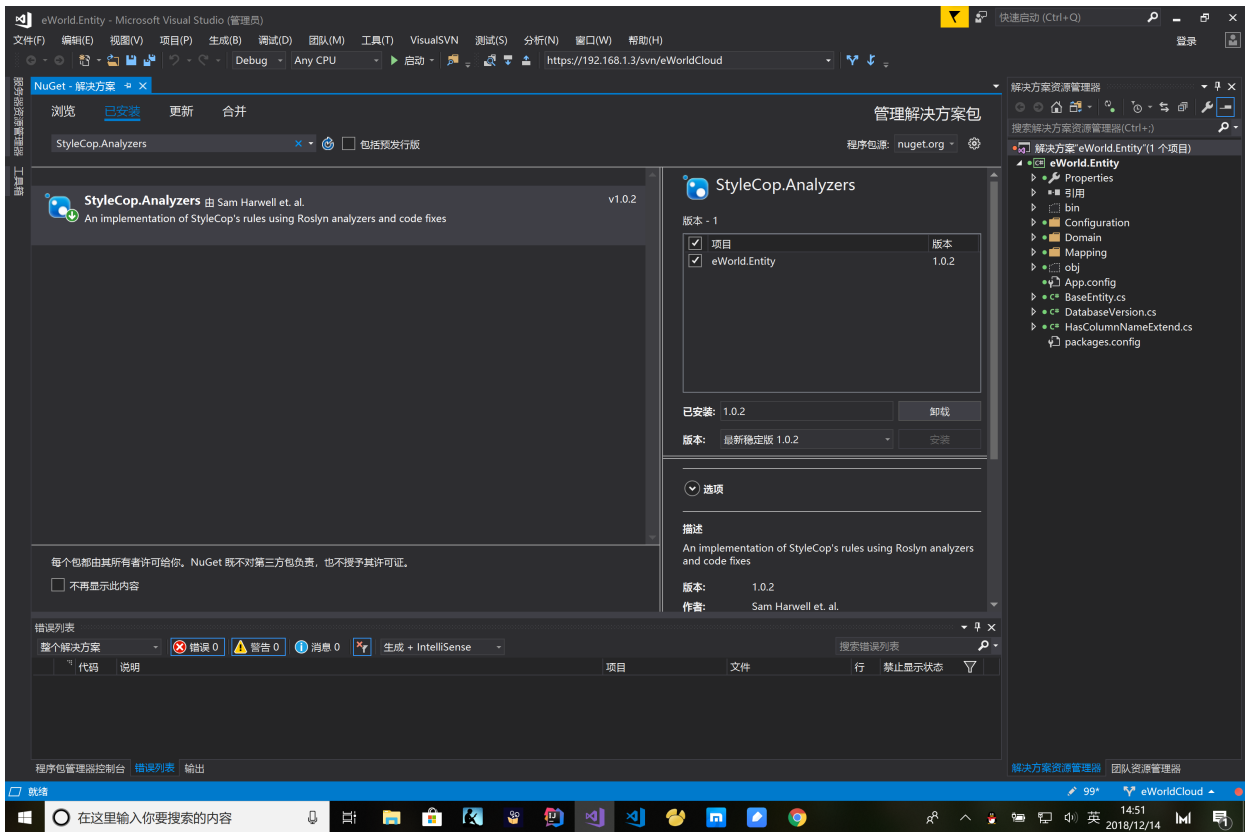
可以使用NuGet命令行或Visual Studio 2017中的NuGet Package Manager来安装StyleCopAnalyzers。

