# Namespace com.absence.variablebanks

## Classes

### FixedVariableComparer

Comparer with a fixed bank.

### FixedVariableSetter

Setter with a fixed bank.

### VariableBank

The scriptable object represents a bank of variables.

### VariableBankAcquirer

A component to reference banks both in editor and runtime.

### VariableBankReference

The class responsible for letting you reference a `VariableBank` both in editor and in runtime. You can use the `VariableBank` class directly if the bank you are referencing is marked as `ForExternalUse`. For more information, read the docs.

### VariableComparer

Comparer with a dynamic bank you select in editor.

### VariableSetter

Setter with a dynamic bank you select in the editor.

Edit this page

# Class FixedVariableComparer

Comparer with a fixed bank.

### Inheritance

↳ object
  ↳ BaseVariableComparer
    ↳ FixedVariableComparer

### Inherited Members

BaseVariableComparer.GetResult()

**Namespace: com.absence.variablebanks**

**Assembly: Assembly-CSharp-firstpass.dll**

### Syntax

```
[Serializable]
public sealed class FixedVariableComparer : BaseVariableComparer
```

## Properties

## HasFixedBank

Will the bank selector be hidden in the editor?

### Declaration

```
public override bool HasFixedBank { get; }
```

### Property Value

| TYPE |
|---|
| bool |

### Overrides

BaseVariableComparer.HasFixedBank

## Methods

### Clone()

Use to clone this comparer.

#### Declaration

```
public FixedVariableComparer Clone()
```

#### Returns

| TYPE | DESCRIPTION |
|---|---|
| FixedVariableComparer | The clone. |

### Clone(string)

Use to clone this comparer.

#### Declaration

```
public FixedVariableComparer Clone(string overrideBankGuid)
```

#### Parameters

| TYPE | NAME | DESCRIPTION |
|---|---|---|
| string | overrideBankGuid | Guid for a new bank. |

#### Returns

| TYPE | DESCRIPTION |
|---|---|
| FixedVariableComparer | The clone. |

### SetFixedBank(string)

Use to set the fixed bank of this fixed comparer.

### Declaration

```
public void SetFixedBank(string fixedBankGuid)
```

### Parameters

| TYPE | NAME | DESCRIPTION |
|---|---|---|
| string | `fixedBankGuid` | Guid for the fixed bank. |

Edit this page

# Class FixedVariableSetter

Setter with a fixed bank.

### Inheritance

↳ object
  ↳ BaseVariableSetter
    ↳ FixedVariableSetter

### Inherited Members

BaseVariableSetter.Perform()

**Namespace: com.absence.variablebanks**

**Assembly: Assembly-CSharp-firstpass.dll**

### Syntax

```
[Serializable]
public sealed class FixedVariableSetter : BaseVariableSetter
```

## Properties

### HasFixedBank

Will the bank selector be hidden in the editor?

#### Declaration

```
public override bool HasFixedBank { get; }
```

#### Property Value

| TYPE |
|---|
| bool |

#### Overrides

BaseVariableSetter.HasFixedBank

## Methods

### Clone()

Use to clone this setter.

#### Declaration

```
public FixedVariableSetter Clone()
```

#### Returns

| TYPE | DESCRIPTION |
|---|---|
| FixedVariableSetter | The clone. |

### Clone(string)

Use to clone this setter.

#### Declaration

```
public FixedVariableSetter Clone(string overrideBankGuid)
```

#### Parameters

| TYPE | NAME | DESCRIPTION |
|---|---|---|
| string | overrideBankGuid | Guid for a new bank. |

#### Returns

| TYPE | DESCRIPTION |
|---|---|
| FixedVariableSetter | The clone. |

### SetFixedBank(string)

Use to set the fixed bank of this fixed setter.

## Declaration

```
public void SetFixedBank(string fixedBankGuid)
```

## Parameters

| TYPE | NAME | DESCRIPTION |
|---|---|---|
| string | `fixedBankGuid` | Guid for the fixed bank. |

Edit this page

# Class VariableBank

The scriptable object represents a bank of variables.

### Inheritance

↳ object
  ↳ Object
    ↳ ScriptableObject
      ↳ VariableBank

### Inherited Members

ScriptableObject.SetDirty()
ScriptableObject.CreateInstance(string)
ScriptableObject.CreateInstance(Type)
ScriptableObject.CreateInstance<T>()

**Namespace: com.absence.variablebanks**

**Assembly: Assembly-CSharp-firstpass.dll**

### Syntax

```
public class VariableBank : ScriptableObject
```

## Fields

### Null

A constant string that represents a null variable name (with the prefix).

#### Declaration

```
public const string Null = "null: null"
```

#### Field Value

| TYPE |
|---|
| string |

## m_booleans

### Declaration

```
[SerializeField]
protected List<Variable_Boolean> m_booleans
```

### Field Value

| TYPE |
|---|
| List<Variable_Boolean> |

## m_floats

### Declaration

```
[SerializeField]
protected List<Variable_Float> m_floats
```

### Field Value

| TYPE |
|---|
| List<Variable_Float> |

## m_ints

### Declaration

```
[SerializeField]
protected List<Variable_Integer> m_ints
```

### Field Value

| TYPE |
|---|
| List<Variable_Integer> |

## m_strings

### Declaration

```
[SerializeField]
protected List<Variable_String> m_strings
```

### Field Value

| TYPE |
|---|
| List<Variable_String> |

# Properties

## Booleans

All of the boolean variables within this bank.

### Declaration

```
public List<Variable_Boolean> Booleans { get; }
```

### Property Value

| TYPE |
|---|
| List<Variable_Boolean> |

## ClonedFrom

Returns null if this is not a clone. Returns the original bank if this is a clone.

