

library ahbsd.lib

1.4

Generated by Doxygen 1.9.1

1 LICENSE	1
2 ahbsd.lib	5
2.1 About generic EventArgs	5
2.2 About generic ChangeEventArgs	5
3 Namespace Index	7
3.1 Packages	7
4 Hierarchical Index	9
4.1 Class Hierarchy	9
5 Class Index	11
5.1 Class List	11
6 Namespace Documentation	13
6.1 ahbsd Namespace Reference	13
6.2 ahbsd.lib Namespace Reference	13
6.2.1 Function Documentation	13
6.2.1.1 ChangeEventHandler< T >()	13
6.3 ahbsd.lib.ApiKey Namespace Reference	14
6.3.1 Function Documentation	14
6.3.1.1 ApiKeyEventHandler< T >()	14
6.4 ahbsd.lib.Exceptions Namespace Reference	15
6.5 ahbsd.lib.NamedCollections Namespace Reference	15
6.6 ahbsd.lib.Password Namespace Reference	15
6.6.1 Enumeration Type Documentation	16
6.6.1.1 Charasteristic	16
6.7 ahbsd.lib.Password.Check Namespace Reference	16
6.8 ahbsd.lib.Products Namespace Reference	16
6.8.1 Enumeration Type Documentation	17
6.8.1.1 AdressType	17
6.9 ahbsd.lib.Tools Namespace Reference	17
6.10 Test_xUnit Namespace Reference	17
7 Class Documentation	19
7.1 ahbsd.lib.ApiKey.ApiKeyEventArgs< T > Class Template Reference	19
7.1.1 Detailed Description	19
7.1.2 Constructor & Destructor Documentation	20
7.1.2.1 ApiKeyEventArgs() [1/2]	20
7.1.2.2 ApiKeyEventArgs() [2/2]	20
7.1.3 Property Documentation	20
7.1.3.1 Index	20
7.2 ahbsd.lib.ApiKey.ApiKeyHolder< T > Class Template Reference	21

7.2.1 Detailed Description	22
7.2.2 Constructor & Destructor Documentation	23
7.2.2.1 ApiKeyHolder() [1/2]	23
7.2.2.2 ApiKeyHolder() [2/2]	23
7.2.3 Member Function Documentation	23
7.2.3.1 Equals() [1/2]	23
7.2.3.2 Equals() [2/2]	24
7.2.3.3 FindApiKey()	24
7.2.3.4 GetApiKey()	25
7.2.3.5 GetHashCode()	25
7.2.3.6 operator!=(())	25
7.2.3.7 operator==(())	26
7.2.4 Member Data Documentation	26
7.2.4.1 KnownApiKeys	26
7.2.5 Property Documentation	26
7.2.5.1 ApiKey	26
7.2.6 Event Documentation	27
7.2.6.1 OnApiKeyAdded	27
7.3 ahbsd.lib.ChangeEventArgs< T > Class Template Reference	27
7.3.1 Detailed Description	28
7.3.2 Constructor & Destructor Documentation	28
7.3.2.1 ChangeEventArgs() [1/3]	28
7.3.2.2 ChangeEventArgs() [2/3]	28
7.3.2.3 ChangeEventArgs() [3/3]	29
7.3.3 Member Function Documentation	29
7.3.3.1 Equals() [1/2]	29
7.3.3.2 Equals() [2/2]	30
7.3.3.3 GetHashCode()	30
7.3.3.4 operator!=(())	30
7.3.3.5 operator==(())	31
7.3.3.6 SetNewValue()	31
7.3.3.7 ToString()	31
7.3.4 Property Documentation	32
7.3.4.1 NewValue	32
7.3.4.2 OldValue	32
7.4 ahbsd.lib.Password.CharacteristicDictionary Class Reference	32
7.4.1 Detailed Description	34
7.4.2 Constructor & Destructor Documentation	34
7.4.2.1 CharacteristicDictionary() [1/6]	34
7.4.2.2 CharacteristicDictionary() [2/6]	34
7.4.2.3 CharacteristicDictionary() [3/6]	34
7.4.2.4 CharacteristicDictionary() [4/6]	35

7.4.2.5 CharacteristicDictionary() [5/6]	35
7.4.2.6 CharacteristicDictionary() [6/6]	35
7.4.3 Member Function Documentation	35
7.4.3.1 Add() [1/2]	36
7.4.3.2 Add() [2/2]	36
7.4.3.3 Clear()	36
7.4.3.4 Contains()	36
7.4.3.5 ContainsKey()	37
7.4.3.6 CopyTo()	37
7.4.3.7 GetEnumerator()	37
7.4.3.8 Remove() [1/2]	37
7.4.3.9 Remove() [2/2]	38
7.4.3.10 ToString()	38
7.4.3.11 TryGetValue()	38
7.4.4 Property Documentation	39
7.4.4.1 Count	39
7.4.4.2 IsReadOnly	39
7.4.4.3 Keys	39
7.4.4.4 Name	39
7.4.4.5 this[Charasteristic key]	39
7.4.4.6 Values	40
7.5 ahbsd.lib.Tools.Checksum Class Reference	40
7.5.1 Detailed Description	41
7.5.2 Member Function Documentation	41
7.5.2.1 GetChecksum() [1/4]	41
7.5.2.2 GetChecksum() [2/4]	41
7.5.2.3 GetChecksum() [3/4]	42
7.5.2.4 GetChecksum() [4/4]	42
7.5.2.5 GetFinalChecksum() [1/4]	42
7.5.2.6 GetFinalChecksum() [2/4]	43
7.5.2.7 GetFinalChecksum() [3/4]	43
7.5.2.8 GetFinalChecksum() [4/4]	43
7.5.2.9 IsPositiv() [1/3]	44
7.5.2.10 IsPositiv() [2/3]	44
7.5.2.11 IsPositiv() [3/3]	45
7.5.2.12 IsPositive() [1/3]	45
7.5.2.13 IsPositive() [2/3]	45
7.5.2.14 IsPositive() [3/3]	46
7.6 ahbsd.lib.NamedCollections.CompanyModelProducts Class Reference	46
7.6.1 Detailed Description	47
7.6.2 Constructor & Destructor Documentation	47
7.6.2.1 CompanyModelProducts()	47

7.6.3 Member Function Documentation	48
7.6.3.1 Add() [1/3]	48
7.6.3.2 Add() [2/3]	49
7.6.3.3 Add() [3/3]	49
7.6.3.4 Clear()	49
7.6.3.5 Contains()	50
7.6.3.6 ContainsKey()	50
7.6.3.7 CopyTo()	50
7.6.3.8 GetEnumerator()	51
7.6.3.9 Remove() [1/2]	51
7.6.3.10 Remove() [2/2]	51
7.6.3.11 TryGetValue()	52
7.6.4 Property Documentation	52
7.6.4.1 Count	52
7.6.4.2 IsReadOnly	52
7.6.4.3 Keys	53
7.6.4.4 this[ICompany key]	53
7.6.4.5 Values	53
7.7 ahbsd.lib.Tools.ConsolePrintTable Class Reference	53
7.7.1 Detailed Description	54
7.7.2 Member Function Documentation	54
7.7.2.1 Print()	54
7.8 ahbsd.lib.NamedCollections.DictionaryOfNamedCollection< K, V > Class Template Reference	54
7.8.1 Detailed Description	55
7.8.2 Member Function Documentation	55
7.8.2.1 Add() [1/3]	55
7.8.2.2 Add() [2/3]	55
7.8.2.3 Add() [3/3]	56
7.8.3 Event Documentation	56
7.8.3.1 OnNamedCollectionAdded	57
7.9 ahbsd.lib.NamedCollections.DictionaryOfNamedList< K, V > Class Template Reference	57
7.9.1 Detailed Description	57
7.9.2 Member Function Documentation	58
7.9.2.1 Add() [1/3]	58
7.9.2.2 Add() [2/3]	58
7.9.2.3 Add() [3/3]	59
7.9.3 Event Documentation	59
7.9.3.1 OnNamedListAdded	59
7.10 ahbsd.lib.EventArgs< T > Class Template Reference	60
7.10.1 Detailed Description	60
7.10.2 Constructor & Destructor Documentation	60
7.10.2.1 EventArgs() [1/2]	60

7.10.2.2 EventArgs() [2/2]	61
7.10.3 Property Documentation	61
7.10.3.1 Value	61
7.11 ahbsd.lib.Exceptions.Exception< T > Class Template Reference	61
7.11.1 Detailed Description	62
7.11.2 Constructor & Destructor Documentation	62
7.11.2.1 Exception()	62
7.11.3 Property Documentation	63
7.11.3.1 Value	63
7.12 ahbsd.lib.Products.IAddress Interface Reference	63
7.12.1 Detailed Description	64
7.12.2 Property Documentation	64
7.12.2.1 Address	64
7.12.2.2 AddressType	64
7.12.2.3 City	65
7.12.2.4 Country	65
7.12.2.5 ZIP	65
7.12.3 Event Documentation	65
7.12.3.1 OnAddressChanged	65
7.12.3.2 OnTypeChanged	65
7.13 ahbsd.lib.ApiKey.IApiKeyEventArgs Interface Reference	66
7.13.1 Detailed Description	66
7.13.2 Property Documentation	66
7.13.2.1 Index	66
7.14 ahbsd.lib.IChangeEventArgs< T > Interface Template Reference	66
7.14.1 Detailed Description	67
7.14.2 Member Function Documentation	67
7.14.2.1 SetNewValue()	67
7.14.2.2 ToString()	68
7.14.3 Property Documentation	68
7.14.3.1 NewValue	68
7.14.3.2 OldValue	68
7.15 ahbsd.lib.Password.ICharacteristicDictionary Interface Reference	68
7.15.1 Detailed Description	69
7.15.2 Member Function Documentation	69
7.15.2.1 ToString()	69
7.15.3 Property Documentation	69
7.15.3.1 Name	69
7.16 ahbsd.lib.Products.ICompany Interface Reference	70
7.16.1 Detailed Description	70
7.16.2 Member Function Documentation	70
7.16.2.1 AddAddress()	70

7.16.2.2 RemoveAdress()	71
7.16.2.3 RemoveAdresses()	71
7.16.3 Property Documentation	71
7.16.3.1 Adresses	72
7.16.3.2 Name	72
7.16.3.3 Note	72
7.16.4 Event Documentation	72
7.16.4.1 OnAdressAdded	72
7.16.4.2 OnAdressRemoved	72
7.16.4.3 OnNoteChanged	73
7.17 ahbsd.lib.NamedCollections.IDictionaryOfNamedCollections< K, V > Interface Template Reference	73
7.17.1 Detailed Description	73
7.17.2 Member Function Documentation	74
7.17.2.1 Add() [1/3]	74
7.17.2.2 Add() [2/3]	74
7.17.2.3 Add() [3/3]	75
7.17.3 Event Documentation	75
7.17.3.1 OnNamedCollectionAdded	75
7.18 ahbsd.lib.NamedCollections.IDictionaryOfNamedList< K, V > Interface Template Reference	75
7.18.1 Detailed Description	76
7.18.2 Member Function Documentation	76
7.18.2.1 Add() [1/3]	76
7.18.2.2 Add() [2/3]	77
7.18.2.3 Add() [3/3]	77
7.18.3 Event Documentation	78
7.18.3.1 OnNamedListAdded	78
7.19 ahbsd.lib.IEventArgs< T > Interface Template Reference	78
7.19.1 Detailed Description	78
7.19.2 Property Documentation	79
7.19.2.1 Value	79
7.20 ahbsd.lib.Exceptions.IGenericException< T > Interface Template Reference	79
7.20.1 Detailed Description	79
7.20.2 Property Documentation	80
7.20.2.1 Value	80
7.21 ahbsd.lib.Products.IModel Interface Reference	80
7.21.1 Detailed Description	80
7.21.2 Property Documentation	80
7.21.2.1 Name	81
7.21.2.2 Notes	81
7.21.2.3 Producer	81
7.21.3 Event Documentation	81
7.21.3.1 OnNotesChanged	81

7.22 ahbsd.lib.NamedCollections.INamedCollection< T > Interface Template Reference	81
7.22.1 Detailed Description	82
7.22.2 Member Function Documentation	82
7.22.2.1 ToString()	82
7.22.3 Property Documentation	82
7.22.3.1 Name	83
7.22.4 Event Documentation	83
7.22.4.1 OnNameChanged	83
7.23 ahbsd.lib.NamedCollections.INamedList< T > Interface Template Reference	83
7.23.1 Detailed Description	83
7.23.2 Member Function Documentation	84
7.23.2.1 ToString()	84
7.23.3 Property Documentation	84
7.23.3.1 Name	84
7.23.4 Event Documentation	84
7.23.4.1 OnNameChanged	84
7.24 ahbsd.lib.Password.IPassword Interface Reference	85
7.24.1 Detailed Description	85
7.24.2 Property Documentation	85
7.24.2.1 Characteristics	86
7.24.2.2 Length	86
7.24.2.3 LowerCases	86
7.24.2.4 Numbers	86
7.24.2.5 SecurityValue	86
7.24.2.6 Spaces	86
7.24.2.7 Specials	87
7.24.2.8 UpperCases	87
7.24.2.9 Value	87
7.24.3 Event Documentation	87
7.24.3.1 OnChange	87
7.25 ahbsd.lib.Products.IProductItem Interface Reference	87
7.25.1 Detailed Description	88
7.25.2 Property Documentation	88
7.25.2.1 Model	88
7.25.2.2 Name	88
7.25.2.3 SerialNumber	88
7.26 ahbsd.lib.Password.Check.ISecurityValue Interface Reference	89
7.26.1 Detailed Description	89
7.26.2 Property Documentation	89
7.26.2.1 Password	89
7.26.2.2 Security	89
7.27 Test_xUnit.ITestClass< T, A > Interface Template Reference	90

7.27.1 Detailed Description	90
7.27.2 Property Documentation	90
7.27.2.1 Variable	90
7.27.3 Event Documentation	90
7.27.3.1 OnChange	91
7.28 ahbsd.lib.NamedCollections.NamedCollection< T > Class Template Reference	91
7.28.1 Detailed Description	91
7.28.2 Constructor & Destructor Documentation	92
7.28.2.1 NamedCollection() [1/4]	92
7.28.2.2 NamedCollection() [2/4]	92
7.28.2.3 NamedCollection() [3/4]	92
7.28.2.4 NamedCollection() [4/4]	92
7.28.3 Member Function Documentation	93
7.28.3.1 ToString()	93
7.28.4 Property Documentation	93
7.28.4.1 Name	93
7.28.5 Event Documentation	93
7.28.5.1 OnNameChanged	94
7.29 ahbsd.lib.NamedCollections.NamedList< T > Class Template Reference	94
7.29.1 Detailed Description	94
7.29.2 Constructor & Destructor Documentation	95
7.29.2.1 NamedList() [1/6]	95
7.29.2.2 NamedList() [2/6]	95
7.29.2.3 NamedList() [3/6]	95
7.29.2.4 NamedList() [4/6]	96
7.29.2.5 NamedList() [5/6]	96
7.29.2.6 NamedList() [6/6]	96
7.29.3 Member Function Documentation	97
7.29.3.1 ToString()	97
7.29.4 Property Documentation	97
7.29.4.1 Name	97
7.29.5 Event Documentation	97
7.29.5.1 OnNameChanged	97
7.30 ahbsd.lib.Password.Password Class Reference	98
7.30.1 Detailed Description	99
7.30.2 Constructor & Destructor Documentation	99
7.30.2.1 Password() [1/4]	99
7.30.2.2 Password() [2/4]	99
7.30.2.3 Password() [3/4]	100
7.30.2.4 Password() [4/4]	100
7.30.3 Member Function Documentation	100
7.30.3.1 Equals() [1/2]	100

7.30.3.2 Equals() [2/2]	101
7.30.3.3 GetCharasteristic()	101
7.30.3.4 GetCharasteristics()	102
7.30.3.5 GetHashCode()	102
7.30.3.6 GetLowerCases()	102
7.30.3.7 GetNumbers()	103
7.30.3.8 GetPassword() [1/2]	103
7.30.3.9 GetPassword() [2/2]	103
7.30.3.10 GetSpaces()	104
7.30.3.11 GetSpecials()	104
7.30.3.12 GetUpperCases()	104
7.30.3.13 operator"!=()	105
7.30.3.14 operator==(())	105
7.30.4 Property Documentation	106
7.30.4.1 Characteristics	106
7.30.4.2 Length	106
7.30.4.3 LowerCases	106
7.30.4.4 Numbers	106
7.30.4.5 SecurityValue	106
7.30.4.6 Spaces	107
7.30.4.7 Specials	107
7.30.4.8 UpperCases	107
7.30.4.9 Value	107
7.30.5 Event Documentation	107
7.30.5.1 OnChange	107
7.31 ahbsd.lib.Tools.PrimeCheck Class Reference	108
7.31.1 Detailed Description	108
7.31.2 Member Function Documentation	108
7.31.2.1 Check()	108
7.31.3 Property Documentation	108
7.31.3.1 Amount	109
7.31.3.2 Time	109
7.32 ahbsd.lib.Password.Check.SecurityValue Class Reference	109
7.32.1 Detailed Description	110
7.32.2 Constructor & Destructor Documentation	110
7.32.2.1 SecurityValue() [1/6]	110
7.32.2.2 SecurityValue() [2/6]	110
7.32.2.3 SecurityValue() [3/6]	111
7.32.2.4 SecurityValue() [4/6]	111
7.32.2.5 SecurityValue() [5/6]	111
7.32.2.6 SecurityValue() [6/6]	111
7.32.3 Member Function Documentation	112

7.32.3.1 Equals() [1/2]	112
7.32.3.2 Equals() [2/2]	112
7.32.3.3 GetHashCode()	112
7.32.3.4 operator!=(())	113
7.32.3.5 operator==(())	113
7.32.3.6 ToString()	113
7.32.4 Property Documentation	114
7.32.4.1 Password	114
7.32.4.2 Security	114
7.33 Test_xUnit.TestClass< T, A > Class Template Reference	114
7.33.1 Detailed Description	115
7.33.2 Constructor & Destructor Documentation	115
7.33.2.1 TestClass() [1/4]	115
7.33.2.2 TestClass() [2/4]	115
7.33.2.3 TestClass() [3/4]	116
7.33.2.4 TestClass() [4/4]	116
7.33.3 Property Documentation	116
7.33.3.1 Variable	116
7.33.4 Event Documentation	116
7.33.4.1 OnChange	117
Index	119

Chapter 1

LICENSE

Apache License Version 2.0, January 2004 <http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. **Grant of Copyright License.** Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. **Grant of Patent License.** Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. **Redistribution.** You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.
5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

Chapter 2

ahbsd.lib

Classes with functionality I miss

- For example generic EventArgs or EventArgs for changing values. These two are in version 1.0.
- In Version 1.0.1 some classes for generic API-Keys were added.
- In Version 1.2.0 some code beautification was done, the license changes to Apache 2.0 and a generic Exception was added.

2.1 About generic EventArgs

The generic EventArgs have a generic Value that could be set with the constructor.

2.2 About generic ChangeEventArgs

The generic ChangeEventArgs have two generic values. One for the old value and one for the new value.