命令行安装

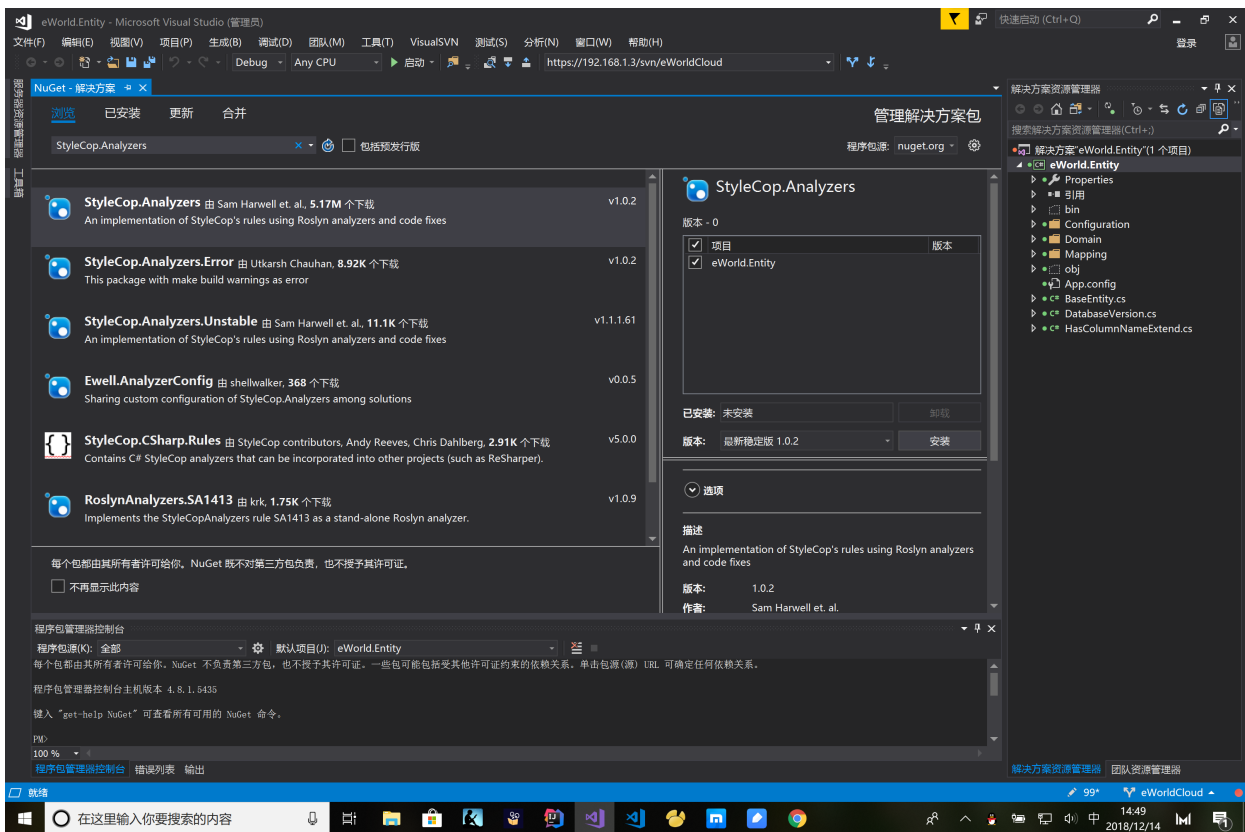
Install-Package StyleCop.Analyzers

NuGet Package Manager安装

在项目->管理NuGet程序包(N) 或者 在项目上右键，点击管理NuGet程序包，显示NuGet包管理界面，如下图所示：



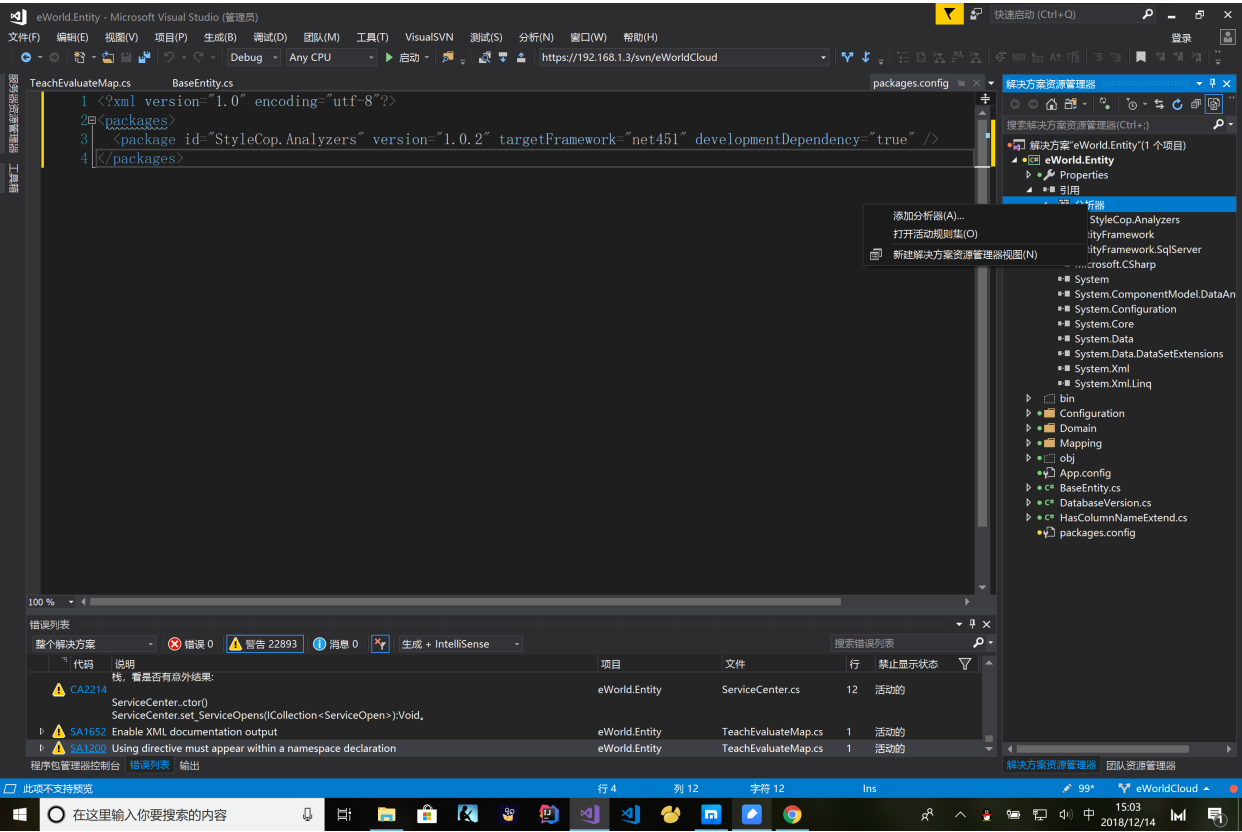
由于我已经安装，所以直接就显示出来，如果没有安装，可以选择“浏览”，在搜索框中输入StyleCop.Analyzers选择安装即可，如下图所示：