### Declaration

```
public VariableBank ClonedFrom { get; }
```

### Property Value

| TYPE |
|---|
| VariableBank |

## Floats

All of the floating point variables within this bank.

### Declaration

```
public List<Variable_Float> Floats { get; }
```

### Property Value

| TYPE |
|---|
| List<Variable_Float> |

## ForExternalUse

If true, this bank won't get cloned in the startup and also will not get shown on the variable bank name lists. Set to true if you'll use direct references of such. For more information, read the docs.

### Declaration

```
public bool ForExternalUse { get; set; }
```

### Property Value

| TYPE |
|---|
| bool |

## Guid

Guid of this bank.

### Declaration

```
public string Guid { get; }
```

### Property Value

| TYPE |
|---|
| string |

## Ints

All of the integer variables within this bank.

### Declaration

```
public List<Variable_Integer> Ints { get; }
```

### Property Value

| TYPE |
|---|
| List<Variable_Integer> |

## IsClone

Use to check if this bank is a clone.

### Declaration

```
public bool IsClone { get; }
```

### Property Value

| TYPE |
|---|
| bool |

## Strings

All of the string variables within this bank.

### Declaration

```
public List<Variable_String> Strings { get; }
```

### Property Value

| TYPE |
|---|
| List<Variable_String> |

## Methods

### AddValueChangeListenerToBoolean(string, Action<VariableValueChangedCallbackContext<bool>>)

Use to add a value change callback to a boolean variable with a specific name.

#### Declaration

```
public void AddValueChangeListenerToBoolean(string variableName, Action<VariableValueChangedCallbackCor
```

#### Parameters

| TYPE | NAME | DESCRIPTION |
|---|---|---|
| string | `variableName` | Target name. |
| Action<VariableValueChangedCallbackContext<bool>> | `callbackAction` | What to do when value of the variable changes. |

### AddValueChangeListenerToFloat(string, Action<VariableValueChangedCallbackContext<float>>)

Use to add a value change callback to a floating point variable with a specific name.

#### Declaration

```
public void AddValueChangeListenerToFloat(string variableName, Action<VariableValueChangedCallbackConte
```

#### Parameters

| TYPE | NAME | DESCRIPTION |
|---|---|---|
| string | `variableName` | Target name. |
| Action<VariableValueChangedCallbackContext<float>> | `callbackAction` | What to do when value of the variable changes. |

## AddValueChangeListenerToInt(string, Action<VariableValueChangedCallbackContext<int>>)

Use to add a value change callback to an integer variable with a specific name.

### Declaration

```
public void AddValueChangeListenerToInt(string variableName, Action<VariableValueChangedCallbackContext
```

### Parameters

| TYPE | NAME | DESCRIPTION |
|---|---|---|
| string | `variableName` | Target name. |
| Action<VariableValueChangedCallbackContext<int>> | `callbackAction` | What to do when value of the variable changes. |

## AddValueChangeListenerToString(string, Action<VariableValueChangedCallbackContext<string>>)

Use to add a value change callback to a string variable with a specific name.

### Declaration

```
public void AddValueChangeListenerToString(string variableName, Action<VariableValueChangedCallbackCont
```

### Parameters

| TYPE | NAME | DESCRIPTION |
|---|---|---|
| string | `variableName` | Target name. |
| Action<VariableValueChangedCallbackContext<string>> | `callbackAction` | What to do when value of the variable changes. |

## Clone()

Use to clone this bank.

### Declaration

```
public VariableBank Clone()
```

### Returns

| TYPE | DESCRIPTION |
|---|---|
| VariableBank | Returns the clone created. |

## GetAllVariableNames()

Use to get a list of all variables' names of this bank.

### Declaration

```
public List<string> GetAllVariableNames()
```

### Returns

| TYPE | DESCRIPTION |
|---|---|
| List<string> | A list of variable names. Example: "example_int" |

## GetAllVariableNamesWithTypes()

Use to get a list of all variables' names of this bank, each one of the names will contain a type prefix. **Those prefixes get trimmed when you pass them to any function of a variable bank.**

### Declaration

```
public List<string> GetAllVariableNamesWithTypes()
```

### Returns

| TYPE | DESCRIPTION |
|---|---|
| List<string> | A list of all variable names with the prefixes. Example: "int: example_int" |

## GetInstance(string)

Use to get a cloned bank with a specific Guid. **Runtime Only.**

### Declaration

```
public static VariableBank GetInstance(string targetGuid)
```

### Parameters

| TYPE | NAME | DESCRIPTION |
|---|---|---|
| string | `targetGuid` | Target Guid. |

### Returns

| TYPE | DESCRIPTION |
|---|---|
| VariableBank | Throws an error if a clone with the target Guid does not exist. Returns the bank otherwise. |

## HasAny(string)

Use to check if a variable with the target name exists within this bank.

### Declaration

```
public bool HasAny(string variableName)
```

### Parameters

| TYPE | NAME | DESCRIPTION |
|---|---|---|
| string | `variableName` | Target name. |

### Returns

| TYPE | DESCRIPTION |
|---|---|
| bool | True if exists, false otherwise. |

## HasBoolean(string)

Use to check if a boolean variable with the target name exists within this bank.

### Declaration

```
public bool HasBoolean(string variableName)
```

### Parameters

| TYPE | NAME | DESCRIPTION |
|---|---|---|
| string | `variableName` | Target name. |

### Returns

| TYPE | DESCRIPTION |
|---|---|
| bool | True if exists, false otherwise. |

## HasFloat(string)

Use to check if a floating point variable with the target name exists within this bank.

