

Namespace ASE_Assignment_Demo

Classes

[AppCanvas](#)

Represents the drawing canvas and its associated operations, including shapes, lines, and text. Implements the BOOSE.ICanvas interface.

[AppCommandFactory](#)

Factory class for creating instances of commands. Implements the Singleton Pattern to ensure a single instance throughout the application.

[CustomCircle](#)

Represents a custom circle command that can be executed on a canvas.

[CustomFor](#)

The CustomFor class extends the functionality of the `For` class from the BOOSE library. This class allows for customization of the `Restrictions` method or any other inherited functionality.

[CustomIf](#)

The `CustomIf` class extends the functionality of the `If` class from the BOOSE library. It allows for customization of the `If` logic and provides additional or modified behavior for conditions and their execution.

[CustomInt](#)

The `CustomInt` class inherits from the `Int` class to provide custom behavior for variable handling, restrictions, and execution logic.

[CustomMethod](#)

The `CustomMethod` class extends the `Method` class from the BOOSE library. This class provides customization for method-related functionalities, including reducing restrictions and managing method execution and compilation behavior.

[CustomRectangle](#)

Represents a custom rectangle command that can be executed on a canvas.

[CustomWhile](#)

The `CustomWhile` class inherits from the `While` class to provide customized behavior for the 'While' loop, with the added functionality of reducing restrictions as needed.

[Form1](#)

Main form class for the ASE Assignment Demo application.

Class AppCanvas

Namespace: [ASE Assignment Demo](#)

Assembly: ASE Assignment Demo.dll

Represents the drawing canvas and its associated operations, including shapes, lines, and text. Implements the BOOSE.ICanvas interface.

```
public class AppCanvas : ICanvas
```








Inheritance

[object](#)  ← AppCanvas

Implements

ICanvas

Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#) 

Constructors

AppCanvas()

Initializes a new instance of the [AppCanvas](#) class with default settings.

```
public AppCanvas()
```

Properties

PenColour

Gets or sets the color of the pen.

```
public object PenColour { get; set; }
```

Property Value

[object](#)

Xpos

Gets or sets the current X position of the pen on the canvas.

```
public int Xpos { get; set; }
```

Property Value

[int](#)

Ypos

Gets or sets the current Y position of the pen on the canvas.

```
public int Ypos { get; set; }
```

Property Value

[int](#)

Methods

Circle(int, bool)

Draws a circle at the current position with the specified radius.

```
public void Circle(int radius, bool filled)
```

Parameters

radius [int](#)

The radius of the circle.

filled [bool](#)

Indicates whether the circle should be filled.

Exceptions

CanvasException

Thrown when the radius is negative.

Clear()

Clears the canvas by filling it with a white background.

```
public void Clear()
```

DrawTo(int, int)

Draws a line from the current pen position to the specified coordinates.

```
public void DrawTo(int x, int y)
```

Parameters

x [int](#)

The X-coordinate of the destination point.

y [int](#)

The Y-coordinate of the destination point.

MoveTo(int, int)

Moves the pen to the specified coordinates without drawing.

```
public void MoveTo(int x, int y)
```

Parameters

x [int](#)

The X-coordinate to move to.

y [int](#)

The Y-coordinate to move to.

Rect(int, int, bool)

Draws a rectangle at the current pen position with specified width and height.

```
public void Rect(int width, int height, bool filled)
```

Parameters

width [int](#)

The width of the rectangle.

height [int](#)

The height of the rectangle.

filled [bool](#)

Indicates whether the rectangle should be filled.

Reset()

Resets the canvas by clearing it, resetting the pen position, and restoring default settings.

```
public void Reset()
```

Set(int, int)

Sets the canvas dimensions and clears it.

```
public void Set(int width, int height)
```

Parameters

width [int](#)

The width of the canvas.

height [int](#)

The height of the canvas.

SetColour(int, int, int)

Sets the pen color using RGB values.

```
public void SetColour(int red, int green, int blue)
```

Parameters

red [int](#)

The red component of the color.

green [int](#)

The green component of the color.

blue [int](#)

The blue component of the color.