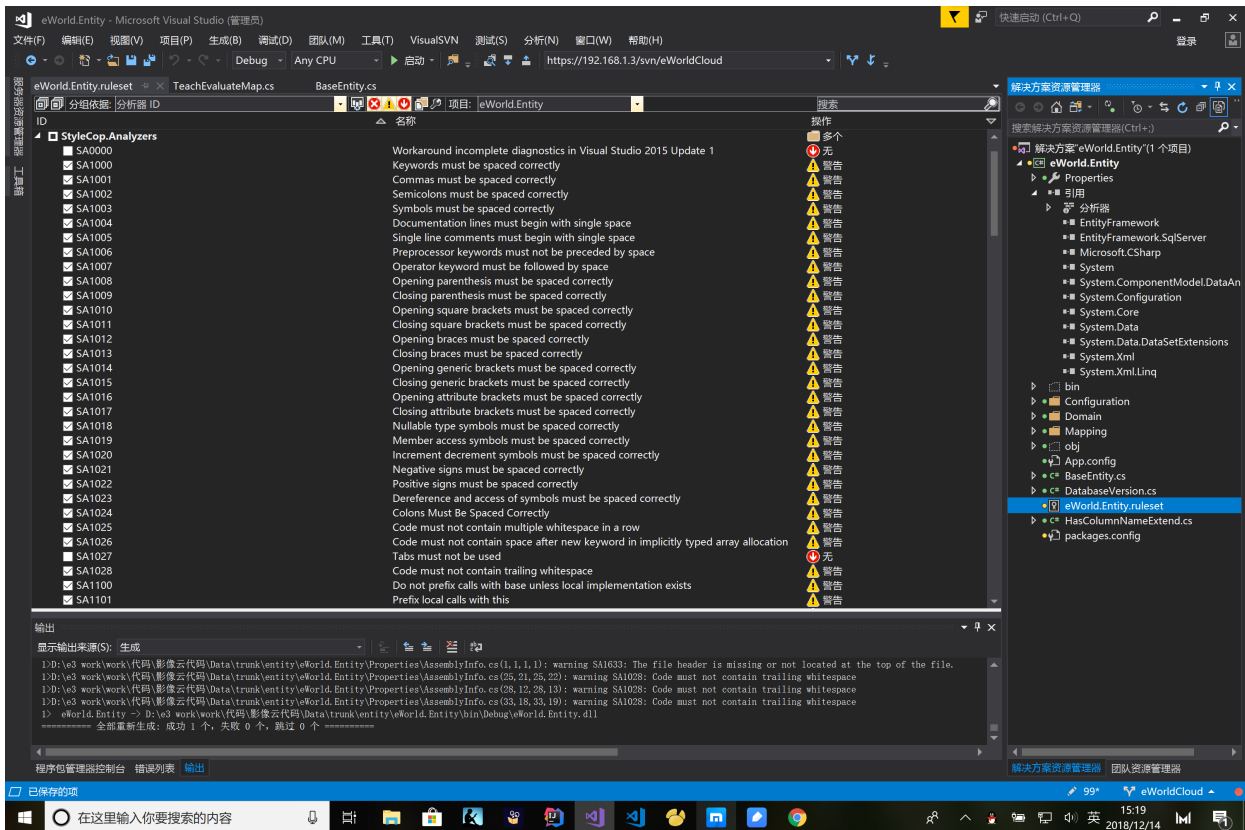


使用

自定义规则

在解决方案视图中，选择项目XXX，“引用”-->"分析器"，右键“打开活动规则集”，选择Style.Analyzers，如下图所示：





可自定义规则，保存之后，项目中会生成XXX.ruleset和packages.config两个文件。

去掉和医网C#编码规范冲突项（可以根据实际做调整）：

SA1027:TabsMustNotBeUsed

SA1124:DoNotUseRegions

SA1200:UsingDirectivesMustBePlacedWithinNamespace

SA1201:ElementsMustAppearInTheCorrectOrder

SA1202:ElementsMustBeOrderedByAccess

SA1208:SystemUsingDirectivesMustBePlacedBeforeOtherUsingDirectives

SA1210:UsingDirectivesMustBeOrderedAlphabeticallyByNamespace

SA1309:FieldNamesMustNotBeginWithUnderscore

SA1310:FieldNamesMustNotContainUnderscore

SA1410:RemoveDelegateParenthesisWhenPossible

SA1623:PropertySummaryDocumentationMustMatchAccessors

SA1633:FileMustHaveHeader