Chapter 3

Namespace Index

3.1 Packages

Here are the packages with brief descriptions (if available):

ahbsd	13
ahbsd.lib	13
ahbsd.lib.ApiKey	14
ahbsd.lib.Exceptions	15
ahbsd.lib.NamedCollections	15
ahbsd.lib.Password	15
ahbsd.lib.Password.Check	16
ahbsd.lib.Products	16
ahbsd.lib.Tools	17
Test_xUnit	17

Chapter 4

Hierarchical Index

4.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

ahbsd.lib.ApiKey.ApiKeyHolder< T >	21
ahbsd.lib.ApiKey.ApiKeyHolder< A >	21
Test_xUnit.TestClass< T, A >	114
ahbsd.lib.Tools.Checksum	40
ahbsd.lib.NamedCollections.CompanyModelProducts	46
ahbsd.lib.Tools.ConsolePrintTable	53
ahbsd.lib.Products.IAdress	63
ahbsd.lib.ApiKey.IApiKeyEventArgs	66
ahbsd.lib.ApiKey.ApiKeyEventArgs< T >	19
ahbsd.lib.IChangeEventArgs< T >	66
ahbsd.lib.ChangeEventArgs< T >	27
ahbsd.lib.Password.ICharacteristicDictionary	68
ahbsd.lib.Password.CharacteristicDictionary	32
ahbsd.lib.Products.ICompany	70
ahbsd.lib.NamedCollections.IDictionaryOfNamedCollections< K, V >	73
ahbsd.lib.NamedCollections.DictionaryOfNamedCollection< K, V >	54
ahbsd.lib.NamedCollections.IDictionaryOfNamedList< K, V >	75
ahbsd.lib.NamedCollections.DictionaryOfNamedList< K, V >	57
ahbsd.lib.IEventArgs< T >	78
ahbsd.lib.EventArgs< T >	60
ahbsd.lib.ApiKey.ApiKeyEventArgs< T >	19
ahbsd.lib.ChangeEventArgs< T >	27
ahbsd.lib.Exceptions.IGenericException< T >	79
ahbsd.lib.Exceptions.Exception< T >	61
ahbsd.lib.Products.IModel	80
ahbsd.lib.NamedCollections.INamedCollection< T >	81
ahbsd.lib.NamedCollections.NamedCollection< T >	91
ahbsd.lib.NamedCollections.INamedList< T >	83
ahbsd.lib.NamedCollections.NamedList< T >	94
ahbsd.lib.Password.IPassword	85
ahbsd.lib.Password.Password	98
ahbsd.lib.Products.IProductItem	87

ahbsd.lib.Password.Check.ISecurityValue	89
ahbsd.lib.Password.Check.SecurityValue	109
Test_xUnit.ITestClass< T, A >	90
Test_xUnit.TestClass< T, A >	114
ahbsd.lib.Tools.PrimeCheck	108

Chapter 5

Class Index

5.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

ahbsd.lib.ApiKey.ApiKeyEventArgs< T >	19
Specialized generic EventArgs for API-Keys	
ahbsd.lib.ApiKey.ApiKeyHolder< T >	21
Class for generic API-Keys	
ahbsd.lib.ChangeEventArgs< T >	27
Generic EventArgs for changing values	
ahbsd.lib.Password.CharacteristicDictionary	32
Characteristic Dictionary	
ahbsd.lib.Tools.Checksum	40
Class to calculate checksums	
ahbsd.lib.NamedCollections.CompanyModelProducts	46
Class for a combination of difined products of a company	
ahbsd.lib.Tools.ConsolePrintTable	53
Static class to print a DataTable to console	
ahbsd.lib.NamedCollections.DictionaryOfNamedCollection< K, V >	54
Default implementation of IDictionaryOfNamedCollections<K, V>	
ahbsd.lib.NamedCollections.DictionaryOfNamedList< K, V >	57
Default implementation of IDictionaryOfNamedList<K, V>	
ahbsd.lib.EventArgs< T >	60
Generic EventArgs	
ahbsd.lib.Exceptions.Exception< T >	61
Class for a generic Exception , which additionally holds a value of T	
ahbsd.lib.Products.IAddress	63
An interface describing an adress in general	
ahbsd.lib.ApiKey.IApiKeyEventArgs	66
Interface for ApiKeyEventArgs<T>	
ahbsd.lib.IChangeEventArgs< T >	66
Interface for generic EventArgs for changing values	
ahbsd.lib.Password.ICharacteristicDictionary	68
An interface for a characteristic dictionary	
ahbsd.lib.Products.ICompany	70
An interface, describing a company / manufacturer	
ahbsd.lib.NamedCollections.IDictionaryOfNamedCollections< K, V >	73
Interface for a dictionary of named collections as value	
ahbsd.lib.NamedCollections.IDictionaryOfNamedList< K, V >	75
Interface for a collection of INamedList<T>	

ahbsd.lib.IEventArgs< T >	
Interface for generic EventArgs	78
ahbsd.lib.Exceptions.IGenericException< T >	
Interface for generic exceptions	79
ahbsd.lib.Products.IModel	
Interface of an model of a product	80
ahbsd.lib.NamedCollections.INamedCollection< T >	
Interface for a named collection	81
ahbsd.lib.NamedCollections.INamedList< T >	
Interface for a named list	83
ahbsd.lib.Password.IPassword	
An interface for a password object in general	85
ahbsd.lib.Products.IProductItem	
Interface of an Product	87
ahbsd.lib.Password.Check.ISecurityValue	
An Interface for getting the security value of a given IPassword	89
Test_xUnit.ITestClass< T, A >	
An interface for a class to demonstrate ChangeEventArgs<T> and ChangeEventHandler<T>	90
ahbsd.lib.NamedCollections.NamedCollection< T >	
Default implementation of INamedCollection<T>	91
ahbsd.lib.NamedCollections.NamedList< T >	
Default implementation of INamedList<T>	94
ahbsd.lib.Password.Password	
A Password class	98
ahbsd.lib.Tools.PrimeCheck	
Class for checking, wheather a given number is a prime number or not	108
ahbsd.lib.Password.Check.SecurityValue	
Component for the Security value of an password	109
Test_xUnit.TestClass< T, A >	
A class to demonstrate ChangeEventArgs<T> and ChangeEventHandler<T>	114

Chapter 6

Namespace Documentation

6.1 ahbsd Namespace Reference

6.2 ahbsd.lib Namespace Reference

Classes

- class [ChangeEventArgs](#)
Generic [EventArgs](#) for changing values.
- class [EventArgs](#)
Generic [EventArgs](#).
- interface [IChangeEventArgs](#)
Interface for generic [EventArgs](#) for changing values.
- interface [IEventArgs](#)
Interface for generic [EventArgs](#).

Functions

- delegate void [ChangeEventHandler](#)< T > (object sender, [ChangeEventArgs](#)< T > e)
A delegate for change events.

6.2.1 Function Documentation

6.2.1.1 [ChangeEventHandler](#)< T >()

```
delegate void ahbsd.lib.ChangeEventHandler< T > (
    object sender,
    ChangeEventArgs< T > e )
```

A delegate for change events.

Template Parameters

<i>T</i>	The type of changing values.
----------	------------------------------

Parameters

<i>sender</i>	Sending object.
<i>e</i>	The changing EventArgs .

6.3 ahbsd.lib.ApiKey Namespace Reference

Classes

- class [ApiKeyEventArgs](#)
Specialized generic [EventArgs](#) for API-Keys.
- class [ApiKeyHolder](#)
Class for generic API-Keys.
- interface [IApiKeyEventArgs](#)
Interface for [ApiKeyEventArgs](#)< T >.

Functions

- delegate void [ApiKeyEventHandler](#)< T > (object sender, [ApiKeyEventArgs](#)< T > e)
Delegate for events with generic API-Keys.

6.3.1 Function Documentation

6.3.1.1 ApiKeyEventHandler< T >()

```
delegate void ahbsd.lib.ApiKey.ApiKeyEventHandler< T > (
    object sender,
    ApiKeyEventArgs< T > e )
```

Delegate for events with generic API-Keys.

Template Parameters

<i>T</i>	The Type of the API-Key.
----------	--------------------------

Parameters

<i>sender</i>	The sending object.
---------------	---------------------

Parameters

<i>e</i>	The event arguments with a generic API-Key.
----------	---

6.4 ahbsd.lib.Exceptions Namespace Reference

Classes

- class [Exception](#)
*Class for a generic [Exception](#), which additionally holds a value of *T**
- interface [IGenericException](#)
Interface for generic exceptions.

6.5 ahbsd.lib.NamedCollections Namespace Reference

Classes

- class [CompanyModelProducts](#)
Class for a combination of difined products of a company.
- class [DictionaryOfNamedCollection](#)
*Default implementation of *IDictionaryOfNamedCollections*<*K*, *V*>.*
- class [DictionaryOfNamedList](#)
*Default implementation of *IDictionaryOfNamedList*<*K*, *V*>.*
- interface [IDictionaryOfNamedCollections](#)
Interface for a dictionary of named collections as value.
- interface [IDictionaryOfNamedList](#)
*Interface for a collection of *INamedList*<*T*>.*
- interface [INamedCollection](#)
Interface for a named collection.
- interface [INamedList](#)
Interface for a named list.
- class [NamedCollection](#)
*Default implementation of *INamedCollection*<*T*>.*
- class [NamedList](#)
*Default implementation of *INamedList*<*T*>.*

6.6 ahbsd.lib.Password Namespace Reference

Classes

- class [CharacteristicDictionary](#)
Characteristic Dictionary.
- interface [ICharacteristicDictionary](#)
An interface for a characteristic dictionary.
- interface [IPassword](#)
An interface for a password object in general.
- class [Password](#)
A [Password](#) class.

Enumerations

- enum class [Charasteristic](#) {
[Numeric](#) , [LowercaseLetter](#) , [CapitalLetter](#) , [SpecialCharacter](#) ,
[Space](#) }

The charasteristic of a password.

6.6.1 Enumeration Type Documentation

6.6.1.1 Charasteristic

```
enum ahbsd.lib.Password.Charasteristic [strong]
```

The charasteristic of a password.

Enumerator

Numeric	A numeric letter.
LowercaseLetter	A lowercase letter.
CapitalLetter	A capital letter.
SpecialCharacter	A special character.
Space	A space.

6.7 [ahbsd.lib.Password.Check](#) Namespace Reference

Classes

- interface [ISecurityValue](#)
An Interface for getting the security value of a given [IPassword](#).
- class [SecurityValue](#)
Component for the Security value of an password.

6.8 [ahbsd.lib.Products](#) Namespace Reference

Classes

- interface [IAdress](#)
An interface describing an adress in general.
- interface [ICompany](#)
An interface, describing a company / manufacturer.
- interface [IModel](#)
Interface of an model of a product.
- interface [IProductItem](#)
Interface of an Product.

Enumerations

- enum class [AdressType](#) {
[Postal](#) , [Delivery](#) , [Store](#) , [Private](#) ,
[Other](#) }

An enum for the type of an Adress.

6.8.1 Enumeration Type Documentation

6.8.1.1 AdressType

```
enum ahbsd.lib.Products.AdressType [strong]
```

An enum for the type of an Adress.

Enumerator

Postal	The postal address.
Delivery	The delivery address.
Store	The store address.
Private	The private address.
Other	An other address. If no other type matches.

6.9 ahbsd.lib.Tools Namespace Reference

Classes

- class [Checksum](#)
Class to calculate checksums.
- class [ConsolePrintTable](#)
Static class to print a DataTable to console.
- class [PrimeCheck](#)
Class for checking, wheather a given number is a prime number or not.

6.10 Test_xUnit Namespace Reference

Classes

- interface [ITestClass](#)
An interface for a class to demonstrate ChangeEventArgs<T> and ChangeEventHandler<T>.
- class [TestClass](#)
A class to demonstrate ChangeEventArgs<T> and ChangeEventHandler<T>.

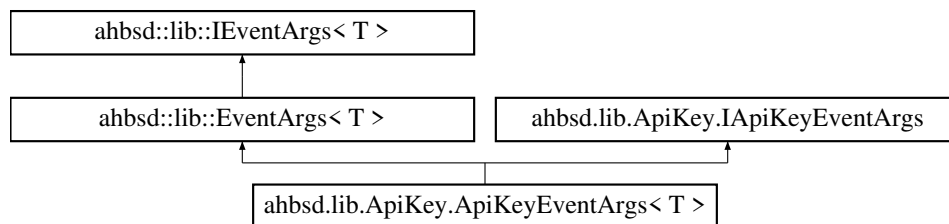
Chapter 7

Class Documentation

7.1 ahbsd.lib.ApiKey.ApiKeyEventArgs< T > Class Template Reference

Specialized generic [EventArgs](#) for API-Keys.

Inheritance diagram for ahbsd.lib.ApiKey.ApiKeyEventArgs< T >:



Public Member Functions

- [ApiKeyEventArgs](#) (T apiKey, int idx)
Constructor with the API-Key and the index of the API-Key.
- [ApiKeyEventArgs](#) (T apiKey)
Constructor with the API-Key.

Properties

- int? [Index](#) [get]
Gets the index of the API-Key from the `ApiKeyHolder< T >`.

7.1.1 Detailed Description

Specialized generic [EventArgs](#) for API-Keys.

Template Parameters

<i>T</i>	Type of API-Key.
----------	------------------

See also

ApiKeyHolder<T>

7.1.2 Constructor & Destructor Documentation

7.1.2.1 ApiKeyEventArgs() [1/2]

```
ahbsd.lib.ApiKey.ApiKeyEventArgs< T >.ApiKeyEventArgs (
    T apiKey,
    int idx )
```

Constructor with the API-Key and the index of the API-Key.

Parameters

<i>apiKey</i>	The API-Key.
<i>idx</i>	The index of the API-Key.

7.1.2.2 ApiKeyEventArgs() [2/2]

```
ahbsd.lib.ApiKey.ApiKeyEventArgs< T >.ApiKeyEventArgs (
    T apiKey )
```

Constructor with the API-Key.

Parameters

<i>apiKey</i>	The API-Key.
---------------	--------------

7.1.3 Property Documentation

7.1.3.1 Index

```
int? ahbsd.lib.ApiKey.ApiKeyEventArgs< T >.Index [get]
```


Gets the index of the API-Key from the ApiKeyHolder<T>.

The index.

The documentation for this class was generated from the following file:

- ahbsd.lib/ApiKey/ApiKeyEventArgs.cs

7.2 ahbsd.lib.ApiKey.ApiKeyHolder< T > Class Template Reference

Class for generic API-Keys.

Inherits IEquatable< ApiKeyHolder< T >>.

Public Member Functions

- [ApiKeyHolder](#) (T apiKey)
Constructor with a given API-Key.
- [ApiKeyHolder](#) ()
Constructor without parameters.
- override bool [Equals](#) (object obj)
Find out, if this object equals another given object.
- override int [GetHashCode](#) ()
Gets the HashCode.
- bool [Equals](#) (ApiKeyHolder< T > other)
Find out, if this object equals another given object.

Static Public Member Functions

- static ? int [FindApiKey](#) (T apiKey)
Looks for a given API-Key.
- static T [GetApiKey](#) (int idx)
Returns an API-Key from a defined index number.
- static bool [operator==](#) (ApiKeyHolder< T > left, ApiKeyHolder< T > right)
Find out if two objects equals.
- static bool [operator!=](#) (ApiKeyHolder< T > left, ApiKeyHolder< T > right)
Find out if two objects do not equals.

Static Package Attributes

- static List< T > [KnownApiKeys](#)
A list of all known API-Keys.

Properties

- T [ApiKey](#) [get]
Gets the API-Key.

Events

- static ApiKeyEventHandler< T > [OnApiKeyAdded](#)
Happens if a new API-Key was added to the static list [KnownApiKeys](#).

7.2.1 Detailed Description

Class for generic API-Keys.

Template Parameters

<i>T</i>	Type of API-Key
----------	-----------------

7.2.2 Constructor & Destructor Documentation

7.2.2.1 ApiKeyHolder() [1/2]

```
ahbsd.lib.ApiKey.ApiKeyHolder< T >.ApiKeyHolder (
    T apiKey )
```

Constructor with a given API-Key.

Parameters

<i>apiKey</i>	The API-Key.
---------------	--------------

7.2.2.2 ApiKeyHolder() [2/2]

```
ahbsd.lib.ApiKey.ApiKeyHolder< T >.ApiKeyHolder ( )
```

Constructor without parameters.

If before an object was created, the last API-Key will be used. Otherwise the default of *T* will be used.

Exceptions

<i>ArgumentNullException</i>	If KnownApiKeys is <code>null</code> or something similar.
<i>InvalidOperationException</i>	If anything regarding KnownApiKeys is an invalid operation.

References [ahbsd.lib.ApiKey.ApiKeyHolder< T >.ApiKey](#), and [ahbsd.lib.ApiKey.ApiKeyHolder< T >.KnownApiKeys](#).

7.2.3 Member Function Documentation

7.2.3.1 Equals() [1/2]

```
bool ahbsd.lib.ApiKey.ApiKeyHolder< T >.Equals (
    ApiKeyHolder< T > other )
```

Find out, if this object equals another given object.

Parameters

<i>other</i>	The other object.
--------------	-------------------

Returns

If both objects equals `TRUE`, otherwise `FALSE`.

References [ahbsd.lib.ApiKey.ApiKeyHolder< T >.ApiKey](#), and [ahbsd.lib.ApiKey.ApiKeyHolder< T >.Equals\(\)](#).

7.2.3.2 Equals() [2/2]

```
override bool ahbsd.lib.ApiKey.ApiKeyHolder< T >.Equals (
    object obj )
```

Find out, if this object equals another given object.

Parameters

<i>obj</i>	The other object.
------------	-------------------

Returns

If both objects equals `TRUE`, otherwise `FALSE`.

Referenced by [ahbsd.lib.ApiKey.ApiKeyHolder< T >.Equals\(\)](#).

7.2.3.3 FindApiKey()

```
static ? int ahbsd.lib.ApiKey.ApiKeyHolder< T >.FindApiKey (
    T apiKey ) [static]
```

Looks for a given API-Key.

Parameters

<i>apiKey</i>	The given API-Key.
---------------	--------------------

Returns

If found it returns the index, if not `null` is returned.

References [ahbsd.lib.ApiKey.ApiKeyHolder< T >.KnownApiKeys](#).

7.2.3.4 GetApiKey()

```
static T ahbsd.lib.ApiKey.ApiKeyHolder< T >.GetApiKey (
    int idx ) [static]
```

Returns an API-Key from a defined index number.

Parameters

<i>idx</i>	The defined index number.
------------	---------------------------

Returns

An API-Key.

References [ahbsd.lib.ApiKey.ApiKeyHolder< T >.KnownApiKeys](#).

7.2.3.5 GetHashCode()

```
override int ahbsd.lib.ApiKey.ApiKeyHolder< T >.GetHashCode ( )
```

Gets the HashCode.

Returns

The HashCode.

References [ahbsd.lib.ApiKey.ApiKeyHolder< T >.ApiKey](#).

7.2.3.6 operator"!="()

```
static bool ahbsd.lib.ApiKey.ApiKeyHolder< T >.operator!= (
    ApiKeyHolder< T > left,
    ApiKeyHolder< T > right ) [static]
```

Find out if two objects do not equals.

Parameters

<i>left</i>	The object on the left side.
<i>right</i>	The object on the right side.

Returns

If both objects do not equals TRUE, otherwise FALSE.

7.2.3.7 operator==()

```
static bool ahbsd.lib.ApiKey.ApiKeyHolder< T >.operator== (
    ApiKeyHolder< T > left,
    ApiKeyHolder< T > right ) [static]
```

Find out if two objects equals.

Parameters

<i>left</i>	The object on the left side.
<i>right</i>	The object on the right side.

Returns

If both objects equals `TRUE`, otherwise `FALSE`.

7.2.4 Member Data Documentation

7.2.4.1 KnownApiKeys

```
List<T> ahbsd.lib.ApiKey.ApiKeyHolder< T >.KnownApiKeys [static], [package]
```

A list of all known API-Keys.

Of current instances. Is eg needed for construction without api-key etc.

Referenced by [ahbsd.lib.ApiKey.ApiKeyHolder< T >.ApiKeyHolder\(\)](#), [ahbsd.lib.ApiKey.ApiKeyHolder< T >.FindApiKey\(\)](#), and [ahbsd.lib.ApiKey.ApiKeyHolder< T >.GetApiKey\(\)](#).

7.2.5 Property Documentation

7.2.5.1 ApiKey

```
T ahbsd.lib.ApiKey.ApiKeyHolder< T >.ApiKey [get], [protected]
```

Gets the API-Key.

The API-Key.

Referenced by [ahbsd.lib.ApiKey.ApiKeyHolder< T >.ApiKeyHolder\(\)](#), [ahbsd.lib.ApiKey.ApiKeyHolder< T >.Equals\(\)](#), and [ahbsd.lib.ApiKey.ApiKeyHolder< T >.GetHashCode\(\)](#).

7.2.6 Event Documentation

7.2.6.1 OnApiKeyAdded

ApiKeyEventHandler<T> [ahbsd.lib.ApiKey.ApiKeyHolder](#)< T >.OnApiKeyAdded [static], [package]

Happens if a new API-Key was added to the static list [KnownApiKeys](#).

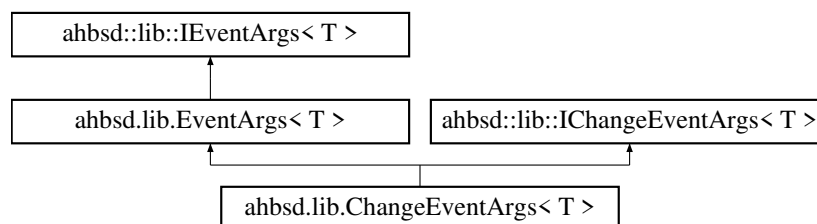
The documentation for this class was generated from the following file:

- ahbsd.lib/ApiKey/ApiKeyHolder.cs

7.3 ahbsd.lib.ChangeEventArgs< T > Class Template Reference

Generic [EventArgs](#) for changing values.

Inheritance diagram for ahbsd.lib.ChangeEventArgs< T >:



Public Member Functions

- [ChangeEventArgs](#) ()
Constructor without any parameters.
- [ChangeEventArgs](#) (T oldValue, T newValue)
Constructor with the old and the new value.
- [ChangeEventArgs](#) (T oldValue)
Constructor with the old value.
- void [SetNewValue](#) (T newValue)
Sets the new value.
- override string [ToString](#) ()
Gets a string representation of the changed value.
- bool [Equals](#) (IChangeEventArgs< T > other)
Finds out wheather an other object of type IChangeEventArgs< T > equals this object.
- override bool [Equals](#) (object obj)
Finds out wheather an other object equals this object.
- override int [GetHashCode](#) ()
Gets the HashCode of this object.

