Type References

1.0.0

Generated by Doxygen 1.9.5

Chapter 1

Namespace Index

1.1 Package List

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

Chapter 2

Hierarchical Index

2.1 Class Hierarchy

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

Chapter 3

Class Index

3.1 Class List

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

Chapter 4

Namespace Documentation

4.1 TypeReferences Namespace Reference

Classes

- class ClassExtendsAttribute

  Constraint that allows selection of classes that extend a specific class when selecting a ClassTypeReference with the Unity inspector.
- class ClassImplementsAttribute

  Constraint that allows selection of classes that implement a specific interface when selecting a ClassTypeReference with the Unity inspector.
- class ClassTypeConstraintAttribute

  Base class for class selection constraints that can be applied when selecting a ClassTypeReference with the Unity inspector.
- class ClassTypeReference

  Reference to a class System.Type with support for Unity serialization.

Enumerations

- enum ClassGrouping

  Indicates how selectable classes should be collated in drop-down menu.

4.2 TypeReferences.Editor Namespace Reference

Classes

- class **ClassTypeReferenceEditorSettings**

  A public static class that provides an easy way for programs to modify included assemblies in 'type references'.
- class ClassTypeReferencePropertyDrawer

  Custom property drawer for ClassTypeReference properties.

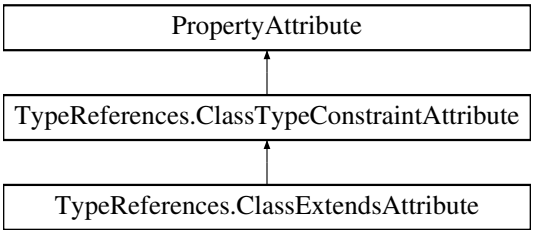
Chapter 5

Class Documentation

5.1 TypeReferences.ClassExtendsAttribute Class Reference

Constraint that allows selection of classes that extend a specific class when selecting a ClassTypeReference with the Unity inspector.

Inheritance diagram for TypeReferences.ClassExtendsAttribute:



Public Member Functions

- **ClassExtendsAttribute** ()

  Initializes a new instance of the ClassExtendsAttribute class.
- ClassExtendsAttribute (Type baseType)

  Initializes a new instance of the ClassExtendsAttribute class.
- override bool IsConstraintSatisfied (Type type)

  Determines whether the specified Type satisfies filter constraint.

Parameters

type	*Type to test.*

Returns

A bool value indicating if the type specified by type satisfies this constraint and should thus be selectable.

Properties

- Type **BaseType** `[get]`

  Gets the type of class that selectable classes must derive from.

5.1.1 Detailed Description

Constraint that allows selection of classes that extend a specific class when selecting a ClassTypeReference with the Unity inspector.

5.1.2 Constructor & Destructor Documentation

5.1.2.1 ClassExtendsAttribute()

```
TypeReferences.ClassExtendsAttribute.ClassExtendsAttribute (
            Type baseType )
```

Initializes a new instance of the ClassExtendsAttribute class.

Parameters

baseType	Type of class that selectable classes must derive from.

5.1.3 Member Function Documentation

5.1.3.1 IsConstraintSatisfied()

```
override bool TypeReferences.ClassExtendsAttribute.IsConstraintSatisfied (
            Type type )  [virtual]
```

Determines whether the specified Type satisfies filter constraint.

Parameters

type	Type to test.

Returns

A bool value indicating if the type specified by *type* satisfies this constraint and should thus be selectable.

Reimplemented from TypeReferences.ClassTypeConstraintAttribute.

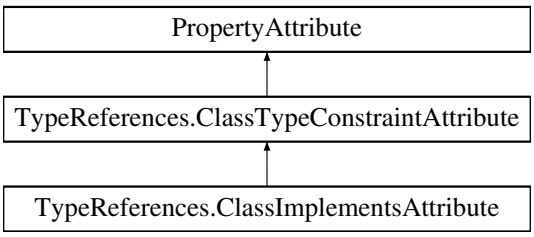
The documentation for this class was generated from the following file:

- ClassTypeConstraintAttribute.cs

5.2 TypeReferences.ClassImplementsAttribute Class Reference

Constraint that allows selection of classes that implement a specific interface when selecting a ClassTypeReference with the Unity inspector.