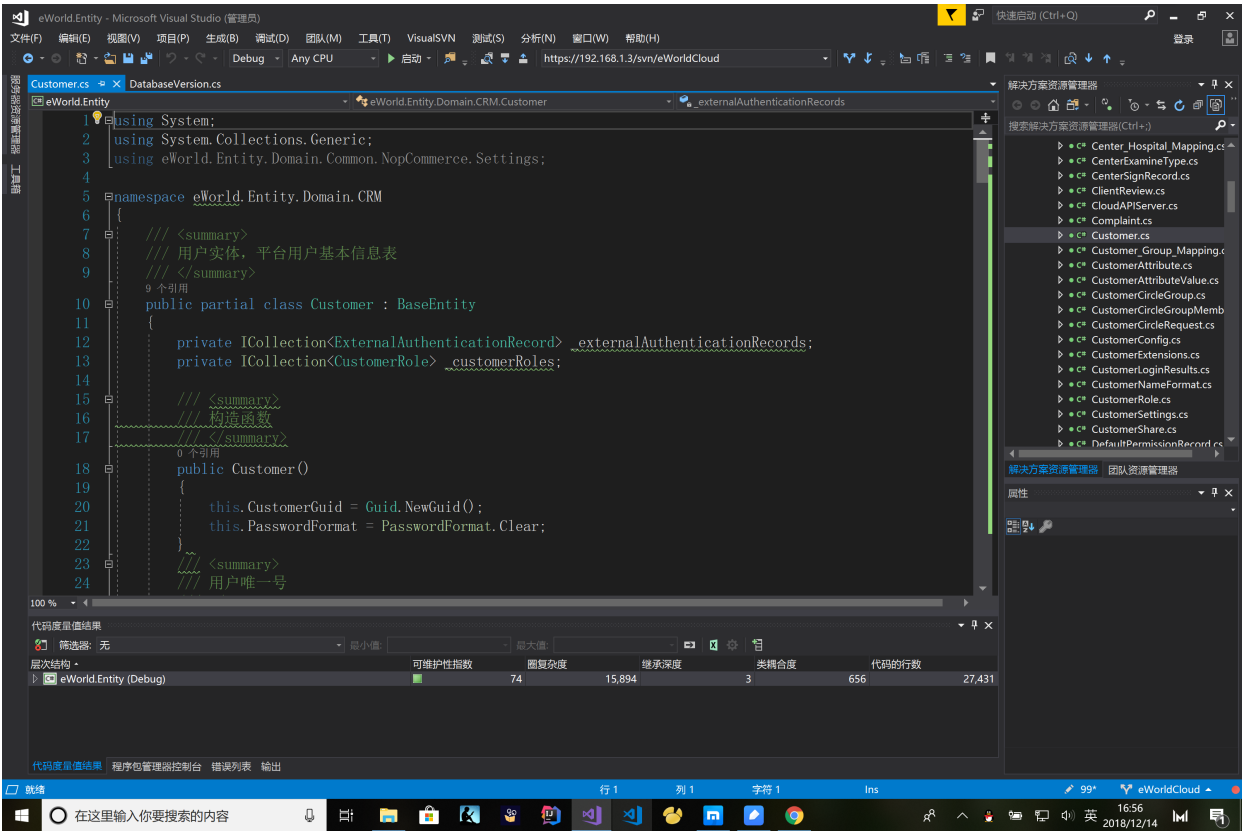
SA1645:IncludedDocumentationFileDoesNotExist

SA1650:ElementDocumentationMustBeSpelledCorrectly

SA1652:Enable XML documentation output

效果

如下图所示：



带有绿色波浪线的就是可能存在编码不规范的地方，可根据自身或公司编码规范进行调整。

在StyleCop中官方检测规则中文解释：

文档规则

- 1、SA1600: ElementsMustBeDocumented 元素必须添加注释
- 2、SA1601: PartialElementsMustBeDocumented Partial修饰的成员必须添加注释
- 3、SA1602: EnumerationItemsMustBeDocumented 枚举必须添加注释
- 4、SA1603: DocumentationMustContainValidXml 注释必须合法(注释中的关键字不能有错误)
- 5、SA1604: ElementDocumentationMustHaveSummary 元素注释必须包含Summary关键字
- 6、SA1605: PartialElementDocumentationMustHaveSummary Partial修饰的成员注释必须包含Summary关键字
- 7、SA1606: ElementDocumentationMustHaveSummaryText Summary节点内部必须添加文本