Static Public Member Functions

- static bool `operator==` (`ChangeEventArgs< T >` left, `ChangeEventArgs< T >` right)
Finds out, if two objects of type `ChangeEventArgs< T >` equals each other.
- static bool `operator!=` (`ChangeEventArgs< T >` left, `ChangeEventArgs< T >` right)
Finds out, if two objects of type `ChangeEventArgs< T >` do not equals each other.

Properties

- T `OldValue` [get]
Gets the old value.
- T `NewValue` [get]
Gets the new value.

7.3.1 Detailed Description

Generic `EventArgs` for changing values.

Template Parameters

<code>T</code>	The type of the changing Values.
----------------	----------------------------------

7.3.2 Constructor & Destructor Documentation

7.3.2.1 `ChangeEventArgs()` [1/3]

```
ahbsd.lib.ChangeEventArgs< T >.ChangeEventArgs ( )
```

Constructor without any parameters.

References `ahbsd.lib.ChangeEventArgs< T >.NewValue`, and `ahbsd.lib.ChangeEventArgs< T >.OldValue`.

7.3.2.2 `ChangeEventArgs()` [2/3]

```
ahbsd.lib.ChangeEventArgs< T >.ChangeEventArgs (
    T oldValue,
    T newValue )
```

Constructor with the old and the new value.

Parameters

<i>oldValue</i>	The old value.
<i>newValue</i>	The new value.

References [ahbsd.lib.ChangeEventArgs< T >.NewValue](#), and [ahbsd.lib.ChangeEventArgs< T >.OldValue](#).

7.3.2.3 ChangeEventArgs() [3/3]

```
ahbsd.lib.ChangeEventArgs< T >.ChangeEventArgs (
    T oldValue )
```

Constructor with the old value.

Parameters

<i>oldValue</i>	The old value.
-----------------	----------------

References [ahbsd.lib.ChangeEventArgs< T >.NewValue](#), and [ahbsd.lib.ChangeEventArgs< T >.OldValue](#).

7.3.3 Member Function Documentation

7.3.3.1 Equals() [1/2]

```
bool ahbsd.lib.ChangeEventArgs< T >.Equals (
    IChangeEventArgs< T > other )
```

Finds out wheather an other object of type IChangeEventArgs<T> equals this object.

Parameters

<i>other</i>	The other object.
--------------	-------------------

Returns

TRUE if the other object equals this object, otherwise FALSE.

References [ahbsd.lib.ChangeEventArgs< T >.NewValue](#), [ahbsd.lib.IChangeEventArgs< T >.NewValue](#), [ahbsd.lib.ChangeEventArgs< T >.OldValue](#), and [ahbsd.lib.IChangeEventArgs< T >.OldValue](#).

Referenced by [ahbsd.lib.ChangeEventArgs< T >.Equals\(\)](#).

7.3.3.2 Equals() [2/2]

```
override bool ahbsd.lib.ChangeEventArgs< T >.Equals (
    object obj )
```

Finds out wheather an other object equals this object.

Parameters

<i>obj</i>	The other object.
------------	-------------------

Returns

TRUE if the other object equals this object, otherwise FALSE.

References [ahbsd.lib.ChangeEventArgs< T >.Equals\(\)](#).

7.3.3.3 GetHashCode()

```
override int ahbsd.lib.ChangeEventArgs< T >.GetHashCode ( )
```

Gets the GetHashCode of this object.

Returns

The GetHashCode.

References [ahbsd.lib.ChangeEventArgs< T >.NewValue](#), and [ahbsd.lib.ChangeEventArgs< T >.OldValue](#).

7.3.3.4 operator"!=()"

```
static bool ahbsd.lib.ChangeEventArgs< T >.operator!= (
    ChangeEventArgs< T > left,
    ChangeEventArgs< T > right ) [static]
```

Finds out, if two objects of type [ChangeEventArgs<T>](#) do not eaquals each other.

Parameters

<i>left</i>	The object on the left side.
<i>right</i>	The object on the right side.

Returns

TRUE if both objects are not eaqual, otherwise FALSE.

7.3.3.5 operator==()

```
static bool ahbsd.lib.ChangeEventArgs< T >.operator== (
    ChangeEventArgs< T > left,
    ChangeEventArgs< T > right ) [static]
```

Finds out, if two objects of type [ChangeEventArgs<T>](#) equals each other.

Parameters

<i>left</i>	The object on the left side.
<i>right</i>	The object on the right side.

Returns

TRUE if both objects are equal, otherwise FALSE.

7.3.3.6 SetNewValue()

```
void ahbsd.lib.ChangeEventArgs< T >.SetNewValue (
    T newValue )
```

Sets the new value.

Parameters

<i>newValue</i>	The new value.
-----------------	----------------

Exceptions

<i>Exception</i>	If the NewValue was already set.
------------------	--

Implements [ahbsd.lib.IChangeEventArgs< T >](#).

References [ahbsd.lib.ChangeEventArgs< T >.NewValue](#).

7.3.3.7 ToString()

```
override string ahbsd.lib.ChangeEventArgs< T >.ToString ( )
```

Gets a string representation of the changed value.

Returns

A string representation of the changed value.

Implements [ahbsd.lib.IChangeEventArgs< T >](#).

References [ahbsd.lib.ChangeEventArgs< T >.NewValue](#), and [ahbsd.lib.ChangeEventArgs< T >.OldValue](#).

7.3.4 Property Documentation**7.3.4.1 NewValue**

```
T ahbsd.lib.ChangeEventArgs< T >.NewValue [get]
```

Gets the new value.

The new value.

Referenced by [ahbsd.lib.ChangeEventArgs< T >.ChangeEventArgs\(\)](#), [ahbsd.lib.ChangeEventArgs< T >.Equals\(\)](#), [ahbsd.lib.ChangeEventArgs< T >.GetHashCode\(\)](#), [ahbsd.lib.ChangeEventArgs< T >.SetNewValue\(\)](#), and [ahbsd.lib.ChangeEventArgs< T >.ToString\(\)](#).

7.3.4.2 OldValue

```
T ahbsd.lib.ChangeEventArgs< T >.OldValue [get]
```

Gets the old value.

The old value.

Referenced by [ahbsd.lib.ChangeEventArgs< T >.ChangeEventArgs\(\)](#), [ahbsd.lib.ChangeEventArgs< T >.Equals\(\)](#), [ahbsd.lib.ChangeEventArgs< T >.GetHashCode\(\)](#), and [ahbsd.lib.ChangeEventArgs< T >.ToString\(\)](#).

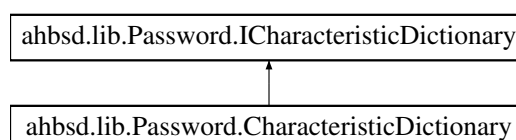
The documentation for this class was generated from the following file:

- [ahbsd.lib/ChangeEventArgs.cs](#)

7.4 ahbsd.lib.Password.CharacteristicDictionary Class Reference

Characteristic Dictionary.

Inheritance diagram for [ahbsd.lib.Password.CharacteristicDictionary](#):



Public Member Functions

- [CharacteristicDictionary](#) ()
Constructor without parameter.
- [CharacteristicDictionary](#) (IContainer container)
Constructor with a given owning container.
- void [Add](#) ([Charasteristic](#) key, bool value)
Throws an Exception.
- void [Add](#) (KeyValuePair< [Charasteristic](#), bool > item)
Throws an Exception.
- void [Clear](#) ()
Re-Initialize the object.
- bool [Contains](#) (KeyValuePair< [Charasteristic](#), bool > item)
- bool [ContainsKey](#) ([Charasteristic](#) key)
- void [CopyTo](#) (KeyValuePair< [Charasteristic](#), bool >[] array, int arrayIndex)
- IEnumerator< KeyValuePair< [Charasteristic](#), bool > > [GetEnumerator](#) ()
- bool [Remove](#) ([Charasteristic](#) key)
- bool [Remove](#) (KeyValuePair< [Charasteristic](#), bool > item)
- bool [TryGetValue](#) ([Charasteristic](#) key, [MaybeNullWhen(false)] out bool value)
- override string [ToString](#) ()
Gets a short info of the characteristic.

Package Functions

- [CharacteristicDictionary](#) (string value)
Constructor with a given password.
- [CharacteristicDictionary](#) (IPassword value)
Constructor with a given password.
- [CharacteristicDictionary](#) (string value, IContainer container)
Constructor with a given password and a given owning container.
- [CharacteristicDictionary](#) (IPassword value, IContainer container)
Constructor with a given password and a given owning container.

Properties

- bool [this\[Charasteristic key\]](#) [get, set]
Gets or sets the value of the given key.
- ICollection< [Charasteristic](#) > [Keys](#) [get]
Gets the Keys.
- ICollection< bool > [Values](#) [get]
Gets all values.
- int [Count](#) [get]
Gets the amount of KeyValuePair< TKey, TValue>s (5).
- bool [IsReadOnly](#) [get]
Gets the readability.
- string [Name](#) [get]
Gets the name of this component.

7.4.1 Detailed Description

Characteristic Dictionary.

7.4.2 Constructor & Destructor Documentation

7.4.2.1 CharacteristicDictionary() [1/6]

```
ahbsd.lib.Password.CharacteristicDictionary.CharacteristicDictionary ( )
```

Constructor without parameter.

7.4.2.2 CharacteristicDictionary() [2/6]

```
ahbsd.lib.Password.CharacteristicDictionary.CharacteristicDictionary (
    string value ) [package]
```

Constructor with a given password.

Parameters

<i>value</i>	The given password.
--------------	---------------------

References [ahbsd.lib.Password.CharacteristicDictionary.ToString\(\)](#).

7.4.2.3 CharacteristicDictionary() [3/6]

```
ahbsd.lib.Password.CharacteristicDictionary.CharacteristicDictionary (
    IPassword value ) [package]
```

Constructor with a given password.

Parameters

<i>value</i>	The given password.
--------------	---------------------

References [ahbsd.lib.Password.CharacteristicDictionary.ToString\(\)](#).

7.4.2.4 CharacteristicDictionary() [4/6]

```
ahbsd.lib.Password.CharacteristicDictionary.CharacteristicDictionary (
    IContainer container )
```

Constructor with a given owning container.

Parameters

<i>container</i>	The given owning container.
------------------	-----------------------------

7.4.2.5 CharacteristicDictionary() [5/6]

```
ahbsd.lib.Password.CharacteristicDictionary.CharacteristicDictionary (
    string value,
    IContainer container ) [package]
```

Constructor with a given password and a given owning container.

Parameters

<i>value</i>	The given password.
<i>container</i>	The given owning container.

References [ahbsd.lib.Password.CharacteristicDictionary.ToString\(\)](#).

7.4.2.6 CharacteristicDictionary() [6/6]

```
ahbsd.lib.Password.CharacteristicDictionary.CharacteristicDictionary (
    IPassword value,
    IContainer container ) [package]
```

Constructor with a given password and a given owning container.

Parameters

<i>value</i>	The given password.
<i>container</i>	The given owning container.

References [ahbsd.lib.Password.CharacteristicDictionary.ToString\(\)](#).

7.4.3 Member Function Documentation

7.4.3.1 Add() [1/2]

```
void ahbsd.lib.Password.CharacteristicDictionary.Add (
    Charasteristic key,
    bool value )
```

Throws an Exception.

Exceptions

<i>Exception</i>	Always, since no add is possible here.
------------------	--

7.4.3.2 Add() [2/2]

```
void ahbsd.lib.Password.CharacteristicDictionary.Add (
    KeyValuePair< Charasteristic, bool > item )
```

Throws an Exception.

Exceptions

<i>Exception</i>	Always, since no add is possible here.
------------------	--

7.4.3.3 Clear()

```
void ahbsd.lib.Password.CharacteristicDictionary.Clear ( )
```

Re-Initialize the object.

7.4.3.4 Contains()

```
bool ahbsd.lib.Password.CharacteristicDictionary.Contains (
    KeyValuePair< Charasteristic, bool > item )
```

Parameters

<i>item</i>	
-------------	--

Returns

7.4.3.5 ContainsKey()

```
bool ahbsd.lib.Password.CharacteristicDictionary.ContainsKey (
    Charasteristic key )
```

Parameters

<i>key</i>	
------------	--

Returns

7.4.3.6 CopyTo()

```
void ahbsd.lib.Password.CharacteristicDictionary.CopyTo (
    KeyValuePair< Charasteristic, bool >[] array,
    int arrayIndex )
```

Parameters

<i>array</i>	
<i>arrayIndex</i>	

7.4.3.7 GetEnumerator()

```
IEnumerator<KeyValuePair<Charasteristic, bool> > ahbsd.lib.Password.CharacteristicDictionary.↵  
GetEnumerator ( )
```

Returns

7.4.3.8 Remove() [1/2]

```
bool ahbsd.lib.Password.CharacteristicDictionary.Remove (
    Charasteristic key )
```

Parameters

<i>key</i>	
------------	--

Returns

7.4.3.9 Remove() [2/2]

```
bool ahbsd.lib.Password.CharacteristicDictionary.Remove (
    KeyValuePair< Charasteristic, bool > item )
```

Parameters

<i>item</i>	
-------------	--

Returns

7.4.3.10 ToString()

```
override string ahbsd.lib.Password.CharacteristicDictionary.ToString ( )
```

Gets a short info of the characteristic.

Returns

A short info of the characteristic.

Implements [ahbsd.lib.Password.ICharacteristicDictionary](#).

Referenced by [ahbsd.lib.Password.CharacteristicDictionary.CharacteristicDictionary\(\)](#).

7.4.3.11 TryGetValue()

```
bool ahbsd.lib.Password.CharacteristicDictionary.TryGetValue (
    Charasteristic key,
    [MaybeNullWhen(false)] out bool value )
```

Parameters

<i>key</i>	
<i>value</i>	

Returns

7.4.4 Property Documentation

7.4.4.1 Count

```
int ahbsd.lib.Password.CharacteristicDictionary.Count [get]
```

Gets the amount of KeyValuePair<TKey, TValue>s (5).

The amount of KeyValuePair<TKey, TValue>s.

7.4.4.2 IsReadOnly

```
bool ahbsd.lib.Password.CharacteristicDictionary.IsReadOnly [get]
```

Gets the readability.

true if readonly, otherwise false.

7.4.4.3 Keys

```
ICollection<Charasteristic> ahbsd.lib.Password.CharacteristicDictionary.Keys [get]
```

Gets the Keys.

The Keys.

Each available [Charasteristic](#).

7.4.4.4 Name

```
string ahbsd.lib.Password.CharacteristicDictionary.Name [get]
```

Gets the name of this component.

The name of this component.

7.4.4.5 this[Charasteristic key]

```
bool ahbsd.lib.Password.CharacteristicDictionary.this[Charasteristic key] [get], [set]
```

Gets or sets the value of the given key.

Parameters

<i>key</i>	The given key.
------------	----------------

Returns

The value of the given key.

7.4.4.6 Values

```
ICollection<bool> ahbsd.lib.Password.CharacteristicDictionary.Values [get]
```

Gets all values.

The values.

The documentation for this class was generated from the following file:

- ahbsd.lib/Password/CharacteristicDictionary.cs

7.5 ahbsd.lib.Tools.Checksum Class Reference

Class to calculate checksums.

Static Public Member Functions

- static long [GetChecksum](#) (long value)
Gets the checksum of the given value.
- static short [GetFinalChecksum](#) (long value)
Gets the final checksum from a given value.
- static ? long [GetChecksum](#) (long? value)
Gets the checksum of the given value.
- static ? short [GetFinalChecksum](#) (long? value)
Gets the final checksum from a given value.
- static ulong [GetChecksum](#) (ulong value)
Gets the checksum of the given value.
- static ushort [GetFinalChecksum](#) (ulong value)
Gets the final checksum from a given value.
- static ? ulong [GetChecksum](#) (ulong? value)
Gets the checksum of the given value.
- static ? ushort [GetFinalChecksum](#) (ulong? value)
Gets the final checksum from a given value.
- static bool [IsPositiv](#) (long value)
Checks wheather value is positive or negative.
- static bool [IsPositive](#) (long? value)
Checks wheather value is positive or negative.
- static bool [IsPositiv](#) (ulong value)
Checks wheather value is positive or negative.
- static bool [IsPositive](#) (ulong? value)
Checks wheather value is positive or negative.
- static bool [IsPositiv](#) (short value)
Checks wheather value is positive or negative.
- static bool [IsPositive](#) (short? value)
Checks wheather value is positive or negative.

7.5.1 Detailed Description

Class to calculate checksums.

7.5.2 Member Function Documentation

7.5.2.1 GetChecksum() [1/4]

```
static long ahbsd.lib.Tools.Checksum.GetChecksum (  
    long value ) [static]
```

Gets the checksum of the given value.

Parameters

<i>value</i>	The given value.
--------------	------------------

Returns

The checksum of the given value.

References [ahbsd.lib.Tools.Checksum.IsPositiv\(\)](#).

Referenced by [ahbsd.lib.Tools.Checksum.GetChecksum\(\)](#), and [ahbsd.lib.Tools.Checksum.GetFinalChecksum\(\)](#).

7.5.2.2 GetChecksum() [2/4]

```
static ? long ahbsd.lib.Tools.Checksum.GetChecksum (  
    long? value ) [static]
```

Gets the checksum of the given value.

Parameters

<i>value</i>	The given value.
--------------	------------------

Returns

The checksum of the given value or `null` if value is as well `null`.

References [ahbsd.lib.Tools.Checksum.GetChecksum\(\)](#).

7.5.2.3 GetChecksum() [3/4]

```
static ulong ahbsd.lib.Tools.Checksum.GetChecksum (
    ulong value ) [static]
```

Gets the checksum of the given value.

Parameters

<i>value</i>	The given value.
--------------	------------------

Returns

The checksum of the given value.

7.5.2.4 GetChecksum() [4/4]

```
static ?  ulong ahbsd.lib.Tools.Checksum.GetChecksum (
    ulong? value ) [static]
```

Gets the checksum of the given value.

Parameters

<i>value</i>	The given value.
--------------	------------------

Returns

The checksum of the given value or `null` if value is as well `null`.

References [ahbsd.lib.Tools.Checksum.GetChecksum\(\)](#).

7.5.2.5 GetFinalChecksum() [1/4]

```
static short ahbsd.lib.Tools.Checksum.GetFinalChecksum (
    long value ) [static]
```

Gets the final checksum from a given value.

Parameters

<i>value</i>	The given value.
--------------	------------------

Returns

The final checksum of the given value.

References [ahbsd.lib.Tools.Checksum.GetChecksum\(\)](#), and [ahbsd.lib.Tools.Checksum.IsPositiv\(\)](#).

7.5.2.6 GetFinalChecksum() [2/4]

```
static ? short ahbsd.lib.Tools.Checksum.GetFinalChecksum (
    long? value ) [static]
```

Gets the final checksum from a given value.

Parameters

<i>value</i>	The given value.
--------------	------------------

Returns

The final checksum of the given value.

References [ahbsd.lib.Tools.Checksum.GetChecksum\(\)](#), and [ahbsd.lib.Tools.Checksum.IsPositiv\(\)](#).

7.5.2.7 GetFinalChecksum() [3/4]

```
static ushort ahbsd.lib.Tools.Checksum.GetFinalChecksum (
    ulong value ) [static]
```

Gets the final checksum from a given value.

Parameters

<i>value</i>	The given value.
--------------	------------------

Returns

The final checksum of the given value.

References [ahbsd.lib.Tools.Checksum.GetChecksum\(\)](#).

7.5.2.8 GetFinalChecksum() [4/4]

```
static ? ushort ahbsd.lib.Tools.Checksum.GetFinalChecksum (
    ulong? value ) [static]
```

Gets the final checksum from a given value.

Parameters

<i>value</i>	The given value.
--------------	------------------

Returns

The final checksum of the given value.

References [ahbsd.lib.Tools.Checksum.GetChecksum\(\)](#).

7.5.2.9 IsPositiv() [1/3]

```
static bool ahbsd.lib.Tools.Checksum.IsPositiv (  
    long value ) [static]
```

Checks wheather value is positive or negative.

Parameters

<i>value</i>	The value to check.
--------------	---------------------

Returns

true if value is positive, otherwise false.

Referenced by [ahbsd.lib.Tools.Checksum.GetChecksum\(\)](#), [ahbsd.lib.Tools.Checksum.GetFinalChecksum\(\)](#), and [ahbsd.lib.Tools.Checksum.IsPositive\(\)](#).

7.5.2.10 IsPositiv() [2/3]

```
static bool ahbsd.lib.Tools.Checksum.IsPositiv (  
    short value ) [static]
```

Checks wheather value is positive or negative.

Parameters

<i>value</i>	The value to check.
--------------	---------------------

Returns

true if value is positive, otherwise false.