### Declaration

```
public bool HasFloat(string variableName)
```

### Parameters

| TYPE | NAME | DESCRIPTION |
|---|---|---|
| string | `variableName` | Target name. |

### Returns

| TYPE | DESCRIPTION |
|---|---|
| bool | True if exists, false otherwise. |

## HasInt(string)

Use to check if an integer variable with the target name exists within this bank.

### Declaration

```
public bool HasInt(string variableName)
```

### Parameters

| TYPE | NAME | DESCRIPTION |
|---|---|---|
| string | `variableName` | Target name. |

### Returns

| TYPE | DESCRIPTION |
|---|---|
| bool | True if exists, false otherwise. |

## HasString(string)

Use to check if a string variable with the target name exists within this bank.

### Declaration

```
public bool HasString(string variableName)
```

### Parameters

| TYPE | NAME | DESCRIPTION |
|---|---|---|
| string | `variableName` | Target name. |

### Returns

| TYPE | DESCRIPTION |
|---|---|
| bool | True if exists, false otherwise. |

## SetBoolean(string, bool)

Use to change a boolean variable's value.

### Declaration

```
public bool SetBoolean(string variableName, bool newValue)
```

### Parameters

| TYPE | NAME | DESCRIPTION |
|---|---|---|
| string | `variableName` | Target name. |
| bool | `newValue` | New value for the variable. |

### Returns

| TYPE | DESCRIPTION |
|---|---|
| bool | True if value changing process ended successfully. False otherwise. |

## SetFloat(string, float)

Use to change a floating point variable's value.

### Declaration

```
public bool SetFloat(string variableName, float newValue)
```

### Parameters

| TYPE | NAME | DESCRIPTION |
|---|---|---|
| string | `variableName` | Target name. |
| float | `newValue` | New value for the variable. |

### Returns

| TYPE | DESCRIPTION |
|---|---|
| bool | True if value changing process ended successfully. False otherwise. |

## SetInt(string, int)

Use to change an integer variable's value.

### Declaration

```
public bool SetInt(string variableName, int newValue)
```

### Parameters

| TYPE | NAME | DESCRIPTION |
|---|---|---|
| string | `variableName` | Target name. |
| int | `newValue` | New value for the variable. |

### Returns

| TYPE | DESCRIPTION |
|---|---|
| bool | True if value changing process ended successfully. False otherwise. |

## SetString(string, string)

Use to change a string variable's value.

### Declaration

```
public bool SetString(string variableName, string newValue)
```

### Parameters

| TYPE | NAME | DESCRIPTION |
|---|---|---|
| string | `variableName` | Target name. |
| string | `newValue` | New value for the variable. |

### Returns

| TYPE | DESCRIPTION |
|---|---|
| bool | True if value changing process ended successfully. False otherwise. |

## TryGetBoolean(string, out bool)

Use to get value of a boolean variable within this bank.

### Declaration

```
public bool TryGetBoolean(string variableName, out bool value)
```

### Parameters

| TYPE | NAME | DESCRIPTION |
|---|---|---|
| string | `variableName` | Target name. |
| bool | `value` | Value of the variable. |

### Returns

| TYPE | DESCRIPTION |
|---|---|
| bool | True if a variable with the target name exists within the bank. |

## TryGetFloat(string, out float)

Use to get value of a floating point variable within this bank.

### Declaration

```
public bool TryGetFloat(string variableName, out float value)
```

### Parameters

| TYPE | NAME | DESCRIPTION |
|---|---|---|
| string | `variableName` | Target name. |
| float | `value` | Value of the variable. |

### Returns

| TYPE | DESCRIPTION |
|---|---|
| bool | True if a variable with the target name exists within the bank. |

## TryGetInt(string, out int)

Use to get value of an integer variable within this bank.

### Declaration

```
public bool TryGetInt(string variableName, out int value)
```

### Parameters

| TYPE | NAME | DESCRIPTION |
|---|---|---|
| string | `variableName` | Target name. |
| int | `value` | Value of the variable. |

### Returns

| TYPE | DESCRIPTION |
|---|---|
| bool | True if a variable with the target name exists within the bank. |

## TryGetString(string, out string)

Use to get value of a string variable within this bank.

### Declaration

```
public bool TryGetString(string variableName, out string value)
```

### Parameters

| TYPE | NAME | DESCRIPTION |
|---|---|---|
| string | `variableName` | Target name. |
| string | `value` | Value of the variable. |

### Returns

| TYPE | DESCRIPTION |
|---|---|
| bool | True if a variable with the target name exists within the bank. |

# Events

## OnDestroyAction

The action gets invoked when this bank gets destroyed.

### Declaration

```
public event Action OnDestroyAction
```

### Event Type

| TYPE |
|---|
| Action |

Edit this page

# Class VariableBankAcquirer

A component to reference banks both in editor and runtime.

### Inheritance

↳ object
  ↳ Object
    ↳ Component
      ↳ Behaviour
        ↳ MonoBehaviour
          ↳ VariableBankAcquirer

**Namespace: com.absence.variablebanks**

**Assembly: Assembly-CSharp-firstpass.dll**

### Syntax

```
public class VariableBankAcquirer : MonoBehaviour
```

## Properties

## Bank

Use to get clone of the referenced bank. **Runtime only.**

### Declaration

```
public VariableBank Bank { get; }
```

### Property Value

| TYPE |
|---|
| VariableBank |

## TargetGuid

Use to get the Guid of the referenced bank.

### Declaration

```
public string TargetGuid { get; }
```

### Property Value

| TYPE |
|---|
| string |

# Class VariableBankReference

The class responsible for letting you reference a `VariableBank` both in editor and in runtime. You can use the `Variable Bank` class directly if the bank you are referencing is marked as `ForExternalUse`. For more information, read the docs.