Tri(int, int)

Draws a triangle at the current pen position with the specified width and height.

```
public void Tri(int width, int height)
```

Parameters

width [int](#)

The base width of the triangle.

height [int](#)

The height of the triangle.

WriteText(string)

Writes text at the current pen position.

```
public void WriteText(string text)
```

Parameters

text [string](#)

The text to write.

getBitmap()

Gets the bitmap representing the current state of the canvas.

```
public object getBitmap()
```

Returns

[object](#)

The bitmap object.

Class AppCommandFactory

Namespace: [ASE Assignment Demo](#)

Assembly: ASE Assignment Demo.dll

Factory class for creating instances of commands. Implements the Singleton Pattern to ensure a single instance throughout the application.

```
public class AppCommandFactory : CommandFactory, ICommandFactory
```








Inheritance

[object](#)  ← [CommandFactory](#) ← [AppCommandFactory](#)

Implements

[ICommandFactory](#)

Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#) 

Constructors

AppCommandFactory()

Private constructor to prevent external instantiation.

```
public AppCommandFactory()
```

Fields

_instance

```
public static AppCommandFactory _instance
```

Field Value

Properties

Instance

Gets the single instance of the AppCommandFactory. Ensures thread safety with a double-check locking mechanism.

```
public static AppCommandFactory Instance { get; }
```

Property Value

[AppCommandFactory](#)

Methods

MakeCommand(string)

Creates an instance of a command based on the specified command type.

```
public override ICommand MakeCommand(string commandType)
```

Parameters

commandType [string](#) 

The type of the command to create (e.g., "circle", "moveto", "drawto").

Returns

ICommand

An instance of the appropriate BOOSE.ICommand implementation, or **null** if the command type is not recognized.

Exceptions

[Exception](#)

Throws an exception if an error occurs during command creation.

Class CustomCircle

Namespace: [ASE Assignment Demo](#)

Assembly: ASE Assignment Demo.dll

Represents a custom circle command that can be executed on a canvas.

```
public class CustomCircle : Circle, ICommand
```









Inheritance

[object](#)  ← [Command](#) ← [CanvasCommand](#) ← [CommandOneParameter](#) ← [Circle](#) ← [CustomCircle](#)

Implements

[ICommand](#)

Inherited Members

[CommandOneParameter.param1](#) , [CommandOneParameter.param1unprocessed](#) ,
[CanvasCommand.yPos](#) , [CanvasCommand.xPos](#) , [CanvasCommand.canvas](#) , [CanvasCommand.Canvas](#) ,
[Command.program](#) , [Command.parameterList](#) , [Command.parameters](#) , [Command.paramsint](#) ,
[Command.Set\(StoredProgram, string\)](#)  , [Command.Compile\(\)](#) , [Command.ProcessParameters\(string\)](#)  ,
[Command.ToString\(\)](#) , [Command.Program](#) , [Command.Name](#) , [Command.ParameterList](#) ,
[Command.Parameters](#) , [Command.Paramsint](#) , [object.Equals\(object\)](#)  , [object.Equals\(object, object\)](#)  ,
[object.GetHashCode\(\)](#)  , [object.GetType\(\)](#)  , [object.MemberwiseClone\(\)](#)  ,
[object.ReferenceEquals\(object, object\)](#) 

Constructors

CustomCircle()

Initializes a new instance of the [CustomCircle](#) class.

```
public CustomCircle()
```

Methods

CheckParameters(string[])

Validates and processes the input parameters for the circle command.

```
public override void CheckParameters(string[] parameter)
```

Parameters

parameter [string](#)[]

Array of parameters where the first parameter is radius and the optional second parameter is fill status.

Exceptions

CommandException

Thrown when parameter validation fails.

Execute()

Executes the circle drawing command on the canvas.

```
public override void Execute()
```

Exceptions

CommandException

Thrown when there is an issue with the command parameters.