7.5.2.11 IsPositiv() [3/3]

```
static bool ahbsd.lib.Tools.Checksum.IsPositiv (  
    ulong value ) [static]
```

Checks wheather value is positive or negative.

Parameters

<i>value</i>	The value to check.
--------------	---------------------

Returns

true if value is positive, otherwise false.

7.5.2.12 IsPositive() [1/3]

```
static bool ahbsd.lib.Tools.Checksum.IsPositive (  
    long? value ) [static]
```

Checks wheather value is positive or negative.

Parameters

<i>value</i>	The value to check.
--------------	---------------------

Returns

true if value is positive, otherwise false.

References [ahbsd.lib.Tools.Checksum.IsPositiv\(\)](#).

7.5.2.13 IsPositive() [2/3]

```
static bool ahbsd.lib.Tools.Checksum.IsPositive (  
    short? value ) [static]
```

Checks wheather value is positive or negative.

Parameters

<i>value</i>	The value to check.
--------------	---------------------

Returns

`true` if value is positive, otherwise `false`.

References [ahbsd.lib.Tools.Checksum.IsPositiv\(\)](#).

7.5.2.14 IsPositive() [3/3]

```
static bool ahbsd.lib.Tools.Checksum.IsPositive (
    ulong? value ) [static]
```

Checks wheather value is positive or negative.

Parameters

<i>value</i>	The value to check.
--------------	---------------------

Returns

`true` if value is positive, otherwise `false`.

The documentation for this class was generated from the following file:

- [ahbsd.lib/Tools/Checksum.cs](#)

7.6 ahbsd.lib.NamedCollections.CompanyModelProducts Class Reference

Class for a combination of difined products of a company.

Inherits Container, and `IDictionary< ICompany, IDictionaryOfNamedList< IModel, IProductItem >>`.

Public Member Functions

- [CompanyModelProducts \(\)](#)
Constructor without any parameters.
- void [Add](#) (ICompany key, IDictionaryOfNamedList< IModel, IProductItem > value)
Adds a given ICompany key and its IDictionaryOfNamedList< IModel, IProductItem> value.
- void [Add](#) (ICompany company)
Adds a Company key.
- void [Add](#) (KeyValuePair< ICompany, IDictionaryOfNamedList< IModel, IProductItem >> item)
Adds a defined KeyValuePair.
- void [Clear](#) ()
Clears this Dictionary.
- bool [Contains](#) (KeyValuePair< ICompany, IDictionaryOfNamedList< IModel, IProductItem >> item)

- Checks wheather a given KeyValuePair is contained or not.*

 - bool **ContainsKey** (ICompany key)

Checks wheather a given ICompany-KEY exists or not.
- void **CopyTo** (KeyValuePair< ICompany, IDictionaryOfNamedList< IModel, IProductItem >>[] array, int arrayIndex)

Copies the elements of this IDictionary<ICompany, IDictionaryOfNamedList<IModel, IProductItem>> to an Array, starting at an particular Array index.
- IEnumerator< KeyValuePair< ICompany, IDictionaryOfNamedList< IModel, IProductItem >> > **GetEnumerator** ()

Returns an enumerator, that iterates through this collection.
- bool **Remove** (ICompany key)

Removes the given KEY and its value.
- bool **Remove** (KeyValuePair< ICompany, IDictionaryOfNamedList< IModel, IProductItem >> item)

Removes the given KeyValuePair<ICompany, IDictionaryOfNamedList<IModel, IProductItem>>.
- bool **TryGetValue** (ICompany key, [MaybeNullWhen(false)] out IDictionaryOfNamedList< IModel, IProductItem > value)

Gets the Value associated with the given Key.

Properties

- IDictionaryOfNamedList< IModel, IProductItem > **this[ICompany key]** [get, set]

Gets or sets a Dictionary of IDictionaryOfNamedList<IModel, IProductItem> for a given ICompany.
- ICollection< ICompany > **Keys** [get]

Gets the Company Keys.
- ICollection< IDictionaryOfNamedList< IModel, IProductItem >> **Values** [get]

Gets the Values.
- int **Count** [get]

Gets the amaunt of Companies.
- bool **IsReadOnly** [get]

Gets the information, if this Dictionary is readonly.

7.6.1 Detailed Description

Class for a combination of difined products of a company.

7.6.2 Constructor & Destructor Documentation

7.6.2.1 CompanyModelProducts()

```
ahbsd.lib.NamedCollections.CompanyModelProducts.CompanyModelProducts ( )
```

Constructor without any parameters.

7.6.3 Member Function Documentation

7.6.3.1 Add() [1/3]

```
void ahbsd.lib.NamedCollections.CompanyModelProducts.Add (  
    ICompany company )
```

Adds a Company key.

Parameters

<i>company</i>	The Company.
----------------	--------------

References [ahbsd.lib.Products.ICompany.Name](#), and [ahbsd.lib.NamedCollections.IDictionaryOfNamedList< K, V >.OnNamedListAd](#)

7.6.3.2 Add() [2/3]

```
void ahbsd.lib.NamedCollections.CompanyModelProducts.Add (
    ICompany key,
    IDictionaryOfNamedList< IModel, IProductItem > value )
```

Adds a given ICompany key and its IDictionaryOfNamedList<IModel, IProductItem> value.

Parameters

<i>key</i>	The Company key.
<i>value</i>	The Dictionary of Model and ProductItem.

References [ahbsd.lib.Products.ICompany.Name](#), and [ahbsd.lib.NamedCollections.IDictionaryOfNamedList< K, V >.OnNamedListAd](#)

7.6.3.3 Add() [3/3]

```
void ahbsd.lib.NamedCollections.CompanyModelProducts.Add (
    KeyValuePair< ICompany, IDictionaryOfNamedList< IModel, IProductItem >> item )
```

Adds a defined KeyValuePair.

Parameters

<i>item</i>	The defined KeyValuePair.
-------------	---------------------------

7.6.3.4 Clear()

```
void ahbsd.lib.NamedCollections.CompanyModelProducts.Clear ( )
```

Clears this Dictionary.

7.6.3.5 Contains()

```
bool ahbsd.lib.NamedCollections.CompanyModelProducts.Contains (
    KeyValuePair< ICompany, IDictionaryOfNamedList< IModel, IProductItem >> item )
```

Checks wheather a given KeyValuePair is contained or not.

Parameters

<i>item</i>	The given KeyValuePair.
-------------	-------------------------

Returns

true, if contained, otherwise false.

7.6.3.6 ContainsKey()

```
bool ahbsd.lib.NamedCollections.CompanyModelProducts.ContainsKey (
    ICompany key )
```

Checks wheather a given ICompany-KEY exists or not.

Parameters

<i>key</i>	The ICompany-KEY to search for.
------------	---------------------------------

Returns

true if the ICompany-KEY exists, otherwise false

7.6.3.7 CopyTo()

```
void ahbsd.lib.NamedCollections.CompanyModelProducts.CopyTo (
    KeyValuePair< ICompany, IDictionaryOfNamedList< IModel, IProductItem >>[] array,
    int arrayIndex )
```

Copies the elements of this IDictionary<ICompany, IDictionaryOfNamedList<IModel, IProductItem>> to an Array, starting at an particular Array index.

Parameters

<i>array</i>	The given Array to copy to.
<i>arrayIndex</i>	The particular Array index to start.

Exceptions

<i>ArgumentNullException</i>	If the Array is <code>null</code> .
<i>ArgumentOutOfRangeException</i>	If the given index is out of range of the array.
<i>ArgumentException</i>	If anything with the given arguments is wrong.

7.6.3.8 GetEnumerator()

```
IEnumerator<KeyValuePair<ICompany, IDictionaryOfNamedList<IModel, IProductItem>>> ahbsd.↵
lib.NamedCollections.CompanyModelProducts.GetEnumerator ( )
```

Returns an enumerator, that iterates through this collection.

Returns

An enumerator that can be used to iterate through this collection.

7.6.3.9 Remove() [1/2]

```
bool ahbsd.lib.NamedCollections.CompanyModelProducts.Remove (
    ICompany key )
```

Removes the given KEY and its value.

Parameters

<i>key</i>	The given KEY to remove.
------------	--------------------------

Returns

`true` if the given KEY was found and removed, otherwise `false`.

7.6.3.10 Remove() [2/2]

```
bool ahbsd.lib.NamedCollections.CompanyModelProducts.Remove (
    KeyValuePair< ICompany, IDictionaryOfNamedList< IModel, IProductItem >> item )
```

Removes the given KeyValuePair<ICompany, IDictionaryOfNamedList<IModel, IProductItem>>.

Parameters

<i>item</i>	The given KeyValuePair to remove.
-------------	-----------------------------------

Returns

`true` if the given `KeyValuePair` was found and removed, otherwise `false`.

7.6.3.11 TryGetValue()

```
bool ahbsd.lib.NamedCollections.CompanyModelProducts.TryGetValue (
    ICompany key,
    [MaybeNullWhen(false)] out IDictionaryOfNamedList< IModel, IProductItem > value )
```

Gets the Value associated with the given Key.

Parameters

<i>key</i>	The given Key.
<i>value</i>	The Value associated with the given Key.

Returns

`true` if the Key was found, otherwise `false`.

Exceptions

<i>ArgumentNullException</i>	If any of the Arguments is <code>null</code> .
------------------------------	--

7.6.4 Property Documentation**7.6.4.1 Count**

```
int ahbsd.lib.NamedCollections.CompanyModelProducts.Count [get]
```

Gets the amount of Companies.

The amount of Companies.

7.6.4.2 IsReadOnly

```
bool ahbsd.lib.NamedCollections.CompanyModelProducts.IsReadOnly [get]
```

Gets the information, if this Dictionary is readonly.

`true`, if readonly, otherwise `false`.

7.6.4.3 Keys

```
ICollection<ICompany> ahbsd.lib.NamedCollections.CompanyModelProducts.Keys [get]
```

Gets the Company Keys.

The Company Keys.

7.6.4.4 this[ICompany key]

```
IDictionaryOfNamedList<IModel, IProductItem> ahbsd.lib.NamedCollections.CompanyModelProducts.↔  
this[ICompany key] [get], [set]
```

Gets or sets a Dictionary of IDictionaryOfNamedList<IModel, IProductItem> for a given ICompany.

Parameters

key	The given ICompany.
-----	---------------------

Returns

The Dictionary for the given Company.

7.6.4.5 Values

```
ICollection<IDictionaryOfNamedList<IModel, IProductItem> > ahbsd.lib.NamedCollections.↔  
CompanyModelProducts.Values [get]
```

Gets the Values.

The Values.

The documentation for this class was generated from the following file:

- ahbsd.lib/NamedCollections/CompanyModelProducts.cs

7.7 ahbsd.lib.Tools.ConsolePrintTable Class Reference

Static class to print a DataTable to console.

Static Public Member Functions

- static void [Print](#) (DataTable table)
Prints a table on console.

7.7.1 Detailed Description

Static class to print a DataTable to console.

7.7.2 Member Function Documentation

7.7.2.1 Print()

```
static void ahbsd.lib.Tools.ConsolePrintTable.Print (
    DataTable table ) [static]
```

Prints a table on console.

Parameters

<i>table</i>	The table to print.
--------------	---------------------

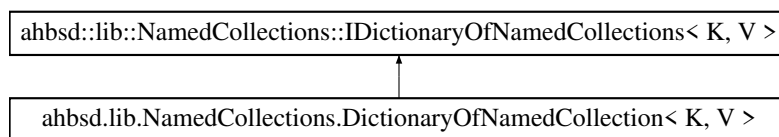
The documentation for this class was generated from the following file:

- ahbsd.lib/Tools/ConsolePrintTable.cs

7.8 ahbsd.lib.NamedCollections.DictionaryOfNamedCollection< K, V > Class Template Reference

Default implementation of IDictionaryOfNamedCollections<K, V>.

Inheritance diagram for ahbsd.lib.NamedCollections.DictionaryOfNamedCollection< K, V >:



Public Member Functions

- void [Add](#) (K key, string name)
Adds a new key with the name of the new INamedCollection< T>.
- void [Add](#) (K key, V value, string name=null)
Adds a value to the INamedCollection< T> of key.
- void [Add](#) (KeyValuePair< K, V > keyValuePair, string name=null)
Adds a KeyValuePair< TKey, TValue>.

Events

- EventHandler< [EventArgs< INamedCollection< V > >](#) > [OnNamedCollectionAdded](#)
Happenes if a new INamedCollection<T> was added.

7.8.1 Detailed Description

Default implementation of IDictionaryOfNamedCollections<K, V>.

Template Parameters

<i>K</i>	Key Type.
<i>V</i>	Value Type of the INamedCollection<T>.

7.8.2 Member Function Documentation

7.8.2.1 Add() [1/3]

```
void ahbsd.lib.NamedCollections.DictionaryOfNamedCollection< K, V >.Add (
    K key,
    string name )
```

Adds a new key with the name of the new INamedCollection<T>.

Parameters

<i>key</i>	The new key.
<i>name</i>	The name of the new INamedCollection<T>.

Exceptions

<i>ArgumentException</i>	If key already exists.
--------------------------	------------------------

Implements [ahbsd.lib.NamedCollections.IDictionaryOfNamedCollections< K, V >](#).

References [ahbsd.lib.NamedCollections.DictionaryOfNamedCollection< K, V >.OnNamedCollectionAdded](#).

Referenced by [ahbsd.lib.NamedCollections.DictionaryOfNamedCollection< K, V >.Add\(\)](#).

7.8.2.2 Add() [2/3]

```
void ahbsd.lib.NamedCollections.DictionaryOfNamedCollection< K, V >.Add (
    K key,
```

```
V value,
string name = null )
```

Adds a value to the `INamedCollection<T>` of key.

Parameters

<i>key</i>	The Key.
<i>value</i>	The Value.
<i>name</i>	The name of the new <code>INamedCollection<T></code> .

If the key already exists the name isn't needed; if the key doesn't exists a name is needed, otherwise a `KeyNotFoundException` will be thrown.

Exceptions

<i>KeyNotFoundException</i>	If the key isn't there AND a name for the new <code>INamedCollection<T></code> was missing.
-----------------------------	---

Implements [ahbsd.lib.NamedCollections.IDictionaryOfNamedCollections< K, V >](#).

References [ahbsd.lib.NamedCollections.DictionaryOfNamedCollection< K, V >.Add\(\)](#), and [ahbsd.lib.NamedCollections.DictionaryOf](#)

7.8.2.3 Add() [3/3]

```
void ahbsd.lib.NamedCollections.DictionaryOfNamedCollection< K, V >.Add (
    KeyValuePair< K, V > keyValuePair,
    string name = null )
```

Adds a `KeyValuePair<TKey, TValue>`.

Parameters

<i>keyValuePair</i>	The <code>KeyValuePair<TKey, TValue></code> .
<i>name</i>	The name of the new <code>INamedCollection<T></code> .

If the key already exists the name isn't needed; if the key doesn't exists a name is needed, otherwise a `KeyNotFoundException` will be thrown.

Exceptions

<i>KeyNotFoundException</i>	If the key isn't there AND a name for the new <code>INamedCollection<T></code> was missing.
-----------------------------	---

Implements [ahbsd.lib.NamedCollections.IDictionaryOfNamedCollections< K, V >](#).

References [ahbsd.lib.NamedCollections.DictionaryOfNamedCollection< K, V >.Add\(\)](#).

7.8.3 Event Documentation

7.8.3.1 OnNamedCollectionAdded

EventHandler<EventArgs<INamedCollection<V> > > [ahbsd.lib.NamedCollections.DictionaryOfNamedCollection< K, V >.OnNamedCollectionAdded](#)

Happenes if a new INamedCollection<T> was added.

Referenced by [ahbsd.lib.NamedCollections.DictionaryOfNamedCollection< K, V >.Add\(\)](#).

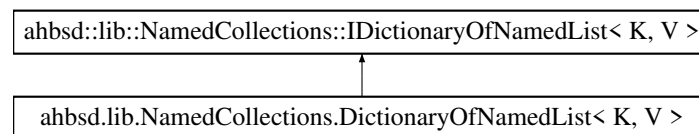
The documentation for this class was generated from the following file:

- [ahbsd.lib/NamedCollections/DictionaryOfNamedCollection.cs](#)

7.9 ahbsd.lib.NamedCollections.DictionaryOfNamedList< K, V > Class Template Reference

Default implementation of IDictionaryOfNamedList<K, V>.

Inheritance diagram for ahbsd.lib.NamedCollections.DictionaryOfNamedList< K, V >:



Public Member Functions

- void [Add](#) (K key, string name)
Adds a key.
- void [Add](#) (K key, V value, string name=null)
Adds a value to the INamedList<T> of key.
- void [Add](#) (KeyValuePair< K, V > keyValuePair, string name=null)
Adds a KeyValuePair<TKey, TValue>.

Events

- EventHandler< [EventArgs< INamedList< V > >](#) > [OnNamedListAdded](#)
Happenes, if a new INamedList<T> was added.

7.9.1 Detailed Description

Default implementation of IDictionaryOfNamedList<K, V>.

Template Parameters

<i>K</i>	The type of the key.
<i>V</i>	The type of the INamedList<T>-ValueType.

7.9.2 Member Function Documentation

7.9.2.1 Add() [1/3]

```
void ahbsd.lib.NamedCollections.DictionaryOfNamedList< K, V >.Add (
    K key,
    string name )
```

Adds a key.

Parameters

<i>key</i>	The key.
<i>name</i>	The name of the new INamedList<T>.

Exceptions

<i>ArgumentException</i>	If key already exists.
--------------------------	------------------------

Implements [ahbsd.lib.NamedCollections.IDictionaryOfNamedList< K, V >](#).

References [ahbsd.lib.NamedCollections.DictionaryOfNamedList< K, V >.OnNamedListAdded](#).

Referenced by [ahbsd.lib.NamedCollections.DictionaryOfNamedList< K, V >.Add\(\)](#).

7.9.2.2 Add() [2/3]

```
void ahbsd.lib.NamedCollections.DictionaryOfNamedList< K, V >.Add (
    K key,
    V value,
    string name = null )
```

Adds a value to the INamedList<T> of key.

Parameters

<i>key</i>	The key.
<i>value</i>	The value.
<i>name</i>	The name of the new INamedList<T>.

If the key already exists the name isn't needed; if the key doesn't exists a name is needed, otherwise a `KeyNotFoundException` will be thrown.

Exceptions

<i>KeyNotFoundException</i>	If the key isn't there AND a name for the new INamedList<T> was missing.
-----------------------------	--

Implements [ahbsd.lib.NamedCollections.IDictionaryOfNamedList< K, V >](#).

References [ahbsd.lib.NamedCollections.DictionaryOfNamedList< K, V >.Add\(\)](#), and [ahbsd.lib.NamedCollections.DictionaryOfNamedList< K, V >.OnNamedListAdded](#).

7.9.2.3 Add() [3/3]

```
void ahbsd.lib.NamedCollections.DictionaryOfNamedList< K, V >.Add (
    KeyValuePair< K, V > keyValuePair,
    string name = null )
```

Adds a KeyValuePair<TKey, TValue>.

Parameters

<i>keyValuePair</i>	The KeyValuePair<TKey, TValue>.
<i>name</i>	The name of the new INamedList<T>.

If the key already exists the name isn't needed; if the key doesn't exist a name is needed, otherwise a KeyNotFoundException will be thrown.

Exceptions

<i>KeyNotFoundException</i>	If the key isn't there AND a name for the new INamedList<T> was missing.
-----------------------------	--

Implements [ahbsd.lib.NamedCollections.IDictionaryOfNamedList< K, V >](#).

References [ahbsd.lib.NamedCollections.DictionaryOfNamedList< K, V >.Add\(\)](#).

7.9.3 Event Documentation**7.9.3.1 OnNamedListAdded**

```
EventHandler<EventArgs<INamedList<V> > > ahbsd.lib.NamedCollections.DictionaryOfNamedList<
K, V >.OnNamedListAdded
```

Happens, if a new INamedList<T> was added.

Referenced by [ahbsd.lib.NamedCollections.DictionaryOfNamedList< K, V >.Add\(\)](#).

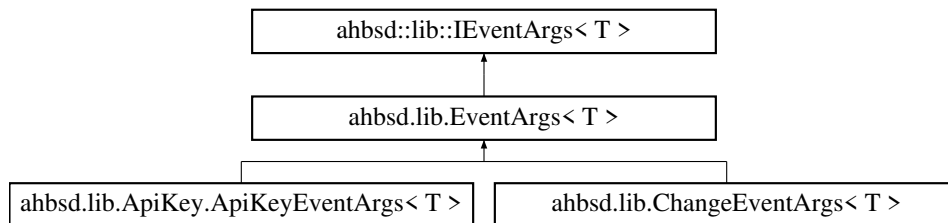
The documentation for this class was generated from the following file:

- ahbsd.lib/NamedCollections/DictionaryOfNamedList.cs

7.10 ahbsd.lib.EventArgs< T > Class Template Reference

Generic [EventArgs](#).