### Inheritance

↳ object
  ↳ VariableBankReference

**Namespace: com.absence.variablebanks**

**Assembly: Assembly-CSharp-firstpass.dll**

### Syntax

```
[Serializable]
public class VariableBankReference
```

## Properties

## Bank

Use to get the bank referenced. **Runtime only.**

### Declaration

```
public VariableBank Bank { get; }
```

### Property Value

| TYPE |
|---|
| VariableBank |

## TargetGuid

Use to get the referenced bank's Guid. Returns an empty string if no banks referenced.

### Declaration

```
public string TargetGuid { get; }
```

### Property Value

| TYPE |
|---|
| string |

Edit this page

# Class VariableComparer

Comparer with a dynamic bank you select in editor.

### Inheritance

↳ object
  ↳ BaseVariableComparer
    ↳ VariableComparer

### Inherited Members

BaseVariableComparer.GetResult()

**Namespace: com.absence.variablebanks**

**Assembly: Assembly-CSharp-firstpass.dll**

### Syntax

```
[Serializable]
public sealed class VariableComparer : BaseVariableComparer
```

## Properties

## HasFixedBank

Will the bank selector be hidden in the editor?

### Declaration

```
public override bool HasFixedBank { get; }
```

### Property Value

| TYPE |
|---|
| bool |

### Overrides

BaseVariableComparer.HasFixedBank

## Methods

### Clone()

Use to clone this comparer.

#### Declaration

```
public VariableComparer Clone()
```

#### Returns

| TYPE | DESCRIPTION |
|---|---|
| VariableComparer | The clone. |

### Clone(string)

Use to clone this comparer.

#### Declaration

```
public VariableComparer Clone(string overrideBankGuid)
```

#### Parameters

| TYPE | NAME | DESCRIPTION |
|---|---|---|
| string | overrideBankGuid | Guid for a new bank. |

#### Returns

| TYPE | DESCRIPTION |
|---|---|
| VariableComparer | The clone. |

### SetBankGuid(string)

Set this comparer's target bank Guid.

### Declaration

```
public void SetBankGuid(string newBankGuid)
```

### Parameters

| TYPE | NAME | DESCRIPTION |
|---|---|---|
| string | `newBankGuid` | New Guid. |

Edit this page

# Class VariableSetter

Setter with a dynamic bank you select in the editor.

### Inheritance

↳ object
  ↳ BaseVariableSetter
    ↳ VariableSetter

### Inherited Members

BaseVariableSetter.Perform()

**Namespace: com.absence.variablebanks**

**Assembly: Assembly-CSharp-firstpass.dll**

### Syntax

```
[Serializable]
public sealed class VariableSetter : BaseVariableSetter
```

## Properties

### HasFixedBank

Will the bank selector be hidden in the editor?

#### Declaration

```
public override bool HasFixedBank { get; }
```

#### Property Value

| TYPE |
|---|
| bool |

#### Overrides

BaseVariableSetter.HasFixedBank

## Methods

### Clone()

Use to clone this setter.

#### Declaration

```
public VariableSetter Clone()
```

#### Returns

| TYPE | DESCRIPTION |
|---|---|
| VariableSetter | The clone. |

### Clone(string)

Use to clone this setter.

#### Declaration

```
public VariableSetter Clone(string overrideBankGuid)
```

#### Parameters

| TYPE | NAME | DESCRIPTION |
|---|---|---|
| string | overrideBankGuid | Guid for a new bank. |

#### Returns

| TYPE | DESCRIPTION |
|---|---|
| VariableSetter | The clone. |

### SetBankGuid(string)

Set this setter's target bank Guid.

### Declaration

```
public void SetBankGuid(string newBankGuid)
```

### Parameters

| TYPE | NAME | DESCRIPTION |
|---|---|---|
| string | `newBankGuid` | New Guid. |

# Namespace com.absence.variablebanks.editor

## Classes

### VariableBankAcquirerCustomEditor

A custom editor script for `VariableBankAcquirer` .

### VariableBankCreationHandler

The static class responsible for handling variable bank creation via editor menu.

### VariableBankDatabase

The static class responsible for holding a list of all `VariableBank` s in the project. **Editor only! For runtime, use `GetInstance(string)` instead.**

### VariableBankReferencePropertyDrawer

A custom property drawer script for `VariableBankReferencePropertyDrawer` .

### VariableComparerDrawer

A custom property drawer for `BaseVariableComparer` .

### VariableSetterDrawer

A custom property drawer script for `BaseVariableSetter` .

Edit this page

# Class VariableBankAcquirerCustomEditor

A custom editor script for `VariableBankAcquirer` .

### Inheritance

↳ object
  ↳ Object
    ↳ ScriptableObject
      ↳ Editor
        ↳ VariableBankAcquirerCustomEditor

### Inherited Members

Editor.SaveChanges()
Editor.DiscardChanges()
Editor.CreateEditorWithContext(Object[], Object, Type)
Editor.CreateEditorWithContext(Object[], Object)
Editor.CreateCachedEditorWithContext(Object, Object, Type, ref Editor)
Editor.CreateCachedEditorWithContext(Object[], Object, Type, ref Editor)
Editor.CreateCachedEditor(Object, Type, ref Editor)
Editor.CreateCachedEditor(Object[], Type, ref Editor)
Editor.CreateEditor(Object)
Editor.CreateEditor(Object, Type)
Editor.CreateEditor(Object[])
Editor.CreateEditor(Object[], Type)
Editor.DrawPropertiesExcluding(SerializedObject, params string[])
Editor.DrawDefaultInspector()
Editor.Repaint()
Editor.CreateInspectorGUI()
Editor.RequiresConstantRepaint()
Editor.DrawHeader()
Editor.OnHeaderGUI()
Editor.ShouldHideOpenButton()
Editor.DrawFoldoutInspector(Object, ref Editor)
Editor.HasPreviewGUI()
Editor.GetPreviewTitle()
Editor.RenderStaticPreview(string, Object[], int, int)
Editor.OnPreviewGUI(Rect, GUIStyle)
Editor.OnInteractivePreviewGUI(Rect, GUIStyle)
Editor.OnPreviewSettings()
Editor.GetInfoString()
Editor.DrawPreview(Rect)
Editor.ReloadPreviewInstances()