Inheritance diagram for TypeReferences.ClassImplementsAttribute:



Public Member Functions

- **ClassImplementsAttribute** ()

  Initializes a new instance of the ClassImplementsAttribute class.
- ClassImplementsAttribute (Type interfaceType)

  Initializes a new instance of the ClassImplementsAttribute class.
- override bool IsConstraintSatisfied (Type type)

  Determines whether the specified Type satisfies filter constraint.

Parameters

type	*Type to test.*

Returns

A bool value indicating if the type specified by type satisfies this constraint and should thus be selectable.

Properties

- Type **InterfaceType** `[get]`

  Gets the type of interface that selectable classes must implement.

5.2.1 Detailed Description

Constraint that allows selection of classes that implement a specific interface when selecting a ClassTypeReference with the Unity inspector.

5.2.2 Constructor & Destructor Documentation

5.2.2.1 ClassImplementsAttribute()

```
TypeReferences.ClassImplementsAttribute.ClassImplementsAttribute (
            Type interfaceType )
```

Initializes a new instance of the ClassImplementsAttribute class.

Parameters

interfaceType	Type of interface that selectable classes must implement.

5.2.3 Member Function Documentation

5.2.3.1 IsConstraintSatisfied()

```
override bool TypeReferences.ClassImplementsAttribute.IsConstraintSatisfied (
            Type type )  [virtual]
```

Determines whether the specified Type satisfies filter constraint.

Parameters

type	Type to test.

Returns

A bool value indicating if the type specified by *type* satisfies this constraint and should thus be selectable.

Reimplemented from TypeReferences.ClassTypeConstraintAttribute.

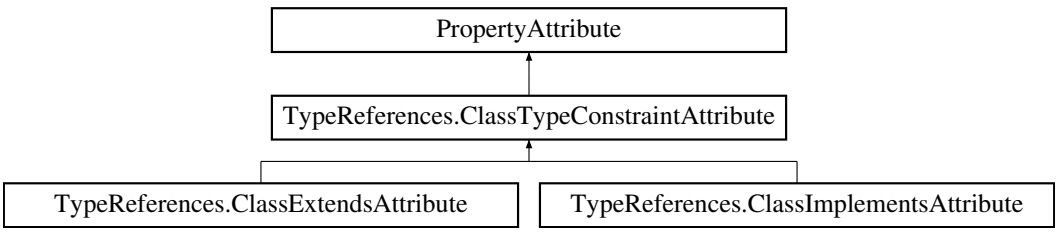
The documentation for this class was generated from the following file:

- ClassTypeConstraintAttribute.cs

5.3 TypeReferences.ClassTypeConstraintAttribute Class Reference

Base class for class selection constraints that can be applied when selecting a ClassTypeReference with the Unity inspector.

Inheritance diagram for TypeReferences.ClassTypeConstraintAttribute:



Public Member Functions

- virtual bool IsConstraintSatisfied (Type type)

  Determines whether the specified Type satisfies filter constraint.

Properties

- ClassGrouping **Grouping** `[get, set]`

  Gets or sets grouping of selectable classes. Defaults to ClassGrouping.ByNamespaceFlat unless explicitly specified.
- bool **AllowAbstract** `[get, set]`

  Gets or sets whether abstract classes can be selected from drop-down. Defaults to a value of `false` *unless explicitly specified.*

5.3.1 Detailed Description

Base class for class selection constraints that can be applied when selecting a ClassTypeReference with the Unity inspector.

5.3.2 Member Function Documentation

5.3.2.1 IsConstraintSatisfied()

```
virtual bool TypeReferences.ClassTypeConstraintAttribute.IsConstraintSatisfied (
            Type type )  [virtual]
```

Determines whether the specified Type satisfies filter constraint.

Parameters

type	Type to test.

Returns

A bool value indicating if the type specified by *type* satisfies this constraint and should thus be selectable.

Reimplemented in TypeReferences.ClassExtendsAttribute, and TypeReferences.ClassImplementsAttribute.

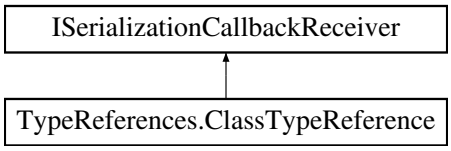
The documentation for this class was generated from the following file:

- ClassTypeConstraintAttribute.cs

5.4 TypeReferences.ClassTypeReference Class Reference

Reference to a class System.Type with support for Unity serialization.