Inheritance diagram for ahbsd.lib.EventArgs< T >:



Public Member Functions

- [EventArgs](#) ()
Constructor without parameters.
- [EventArgs](#) (T val)
Constructor with a value.

Properties

- T [Value](#) [get]
Gets a value.

7.10.1 Detailed Description

Generic [EventArgs](#).

Template Parameters

<i>T</i>	Type of Value .
----------	---------------------------------

7.10.2 Constructor & Destructor Documentation

7.10.2.1 EventArgs() [1/2]

`ahbsd.lib.EventArgs< T >.EventArgs ()`

Constructor without parameters.

References [ahbsd.lib.EventArgs< T >.Value](#).

7.10.2.2 EventArgs() [2/2]

```
ahbsd.lib.EventArgs< T >.EventArgs (
    T val )
```

Constructor with a value.

Parameters

<i>val</i>	A value.
------------	----------

References [ahbsd.lib.EventArgs< T >.Value](#).

7.10.3 Property Documentation

7.10.3.1 Value

```
T ahbsd.lib.EventArgs< T >.Value [get]
```

Gets a value.

A value.

Referenced by [ahbsd.lib.EventArgs< T >.EventArgs\(\)](#).

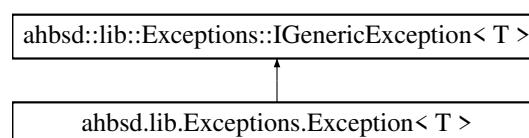
The documentation for this class was generated from the following file:

- ahbsd.lib/EventArgs.cs

7.11 ahbsd.lib.Exceptions.Exception< T > Class Template Reference

Class for a generic [Exception](#), which additionally holds a value of T

Inheritance diagram for ahbsd.lib.Exceptions.Exception< T >:



Protected Member Functions

- [Exception\(\)](#) [Exception\(T value\)](#) [Exception\(string message\)](#) [Exception\(string message, T value\)](#) [Exception\(string message, Exception innerException\)](#) [Exception \(SerializationInfo info, StreamingContext context\)](#)
Constructor without any parameters.

Properties

- [Exception](#)(string message, T value, [Exception](#) innerException) T [Value](#) [get]
Constructor with a message, a value and an inner exception.

7.11.1 Detailed Description

Class for a generic [Exception](#), which additionally holds a value of T

Template Parameters

<i>T</i>	The type.
----------	-----------

7.11.2 Constructor & Destructor Documentation

7.11.2.1 Exception()

```
Exception ()
Exception (T value)
Exception (string message)
Exception (string message, T value)
Exception (string message, Exception innerException)
ahbsd.lib.Exceptions.Exception< T >.Exception
(
    SerializationInfo info,
    StreamingContext context ) [protected]
```

Constructor without any parameters.

Constructor with a value.

Parameters

<i>value</i>	The value.
--------------	------------

Constructor with a message.

Parameters

<i>message</i>	The Message.
----------------	--------------

Constructor with a message and a value.

Parameters

<i>message</i>	The Message.
<i>value</i>	The value.

Constructor with a message and an inner exception.

Parameters

<i>message</i>	The Message.
<i>innerException</i>	The inner Exception .

Constructor with serialized data.

Parameters

<i>info</i>	The serialization info.
<i>context</i>	The straming content.

References [ahbsd.lib.Exceptions.Exception< T >.Value](#), and [ahbsd.lib.Exceptions.IGenericException< T >.Value](#).

7.11.3 Property Documentation

7.11.3.1 Value

```
Exception (string message, T value, Exception innerException) T ahbsd.lib.Exceptions.Exception< T >.Value [get]
```

Constructor with a message, a value and an inner exception.

Parameters

<i>message</i>	The Message.
<i>value</i>	The value.
<i>innerException</i>	The inner Exception .

Gets the value of type T.

The value of type T.

Referenced by [ahbsd.lib.Exceptions.Exception< T >.Exception\(\)](#).

The documentation for this class was generated from the following file:

- [ahbsd.lib/Exceptions/Exception.cs](#)

7.12 ahbsd.lib.Products.IAdress Interface Reference

An interface describing an adress in general.

Properties

- [AdressType](#) [AdressType](#) [get, set]
Gets or sets the AdressType.
- string [Address](#) [get, set]
Gets or sets the Address.
- string [ZIP](#) [get]
Gets the ZIP.
- string [City](#) [get]
Gets the City.
- CultureInfo [Country](#) [get]
Gets the Country.

Events

- ChangeEventHandler< [AdressType](#) > [OnTypeChanged](#)
Happenes, if the [AdressType](#) has changed.
- ChangeEventHandler< string > [OnAdressChanged](#)
Happenes, if the Address has changed.

7.12.1 Detailed Description

An interface describing an adress in general.

7.12.2 Property Documentation

7.12.2.1 Address

```
string ahbsd.lib.Products.IAddress.Address [get], [set]
```

Gets or sets the Address.

The Address.

7.12.2.2 AdressType

```
AdressType ahbsd.lib.Products.IAddress.AdressType [get], [set]
```

Gets or sets the AdressType.

The AdressType.

7.12.2.3 City

```
string ahbsd.lib.Products.IAdress.City [get]
```

Gets the City.

The City.

Is only settable by the constructor.

7.12.2.4 Country

```
CultureInfo ahbsd.lib.Products.IAdress.Country [get]
```

Gets the Country.

The Country.

Is only settable by the constructor.

7.12.2.5 ZIP

```
string ahbsd.lib.Products.IAdress.ZIP [get]
```

Gets the ZIP.

The ZIP.

Is only settable by the constructor.

7.12.3 Event Documentation

7.12.3.1 OnAdressChanged

```
ChangeEventHandler<string> ahbsd.lib.Products.IAdress.OnAdressChanged
```

Happenes, if the Adress has changed.

7.12.3.2 OnTypeChanged

```
ChangeEventHandler<AdressType> ahbsd.lib.Products.IAdress.OnTypeChanged
```

Happenes, if the [AdressType](#) has changed.

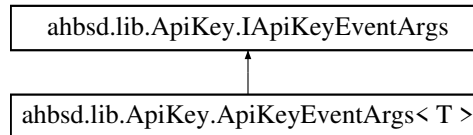
The documentation for this interface was generated from the following file:

- ahbsd.lib/Products/IAdress.cs

7.13 ahbsd.lib.ApiKey.IApiKeyEventArgs Interface Reference

Interface for ApiKeyEventArgs<T>.

Inheritance diagram for ahbsd.lib.ApiKey.IApiKeyEventArgs:



Properties

- `int? Index` [get]
Gets the index of the API-Key.

7.13.1 Detailed Description

Interface for ApiKeyEventArgs<T>.

7.13.2 Property Documentation

7.13.2.1 Index

```
int? ahbsd.lib.ApiKey.IApiKeyEventArgs.Index [get]
```

Gets the index of the API-Key.

The index.

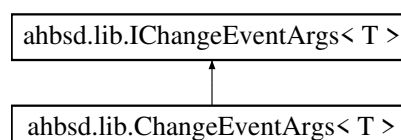
The documentation for this interface was generated from the following file:

- `ahbsd.lib/ApiKey/IApiKeyEventArgs.cs`

7.14 ahbsd.lib.IChangeEventArgs< T > Interface Template Reference

Interface for generic EventArgs for changing values.

Inheritance diagram for ahbsd.lib.IChangeEventArgs< T >:



Public Member Functions

- void [SetNewValue](#) (T newValue)
Sets the new value.
- string [ToString](#) ()
Gets a string representation of the changed value.

Properties

- T [OldValue](#) [get]
Gets the old value.
- T [NewValue](#) [get]
Gets the new value.

7.14.1 Detailed Description

Interface for generic [EventArgs](#) for changing values.

Template Parameters

<i>T</i>	The type of the changing Values.
----------	----------------------------------

7.14.2 Member Function Documentation

7.14.2.1 SetNewValue()

```
void ahbsd.lib.IChangeEventArgs< T >.SetNewValue (  
    T newValue )
```

Sets the new value.

Parameters

<i>newValue</i>	The new value.
-----------------	----------------

Exceptions

<i>Exception</i>	If the NewValue was already set.
------------------	--

Implemented in [ahbsd.lib.ChangeEventArgs< T >](#).

7.14.2.2 ToString()

```
string ahbsd.lib.IChangeEventArgs< T >.ToString ( )
```

Gets a string representation of the changed value.

Returns

A string representation of the changed value.

Implemented in [ahbsd.lib.ChangeEventArgs< T >](#).

7.14.3 Property Documentation

7.14.3.1 NewValue

```
T ahbsd.lib.IChangeEventArgs< T >.NewValue [get]
```

Gets the new value.

The new value.

Referenced by [ahbsd.lib.ChangeEventArgs< T >.Equals\(\)](#).

7.14.3.2 OldValue

```
T ahbsd.lib.IChangeEventArgs< T >.OldValue [get]
```

Gets the old value.

The old value.

Referenced by [ahbsd.lib.ChangeEventArgs< T >.Equals\(\)](#).

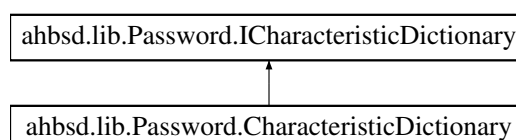
The documentation for this interface was generated from the following file:

- [ahbsd.lib/IChangeEventArgs.cs](#)

7.15 ahbsd.lib.Password.ICharacteristicDictionary Interface Reference

An interface for a characteristic dictionary.

Inheritance diagram for [ahbsd.lib.Password.ICharacteristicDictionary](#):



Public Member Functions

- string [ToString](#) ()
Gets a short info of the characteristic.

Properties

- string [Name](#) [get]
Gets the name of this component.

7.15.1 Detailed Description

An interface for a characteristic dictionary.

7.15.2 Member Function Documentation

7.15.2.1 ToString()

```
string ahbsd.lib.Password.ICharacteristicDictionary.ToString ( )
```

Gets a short info of the characteristic.

Returns

A short info of the characteristic.

Implemented in [ahbsd.lib.Password.CharacteristicDictionary](#).

7.15.3 Property Documentation

7.15.3.1 Name

```
string ahbsd.lib.Password.ICharacteristicDictionary.Name [get]
```

Gets the name of this component.

The name of this component.

The documentation for this interface was generated from the following file:

- [ahbsd.lib/Password/ICharacteristicDictionary.cs](#)

7.16 ahbsd.lib.Products.ICompany Interface Reference

An interface, describing a company / manufacturer.

Inherits IComponent.

Public Member Functions

- bool [AddAddress](#) ([IAddress](#) address)
Adds an Adress.
- bool [RemoveAddress](#) ([IAddress](#) address)
Removes an adress.
- bool [RemoveAddresses](#) ([AdressType](#) type)
Removes all addresses of the given type.

Properties

- string [Name](#) [get]
Gets the company name.
- IList< [IAddress](#) > [Addresses](#) [get]
Gets the list of addresses.
- string [Note](#) [get, set]
Gets or sets the notes to the company.

Events

- EventHandler< [EventArgs](#)< [IAddress](#) > > [OnAddressAdded](#)
Happenes, if an adress was added.
- EventHandler< [EventArgs](#)< [IAddress](#) > > [OnAddressRemoved](#)
Happenes, if an adress was removed.
- ChangeEventHandler< string > [OnNoteChanged](#)
Happenes, if the note has changed.

7.16.1 Detailed Description

An interface, describing a company / manufacturer.

7.16.2 Member Function Documentation

7.16.2.1 AddAddress()

```
bool ahbsd.lib.Products.ICompany.AddAddress (  
    IAddress address )
```

Adds an Adress.

Parameters

<i>adress</i>	The adress to be added.
---------------	-------------------------

Returns

`true` if the adress was sucessfully added; otherwise `false`.

7.16.2.2 RemoveAdress()

```
bool ahbsd.lib.Products.ICompany.RemoveAdress (
    IAdress adress )
```

Removes an adress.

Parameters

<i>adress</i>	The Adress to be removed.
---------------	---------------------------

Returns

`true` if the adress was successfully removed; otherwise `false`.

7.16.2.3 RemoveAdresses()

```
bool ahbsd.lib.Products.ICompany.RemoveAdresses (
    AdressType type )
```

Removes all addresses of the given type.

Parameters

<i>type</i>	The given type of addresses to remove.
-------------	--

Returns

`true` if the Adresses of the given type were found and removed, otherwise `false`.

7.16.3 Property Documentation

7.16.3.1 Adresses

```
IList<IAdress> ahbsd.lib.Products.ICompany.Adresses [get]
```

Gets the list of adresses.

The list of adresses.

7.16.3.2 Name

```
string ahbsd.lib.Products.ICompany.Name [get]
```

Gets the company name.

The company name.

Referenced by [ahbsd.lib.NamedCollections.CompanyModelProducts.Add\(\)](#).

7.16.3.3 Note

```
string ahbsd.lib.Products.ICompany.Note [get], [set]
```

Gets or sets the notes to the company.

The notes to the company.

7.16.4 Event Documentation

7.16.4.1 OnAdressAdded

```
EventHandler<EventArgs<IAdress>> > ahbsd.lib.Products.ICompany.OnAdressAdded
```

Happenes, if an adress was added.

7.16.4.2 OnAdressRemoved

```
EventHandler<EventArgs<IAdress>> > ahbsd.lib.Products.ICompany.OnAdressRemoved
```

Happenes, if an adress was removed.

7.16.4.3 OnNoteChanged

ChangeEventHandler<string> ahbsd.lib.Products.ICompany.OnNoteChanged

Happenes, if the note has changed.

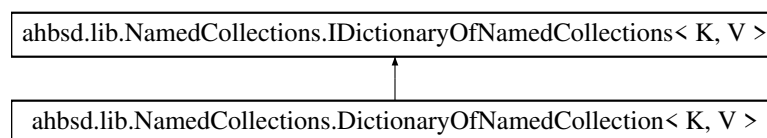
The documentation for this interface was generated from the following file:

- ahbsd.lib/Products/ICompany.cs

7.17 ahbsd.lib.NamedCollections.IDictionaryOfNamedCollections< K, V > Interface Template Reference

Interface for a dictionary of named collections as value.

Inheritance diagram for ahbsd.lib.NamedCollections.IDictionaryOfNamedCollections< K, V >:



Public Member Functions

- void [Add](#) (K key, string name)
Adds a new key with the name of the new INamedCollection< T>.
- void [Add](#) (K key, V value, string name=null)
Adds a value to the INamedCollection< T> of key.
- void [Add](#) (KeyValuePair< K, V > keyValuePair, string name=null)
Adds a KeyValuePair< TKey, TValue>.

Events

- EventHandler< [EventArgs](#) < [INamedCollection](#) < V > > > [OnNamedCollectionAdded](#)
Happenes if a new INamedCollection< T> was added.

7.17.1 Detailed Description

Interface for a dictionary of named collections as value.

Template Parameters

<i>K</i>	Key Type.
<i>V</i>	Value Type of the INamedCollection< T>.

7.17.2 Member Function Documentation

7.17.2.1 Add() [1/3]

```
void ahbsd.lib.NamedCollections.IDictionaryOfNamedCollections< K, V >.Add (
    K key,
    string name )
```

Adds a new key with the name of the new INamedCollection<T>.

Parameters

<i>key</i>	The new key.
<i>name</i>	The name of the new INamedCollection<T>.

Exceptions

<i>ArgumentException</i>	If key already exists.
--------------------------	------------------------

Implemented in [ahbsd.lib.NamedCollections.DictionaryOfNamedCollection< K, V >](#).

7.17.2.2 Add() [2/3]

```
void ahbsd.lib.NamedCollections.IDictionaryOfNamedCollections< K, V >.Add (
    K key,
    V value,
    string name = null )
```

Adds a value to the INamedCollection<T> of key.

Parameters

<i>key</i>	The Key.
<i>value</i>	The Value.
<i>name</i>	The name of the new INamedCollection<T>.

If the key already exists the name isn't needed; if the key doesn't exists a name is needed, otherwise a KeyNotFoundException will be thrown.

Exceptions

<i>KeyNotFoundException</i>	If the key isn't there AND a name for the new INamedCollection<T> was missing.
-----------------------------	--

Implemented in [ahbsd.lib.NamedCollections.DictionaryOfNamedCollection< K, V >](#).

7.17.2.3 Add() [3/3]

```
void ahbsd.lib.NamedCollections.IDictionaryOfNamedCollections< K, V >.Add (
    KeyValuePair< K, V > keyValuePair,
    string name = null )
```

Adds a KeyValuePair<TKey, TValue>.

Parameters

<i>keyValuePair</i>	The KeyValuePair<TKey, TValue>.
<i>name</i>	The name of the new INamedCollection<T>.

If the key already exists the name isn't needed; if the key doesn't exists a name is needed, otherwise a KeyNotFoundException will be thrown.

Exceptions

<i>KeyNotFoundException</i>	If the key isn't there AND a name for the new INamedCollection<T> was missing.
-----------------------------	--

Implemented in [ahbsd.lib.NamedCollections.DictionaryOfNamedCollection< K, V >](#).

7.17.3 Event Documentation**7.17.3.1 OnNamedCollectionAdded**

```
EventHandler<EventArgs<INamedCollection<V> > > ahbsd.lib.NamedCollections.IDictionaryOfNamedCollections<
K, V >.OnNamedCollectionAdded
```

Happenes if a new INamedCollection<T> was added.

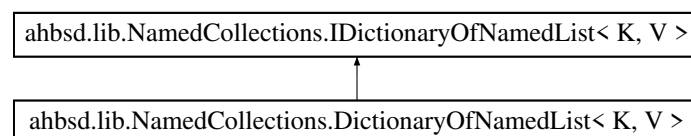
The documentation for this interface was generated from the following file:

- [ahbsd.lib/NamedCollections/IDictionaryOfNamedCollections.cs](#)

7.18 ahbsd.lib.NamedCollections.IDictionaryOfNamedList< K, V > Interface Template Reference

Interface for a collection of INamedList<T>.

Inheritance diagram for ahbsd.lib.NamedCollections.IDictionaryOfNamedList< K, V >:



Public Member Functions

- void [Add](#) (K key, string name)
Adds a key.
- void [Add](#) (K key, V value, string name=null)
Adds a value to the `INamedList<T>` of key.
- void [Add](#) (KeyValuePair< K, V > keyValuePair, string name=null)
Adds a `KeyValuePair< TKey, TValue>`.

Events

- EventHandler< [EventArgs](#)< [INamedList](#)< V > > > [OnNamedListAdded](#)
Happenes, if a new `INamedList<T>` was added.

7.18.1 Detailed Description

Interface for a collection of `INamedList<T>`.

Template Parameters

<i>K</i>	The type of the key.
<i>V</i>	The type of the <code>INamedList<T></code> -ValueType.

7.18.2 Member Function Documentation

7.18.2.1 Add() [1/3]

```
void ahbsd.lib.NamedCollections.IDictionaryOfNamedList< K, V >.Add (
    K key,
    string name )
```

Adds a key.

Parameters

<i>key</i>	The key.
<i>name</i>	The name of the new <code>INamedList<T></code> .

Exceptions

<i>ArgumentException</i>	If key already exists.
--------------------------	------------------------

Implemented in [ahbsd.lib.NamedCollections.DictionaryOfNamedList](#)< K, V >.

7.18.2.2 Add() [2/3]

```
void ahbsd.lib.NamedCollections.IDictionaryOfNamedList< K, V >.Add (
    K key,
    V value,
    string name = null )
```

Adds a value to the INamedList<T> of key.

Parameters

<i>key</i>	The key.
<i>value</i>	The value.
<i>name</i>	The name of the new INamedList<T>.

If the key already exists the name isn't needed; if the key doesn't exists a name is needed, otherwise a KeyNotFoundException will be thrown.

Exceptions

<i>KeyNotFoundException</i>	If the key isn't there AND a name for the new INamedList<T> was missing.
-----------------------------	--

Implemented in [ahbsd.lib.NamedCollections.DictionaryOfNamedList< K, V >](#).

7.18.2.3 Add() [3/3]

```
void ahbsd.lib.NamedCollections.IDictionaryOfNamedList< K, V >.Add (
    KeyValuePair< K, V > keyValuePair,
    string name = null )
```

Adds a KeyValuePair<TKey, TValue>.

Parameters

<i>keyValuePair</i>	The KeyValuePair<TKey, TValue>.
<i>name</i>	The name of the new INamedList<T>.

If the key already exists the name isn't needed; if the key doesn't exists a name is needed, otherwise a KeyNotFoundException will be thrown.

Exceptions

<i>KeyNotFoundException</i>	If the key isn't there AND a name for the new INamedList<T> was missing.
-----------------------------	--

Implemented in [ahbsd.lib.NamedCollections.DictionaryOfNamedList< K, V >](#).

7.18.3 Event Documentation

7.18.3.1 OnNamedListAdded

EventHandler<EventArgs<INamedList<V> > > ahbsd.lib.NamedCollections.IDictionaryOfNamedList<K, V>.OnNamedListAdded

Happenes, if a new INamedList<T> was added.