Editor.UseDefaultMargins()
Editor.MoveNextTarget()
Editor.ResetTarget()
Editor.hasUnsavedChanges
Editor.saveChangesMessage
Editor.target
Editor.targets
Editor.serializedObject
Editor.finishedDefaultHeaderGUI
ScriptableObject.SetDirty()
ScriptableObject.CreateInstance(string)
ScriptableObject.CreateInstance(Type)
ScriptableObject.CreateInstance<T>()

**Namespace: com.absence.variablebanks.editor**

**Assembly: Assembly-CSharp-Editor-firstpass.dll**

### Syntax

```
[CustomEditor(typeof(VariableBankAcquirer), true)]
public class VariableBankAcquirerCustomEditor : Editor
```

# Methods

## OnInspectorGUI()

Implement this function to make a custom inspector.

### Declaration

```
public override void OnInspectorGUI()
```

### Overrides

UnityEditor.Editor.OnInspectorGUI()

Edit this page

# Class VariableBankCreationHandler

The static class responsible for handling variable bank creation via editor menu.

## Inheritance

↳ object
  ↳ VariableBankCreationHandler

**Namespace: com.absence.variablebanks.editor**

**Assembly: Assembly-CSharp-Editor-firstpass.dll**

## Syntax

```
public static class VariableBankCreationHandler
```

# Class VariableBankDatabase

The static class responsible for holding a list of all `VariableBank` s in the project. **Editor only! For runtime, use `Get Instance(string)` instead.**

### Inheritance

↳ object
  ↳ VariableBankDatabase

**Namespace: com.absence.variablebanks.editor**

**Assembly: Assembly-CSharp-Editor-firstpass.dll**

### Syntax

```
[InitializeOnLoad]
public static class VariableBankDatabase
```

## Properties

## BanksInAssets

All of the banks in the project.

### Declaration

```
public static List<VariableBank> BanksInAssets { get; }
```

### Property Value

| TYPE |
|---|
| List<VariableBank> |

## NoBanks

Returns true when there are no variable banks in the project's assets.

### Declaration

```
public static bool NoBanks { get; }
```

### Property Value

| TYPE |
|---|
| bool |

## Methods

### Exists(string)

Use to check if a bank with the target Guid exists.

#### Declaration

```
public static bool Exists(string bankGuid)
```

#### Parameters

| TYPE | NAME | DESCRIPTION |
|---|---|---|
| string | `bankGuid` | Target Guid. |

#### Returns

| TYPE | DESCRIPTION |
|---|---|
| bool | True if exists, false otherwise. |

### GetBankIfExists(string)

#### Declaration

```
public static VariableBank GetBankIfExists(string bankGuid)
```

#### Parameters

| TYPE | NAME |
|---|---|
| string | `bankGuid` |

**Returns**

| TYPE |
|---|
| VariableBank |

## GetBankNameList()

Use to get a list of all variable banks' names.

**Declaration**

```
public static List<string> GetBankNameList()
```

**Returns**

| TYPE | DESCRIPTION |
|---|---|
| List<string> | Returns a list of all variable banks' (except of the ones marked as `ForExternalUse` ) names. |

## GetIndexOf(string)

Get the index of the bank with the target Guid.

**Declaration**

```
public static int GetIndexOf(string bankGuid)
```

**Parameters**

| TYPE | NAME | DESCRIPTION |
|---|---|---|
| string | `bankGuid` | Target Guid. |

**Returns**

| TYPE | DESCRIPTION |
|---|---|
| int | Returns **-1** if the bank with the target Guid does not exits. Returns the index otherwise. |

## NameToGuid(string)

Use to get Guid of a bank with a specific name.

### Declaration

```
public static string NameToGuid(string bankName)
```

### Parameters

| TYPE | NAME | DESCRIPTION |
|---|---|---|
| string | `bankName` | Target name. |

### Returns

| TYPE | DESCRIPTION |
|---|---|
| string | Returns null if a bank with the target name does not exist. Returns the Guid otherwise. |

## Refresh()

Use to refresh the variable bank database.

### Declaration

```
public static void Refresh()
```

Edit this page

# Class VariableBankReferencePropertyDrawer

A custom property drawer script for `VariableBankReferencePropertyDrawer` .

### Inheritance

↳ object
  ↳ GUIDrawer
    ↳ PropertyDrawer
      ↳ VariableBankReferencePropertyDrawer

### Inherited Members

PropertyDrawer.CreatePropertyGUI(SerializedProperty)
PropertyDrawer.CanCacheInspectorGUI(SerializedProperty)
PropertyDrawer.attribute
PropertyDrawer.fieldInfo
PropertyDrawer.preferredLabel

**Namespace: com.absence.variablebanks.editor**

**Assembly: Assembly-CSharp-Editor-firstpass.dll**

### Syntax

```
[CustomPropertyDrawer(typeof(VariableBankReference), true)]
public class VariableBankReferencePropertyDrawer : PropertyDrawer
```

## Methods

### GetPropertyHeight(SerializedProperty, GUIContent)

Override this method to specify how tall the GUI for this field is in pixels.