Inheritance diagram for TypeReferences.ClassTypeReference:



Public Member Functions

- **ClassTypeReference** ()

  Initializes a new instance of the ClassTypeReference class.
- ClassTypeReference (string assemblyQualifiedClassName)

  Initializes a new instance of the ClassTypeReference class.
- ClassTypeReference (Type type)

  Initializes a new instance of the ClassTypeReference class.
- override string **ToString** ()

Static Public Member Functions

- static string **GetClassRef** (Type type)
- static implicit **operator string** (ClassTypeReference typeReference)
- static implicit **operator Type** (ClassTypeReference typeReference)
- static implicit **operator ClassTypeReference** (Type type)

Properties

- Type Type `[get, set]`

  Gets or sets type of class reference.

5.4.1 Detailed Description

Reference to a class System.Type with support for Unity serialization.

5.4.2 Constructor & Destructor Documentation

5.4.2.1 ClassTypeReference() `[1/2]`

```
TypeReferences.ClassTypeReference.ClassTypeReference (
            string assemblyQualifiedClassName )
```

Initializes a new instance of the ClassTypeReference class.

Parameters

assemblyQualifiedClassName	Assembly qualified class name.

5.4.2.2 ClassTypeReference() [2/2]

```
TypeReferences.ClassTypeReference.ClassTypeReference (
            Type type )
```

Initializes a new instance of the ClassTypeReference class.

Parameters

type	Class type.

Exceptions

System.ArgumentException	If *type* is not a class type.

5.4.3 Property Documentation

5.4.3.1 Type

```
Type TypeReferences.ClassTypeReference.Type  [get], [set]
```

Gets or sets type of class reference.

Exceptions

System.ArgumentException	If *value* is not a class type.

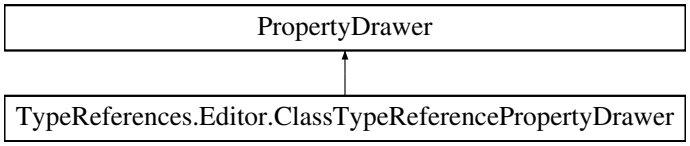
The documentation for this class was generated from the following file:

- ClassTypeReference.cs

5.5 TypeReferences.Editor.ClassTypeReferencePropertyDrawer Class Reference

Custom property drawer for ClassTypeReference properties.

Inheritance diagram for TypeReferences.Editor.ClassTypeReferencePropertyDrawer:



Public Member Functions

- override float **GetPropertyHeight** (SerializedProperty property, GUIContent label)
- override void **OnGUI** (Rect position, SerializedProperty property, GUIContent label)

Properties

- static Func< ICollection< Type > > ExcludedTypeCollectionGetter `[get, set]`

  Gets or sets a function that returns a collection of types that are to be excluded from drop-down. A value of `null` *specifies that no types are to be excluded.*

5.5.1 Detailed Description

Custom property drawer for ClassTypeReference properties.

5.5.2 Property Documentation

5.5.2.1 ExcludedTypeCollectionGetter

`Func<ICollection<Type> > TypeReferences.Editor.ClassTypeReferencePropertyDrawer.Excluded↩TypeCollectionGetter [static], [get], [set]`

Gets or sets a function that returns a collection of types that are to be excluded from drop-down. A value of `null` specifies that no types are to be excluded.

This property must be set immediately before presenting a class type reference property field using EditorGUI.↩PropertyField or EditorGUILayout.PropertyField since the value of this property is reset to `null` each time the control is drawn.

Since filtering makes extensive use of ICollection<Type>.Contains it is recommended to use a collection that is optimized for fast lookups such as HashSet<Type> for better performance.

Exclude a specific type from being selected:

```
private SerializedProperty _someClassTypeReferenceProperty;
public override void OnInspectorGUI() {
    serializedObject.Update();
    ClassTypeReferencePropertyDrawer.ExcludedTypeCollectionGetter = GetExcludedTypeCollection;
    EditorGUILayout.PropertyField(_someClassTypeReferenceProperty);
    serializedObject.ApplyModifiedProperties();
}
private ICollection\<Type\> GetExcludedTypeCollection() {
    var set = new HashSet\<Type\>();
    set.Add(typeof(SpecialClassToHideInDropdown));
    return set;
}
```

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

- ClassTypeReferencePropertyDrawer.cs

Index