Referenced by [ahbsd.lib.NamedCollections.CompanyModelProducts.Add\(\)](#).

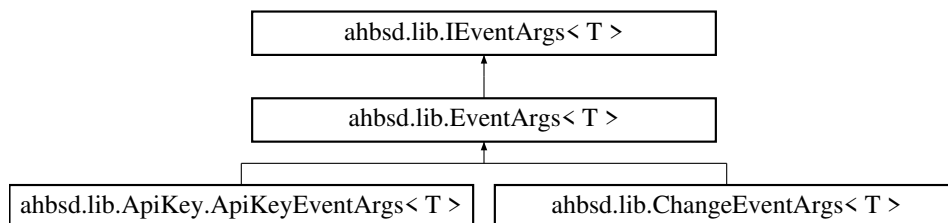
The documentation for this interface was generated from the following file:

- ahbsd.lib/NamedCollections/IDictionaryOfNamedList.cs

7.19 ahbsd.lib.IEventArgs< T > Interface Template Reference

Interface for generic [EventArgs](#).

Inheritance diagram for ahbsd.lib.IEventArgs< T >:



Properties

- `T Value` [get]
Gets a value.

7.19.1 Detailed Description

Interface for generic [EventArgs](#).

Template Parameters

<code>T</code>	Type of Value.
----------------	----------------

The difference to usual [EventArgs](#) is, that a generic value is added.

7.19.2 Property Documentation

7.19.2.1 Value

`T ahbsd.lib.IEventArgs< T >.Value [get]`

Gets a value.

A value.

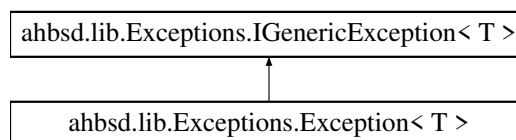
The documentation for this interface was generated from the following file:

- ahbsd.lib/IEventArgs.cs

7.20 ahbsd.lib.Exceptions.IGenericException< T > Interface Template Reference

Interface for generic exceptions.

Inheritance diagram for ahbsd.lib.Exceptions.IGenericException< T >:



Properties

- `T Value [get]`
Gets the value of type T.

7.20.1 Detailed Description

Interface for generic exceptions.

Template Parameters

<i>T</i>	Selectable type.
----------	------------------

7.20.2 Property Documentation

7.20.2.1 Value

`T ahbsd.lib.Exceptions.IGenericException< T >.Value [get]`

Gets the value of type T.

The value of type T.

Referenced by [ahbsd.lib.Exceptions.Exception< T >.Exception\(\)](#).

The documentation for this interface was generated from the following file:

- [ahbsd.lib/Exceptions/IGenericException.cs](#)

7.21 ahbsd.lib.Products.IModel Interface Reference

Interface of an model of a product.

Inherits IComponent.

Properties

- [ICompany Producer](#) [get]
Gets the producer of the Model.
- string [Name](#) [get]
Gets the name of the model.
- string [Notes](#) [get, set]
Gets or sets the notes to the model.

Events

- [ChangeEventHandler< string > OnNotesChanged](#)
Happenes, if the notes have changed.

7.21.1 Detailed Description

Interface of an model of a product.

7.21.2 Property Documentation

7.21.2.1 Name

```
string ahbsd.lib.Products.IModel.Name [get]
```

Gets the name of the model.

The name of the model.

7.21.2.2 Notes

```
string ahbsd.lib.Products.IModel.Notes [get], [set]
```

Gets or sets the notes to the model.

The notes to the model.

7.21.2.3 Producer

```
ICompany ahbsd.lib.Products.IModel.Producer [get]
```

Gets the producer of the Model.

The producer of the Model.

7.21.3 Event Documentation

7.21.3.1 OnNotesChanged

```
ChangeEventHandler<string> ahbsd.lib.Products.IModel.OnNotesChanged
```

Happenes, if the notes have changed.

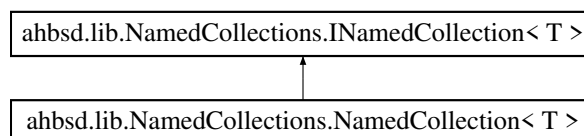
The documentation for this interface was generated from the following file:

- ahbsd.lib/Products/IModel.cs

7.22 ahbsd.lib.NamedCollections.INamedCollection< T > Interface Template Reference

Interface for a named collection.

Inheritance diagram for ahbsd.lib.NamedCollections.INamedCollection< T >:



Public Member Functions

- string [ToString](#) ()
Gets a string representation of this object.

Properties

- string [Name](#) [get, set]
Gets or sets the name of the collection.

Events

- ChangeEventHandler< string > [OnNameChanged](#)
Happenes, when the [Name](#) has changed.

7.22.1 Detailed Description

Interface for a named collection.

Template Parameters

<i>T</i>	The type of the collected objects.
----------	------------------------------------

7.22.2 Member Function Documentation

7.22.2.1 ToString()

```
string ahbsd.lib.NamedCollections.INamedCollection< T >.ToString ( )
```

Gets a string representation of this object.

Returns

The string representation of this object.

Implemented in [ahbsd.lib.NamedCollections.NamedCollection](#)< T >.

7.22.3 Property Documentation

7.22.3.1 Name

```
string ahbsd.lib.NamedCollections.INamedCollection< T >.Name [get], [set]
```

Gets or sets the name of the collection.

The name of the collection.

7.22.4 Event Documentation

7.22.4.1 OnNameChanged

```
ChangeEventHandler<string> ahbsd.lib.NamedCollections.INamedCollection< T >.OnNameChanged
```

Happenes, when the [Name](#) has changed.

The documentation for this interface was generated from the following file:

- ahbsd.lib/NamedCollections/INamedCollection.cs

7.23 ahbsd.lib.NamedCollections.INamedList< T > Interface Template Reference

Interface for a named list.

Inheritance diagram for ahbsd.lib.NamedCollections.INamedList< T >:



Public Member Functions

- string [ToString](#) ()
Gets a string representation of this object.

Properties

- string [Name](#) [get, set]
Gets or sets the name of the list.

Events

- ChangeEventHandler< string > [OnNameChanged](#)
Happenes, if the name of the list has changed.

7.23.1 Detailed Description

Interface for a named list.

Template Parameters

<i>T</i>	The type of the listed objects.
----------	---------------------------------

7.23.2 Member Function Documentation

7.23.2.1 ToString()

```
string ahbsd.lib.NamedCollections.INamedList< T >.ToString ( )
```

Gets a string representation of this object.

Returns

The string representation of this object.

Implemented in [ahbsd.lib.NamedCollections.NamedList< T >](#).

7.23.3 Property Documentation

7.23.3.1 Name

```
string ahbsd.lib.NamedCollections.INamedList< T >.Name [get], [set]
```

Gets or sets the name of the list.

The name of the list.

7.23.4 Event Documentation

7.23.4.1 OnNameChanged

```
ChangeEventHandler<string> ahbsd.lib.NamedCollections.INamedList< T >.OnNameChanged
```

Happenes, if the name of the list has changed.

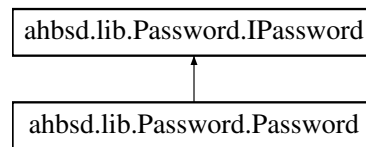
The documentation for this interface was generated from the following file:

- [ahbsd.lib/NamedCollections/INamedList.cs](#)

7.24 ahbsd.lib.Password.IPassword Interface Reference

An interface for a password object in general.

Inheritance diagram for ahbsd.lib.Password.IPassword:



Properties

- string [Value](#) [get, set]
Gets or sets the value.
- int [Length](#) [get]
Gets the length of the password.
- int [LowerCases](#) [get]
Gets the amount of lower cases.
- int [UpperCases](#) [get]
Gets the amount of upper cases.
- int [Numbers](#) [get]
Gets the amount of numbers.
- int [Spaces](#) [get]
Gets the amount of spaces.
- int [Specials](#) [get]
Gets the amount of special cases.
- [ICharacteristicDictionary Characteristics](#) [get]
Gets the characteristics of a password.
- [Check.ISecurityValue SecurityValue](#) [get]
Gets the SecurityValue.

Events

- [ChangeEventHandler< IPassword > OnChange](#)
Happenes, when the [Value](#) changes.

7.24.1 Detailed Description

An interface for a password object in general.

7.24.2 Property Documentation

7.24.2.1 Characteristics

`ICharacteristicDictionary` `ahbsd.lib.Password.IPassword.Characteristics` [get]

Gets the characteristics of a password.

7.24.2.2 Length

`int` `ahbsd.lib.Password.IPassword.Length` [get]

Gets the length of the password.

The length of the password.

7.24.2.3 LowerCases

`int` `ahbsd.lib.Password.IPassword.LowerCases` [get]

Gets the amount of lower cases.

The amount of lower cases.

7.24.2.4 Numbers

`int` `ahbsd.lib.Password.IPassword.Numbers` [get]

Gets the amount of numbers.

The amount of numbers.

7.24.2.5 SecurityValue

`Check.ISecurityValue` `ahbsd.lib.Password.IPassword.SecurityValue` [get]

Gets the SecurityValue.

The SecurityValue.

7.24.2.6 Spaces

`int` `ahbsd.lib.Password.IPassword.Spaces` [get]

Gets the amount of spaces.

The amount of spaces.

7.24.2.7 Specials

```
int ahbsd.lib.Password.IPassword.Specials [get]
```

Gets the amount of special cases.

The amount of special cases.

7.24.2.8 UpperCases

```
int ahbsd.lib.Password.IPassword.UpperCases [get]
```

Gets the amount of upper cases.

The amount of upper cases.

7.24.2.9 Value

```
string ahbsd.lib.Password.IPassword.Value [get], [set]
```

Gets or sets the value.

The value.

Referenced by [ahbsd.lib.Password.Password.Equals\(\)](#).

7.24.3 Event Documentation

7.24.3.1 OnChange

```
ChangeEventHandler<IPassword> ahbsd.lib.Password.IPassword.OnChange
```

Happenes, when the [Value](#) changes.

The documentation for this interface was generated from the following file:

- [ahbsd.lib/Password/IPassword.cs](#)

7.25 ahbsd.lib.Products.IProductItem Interface Reference

Interface of an Product.

Inherits IComponent.

Properties

- `IModel Model` [get]
Gets the model of the product.
- `string Name` [get]
Gets the name of the product.
- `string SerialNumber` [get]
Gets the serial number.

7.25.1 Detailed Description

Interface of an Product.

7.25.2 Property Documentation

7.25.2.1 Model

```
IModel ahbsd.lib.Products.IProductItem.Model [get]
```

Gets the model of the product.

The model of the product.

7.25.2.2 Name

```
string ahbsd.lib.Products.IProductItem.Name [get]
```

Gets the name of the product.

The name of the product.

7.25.2.3 SerialNumber

```
string ahbsd.lib.Products.IProductItem.SerialNumber [get]
```

Gets the serial number.

The serial number.

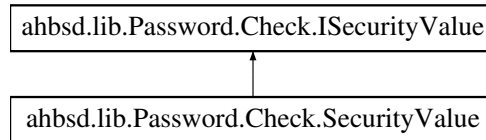
The documentation for this interface was generated from the following file:

- `ahbsd.lib/Products/IProduct.cs`

7.26 ahbsd.lib.Password.Check.ISecurityValue Interface Reference

An Interface for getting the security value of a given [IPassword](#).

Inheritance diagram for ahbsd.lib.Password.Check.ISecurityValue:



Properties

- [IPassword Password](#) [get]
Gets the password.
- ushort [Security](#) [get]
Gets the security.

7.26.1 Detailed Description

An Interface for getting the security value of a given [IPassword](#).

7.26.2 Property Documentation

7.26.2.1 Password

```
IPassword ahbsd.lib.Password.Check.ISecurityValue.Password [get]
```

Gets the password.

The password.

7.26.2.2 Security

```
ushort ahbsd.lib.Password.Check.ISecurityValue.Security [get]
```

Gets the security.

The security.

The higher, the better.

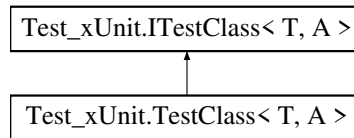
The documentation for this interface was generated from the following file:

- ahbsd.lib/Password/Check/ISecurityValue.cs

7.27 Test_xUnit.ITestClass< T, A > Interface Template Reference

An interface for a class to demonstrate ChangeEventArgs<T> and ChangeEventHandler<T>.

Inheritance diagram for Test_xUnit.ITestClass< T, A >:



Properties

- T [Variable](#) [get, set]
Gets or sets a variable.

Events

- ChangeEventHandler< T > [OnChange](#)
Happenes when [Variable](#) has changed.

7.27.1 Detailed Description

An interface for a class to demonstrate ChangeEventArgs<T> and ChangeEventHandler<T>.

Template Parameters

<i>T</i>	Type of Variable .
<i>A</i>	Type of API-Key.

7.27.2 Property Documentation

7.27.2.1 Variable

T [Test_xUnit.ITestClass< T, A >.Variable](#) [get], [set]

Gets or sets a variable.

7.27.3 Event Documentation

7.27.3.1 OnChange

ChangeEventHandler<T> [Test_xUnit.ITestClass](#)< T, A >.OnChange

Happenes when [Variable](#) has changed.

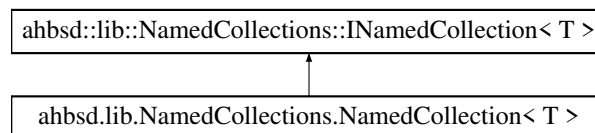
The documentation for this interface was generated from the following file:

- [Test_xUnit/ITestClass.cs](#)

7.28 ahbsd.lib.NamedCollections.NamedCollection< T > Class Template Reference

Default implementation of INamedCollection<T>.

Inheritance diagram for ahbsd.lib.NamedCollections.NamedCollection< T >:



Public Member Functions

- [NamedCollection](#) ()
Constructor without any parameters.
- [NamedCollection](#) (IList< T > list)
Constructor with a given IList<T> to wrap.
- [NamedCollection](#) (string name)
Constructor with a given name for the collection.
- [NamedCollection](#) (string name, IList< T > list)
Constructor with a given name for the collection and a IList<T> to wrap.
- override string [ToString](#) ()
Gets a string representation of this object.

Properties

- string?? [Name](#) [get, set]
Gets or sets the name of the collection.

Events

- ChangeEventHandler< string > [OnNameChanged](#)
Happenes, when the [Name](#) has changed.

7.28.1 Detailed Description

Default implementation of INamedCollection<T>.

Template Parameters

<i>T</i>	Type of the collected objects.
----------	--------------------------------

7.28.2 Constructor & Destructor Documentation

7.28.2.1 NamedCollection() [1/4]

```
ahbsd.lib.NamedCollections.NamedCollection< T >.NamedCollection ( )
```

Constructor without any parameters.

7.28.2.2 NamedCollection() [2/4]

```
ahbsd.lib.NamedCollections.NamedCollection< T >.NamedCollection (
    IList< T > list )
```

Constructor with a given IList<T> to wrap.

Parameters

<i>list</i>	The list to wrap.
-------------	-------------------

7.28.2.3 NamedCollection() [3/4]

```
ahbsd.lib.NamedCollections.NamedCollection< T >.NamedCollection (
    string name )
```

Constructor with a given name for the collection.

Parameters

<i>name</i>	The given name for the collection.
-------------	------------------------------------

7.28.2.4 NamedCollection() [4/4]

```
ahbsd.lib.NamedCollections.NamedCollection< T >.NamedCollection (
```



```
string name,  
IList< T > list )
```

Constructor with a given name for the collection and a `IList<T>` to wrap.

Parameters

<i>name</i>	The given name for the collection.
<i>list</i>	The list to wrap.

7.28.3 Member Function Documentation

7.28.3.1 ToString()

```
override string ahbsd.lib.NamedCollections.NamedCollection< T >.ToString ( )
```

Gets a string representation of this object.

Returns

The string representation of this object.

Implements [ahbsd.lib.NamedCollections.INamedCollection< T >](#).

7.28.4 Property Documentation

7.28.4.1 Name

```
string?? ahbsd.lib.NamedCollections.NamedCollection< T >.Name [get], [set]
```

Gets or sets the name of the collection.

The name of the collection.

7.28.5 Event Documentation

7.28.5.1 OnNameChanged

`ChangeEventHandler<string> ahbsd.lib.NamedCollections.NamedCollection< T >.OnNameChanged`

Happenes, when the [Name](#) has changed.

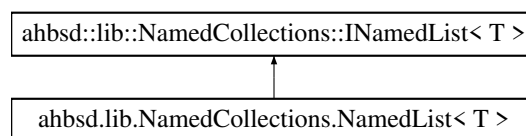
The documentation for this class was generated from the following file:

- `ahbsd.lib/NamedCollections/NamedCollection.cs`

7.29 ahbsd.lib.NamedCollections.NamedList< T > Class Template Reference

Default implementation of `INamedList<T>`.

Inheritance diagram for `ahbsd.lib.NamedCollections.NamedList< T >`:



Public Member Functions

- [NamedList](#) ()
Constructor without any parameters.
- [NamedList](#) (int capacity)
Constructor with a base capacity of the list.
- [NamedList](#) (IEnumerable< T > collection)
Constructor with a given collection.
- [NamedList](#) (string name)
Constructor with a given name.
- [NamedList](#) (string name, int capacity)
Constructor with a given name and a base capacity of the list.
- [NamedList](#) (string name, IEnumerable< T > collection)
Constructor with a given name and a given collection.
- override string [ToString](#) ()
Gets a string representation of this object.

Properties

- string?? [Name](#) [get, set]
Gets or sets the name of the list.

Events

- `ChangeEventHandler< string >` [OnNameChanged](#)
Happenes, if the name of the list has changed.

7.29.1 Detailed Description

Default implementation of `INamedList<T>`.

Template Parameters

<i>T</i>	Type of the listed objects.
----------	-----------------------------

7.29.2 Constructor & Destructor Documentation

7.29.2.1 NamedList() [1/6]

```
ahbsd.lib.NamedCollections.NamedList< T >.NamedList ( )
```

Constructor without any parameters.

7.29.2.2 NamedList() [2/6]

```
ahbsd.lib.NamedCollections.NamedList< T >.NamedList (
    int capacity )
```

Constructor with a base capacity of the list.

Parameters

<i>capacity</i>	The base capacity of the list.
-----------------	--------------------------------

Exceptions

<i>ArgumentOutOfRangeException</i>	If the capacity is out of range.
------------------------------------	----------------------------------

7.29.2.3 NamedList() [3/6]

```
ahbsd.lib.NamedCollections.NamedList< T >.NamedList (
    IEnumerable< T > collection )
```

Constructor with a given collection.

Parameters

<i>collection</i>	The given collection.
-------------------	-----------------------

7.29.2.4 NamedList() [4/6]

```
ahbsd.lib.NamedCollections.NamedList< T >.NamedList (
    string name )
```

Constructor with a given name.

Parameters

<i>name</i>	The given name.
-------------	-----------------

7.29.2.5 NamedList() [5/6]

```
ahbsd.lib.NamedCollections.NamedList< T >.NamedList (
    string name,
    int capacity )
```

Constructor with a given name and a base capacity of the list.

Parameters

<i>name</i>	The given name.
<i>capacity</i>	The base capacity of the list.

Exceptions

<i>ArgumentOutOfRangeException</i>	If the capacity is out of range.
------------------------------------	----------------------------------

7.29.2.6 NamedList() [6/6]

```
ahbsd.lib.NamedCollections.NamedList< T >.NamedList (
    string name,
    IEnumerable< T > collection )
```

Constructor with a given name and a given collection.

Parameters

<i>name</i>	The given name.
<i>collection</i>	The given collection.

7.29.3 Member Function Documentation

7.29.3.1 ToString()

```
override string ahbsd.lib.NamedCollections.NamedList< T >.ToString ( )
```

Gets a string representation of this object.

Returns

The string representation of this object.

Implements [ahbsd.lib.NamedCollections.INamedList< T >](#).

7.29.4 Property Documentation

7.29.4.1 Name

```
string?? ahbsd.lib.NamedCollections.NamedList< T >.Name [get], [set]
```

Gets or sets the name of the list.

The name of the list.

7.29.5 Event Documentation

7.29.5.1 OnNameChanged

```
ChangeEventHandler<string> ahbsd.lib.NamedCollections.NamedList< T >.OnNameChanged
```

Happenes, if the name of the list has changed.

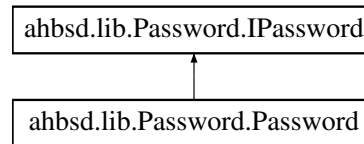
The documentation for this class was generated from the following file:

- [ahbsd.lib/NamedCollections/NamedList.cs](#)

7.30 ahbsd.lib.Password.Password Class Reference

A [Password](#) class.