#### Declaration

```
public override float GetPropertyHeight(SerializedProperty property, GUIContent label)
```

#### Parameters

| TYPE | NAME | DESCRIPTION |
|---|---|---|
| SerializedProperty | `property` | The SerializedProperty to make the custom GUI for. |
| GUIContent | `label` | The label of this property. |

### Returns

| TYPE | DESCRIPTION |
|---|---|
| float | The height in pixels. |

### Overrides

UnityEditor.PropertyDrawer.GetPropertyHeight(UnityEditor.SerializedProperty, UnityEngine.GUIContent)

## OnGUI(Rect, SerializedProperty, GUIContent)

Override this method to make your own IMGUI based GUI for the property.

### Declaration

```
public override void OnGUI(Rect position, SerializedProperty property, GUIContent label)
```

### Parameters

| TYPE | NAME | DESCRIPTION |
|---|---|---|
| Rect | `position` | Rectangle on the screen to use for the property GUI. |
| SerializedProperty | `property` | The SerializedProperty to make the custom GUI for. |
| GUIContent | `label` | The label of this property. |

### Overrides

UnityEditor.PropertyDrawer.OnGUI(UnityEngine.Rect, UnityEditor.SerializedProperty, UnityEngine.GUIContent)

# Class VariableComparerDrawer

A custom property drawer for `BaseVariableComparer`.

### Inheritance

↳ object
  ↳ GUIDrawer
    ↳ PropertyDrawer
      ↳ VariableComparerDrawer

### Inherited Members

PropertyDrawer.CanCacheInspectorGUI(SerializedProperty)
PropertyDrawer.attribute
PropertyDrawer.fieldInfo
PropertyDrawer.preferredLabel

**Namespace: com.absence.variablebanks.editor**

**Assembly: Assembly-CSharp-Editor-firstpass.dll**

### Syntax

```
[CustomPropertyDrawer(typeof(BaseVariableComparer), true)]
public class VariableComparerDrawer : PropertyDrawer
```

## Fields

## StyleSheetPath

Path of the uss file.

### Declaration

```
protected static readonly string StyleSheetPath
```

### Field Value

| TYPE |
|---|
| string |

## Methods

### CreatePropertyGUI(SerializedProperty)

Override this method to make your own UI Toolkit based GUI for the property.

#### Declaration

```
public override VisualElement CreatePropertyGUI(SerializedProperty property)
```

#### Parameters

| TYPE | NAME | DESCRIPTION |
|---|---|---|
| SerializedProperty | `property` | The SerializedProperty to make the custom GUI for. |

#### Returns

| TYPE | DESCRIPTION |
|---|---|
| VisualElement | The element containing the custom GUI. |

#### Overrides

UnityEditor.PropertyDrawer.CreatePropertyGUI(UnityEditor.SerializedProperty)

### DrawGUI(VisualElement, SerializedProperty)

#### Declaration

```
protected virtual VisualElement DrawGUI(VisualElement container, SerializedProperty property)
```

#### Parameters

| TYPE | NAME |
|---|---|
| VisualElement | `container` |
| SerializedProperty | `property` |

#### Returns

| TYPE |
|---|
| VisualElement |

## GetPropertyHeight(SerializedProperty, GUIContent)

Override this method to specify how tall the GUI for this field is in pixels.

#### Declaration

```
public override float GetPropertyHeight(SerializedProperty property, GUIContent label)
```

#### Parameters

| TYPE | NAME | DESCRIPTION |
|---|---|---|
| SerializedProperty | `property` | The SerializedProperty to make the custom GUI for. |
| GUIContent | `label` | The label of this property. |

#### Returns

| TYPE | DESCRIPTION |
|---|---|
| float | The height in pixels. |

#### Overrides

UnityEditor.PropertyDrawer.GetPropertyHeight(UnityEditor.SerializedProperty, UnityEngine.GUIContent)

## OnGUI(Rect, SerializedProperty, GUIContent)

Override this method to make your own IMGUI based GUI for the property.

#### Declaration

```
public override void OnGUI(Rect position, SerializedProperty property, GUIContent label)
```

#### Parameters

| TYPE | NAME | DESCRIPTION |
|---|---|---|
| Rect | `position` | Rectangle on the screen to use for the property GUI. |
| SerializedProperty | `property` | The SerializedProperty to make the custom GUI for. |

| TYPE | NAME | DESCRIPTION |
|---|---|---|
| GUIContent | `label` | The label of this property. |

### Overrides

UnityEditor.PropertyDrawer.OnGUI(UnityEngine.Rect, UnityEditor.SerializedProperty, UnityEngine.GUIContent)

# Class VariableSetterDrawer

A custom property drawer script for `BaseVariableSetter` .

### Inheritance

↳ object
  ↳ GUIDrawer
    ↳ PropertyDrawer
      ↳ VariableSetterDrawer

### Inherited Members

PropertyDrawer.CanCacheInspectorGUI(SerializedProperty)
PropertyDrawer.attribute
PropertyDrawer.fieldInfo
PropertyDrawer.preferredLabel

**Namespace: com.absence.variablebanks.editor**

**Assembly: Assembly-CSharp-Editor-firstpass.dll**

### Syntax

```
[CustomPropertyDrawer(typeof(BaseVariableSetter), true)]
public class VariableSetterDrawer : PropertyDrawer
```

## Fields

## StyleSheetPath

Path of the uss file.

### Declaration

```
protected static readonly string StyleSheetPath
```

### Field Value

| TYPE |
|---|
| string |

## Methods

### CreatePropertyGUI(SerializedProperty)

Override this method to make your own UI Toolkit based GUI for the property.