- 8、SA1607:PartialElementDocumentationMustHaveSummaryText Partial修饰的成员注释Summary节点内部必须添加文本
- 9、SA1608:ElementDocumentationMustNotHaveDefaultSummary Summary注释不能使用编译器自带的注释文本
- 10、SA1609:PropertyDocumentationMustHaveValue 属性的注释中必须包含<Value>节点
- 11、SA1610:PropertyDocumentationMustHaveValueText 属性的注释<Value>节点必须包含文本值
- 12、SA1611:ElementParametersMustBeDocumented 参数必须注释
- 13、SA1612:ElementParameterDocumentationMustMatchElementParameters 参数的个数必须与注释里的个数一致
- 14、SA1613:ElementParameterDocumentationMustDeclareParameterName 参数的注释里必须有参数的名称
- 15、SA1614:ElementParameterDocumentationMustHaveText 参数注释节点里不能空着
- 16、SA1615:ElementReturnValueMustBeDocumented 返回值必须添加注释
- 17、SA1616:ElementReturnValueDocumentationMustHaveText 返回值注释的节点内必须有文本值
- 18、SA1617:VoidReturnValueMustNotBeDocumented 空返回值一定不能有返回值注释
- 19、SA1618:GenericTypeParametersMustBeDocumented 生成的类型(泛型)参数必须有注释
- 20、SA1619:GenericTypeParametersMustBeDocumentedPartialClass SA1618情况如果有Partial类存在都要有注释
- 21、SA1620:GenericTypeParameterDocumentationMustMatchTypeParameters 注释与泛型必须能匹配上
- 22、SA1621:GenericTypeParameterDocumentationMustDeclareParameterName 泛型的注释必须包含泛型定义是的关键字(如'T')
- 23、SA1622:GenericTypeParameterDocumentationMustHaveText 泛型的注释节点中必须包含文本值
- 24、SA1623:PropertySummaryDocumentationMustMatchAccessors 属性的注释必须与属性的读写权限匹配, private类型的属性不能出现在注释里
- 25、SA1624:PropertySummaryDocumentationMustOmitSetAccessorWithRestrictedAccess 属性的注释里必须忽略protected关键字, 当成public类型当注释
- 26、SA1625:ElementDocumentationMustNotBeCopiedAndPasted 各个参数的注解不能完全相同(避免copy、post行为)
- 27、SA1626:SingleLineCommentsMustNotUseDocumentationStyleSlashes 单行注释不能采取三个斜线的注释方式, 斜线的个数必须是二的倍数
- 28、SA1627:DocumentationTextMustNotBeEmpty 注释节点内部不能为空(必须有文本值)

- 29、SA1628:DocumentationTextMustBeginWithACapitalLetter 注释节点内部的文本必须以大写字母开头
- 30、SA1629:DocumentationTextMustEndWithAPeriod 注释节点内部的文本必须英文的句号结束
- 31、SA1630:DocumentationTextMustContainWhitespace 注释节点内部的文本必须包含空格
- 32、SA1631:DocumentationMustMeetCharacterPercentage 注释节点内部的文本中不能包含过多的字符(举例:包含'-----'是不允许的)
- 33、SA1632:DocumentationTextMustMeetMinimumCharacterLength 已经作废的规则,不允许有太短的字符串(如'A name' 中的A字母)
- 34、SA1633:FileMustHaveHeader 代码文件头部必须有说明,一般放置版权信息
- 35、SA1634:FileHeaderMustShowCopyright 代码文件头部注解中必须包含版权关键字
- 36、SA1635:FileHeaderMustHaveCopyrightText代码文件头部注解中必须包含版权信息内容
- 37、SA1636:FileHeaderCopyrightTextMustMatch 代码文件头部注解中版权信息必须与设置画面设置的内容相匹配,在"style seting"画面的" Company Information tab"进行设置
- 38、SA1637:FileHeaderMustContainFileName 代码文件头部注解中必须包含文件名称
- 39、SA1638:FileHeaderFileNameDocumentationMustMatchFileName 代码文件头部注解中必须包含文件名称必须与实际名称匹配
- 40、SA1639:FileHeaderMustHaveSummary 代码文件头部注解中必须包含Summary节点
- 41、SA1640:FileHeaderMustHaveValidCompanyText 代码文件头部注解中版权信息必须含有合理的公司名字
- 42、SA1641:FileHeaderCompanyNameTextMustMatch 代码文件头部注解中的公司名字必须与设置画面设置的内容相匹配,在"style seting"画面的" Company Information tab"进行设置
- 43、SA1642:ConstructorSummaryDocumentationMustBeginWithStandardText 构造函数注释标准: "Initializes a new instance of the <see cref="Customer{T}"/> class."
- 44、SA1643:DestructorSummaryDocumentationMustBeginWithStandardText 析构函数注释标准: "Finalizes an instance of the <see cref="Customer"/> class."
- 45、SA1644:DocumentationHeadersMustNotContainBlankLines 注释中不能出现空白行
- 46、SA1645:IncludedDocumentationFileDoesNotExist 如果注释中包含文件,要确定这个文件存在,举例: "`/// <include file="IncludedDocumentation.xml" path="root/EnabledMethodDocs" />`" 这一项过于复杂不建议用,但是如果将来想规范化相同共性东西的注释的时候倒是可以考虑引入。
- 47、SA1646:IncludedDocumentationXPathDoesNotExist 如果注释中包含文件,路径不存在。
- 48、SA1647:IncludeNodeDoesNotContainValidFileAndPath 如果注释中包含文件 文件与路径都要合法

49、SA1648:InheritDocMustBeUsedWithInheritingClass 这项不明白是什么意思，还请知道的分享给我，谢谢。

50、SA1649:FileHeaderFileNameDocumentationMustMatchTypeName 文件头注释必须匹配类型

51、SA1650:ElementDocumentationMustBeSpelledCorrectly 注释不能有拼写错误(对中文支持不好，不建议使用该项)