Inheritance diagram for ahbsd.lib.Password.Password:



Public Member Functions

- [Password](#) ()
Constructor without any parameter.
- [Password](#) (string passwd)
Constructor with a given password.
- [Password](#) (IContainer container)
Constructor with a given owning container.
- [Password](#) (string passwd, IContainer container)
Constructor with a given password and a given owning container.
- override bool [Equals](#) (object obj)
Compares an other object with this object.
- bool [Equals](#) (IPassword other)
Compares an other [IPassword](#) with this object.
- override int [GetHashCode](#) ()
Gets the HashCode.

Static Public Member Functions

- static [ICharacteristicDictionary](#) [GetCharasteristics](#) (string value)
Gets the password characteristics of the given string.
- static [IPassword](#) [GetPassword](#) (string value)
Gets a [Password](#) from a given string.
- static [IPassword](#) [GetPassword](#) (string value, IContainer container)
Gets a [Password](#) from a given string.
- static int [GetLowerCases](#) (string value)
Gets the amount of lower cases in the given string.
- static int [GetUpperCases](#) (string value)
Gets the amount of upper cases in the given string.
- static int [GetSpaces](#) (string value)
Gets the amount of spaces in the given string.
- static int [GetNumbers](#) (string value)
Gets the amount of numbers in the given string.
- static int [GetSpecials](#) (string value)
Gets the amount of special chars in the given string.
- static [Charasteristic](#) [GetCharasteristic](#) (char c)
Gets the [Charasteristic](#) of a given char.
- static bool [operator==](#) ([Password](#) left, [Password](#) right)
Compares two [Passwords](#).
- static bool [operator!=](#) ([Password](#) left, [Password](#) right)
Compares two [Passwords](#).

Properties

- string? [Value](#) [get, set]
Gets or sets the value.
- int [Length](#) [get]
Gets the length of the password.
- int [LowerCases](#) [get]
Gets the amount of lower cases.
- int [UpperCases](#) [get]
Gets the amount of upper cases.
- int [Numbers](#) [get]
Gets the amount of numbers.
- int [Spaces](#) [get]
Gets the amount of spaces.
- int [Specials](#) [get]
Gets the amount of special cases.
- [ICharacteristicDictionary Characteristics](#) [get]
Gets the characteristics of a password.
- [Check.ISecurityValue SecurityValue](#) [get]
Gets the SecurityValue.

Events

- [ChangeEventHandler< IPassword > OnChange](#)
Happenes, when the [Value](#) changes.

7.30.1 Detailed Description

A [Password](#) class.

7.30.2 Constructor & Destructor Documentation

7.30.2.1 Password() [1/4]

```
ahbsd.lib.Password.Password.Password ( )
```

Constructor without any parameter.

References [ahbsd.lib.Password.Password.OnChange](#).

Referenced by [ahbsd.lib.Password.Password.GetPassword\(\)](#), and [ahbsd.lib.Password.Check.SecurityValue.SecurityValue\(\)](#).

7.30.2.2 Password() [2/4]

```
ahbsd.lib.Password.Password.Password (
    string passwd )
```

Constructor with a given password.

Parameters

<i>passwd</i>	The given password.
---------------	---------------------

References [ahbsd.lib.Password.Password.OnChange](#).

7.30.2.3 Password() [3/4]

```
ahbsd.lib.Password.Password.Password (
    IContainer container )
```

Constructor with a given owning container.

Parameters

<i>container</i>	The given owning container.
------------------	-----------------------------

References [ahbsd.lib.Password.Password.OnChange](#).

7.30.2.4 Password() [4/4]

```
ahbsd.lib.Password.Password.Password (
    string passwd,
    IContainer container )
```

Constructor with a given password and a given owning container.

Parameters

<i>passwd</i>	The given password.
<i>container</i>	The given owning container.

References [ahbsd.lib.Password.Password.OnChange](#).

7.30.3 Member Function Documentation**7.30.3.1 Equals()** [1/2]

```
bool ahbsd.lib.Password.Password.Equals (
    IPassword other )
```

Compares an other [IPassword](#) with this object.

Parameters

<i>other</i>	The other IPassword .
--------------	---------------------------------------

Returns

`true` if the other [IPassword](#) equals this password, otherwise `false`.

References [ahbsd.lib.Password.IPassword.Value](#), and [ahbsd.lib.Password.Password.Value](#).

7.30.3.2 Equals() [2/2]

```
override bool ahbsd.lib.Password.Password.Equals (  
    object obj )
```

Compares an other object with this object.

Parameters

<i>obj</i>	The other object.
------------	-------------------

Returns

`true` if the other object equals this password, otherwise `false`.

7.30.3.3 GetCharasteristic()

```
static Characteristic ahbsd.lib.Password.Password.GetCharasteristic (  
    char c ) [static]
```

Gets the [Characteristic](#) of a given char.

Parameters

<i>c</i>	The given char.
----------	-----------------

Returns

The [Characteristic](#).

Referenced by [ahbsd.lib.Password.Password.GetSpecials\(\)](#).

7.30.3.4 GetCharasteristics()

```
static ICharacteristicDictionary ahbsd.lib.Password.Password.GetCharasteristics (
    string value ) [static]
```

Gets the password characteristics of the given string.

Parameters

<i>value</i>	The given string.
--------------	-------------------

Returns

The password characteristics.

7.30.3.5 GetHashCode()

```
override int ahbsd.lib.Password.Password.GetHashCode ( )
```

Gets the GetHashCode.

Returns

The GetHashCode.

References [ahbsd.lib.Password.Password.Value](#).

7.30.3.6 GetLowerCases()

```
static int ahbsd.lib.Password.Password.GetLowerCases (
    string value ) [static]
```

Gets the amount of lower cases in the given string.

Parameters

<i>value</i>	The given string.
--------------	-------------------

Returns

The amount of lower cases.

7.30.3.7 GetNumbers()

```
static int ahbsd.lib.Password.Password.GetNumbers (
    string value ) [static]
```

Gets the amount of numbers in the given string.

Parameters

<i>value</i>	The given string.
--------------	-------------------

Returns

The amount of numbers.

7.30.3.8 GetPassword() [1/2]

```
static IPassword ahbsd.lib.Password.Password.GetPassword (
    string value ) [static]
```

Gets a [Password](#) from a given string.

Parameters

<i>value</i>	The given string.
--------------	-------------------

Returns

The [Password](#).

References [ahbsd.lib.Password.Password.Password\(\)](#).

7.30.3.9 GetPassword() [2/2]

```
static IPassword ahbsd.lib.Password.Password.GetPassword (
    string value,
    IContainer container ) [static]
```

Gets a [Password](#) from a given string.

Parameters

<i>value</i>	The given string.
<i>container</i>	An owning container.

Returns

The [Password](#).

References [ahbsd.lib.Password.Password.Password\(\)](#).

7.30.3.10 GetSpaces()

```
static int ahbsd.lib.Password.Password.GetSpaces (
    string value ) [static]
```

Gets the amount of spaces in the given string.

Parameters

<i>value</i>	The given string.
--------------	-------------------

Returns

The amount of spaces.

7.30.3.11 GetSpecials()

```
static int ahbsd.lib.Password.Password.GetSpecials (
    string value ) [static]
```

Gets the amount of special chars in the given string.

Parameters

<i>value</i>	The given string.
--------------	-------------------

Returns

The amount of special chars.

References [ahbsd.lib.Password.Password.GetCharasteristic\(\)](#).

7.30.3.12 GetUpperCases()

```
static int ahbsd.lib.Password.Password.GetUpperCases (
    string value ) [static]
```

Gets the amount of upper cases in the given string.

Parameters

<i>value</i>	The given string.
--------------	-------------------

Returns

The amount of upper cases.

7.30.3.13 operator"!=()"

```
static bool ahbsd.lib.Password.Password.operator!= (
    Password left,
    Password right ) [static]
```

Compares two [Passwords](#).

Parameters

<i>left</i>	The password on the left side.
<i>right</i>	The password on the right side.

Returns

`true` if both passwords do not equals, otherwise `false`.

7.30.3.14 operator=="()

```
static bool ahbsd.lib.Password.Password.operator== (
    Password left,
    Password right ) [static]
```

Compares two [Passwords](#).

Parameters

<i>left</i>	The password on the left side.
<i>right</i>	The password on the right side.

Returns

`true` if both passwords equals, otherwise `false`.

7.30.4 Property Documentation

7.30.4.1 Characteristics

`ICharacteristicDictionary` `ahbsd.lib.Password.Password.Characteristics` [get]

Gets the characteristics of a password.

7.30.4.2 Length

`int` `ahbsd.lib.Password.Password.Length` [get]

Gets the length of the password.

The length of the password.

7.30.4.3 LowerCases

`int` `ahbsd.lib.Password.Password.LowerCases` [get]

Gets the amount of lower cases.

The amount of lower cases.

7.30.4.4 Numbers

`int` `ahbsd.lib.Password.Password.Numbers` [get]

Gets the amount of numbers.

The amount of numbers.

7.30.4.5 SecurityValue

`Check.ISecurityValue` `ahbsd.lib.Password.Password.SecurityValue` [get]

Gets the SecurityValue.

The SecurityValue.

7.30.4.6 Spaces

```
int ahbsd.lib.Password.Password.Spaces [get]
```

Gets the amount of spaces.

The amount of spaces.

7.30.4.7 Specials

```
int ahbsd.lib.Password.Password.Specials [get]
```

Gets the amount of special cases.

The amount of special cases.

7.30.4.8 UpperCases

```
int ahbsd.lib.Password.Password.UpperCases [get]
```

Gets the amount of upper cases.

The amount of upper cases.

7.30.4.9 Value

```
string? ahbsd.lib.Password.Password.Value [get], [set]
```

Gets or sets the value.

The value.

Referenced by [ahbsd.lib.Password.Password.Equals\(\)](#), and [ahbsd.lib.Password.Password.GetHashCode\(\)](#).

7.30.5 Event Documentation

7.30.5.1 OnChange

```
ChangeEventHandler<IPassword> ahbsd.lib.Password.Password.OnChange
```

Happenes, when the [Value](#) changes.

Referenced by [ahbsd.lib.Password.Password.Password\(\)](#).

The documentation for this class was generated from the following file:

- [ahbsd.lib/Password/Password.cs](#)

7.31 ahbsd.lib.Tools.PrimeCheck Class Reference

Class for checking, wheather a given number is a prime number or not.

Static Public Member Functions

- static bool [Check](#) (ulong nr)
Checks wheather nr is a prime number or not.

Properties

- static TimeSpan [Time](#) [get]
Gets the time it took for the last [Check\(ulong\)](#).
- static uint [Amount](#) [get]
Gets the amount of loops it took for the last [Check\(ulong\)](#).

7.31.1 Detailed Description

Class for checking, wheather a given number is a prime number or not.

7.31.2 Member Function Documentation

7.31.2.1 Check()

```
static bool ahbsd.lib.Tools.PrimeCheck.Check (  
    ulong nr ) [static]
```

Checks wheather nr is a prime number or not.

Parameters

<i>nr</i>	The number to check.
-----------	----------------------

Returns

`true` when nr is a prime number, otherwise `false`.

References [ahbsd.lib.Tools.PrimeCheck.Amount](#), and [ahbsd.lib.Tools.PrimeCheck.Time](#).

7.31.3 Property Documentation

7.31.3.1 Amount

```
uint ahbsd.lib.Tools.PrimeCheck.Amount [static], [get]
```

Gets the amount of loops it took for the last [Check\(ulong\)](#).

The amount of loops it took for the last [Check\(ulong\)](#).

Referenced by [ahbsd.lib.Tools.PrimeCheck.Check\(\)](#).

7.31.3.2 Time

```
TimeSpan ahbsd.lib.Tools.PrimeCheck.Time [static], [get]
```

Gets the time it took for the last [Check\(ulong\)](#).

The time it took for the last [Check\(ulong\)](#).

Referenced by [ahbsd.lib.Tools.PrimeCheck.Check\(\)](#).

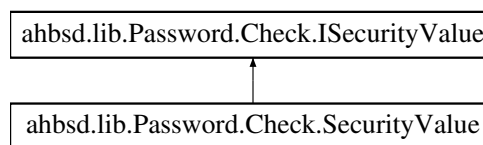
The documentation for this class was generated from the following file:

- [ahbsd.lib/Tools/PrimeCheck.cs](#)

7.32 ahbsd.lib.Password.Check.SecurityValue Class Reference

Component for the Security value of an password.

Inheritance diagram for [ahbsd.lib.Password.Check.SecurityValue](#):



Public Member Functions

- [SecurityValue](#) ()
Constructor without any parameters.
- [SecurityValue](#) (string password)
Constructor with a given password.
- [SecurityValue](#) (IPassword password)
Constructor with a given password.
- [SecurityValue](#) (IContainer container)
Constructor with a given owning container.
- [SecurityValue](#) (string password, IContainer container)
Constructor with a given password and a given owning container.
- [SecurityValue](#) (IPassword password, IContainer container)
Constructor with a given password and a given owning container.
- override string [ToString](#) ()
Gets the Security Value.
- override bool [Equals](#) (object obj)
Checks wheather an other object is equal to this object.
- bool [Equals](#) (ISecurityValue other)
Checks wheather an other ISecurityValue object is equal to this object.
- override int [GetHashCode](#) ()
Gets the hash code.

Static Public Member Functions

- static bool `operator==` ([SecurityValue](#) left, [SecurityValue](#) right)
Checks wheather two objects of type [SecurityValue](#) do eaquals each other.
- static bool `operator!=` ([SecurityValue](#) left, [SecurityValue](#) right)
Checks wheather two objects of type [SecurityValue](#) do not eaquals each other.

Properties

- [IPassword](#) `Password` [get]
Gets the password.
- ushort [Security](#) [get]
Gets the security.

7.32.1 Detailed Description

Component for the Security value of an password.

7.32.2 Constructor & Destructor Documentation

7.32.2.1 `SecurityValue()` [1/6]

```
ahbsd.lib.Password.Check.SecurityValue.SecurityValue ( )
```

Constructor without any parameters.

7.32.2.2 `SecurityValue()` [2/6]

```
ahbsd.lib.Password.Check.SecurityValue.SecurityValue (
    string password )
```

Constructor with a given password.

Parameters

<code>password</code>	The given password.
-----------------------	---------------------

References [ahbsd.lib.Password.Password.Password\(\)](#).

7.32.2.3 SecurityValue() [3/6]

```
ahbsd.lib.Password.Check.SecurityValue.SecurityValue (
    IPassword password )
```

Constructor with a given password.

Parameters

<i>password</i>	The given password.
-----------------	---------------------

7.32.2.4 SecurityValue() [4/6]

```
ahbsd.lib.Password.Check.SecurityValue.SecurityValue (
    IContainer container )
```

Constructor with a given owning container.

Parameters

<i>container</i>	The given owning container.
------------------	-----------------------------

7.32.2.5 SecurityValue() [5/6]

```
ahbsd.lib.Password.Check.SecurityValue.SecurityValue (
    string password,
    IContainer container )
```

Constructor with a given password and a given owning container.

Parameters

<i>password</i>	The given password.
<i>container</i>	The given owning container.

References [ahbsd.lib.Password.Password.Password\(\)](#).

7.32.2.6 SecurityValue() [6/6]

```
ahbsd.lib.Password.Check.SecurityValue.SecurityValue (
    IPassword password,
    IContainer container )
```

Constructor with a given password and a given owning container.

Parameters

<i>password</i>	The given password.
<i>container</i>	The given owning container.

7.32.3 Member Function Documentation

7.32.3.1 Equals() [1/2]

```
bool ahbsd.lib.Password.Check.SecurityValue.Equals (
    ISecurityValue other )
```

Checks wheather an other [ISecurityValue](#) object is equal to this object.

Parameters

<i>other</i>	The other ISecurityValue object.
--------------	--

Returns

`true` if the other [ISecurityValue](#) object eaquals this object, otherwise `false`.

7.32.3.2 Equals() [2/2]

```
override bool ahbsd.lib.Password.Check.SecurityValue.Equals (
    object obj )
```

Checks wheather an other object is equal to this object.

Parameters

<i>obj</i>	The other object.
------------	-------------------

Returns

`true` if the other object eaquals this object, otherwise `false`.

7.32.3.3 GetHashCode()

```
override int ahbsd.lib.Password.Check.SecurityValue.GetHashCode ( )
```

Gets the hash code.

Returns

The hash code.

7.32.3.4 operator"!="()

```
static bool ahbsd.lib.Password.Check.SecurityValue.operator!= (
    SecurityValue left,
    SecurityValue right ) [static]
```

Checks wheather two objects of type [SecurityValue](#) do not eaquals each other.

Parameters

<i>left</i>	The object on the left side.
<i>right</i>	The object on the right side.

Returns

`true` if both objects are not equal to each other, otherwise `false`.

7.32.3.5 operator=="()

```
static bool ahbsd.lib.Password.Check.SecurityValue.operator== (
    SecurityValue left,
    SecurityValue right ) [static]
```

Checks wheather two objects of type [SecurityValue](#) do eaquals each other.

Parameters

<i>left</i>	The object on the left side.
<i>right</i>	The object on the right side.

Returns

`true` if both objects are equal to each other, otherwise `false`.

7.32.3.6 ToString()

```
override string ahbsd.lib.Password.Check.SecurityValue.ToString ( )
```

Gets the Security Value.

Returns

The Security Value.

7.32.4 Property Documentation**7.32.4.1 Password**

```
IPassword ahbsd.lib.Password.Check.SecurityValue.Password [get]
```

Gets the password.

The password.

7.32.4.2 Security

```
ushort ahbsd.lib.Password.Check.SecurityValue.Security [get]
```

Gets the security.

The security value.

The higher, the better.

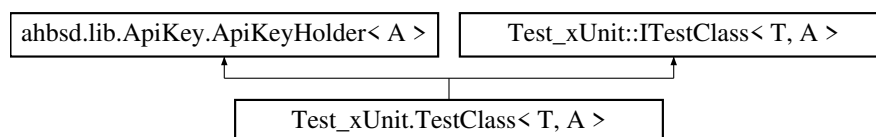
The documentation for this class was generated from the following file:

- ahbsd.lib/Password/Check/SecurityValue.cs

7.33 Test_xUnit.TestClass< T, A > Class Template Reference

A class to demonstrate ChangeEventArgs<T> and ChangeEventHandler<T>.

Inheritance diagram for Test_xUnit.TestClass< T, A >:

**Public Member Functions**

- [TestClass](#) ()
Constructor without parameters.
- [TestClass](#) (T v)
Constructor with a given variable.
- [TestClass](#) (A apiKey)
Constructor with a given API-Key.
- [TestClass](#) (T v, A apiKey)
Constructor with a given value and a given API-Key.

Properties

- T?? [Variable](#) [get, set]
Gets or sets a variable.

Events

- ChangeEventHandler< T > [OnChange](#)
Happenes when [Variable](#) has changed.

Additional Inherited Members

7.33.1 Detailed Description

A class to demonstrate ChangeEventArgs<T> and ChangeEventHandler<T>.

Template Parameters

<i>T</i>	Type of Variable .
<i>A</i>	Type of API-Key.

7.33.2 Constructor & Destructor Documentation

7.33.2.1 TestClass() [1/4]

```
Test_xUnit.TestClass< T, A >.TestClass ( )
```

Constructor without parameters.

References [ahbsd.lib.ApiKey.ApiKeyHolder< A >.OnApiKeyAdded](#).

7.33.2.2 TestClass() [2/4]

```
Test_xUnit.TestClass< T, A >.TestClass (
    T v )
```

Constructor with a given variable.

Parameters

<i>v</i>	The given variable.
----------	---------------------

References [ahbsd.lib.ApiKey.ApiKeyHolder< A >.OnApiKeyAdded](#).

7.33.2.3 TestClass() [3/4]

```
Test_xUnit.TestClass< T, A >.TestClass (
    A apiKey )
```

Constructor with a given API-Key.

Parameters

<i>apiKey</i>	The given API-Key.
---------------	--------------------

References [ahbsd.lib.ApiKey.ApiKeyHolder< A >.OnApiKeyAdded](#).

7.33.2.4 TestClass() [4/4]

```
Test_xUnit.TestClass< T, A >.TestClass (
    T v,
    A apiKey )
```

Constructor with a given value and a given API-Key.

Parameters

<i>v</i>	The given variable.
<i>apiKey</i>	The given API-Key.

References [ahbsd.lib.ApiKey.ApiKeyHolder< A >.OnApiKeyAdded](#).

7.33.3 Property Documentation

7.33.3.1 Variable

```
T?? Test_xUnit.TestClass< T, A >.Variable [get], [set]
```

Gets or sets a variable.

7.33.4 Event Documentation

7.33.4.1 OnChange

ChangeEventHandler<T> [Test_xUnit.TestClass](#)< T, A >.OnChange

Happenes when [Variable](#) has changed.