#### Declaration

```
public override VisualElement CreatePropertyGUI(SerializedProperty property)
```

#### Parameters

| TYPE | NAME | DESCRIPTION |
|---|---|---|
| SerializedProperty | `property` | The SerializedProperty to make the custom GUI for. |

#### Returns

| TYPE | DESCRIPTION |
|---|---|
| VisualElement | The element containing the custom GUI. |

#### Overrides

UnityEditor.PropertyDrawer.CreatePropertyGUI(UnityEditor.SerializedProperty)

### GetPropertyHeight(SerializedProperty, GUIContent)

Override this method to specify how tall the GUI for this field is in pixels.

#### Declaration

```
public override float GetPropertyHeight(SerializedProperty property, GUIContent label)
```

#### Parameters

| TYPE | NAME | DESCRIPTION |
|---|---|---|
| SerializedProperty | `property` | The SerializedProperty to make the custom GUI for. |

| TYPE | NAME | DESCRIPTION |
|---|---|---|
| GUIContent | `label` | The label of this property. |

### Returns

| TYPE | DESCRIPTION |
|---|---|
| float | The height in pixels. |

### Overrides

UnityEditor.PropertyDrawer.GetPropertyHeight(UnityEditor.SerializedProperty, UnityEngine.GUIContent)

## OnGUI(Rect, SerializedProperty, GUIContent)

Override this method to make your own IMGUI based GUI for the property.

### Declaration

```
public override void OnGUI(Rect position, SerializedProperty property, GUIContent label)
```

### Parameters

| TYPE | NAME | DESCRIPTION |
|---|---|---|
| Rect | `position` | Rectangle on the screen to use for the property GUI. |
| SerializedProperty | `property` | The SerializedProperty to make the custom GUI for. |
| GUIContent | `label` | The label of this property. |

### Overrides

UnityEditor.PropertyDrawer.OnGUI(UnityEngine.Rect, UnityEditor.SerializedProperty, UnityEngine.GUIContent)

API Documentation / com.absence.variablebanks.internals

# Namespace com.absence.variablebanks.internals

## Classes

### BaseVariableComparer

The base class for comparers.

### BaseVariableSetter

The base class for setters.

### Constants

The static class responsible for holding the constants variables of the package.

### VariableBanksCloningHandler

The static class responsible for cloning the banks at startup.

## Enums

### BaseVariableComparer.ComparisonType

An enum for deciding how the comparison will get performed.

### BaseVariableSetter.SetType

An enum for deciding which way the setting will work.

Edit this page

# Class BaseVariableComparer

The base class for comparers.

### Inheritance

↳ object
  ↳ BaseVariableComparer
    ↳ FixedVariableComparer
    ↳ VariableComparer

**Namespace: com.absence.variablebanks.internals**

**Assembly: Assembly-CSharp-firstpass.dll**

### Syntax

```
[Serializable]
public abstract class BaseVariableComparer
```

## Fields

### m_boolValue

#### Declaration

```
[SerializeField]
protected bool m_boolValue
```

#### Field Value

| TYPE |
|---|
| bool |

### m_comparisonType

#### Declaration

```
[SerializeField]
protected BaseVariableComparer.ComparisonType m_comparisonType
```

#### Field Value

| TYPE |
|---|
| BaseVariableComparer.ComparisonType |

## m_floatValue

#### Declaration

```
[SerializeField]
protected float m_floatValue
```

#### Field Value

| TYPE |
|---|
| float |

## m_intValue

#### Declaration

```
[SerializeField]
protected int m_intValue
```

#### Field Value

| TYPE |
|---|
| int |

## m_stringValue

#### Declaration

```
[SerializeField]
protected string m_stringValue
```

### Field Value

| TYPE |
|---|
| string |

## m_targetBankGuid

### Declaration

```
[SerializeField]
protected string m_targetBankGuid
```

### Field Value

| TYPE |
|---|
| string |

## m_targetVariableName

### Declaration

```
[SerializeField]
protected string m_targetVariableName
```

### Field Value

| TYPE |
|---|
| string |

## Properties

## HasFixedBank

Will the bank selector be hidden in the editor?

### Declaration

```
public abstract bool HasFixedBank { get; }
```

### Property Value

| TYPE |
|---|
| bool |

## Methods

### GetResult()

Use to get the result of the comparer. **Runtime only.**

#### Declaration

```
public virtual bool GetResult()
```

#### Returns

| TYPE | DESCRIPTION |
|---|---|
| bool | Result of the comparer. Returns true directly if anything goes wrong. |

### GetRuntimeBank()

Override to define how this comparer will find it's runtime bank.

#### Declaration

```
protected virtual VariableBank GetRuntimeBank()
```

#### Returns

| TYPE | DESCRIPTION |
|---|---|
| VariableBank | The runtime bank or null |

# Enum BaseVariableComparer.ComparisonType

An enum for deciding how the comparison will get performed.

**Namespace: com.absence.variablebanks.internals**

**Assembly: Assembly-CSharp-firstpass.dll**

### Syntax

```
public enum BaseVariableComparer.ComparisonType
```

## Fields

| NAME |
|---|
| EqualsTo |
| GreaterOrEqual |
| GreaterThan |
| LessOrEqual |
| LessThan |
| NotEquals |

Edit this page

# Class BaseVariableSetter

The base class for setters.