布局规则

1、SA1500:CurlyBracketsForMultiLineStatementsMustNotShareLine 花括号不能再同一行。

标准格式： `public StRsvrRFun()`

```
{  
}
```

2、SA1501:StatementMustNotBeOnASingleLine 语句不能共享一行

3、SA1502:ElementMustNotBeOnASingleLine 语句不能共享一行

4、SA1503:CurlyBracketsMustNotBeOmitted 花括弧即使在单行代码的情况下也不能省略。

5、SA1504:AllAccessorsMustBeSingleLineOrMultiLine 读写属性，要么在同一行要么在分多行写，不可规则不统一(个人建议分多行写)。

6、SA1505:OpeningCurlyBracketsMustNotBeFollowedByBlankLine 开始的花括弧后面不能有空白行

7、SA1506:ElementDocumentationHeadersMustNotBeFollowedByBlankLine 元素头部注释跟元素之间不能有空白行

8、SA1507:CodeMustNotContainMultipleBlankLinesInARow 不允许有多行空白行紧挨着(个人建议写一个空白行即可)

9、SA1508:ClosingCurlyBracketsMustNotBePrecededByBlankLine 结尾的花括弧不能再一个空白行之前

10、SA1509:OpeningCurlyBracketsMustNotBePrecededByBlankLine 开始的花括弧不能再一个空白行之前

11、SA1510:ChainedStatementBlocksMustNotBePrecededByBlankLine 相连的语句之间不能有空白行，如try语句与catch语句之间不能有空白行

12、SA1511:WhileDoFooterMustNotBePrecededByBlankLine 与SA1510相同，Do语句与While语句之间不能有空白行

13、SA1512:SingleLineCommentsMustNotBeFollowedByBlankLine 单行注释不能后跟空白行

14、SA1513:ClosingCurlyBracketMustBeFollowedByBlankLine 结束花括弧之后必须有一个空白行

- 15、SA1514:ElementDocumentationHeaderMustBePrecededByBlankLine 头部注解之前必须有一个空白行
- 16、SA1515:SingleLineCommentMustBePrecededByBlankLine 单行注释之前要有一个空白行，还一种方法是不加空白行而用四个斜线注释“////”，建议采取第一种方法
- 17、SA1516:ElementsMustBeSeparatedByBlankLine 邻近的元素之间要有一个空白行
- 18、SA1517:CodeMustNotContainBlankLinesAtStartOfFile 代码文件头部字符之前不能出现空白行
- 19、SA1518:CodeMustNotContainBlankLinesAtEndOfFile 代码文件尾部字符之后不能出现空白行

可维护规则

- 1、SA1119:StatementMustNotUseUnnecessaryParenthesis 语句中不可以出现多余的括弧，无意义的括弧增加了代码的可读性
- 2、SA1400:AccessModifierMustBeDeclared 必须定义访问修饰符
- 3、SA1401:FieldsMustBePrivate 字段必须定义为私有的
- 4、SA1402:FileMayOnlyContainASingleClass 一个CS文件里只定义一个类
- 5、SA1403:FileMayOnlyContainASingleNamespace 一个CS文件只包含一个命名空间
- 6、SA1404:CodeAnalysisSuppressionMustHaveJustification Suppression特性(取消报告特定的静态分析工具规则冲突，允许一个代码项目上应用多个取消报告设置)必须要有合理理由
- 7、SA1405:DebugAssertMustProvideMessageText 参见代码：“Debug.Assert(value != true, "The value must always be true.");”
- 8、SA1406:DebugFailMustProvideMessageText 参见代码：“Debug.Fail("The code should never reach this point.");”
- 9、SA1407:ArithmeticExpressionsMustDeclarePrecedence 算数表达式必须用明确的标明其优先级(此条与SA1119冲突)
- 10、SA1408:ConditionalExpressionsMustDeclarePrecedence 条件表达式必须明确的标明其优先级(此条与SA1119冲突) 举例：“if (x || (y && z && a) || b)”
- 11、SA1409:RemoveUnnecessaryCode 移除无用的代码
- 12、SA1410:RemoveDelegateParenthesisWhenPossible 调用一个c#匿名方法不包含任何方法参数,必须要包括一个空括号(本人不建议使用匿名方法、匿名委托的语法，具体原因后续做分享)
- 13、SA1411:AttributeConstructorMustNotUseUnnecessaryParenthesis 特性构造函数如果是空参的形式，不要包括那个空括号

命名规则

- 1、SA1300:ElementMustBeginWithUpperCaseLetter 首字母必须大写(个人理解应该是字段以外的东西)

- 2、SA1301:ElementMustBeginWithLowerCaseLetter 不会出现的情况
- 3、SA1302:InterfaceNamesMustBeginWithI 接口必须以"I"字母开头
- 4、SA1303:ConstFieldNamesMustBeginWithUpperCaseLetter Const常量首字母大写
- 5、SA1304:NonPrivateReadOnlyFieldsMustBeginWithUpperCaseLetter 非私有非制度的字段必须首字母大写
- 6、SA1305:FieldNamesMustNotUseHungarianNotation 除了列表外的字符不可以出现匈牙利命名规则,在"Style Setting"里的" Hungarian"Tab里设置
- 7、SA1306:FieldNamesMustBeginWithLowerCaseLetter 字段名必须首字母小写
- 8、SA1307:AccessibleFieldsMustBeginWithUpperCaseLetter public或者internal字段首字母大写
- 9、SA1308:VariableNamesMustNotBePrefixed 变量名不能有前缀
- 10、SA1309:FieldNamesMustNotBeginWithUnderscore 字段名不能以下划线开头
- 11、SA1310:FieldNamesMustNotContainUnderscore 字段名不能包括下划线
- 12、SA1311:StaticReadOnlyFieldsMustBeginWithUpperCaseLetter static与readonly字段首字母大写