The documentation for this class was generated from the following file:

- [Test_xUnit/TestClass.cs](#)

Index

Add

- ahbsd.lib.NamedCollections.CompanyModelProducts, 48, 49
- ahbsd.lib.NamedCollections.DictionaryOfNamedCollection< K, V >, 55, 56
- ahbsd.lib.NamedCollections.DictionaryOfNamedList< K, V >, 58, 59
- ahbsd.lib.NamedCollections.IDictionaryOfNamedCollection< K, V >, 74, 75
- ahbsd.lib.NamedCollections.IDictionaryOfNamedList< K, V >, 76, 77
- ahbsd.lib.Password.CharacteristicDictionary, 35, 36
- AddAddress
 - ahbsd.lib.Products.ICompany, 70
- Adress
 - ahbsd.lib.Products.IAddress, 64
- Adresses
 - ahbsd.lib.Products.ICompany, 71
- AdressType
 - ahbsd.lib.Products, 17
 - ahbsd.lib.Products.IAddress, 64
- ahbsd, 13
- ahbsd.lib, 13
 - ChangeEventHandler< T >, 13
- ahbsd.lib.ApiKey, 14
 - ApiKeyEventHandler< T >, 14
- ahbsd.lib.ApiKey.ApiKeyEventArgs< T >, 19
 - ApiKeyEventArgs, 20
 - Index, 20
- ahbsd.lib.ApiKey.ApiKeyHolder< T >, 21
 - ApiKey, 26
 - ApiKeyHolder, 23
 - Equals, 23, 24
 - FindApiKey, 24
 - GetApiKey, 24
 - GetHashCode, 25
 - KnownApiKeys, 26
 - OnApiKeyAdded, 27
 - operator!=, 25
 - operator==, 26
- ahbsd.lib.ApiKey.IApiKeyEventArgs, 66
 - Index, 66
- ahbsd.lib.ChangeEventArgs< T >, 27
 - ChangeEventArgs, 28, 29
 - Equals, 29
 - GetHashCode, 30
 - NewValue, 32
 - OldValue, 32
 - operator!=, 30
 - operator==, 31
 - SetNewValue, 31
 - ToString, 31
- ahbsd.lib.EventArgs< T >, 60
 - EventArgs, 60
 - Value, 61
- ahbsd.lib.Exceptions, 15
 - ahbsd.lib.Exceptions.Exception< T >, 61
 - Exception, 62
 - Value, 63
 - ahbsd.lib.Exceptions.IGenericException< T >, 79
 - Value, 80
- ahbsd.lib.IChangeEventArgs< T >, 66
 - NewValue, 68
 - OldValue, 68
 - SetNewValue, 67
 - ToString, 67
- ahbsd.lib.IEventArgs< T >, 78
 - Value, 79
- ahbsd.lib.NamedCollections, 15
- ahbsd.lib.NamedCollections.CompanyModelProducts, 46
 - Add, 48, 49
 - Clear, 49
 - CompanyModelProducts, 47
 - Contains, 49
 - ContainsKey, 50
 - CopyTo, 50
 - Count, 52
 - GetEnumerator, 51
 - IsReadOnly, 52
 - Keys, 52
 - Remove, 51
 - this[ICompany key], 53
 - TryGetValue, 52
 - Values, 53
- ahbsd.lib.NamedCollections.DictionaryOfNamedCollection< K, V >, 54
 - Add, 55, 56
 - OnNamedCollectionAdded, 56
- ahbsd.lib.NamedCollections.DictionaryOfNamedList< K, V >, 57
 - Add, 58, 59
 - OnNamedListAdded, 59
- ahbsd.lib.NamedCollections.IDictionaryOfNamedCollections< K, V >, 73
 - Add, 74, 75
 - OnNamedCollectionAdded, 75

- ahbsd.lib.NamedCollections.IDictionaryOfNamedList< K, V >, 75
 - Add, 76, 77
 - OnNamedListAdded, 78
- ahbsd.lib.NamedCollections.INamedCollection< T >, 81
 - Name, 82
 - OnNameChanged, 83
 - ToString, 82
- ahbsd.lib.NamedCollections.INamedList< T >, 83
 - Name, 84
 - OnNameChanged, 84
 - ToString, 84
- ahbsd.lib.NamedCollections.NamedCollection< T >, 91
 - Name, 93
 - NamedCollection, 92
 - OnNameChanged, 93
 - ToString, 93
- ahbsd.lib.NamedCollections.NamedList< T >, 94
 - Name, 97
 - NamedList, 95, 96
 - OnNameChanged, 97
 - ToString, 97
- ahbsd.lib.Password, 15
 - CapitalLetter, 16
 - Charasteristic, 16
 - LowercaseLetter, 16
 - Numeric, 16
 - Space, 16
 - SpecialCharacter, 16
- ahbsd.lib.Password.CharacteristicDictionary, 32
 - Add, 35, 36
 - CharacteristicDictionary, 34, 35
 - Clear, 36
 - Contains, 36
 - ContainsKey, 36
 - CopyTo, 37
 - Count, 39
 - GetEnumerator, 37
 - IsReadOnly, 39
 - Keys, 39
 - Name, 39
 - Remove, 37, 38
 - this[Charasteristic key], 39
 - ToString, 38
 - TryGetValue, 38
 - Values, 40
- ahbsd.lib.Password.Check, 16
- ahbsd.lib.Password.Check.ISecurityValue, 89
 - Password, 89
 - Security, 89
- ahbsd.lib.Password.Check.SecurityValue, 109
 - Equals, 112
 - GetHashCode, 112
 - operator!=, 113
 - operator==, 113
 - Password, 114
 - Security, 114
 - SecurityValue, 110, 111
 - ToString, 113
- ahbsd.lib.Password.ICharacteristicDictionary, 68
 - Name, 69
 - ToString, 69
- ahbsd.lib.Password.IPassword, 85
 - Characteristics, 85
 - Length, 86
 - LowerCases, 86
 - Numbers, 86
 - OnChange, 87
 - SecurityValue, 86
 - Spaces, 86
 - Specials, 86
 - UpperCases, 87
 - Value, 87
- ahbsd.lib.Password.Password, 98
 - Characteristics, 106
 - Equals, 100, 101
 - GetCharasteristic, 101
 - GetCharasteristics, 101
 - GetHashCode, 102
 - GetLowerCases, 102
 - GetNumbers, 102
 - GetPassword, 103
 - GetSpaces, 104
 - GetSpecials, 104
 - GetUpperCases, 104
 - Length, 106
 - LowerCases, 106
 - Numbers, 106
 - OnChange, 107
 - operator!=, 105
 - operator==, 105
 - Password, 99, 100
 - SecurityValue, 106
 - Spaces, 106
 - Specials, 107
 - UpperCases, 107
 - Value, 107
- ahbsd.lib.Products, 16
 - AdressType, 17
 - Delivery, 17
 - Other, 17
 - Postal, 17
 - Private, 17
 - Store, 17
- ahbsd.lib.Products.IAdress, 63
 - Address, 64
 - AdressType, 64
 - City, 64
 - Country, 65
 - OnAdressChanged, 65
 - OnTypeChanged, 65
 - ZIP, 65
- ahbsd.lib.Products.ICompany, 70
 - AddAddress, 70
 - Addresses, 71

- Name, [72](#)
- Note, [72](#)
- OnAdressAdded, [72](#)
- OnAdressRemoved, [72](#)
- OnNoteChanged, [72](#)
- RemoveAdress, [71](#)
- RemoveAdresses, [71](#)
- ahbsd.lib.Products.IModel, [80](#)
 - Name, [80](#)
 - Notes, [81](#)
 - OnNotesChanged, [81](#)
 - Producer, [81](#)
- ahbsd.lib.Products.IProductItem, [87](#)
 - Model, [88](#)
 - Name, [88](#)
 - SerialNumber, [88](#)
- ahbsd.lib.Tools, [17](#)
- ahbsd.lib.Tools.Checksum, [40](#)
 - GetChecksum, [41](#), [42](#)
 - GetFinalChecksum, [42](#), [43](#)
 - IsPositiv, [44](#)
 - IsPositive, [45](#), [46](#)
- ahbsd.lib.Tools.ConsolePrintTable, [53](#)
 - Print, [54](#)
- ahbsd.lib.Tools.PrimeCheck, [108](#)
 - Amount, [108](#)
 - Check, [108](#)
 - Time, [109](#)
- Amount
 - ahbsd.lib.Tools.PrimeCheck, [108](#)
- ApiKey
 - ahbsd.lib.ApiKey.ApiKeyHolder< T >, [26](#)
- ApiKeyEventArgs
 - ahbsd.lib.ApiKey.ApiKeyEventArgs< T >, [20](#)
- ApiKeyEventHandler< T >
 - ahbsd.lib.ApiKey, [14](#)
- ApiKeyHolder
 - ahbsd.lib.ApiKey.ApiKeyHolder< T >, [23](#)
- CapitalLetter
 - ahbsd.lib.Password, [16](#)
- ChangeEventArgs
 - ahbsd.lib.ChangeEventArgs< T >, [28](#), [29](#)
- ChangeEventHandler< T >
 - ahbsd.lib, [13](#)
- CharacteristicDictionary
 - ahbsd.lib.Password.CharacteristicDictionary, [34](#), [35](#)
- Characteristics
 - ahbsd.lib.Password.IPassword, [85](#)
 - ahbsd.lib.Password.Password, [106](#)
- Charasteristic
 - ahbsd.lib.Password, [16](#)
- Check
 - ahbsd.lib.Tools.PrimeCheck, [108](#)
- City
 - ahbsd.lib.Products.IAdress, [64](#)
- Clear
 - ahbsd.lib.NamedCollections.CompanyModelProducts, [49](#)
 - ahbsd.lib.Password.CharacteristicDictionary, [36](#)
- CompanyModelProducts
 - ahbsd.lib.NamedCollections.CompanyModelProducts, [47](#)
- Contains
 - ahbsd.lib.NamedCollections.CompanyModelProducts, [49](#)
 - ahbsd.lib.Password.CharacteristicDictionary, [36](#)
- ContainsKey
 - ahbsd.lib.NamedCollections.CompanyModelProducts, [50](#)
 - ahbsd.lib.Password.CharacteristicDictionary, [36](#)
- CopyTo
 - ahbsd.lib.NamedCollections.CompanyModelProducts, [50](#)
 - ahbsd.lib.Password.CharacteristicDictionary, [37](#)
- Count
 - ahbsd.lib.NamedCollections.CompanyModelProducts, [52](#)
 - ahbsd.lib.Password.CharacteristicDictionary, [39](#)
- Country
 - ahbsd.lib.Products.IAdress, [65](#)
- Delivery
 - ahbsd.lib.Products, [17](#)
- Equals
 - ahbsd.lib.ApiKey.ApiKeyHolder< T >, [23](#), [24](#)
 - ahbsd.lib.ChangeEventArgs< T >, [29](#)
 - ahbsd.lib.Password.Check.SecurityValue, [112](#)
 - ahbsd.lib.Password.Password, [100](#), [101](#)
- EventArgs
 - ahbsd.lib.EventArgs< T >, [60](#)
- Exception
 - ahbsd.lib.Exceptions.Exception< T >, [62](#)
- FindApiKey
 - ahbsd.lib.ApiKey.ApiKeyHolder< T >, [24](#)
- GetApiKey
 - ahbsd.lib.ApiKey.ApiKeyHolder< T >, [24](#)
- GetCharasteristic
 - ahbsd.lib.Password.Password, [101](#)
- GetCharasteristics
 - ahbsd.lib.Password.Password, [101](#)
- GetChecksum
 - ahbsd.lib.Tools.Checksum, [41](#), [42](#)
- GetEnumerator
 - ahbsd.lib.NamedCollections.CompanyModelProducts, [51](#)
 - ahbsd.lib.Password.CharacteristicDictionary, [37](#)
- GetFinalChecksum
 - ahbsd.lib.Tools.Checksum, [42](#), [43](#)
- GetHashCode
 - ahbsd.lib.ApiKey.ApiKeyHolder< T >, [25](#)
 - ahbsd.lib.ChangeEventArgs< T >, [30](#)
 - ahbsd.lib.Password.Check.SecurityValue, [112](#)

- ahbsd.lib.Password.Password, 102
- GetLowerCases
 - ahbsd.lib.Password.Password, 102
- GetNumbers
 - ahbsd.lib.Password.Password, 102
- GetPassword
 - ahbsd.lib.Password.Password, 103
- GetSpaces
 - ahbsd.lib.Password.Password, 104
- GetSpecials
 - ahbsd.lib.Password.Password, 104
- GetUpperCases
 - ahbsd.lib.Password.Password, 104
- Index
 - ahbsd.lib.ApiKey.ApiKeyEventArgs< T >, 20
 - ahbsd.lib.ApiKey.IApiKeyEventArgs, 66
- IsPositiv
 - ahbsd.lib.Tools.Checksum, 44
- IsPositive
 - ahbsd.lib.Tools.Checksum, 45, 46
- IsReadOnly
 - ahbsd.lib.NamedCollections.CompanyModelProducts, 52
 - ahbsd.lib.Password.CharacteristicDictionary, 39
- Keys
 - ahbsd.lib.NamedCollections.CompanyModelProducts, 52
 - ahbsd.lib.Password.CharacteristicDictionary, 39
- KnownApiKeys
 - ahbsd.lib.ApiKey.ApiKeyHolder< T >, 26
- Length
 - ahbsd.lib.Password.IPassword, 86
 - ahbsd.lib.Password.Password, 106
- LowercaseLetter
 - ahbsd.lib.Password, 16
- LowerCases
 - ahbsd.lib.Password.IPassword, 86
 - ahbsd.lib.Password.Password, 106
- Model
 - ahbsd.lib.Products.IProductItem, 88
- Name
 - ahbsd.lib.NamedCollections.INamedCollection< T >, 82
 - ahbsd.lib.NamedCollections.INamedList< T >, 84
 - ahbsd.lib.NamedCollections.NamedCollection< T >, 93
 - ahbsd.lib.NamedCollections.NamedList< T >, 97
 - ahbsd.lib.Password.CharacteristicDictionary, 39
 - ahbsd.lib.Password.ICharacteristicDictionary, 69
 - ahbsd.lib.Products.ICompany, 72
 - ahbsd.lib.Products.IModel, 80
 - ahbsd.lib.Products.IProductItem, 88
- NamedCollection
 - ahbsd.lib.NamedCollections.NamedCollection< T >, 92
- NamedList
 - ahbsd.lib.NamedCollections.NamedList< T >, 95, 96
- NewValue
 - ahbsd.lib.ChangeEventArgs< T >, 32
 - ahbsd.lib.IChangeEventArgs< T >, 68
- Note
 - ahbsd.lib.Products.ICompany, 72
- Notes
 - ahbsd.lib.Products.IModel, 81
- Numbers
 - ahbsd.lib.Password.IPassword, 86
 - ahbsd.lib.Password.Password, 106
- Numeric
 - ahbsd.lib.Password, 16
- OldValue
 - ahbsd.lib.ChangeEventArgs< T >, 32
 - ahbsd.lib.IChangeEventArgs< T >, 68
- OnAdressAdded
 - ahbsd.lib.Products.ICompany, 72
- OnAdressChanged
 - ahbsd.lib.Products.IAdress, 65
- OnAdressRemoved
 - ahbsd.lib.Products.ICompany, 72
- OnApiKeyAdded
 - ahbsd.lib.ApiKey.ApiKeyHolder< T >, 27
- OnChange
 - ahbsd.lib.Password.IPassword, 87
 - ahbsd.lib.Password.Password, 107
 - Test_xUnit.ITestClass< T, A >, 90
 - Test_xUnit.TestClass< T, A >, 116
- OnNameChanged
 - ahbsd.lib.NamedCollections.INamedCollection< T >, 83
 - ahbsd.lib.NamedCollections.INamedList< T >, 84
 - ahbsd.lib.NamedCollections.NamedCollection< T >, 93
 - ahbsd.lib.NamedCollections.NamedList< T >, 97
- OnNamedCollectionAdded
 - ahbsd.lib.NamedCollections.DictionaryOfNamedCollection< K, V >, 56
 - ahbsd.lib.NamedCollections.IDictionaryOfNamedCollections< K, V >, 75
- OnNamedListAdded
 - ahbsd.lib.NamedCollections.DictionaryOfNamedList< K, V >, 59
 - ahbsd.lib.NamedCollections.IDictionaryOfNamedList< K, V >, 78
- OnNoteChanged
 - ahbsd.lib.Products.ICompany, 72
- OnNotesChanged
 - ahbsd.lib.Products.IModel, 81
- OnTypeChanged
 - ahbsd.lib.Products.IAdress, 65
- operator!=
 - ahbsd.lib.ApiKey.ApiKeyHolder< T >, 25
 - ahbsd.lib.ChangeEventArgs< T >, 30
 - ahbsd.lib.Password.Check.SecurityValue, 113

- ahbsd.lib.Password.Password, 105
- operator==
 - ahbsd.lib.ApiKey.ApiKeyHolder< T >, 26
 - ahbsd.lib.ChangeEventArgs< T >, 31
 - ahbsd.lib.Password.Check.SecurityValue, 113
 - ahbsd.lib.Password.Password, 105
- Other
 - ahbsd.lib.Products, 17
- Password
 - ahbsd.lib.Password.Check.ISecurityValue, 89
 - ahbsd.lib.Password.Check.SecurityValue, 114
 - ahbsd.lib.Password.Password, 99, 100
- Postal
 - ahbsd.lib.Products, 17
- Print
 - ahbsd.lib.Tools.ConsolePrintTable, 54
- Private
 - ahbsd.lib.Products, 17
- Producer
 - ahbsd.lib.Products.IModel, 81
- Remove
 - ahbsd.lib.NamedCollections.CompanyModelProducts, 51
 - ahbsd.lib.Password.CharacteristicDictionary, 37, 38
- RemoveAdress
 - ahbsd.lib.Products.ICompany, 71
- RemoveAdresses
 - ahbsd.lib.Products.ICompany, 71
- Security
 - ahbsd.lib.Password.Check.ISecurityValue, 89
 - ahbsd.lib.Password.Check.SecurityValue, 114
- SecurityValue
 - ahbsd.lib.Password.Check.SecurityValue, 110, 111
 - ahbsd.lib.Password.IPassword, 86
 - ahbsd.lib.Password.Password, 106
- SerialNumber
 - ahbsd.lib.Products.IProductItem, 88
- SetNewValue
 - ahbsd.lib.ChangeEventArgs< T >, 31
 - ahbsd.lib.IChangeEventArgs< T >, 67
- Space
 - ahbsd.lib.Password, 16
- Spaces
 - ahbsd.lib.Password.IPassword, 86
 - ahbsd.lib.Password.Password, 106
- SpecialCharacter
 - ahbsd.lib.Password, 16
- Specials
 - ahbsd.lib.Password.IPassword, 86
 - ahbsd.lib.Password.Password, 107
- Store
 - ahbsd.lib.Products, 17
- Test_xUnit, 17
- Test_xUnit.ITestClass< T, A >, 90
- OnChange, 90
- Variable, 90
- Test_xUnit.TestClass< T, A >, 114
 - OnChange, 116
 - TestClass, 115, 116
 - Variable, 116
- TestClass
 - Test_xUnit.TestClass< T, A >, 115, 116
- this[Charasteristic key]
 - ahbsd.lib.Password.CharacteristicDictionary, 39
- this[ICompany key]
 - ahbsd.lib.NamedCollections.CompanyModelProducts, 53
- Time
 - ahbsd.lib.Tools.PrimeCheck, 109
- ToString
 - ahbsd.lib.ChangeEventArgs< T >, 31
 - ahbsd.lib.IChangeEventArgs< T >, 67
 - ahbsd.lib.NamedCollections.INamedCollection< T >, 82
 - ahbsd.lib.NamedCollections.INamedList< T >, 84
 - ahbsd.lib.NamedCollections.NamedCollection< T >, 93
 - ahbsd.lib.NamedCollections.NamedList< T >, 97
 - ahbsd.lib.Password.CharacteristicDictionary, 38
 - ahbsd.lib.Password.Check.SecurityValue, 113
 - ahbsd.lib.Password.ICharacteristicDictionary, 69
- TryGetValue
 - ahbsd.lib.NamedCollections.CompanyModelProducts, 52
 - ahbsd.lib.Password.CharacteristicDictionary, 38
- UpperCases
 - ahbsd.lib.Password.IPassword, 87
 - ahbsd.lib.Password.Password, 107
- Value
 - ahbsd.lib.EventArgs< T >, 61
 - ahbsd.lib.Exceptions.Exception< T >, 63
 - ahbsd.lib.Exceptions.IGenericException< T >, 80
 - ahbsd.lib.IEventArgs< T >, 79
 - ahbsd.lib.Password.IPassword, 87
 - ahbsd.lib.Password.Password, 107
- Values
 - ahbsd.lib.NamedCollections.CompanyModelProducts, 53
 - ahbsd.lib.Password.CharacteristicDictionary, 40
- Variable
 - Test_xUnit.ITestClass< T, A >, 90
 - Test_xUnit.TestClass< T, A >, 116
- ZIP
 - ahbsd.lib.Products.IAdress, 65