## Inheritance

↳ object
  ↳ BaseVariableSetter
    ↳ FixedVariableSetter
    ↳ VariableSetter

**Namespace: com.absence.variablebanks.internals**

**Assembly: Assembly-CSharp-firstpass.dll**

## Syntax

```
[Serializable]
public abstract class BaseVariableSetter
```

## Fields

### m_boolValue

#### Declaration

```
[SerializeField]
protected bool m_boolValue
```

#### Field Value

| TYPE |
|---|
| bool |

### m_floatValue

### Declaration

```
[SerializeField]
protected float m_floatValue
```

### Field Value

| TYPE |
|---|
| float |

## m_intValue

### Declaration

```
[SerializeField]
protected int m_intValue
```

### Field Value

| TYPE |
|---|
| int |

## m_setType

### Declaration

```
[SerializeField]
protected BaseVariableSetter.SetType m_setType
```

### Field Value

| TYPE |
|---|
| BaseVariableSetter.SetType |

## m_stringValue

### Declaration

```
[SerializeField]
protected string m_stringValue
```

#### Field Value

| TYPE |
|---|
| string |

### m_targetBankGuid

#### Declaration

```
[SerializeField]
protected string m_targetBankGuid
```

#### Field Value

| TYPE |
|---|
| string |

### m_targetVariableName

#### Declaration

```
[SerializeField]
protected string m_targetVariableName
```

#### Field Value

| TYPE |
|---|
| string |

## Properties

### HasFixedBank

Will the bank selector be hidden in the editor?

### Declaration

```
public abstract bool HasFixedBank { get; }
```

### Property Value

| TYPE |
|---|
| bool |

# Methods

## GetRuntimeBank()

Override to define how this setter will find it's runtime bank.

### Declaration

```
protected virtual VariableBank GetRuntimeBank()
```

### Returns

| TYPE | DESCRIPTION |
|---|---|
| VariableBank | The runtime bank or null |

## Perform()

Sets the target variable in target `VariableBank` to intended value.

### Declaration

```
public virtual void Perform()
```

## Perform_Boolean(VariableBank)

Override to define the logic for booleans.

#### Declaration

```
protected virtual void Perform_Boolean(VariableBank bank)
```

#### Parameters

| TYPE | NAME | DESCRIPTION |
|---|---|---|
| VariableBank | bank | Runtime bank. |

## Perform_Float(VariableBank)

Override to define the logic for floating points.

#### Declaration

```
protected virtual void Perform_Float(VariableBank bank)
```

#### Parameters

| TYPE | NAME | DESCRIPTION |
|---|---|---|
| VariableBank | bank | Runtime bank. |

## Perform_Int(VariableBank)

Override to define the logic for integers.

#### Declaration

```
protected virtual void Perform_Int(VariableBank bank)
```

#### Parameters

| TYPE | NAME | DESCRIPTION |
|---|---|---|
| VariableBank | bank | Runtime bank. |

## Perform_String(VariableBank)

Override to define the logic for strings.

### Declaration

```
protected virtual void Perform_String(VariableBank bank)
```

### Parameters

| TYPE | NAME | DESCRIPTION |
|---|---|---|
| VariableBank | bank | Runtime bank. |

# Enum BaseVariableSetter.SetType

An enum for deciding which way the setting will work.

**Namespace: com.absence.variablebanks.internals**

**Assembly: Assembly-CSharp-firstpass.dll**

### Syntax

```
public enum BaseVariableSetter.SetType
```

## Fields

| NAME |
|---|
| DecrementBy |
| DivideBy |
| IncrementBy |
| MultipltyBy |
| SetTo |

Edit this page

# Class Constants

The static class responsible for holding the constants variables of the package.

### Inheritance

↳ object
  ↳ Constants

**Namespace: com.absence.variablebanks.internals**

**Assembly: Assembly-CSharp-firstpass.dll**

### Syntax

```
public static class Constants
```

## Fields

### K_ADDRESSABLES_TAG

The addressables label of variable banks if you're using **Addressables** as the asset management tool.

#### Declaration

```
public const string K_ADDRESSABLES_TAG = "variable-banks"
```

#### Field Value

| TYPE |
|---|
| string |

### K_RESOURCES_PATH

The resources path of variable banks if you're using **Resources API** as the asset management tool.

#### Declaration

```
public const string K_RESOURCES_PATH = "VariableBanks"
```

#### Field Value

| TYPE |
|---|
| string |

Edit this page

# Class VariableBanksCloningHandler

The static class responsible for cloning the banks at startup.

### Inheritance

↳ object
  ↳ VariableBanksCloningHandler

**Namespace: com.absence.variablebanks.internals**

**Assembly: Assembly-CSharp-firstpass.dll**

### Syntax

```
public static class VariableBanksCloningHandler
```

## Properties

### CloningCompleted

Use to check if the cloning process got completed successfully.

### Declaration

```
public static bool CloningCompleted { get; }
```

### Property Value

| TYPE |
|---|
| bool |

## Methods

## AddCloningCompleteCallbackOrInvoke(Action)

Adds the action passed to `OnCloningCompleted` if the cloning process is not ended yet. If it is ended already, the action passed gets invoked instantly.

**Declaration**

```
public static bool AddCloningCompleteCallbackOrInvoke(Action callbackContext)
```

**Parameters**

| TYPE | NAME |
|---|---|
| Action | `callbackContext` |

**Returns**

| TYPE |
|---|
| bool |

## Events

## OnCloningCompleted

Action which will get invoked when cloning process gets completed successfully. It gets cleared automatically after invoking.

**Declaration**

```
public static event Action OnCloningCompleted
```

**Event Type**

| TYPE |
|---|
| Action |

# Namespace com.absence.variablebanks.testing

## Classes

**Varcaster**

Edit this page

# Class Varcaster

### Inheritance

↳ object
  ↳ Object
    ↳ Component
      ↳ Behaviour
        ↳ MonoBehaviour
          ↳ Varcaster

**Namespace: com.absence.variablebanks.testing**

**Assembly: Assembly-CSharp-firstpass.dll**

### Syntax

```
public class Varcaster : MonoBehaviour
```