排序规则

- 1、SA1200:UsingDirectivesMustBePlacedWithinNamespace using部分必须在Namespace内部(个人不建议用这个规则)
- 2、SA1201:ElementsMustAppearInTheCorrectOrder 所有的元素必须出现在正确的位置上(个人不建议使用该规则,用region把一个类文件里的元素分好组即可,字段、属性、对外公布的接口、私有方法等组)
- 3、SA1202:ElementsMustBeOrderedByAccess 元素的修饰符有一个固定的顺序(个人不建议使用该规则,用region把一个类文件里的元素分好组即可,字段、属性、对外公布的接口、私有方法等组)
- 4、SA1203:ConstantsMustAppearBeforeFields 常量字段摆放在其它字段之上
- 5、SA1204:StaticElementsMustAppearBeforeInstanceElements Static 元素摆放在前面
- 6、SA1205:PartialElementsMustDeclareAccess Partial修饰的元素必须声明读写权限
- 7、SA1206:DeclarationKeywordsMustFollowOrder 举例:
keywords must appear in the following order:
Access modifiers
static
All other keywords
- 8、SA1207:ProtectedMustComeBeforeInternal Protected在Internal之前
- 9、SA1208:SystemUsingDirectivesMustBePlacedBeforeOtherUsingDirectives 系统级的引用在其他引用之前(格式化代码就会自动按照这个规则编排using部分)

- 10、SA1209:UsingAliasDirectivesMustBePlacedAfterOtherUsingDirectives using部分起别名的在最后排布
- 11、SA1210:UsingDirectivesMustBeOrderedAlphabeticallyByNamespace using部分按照字母顺序编排
- 12、SA1211:UsingAliasDirectivesMustBeOrderedAlphabeticallyByAliasName 别名的按照别名的字母编排
- 13、SA1212:PropertyAccessorsMustFollowOrder 属性按照先get在set
- 14、SA1213:EventAccessorsMustFollowOrder 时间先remove在add
- 15、SA1214:StaticReadOnlyElementsMustAppearBeforeStaticNonReadOnlyElements StaticReadOnly元素在StaticNonReadOnly元素之前
- 16、SA1215:InstanceReadOnlyElementsMustAppearBeforeInstanceNonReadOnlyElements 一个readonly 实例元素要在一个非readonly 实例元素

可读性规则

- 1、SA1100:DoNotPrefixCallsWithBaseUnlessLocalImplementationExists change the 'base.' prefix to 'this.'
- 2、SA1101:PrefixLocalCallsWithThis 'this.'前缀不能省略
- 3、SA1102:QueryClauseMustFollowPreviousClause linq语句要么同一行，要么关键字分布在多行，关键字对齐
- 4、SA1103:QueryClausesMustBeOnSeparateLinesOrAllOnOneLine 不是很理解
- 5、SA1104:QueryClauseMustBeginOnNewLineWhenPreviousClauseSpansMultipleLines不是很理解
- 6、SA1105:QueryClausesSpanningMultipleLinesMustBeginOnOwnLine不是很理解
- 7、SA1106:CodeMustNotContainEmptyStatements 不允许有空白的声明
- 8、SA1107:CodeMustNotContainMultipleStatementsOnOneLine 同一行里不允许出现多个声明语句
- 9、SA1108:BlockStatementsMustNotContainEmbeddedComments 非法用例：

```
if (x != y)
    // Make sure x does not equal y
    {
    }
```
- 10、SA1109:BlockStatementsMustNotContainEmbeddedRegions 非法用例：

```
if (x != y)
    #region
    {
```

```
}
```

```
#endregion
```

11、SA1110:OpeningParenthesisMustBeOnDeclarationLine 方法开始的括弧要在声明的那一行

12、SA1111:ClosingParenthesisMustBeOnLineOfLastParameter 方法结束括弧与最后一个参数同行

13、SA1112:ClosingParenthesisMustBeOnLineOfOpeningParenthesis 如果是空参的情况，开始于结束括弧在同一行

14、SA1113:CommaMustBeOnSameLineAsPreviousParameter 逗号必须与前一个参数同行

15、SA1114:ParameterListMustFollowDeclaration

确保参数列表与开始括号同行，或者另起一行 参数整体要在同一行

16、SA1115:ParameterMustFollowComma 参数与参数之间不能隔空白行

17、SA1116:SplitParametersMustStartOnLineAfterDeclaration 所有参数在同一行或者如下示例：

```
public string JoinName(  
    string first,  
    string last)
```

18、SA1117:ParametersMustBeOnSameLineOrSeparateLines 同上

19、SA1118:ParameterMustNotSpanMultipleLines 实际调用是参数要在同一行

20、SA1120:CommentsMustContainText 注释必须节点内部必须有文本

22、SA1122:UseStringEmptyForEmptyStrings

示例代码：“string s = string.Empty;”不能写成“string s = “”; ”

23、SA1123:DoNotPlaceRegionsWithinElements 不使用region(个人较反对这条，个人倾向于用region规整类里的元素)

24、SA1124:DoNotUseRegions 不使用region(个人较反对这条，个人倾向于用region规整类里的元素)

25、SA1125:UseShorthandForNullableTypes 使用简写版的可空类型 如“int?”

26、SA1126:PrefixCallsCorrectly 调用类成员的时候要带上“this”前缀

拼写规则

1、SA1000:KeywordsMustBeSpacedCorrectly 关键字不能拼写错误

2、SA1001:CommasMustBeSpacedCorrectly 逗号前面摆放一个空格

3、SA1002:SemicolonsMustBeSpacedCorrectly 分号前面摆放一个空格

4、SA1003:SymbolsMustBeSpacedCorrectly 操作符与元素之间不加空格

5、SA1004:DocumentationLinesMustBeginWithSingleSpace “///”注释的每一行以一个空格开头

- 6、SA1005:SingleLineCommentsMustBeginWithSingleSpace 单行注释以一个空格开头
- 7、SA1006:PreprocessorKeywordsMustNotBePrecededBySpace 预处理关键字内部不允许出现空格 如“# if Debug”是不允许的
- 8、SA1007:OperatorKeywordMustBeFollowedBySpace 操作必须要在一个空格之后
- 9、SA1008:OpeningParenthesisMustBeSpacedCorrectly 开始括弧放置合理的空格
- 10、SA1009:ClosingParenthesisMustBeSpacedCorrectly 结尾括弧放置合理的空格
- 11、SA1010:OpeningSquareBracketsMustBeSpacedCorrectly 开始方括号放置合理的空格
- 12、SA1011:ClosingSquareBracketsMustBeSpacedCorrectly 结束方括号放置合理的空格
- 13、SA1012:OpeningCurlyBracketsMustBeSpacedCorrectly 开始花括号放置合理的空格
- 14、SA1013:ClosingCurlyBracketsMustBeSpacedCorrectly 结束花括号放置合理的空格
- 15、SA1014:OpeningGenericBracketsMustBeSpacedCorrectly 同上
- 16、SA1015:ClosingGenericBracketsMustBeSpacedCorrectly 同上
- 17、SA1016:OpeningAttributeBracketsMustBeSpacedCorrectly 特性开始括号同上
- 18、SA1017:ClosingAttributeBracketsMustBeSpacedCorrectly 特性结尾括号同上
- 19、SA1018:NullableTypeSymbolsMustNotBePrecededBySpace 空类型前不放置空格
- 20、SA1019:MemberAccessSymbolsMustBeSpacedCorrectly 成员访问关键字前后有空格
- 21、SA1020:IncrementDecrementSymbolsMustBeSpacedCorrectly 自增、自减关键字要有空格
- 22、SA1021:NegativeSignsMustBeSpacedCorrectly 负号要有空格
- 23、SA1022:PositiveSignsMustBeSpacedCorrectly 正号要有空格
- 24、SA1023:DereferenceAndAccessOfMustBeSpacedCorrectly
- 25、SA1024:ColonsMustBeSpacedCorrectly 冒号运算符要有空格
- 26、SA1025:CodeMustNotContainMultipleWhitespaceInARow 同一行不允许出现多个空格
- 27、SA1026:CodeMustNotContainSpaceAfterNewKeywordInImplicitlyTypedArrayAllocation
new关键字之后不允许出现空格
- 28、SA1027:TabsMustNotBeUsed 不用tab字符(个人不建议采用这条)

结束语

编码规范是一个Coder应该具有的基本素养，如何能编写出简洁、规范的代码显得尤为重要，借用Python之禅与各位共勉：

优美胜于丑陋（以编写优美的代码为目标）

明了胜于晦涩（优美的代码应当是明了的，命名规范，风格相似）

简洁胜于复杂（优美的代码应当是简洁的，不要有复杂的内部实现）

复杂胜于凌乱（如果复杂不可避免，那代码间也不能有难懂的关系，要保持接口简洁）

扁平胜于嵌套（优美的代码应当是扁平的，不能有太多的嵌套）

间隔胜于紧凑（优美的代码有适当的间隔，不要奢望一行代码解决问题）

可读性很重要（优美的代码是可读的）

即便假借特例的实用性之名，也不可违背这些规则（这些规则至高无上）

不要包容所有错误，除非你确定需要这样做（精准地捕获异常，不写 `except:pass` 风格的代码）

当存在多种可能，不要尝试去猜测 而是尽量找一种，最好是唯一一种明显的解决方案（如果不确定，就用穷举法）

虽然这并不容易，因为你不是 Python 之父（这里的 Dutch 是指 Guido）
做也许好过不做，但不假思索就动手还不如不做（动手之前要细思量）

如果你无法向人描述你的方案，那肯定不是一个好方案；反之亦然（方案测评标准）

命名空间是一种绝妙的理念，我们应当多加利用（倡导与号召）