Class CustomFor


Namespace: [ASE Assignment Demo](#)

Assembly: ASE Assignment Demo.dll

The CustomFor class extends the functionality of the **For** class from the BOOSE library. This class allows for customization of the **Restrictions** method or any other inherited functionality.

```
public class CustomFor : For, ICommand
```











Inheritance

[object](#)  ← [Command](#) ← [Evaluation](#) ← [Boolean](#) ← [ConditionalCommand](#) ← [For](#) ← CustomFor

Implements

ICommand

Inherited Members

For.Compile() , For.Execute() , For.LoopControlV , For.From , For.To , For.Step ,
ConditionalCommand.EndLineNumber , ConditionalCommand.EndLineNumber ,
ConditionalCommand.Condition , ConditionalCommand.LineNumber , ConditionalCommand.CondType ,
ConditionalCommand.ReturnLineNumber , Boolean.BoolValue , Evaluation.expression ,
Evaluation.evaluatedExpression , Evaluation.varName , Evaluation.value ,
[Evaluation.CheckParameters\(string\[\]\)](#)  , [Evaluation.ProcessExpression\(string\)](#)  , Evaluation.Expression ,
Evaluation.VarName , Evaluation.Value , Evaluation.Local , Command.program , Command.parameterList ,
Command.parameters , Command.paramsint , [Command.Set\(StoredProgram, string\)](#)  ,
[Command.ProcessParameters\(string\)](#)  , Command.ToString() , Command.Program , Command.Name ,
Command.ParameterList , Command.Parameters , Command.Paramsint , [object.Equals\(object\)](#)  ,
[object.Equals\(object, object\)](#)  , [object.GetHashCode\(\)](#)  , [object.GetType\(\)](#)  ,
[object.MemberwiseClone\(\)](#)  , [object.ReferenceEquals\(object, object\)](#) 

Methods

Restrictions()

Overrides the Restrictions method from the **For** class in the BOOSE library. Calls the base **Restrictions** method to maintain the original behavior. Custom restrictions or logic can be added here if needed.

```
public override void Restrictions()
```


Class CustomIf

Namespace: [ASE Assignment Demo](#)

Assembly: ASE Assignment Demo.dll

The **CustomIf** class extends the functionality of the **If** class from the BOOSE library. It allows for customization of the **If** logic and provides additional or modified behavior for conditions and their execution.

```
public class CustomIf : If, ICommand
```

Inheritance

[object](#) ← Command ← Evaluation ← Boolean ← ConditionalCommand ← CompoundCommand ← If ← CustomIf

Implements

ICommand

Inherited Members

CompoundCommand.ReduceRestrictions() , [CompoundCommand.CheckParameters\(string\[\]\)](#) ,
CompoundCommand.Compile() , CompoundCommand.CorrespondingCommand ,
ConditionalCommand.endLineNumber , ConditionalCommand.EndLineNumber ,
ConditionalCommand.Condition , ConditionalCommand.LineNumber , ConditionalCommand.CondType ,
ConditionalCommand.ReturnLineNumber , Boolean.Restrictions() , Boolean.BoolValue ,
Evaluation.expression , Evaluation.evaluatedExpression , Evaluation.varName , Evaluation.value ,
[Evaluation.ProcessExpression\(string\)](#) , Evaluation.Expression , Evaluation.VarName , Evaluation.Value ,
Evaluation.Local , Command.program , Command.parameterList , Command.parameters ,
Command.paramsint , [Command.Set\(StoredProgram, string\)](#) , [Command.ProcessParameters\(string\)](#) ,
Command.ToString() , Command.Program , Command.Name , Command.ParameterList ,
Command.Parameters , Command.Paramsint , [object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) ,
[object.GetHashCode\(\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) ,
[object.ReferenceEquals\(object, object\)](#)

Constructors

CustomIf()

Initializes a new instance of the **CustomIf** class. Automatically invokes the **ReduceRestrictions** method from the base **If** class to modify or simplify restrictions as needed.

```
public CustomIf()
```

Methods

Execute()

Executes the conditional logic as defined in the base `If` class. Can be further extended or overridden with custom behavior.

```
public override void Execute()
```


Class CustomInt

Namespace: [ASE Assignment Demo](#)

Assembly: ASE Assignment Demo.dll

The `CustomInt` class inherits from the `Int` class to provide custom behavior for variable handling, restrictions, and execution logic.

```
public class CustomInt : Int, ICommand
```











Inheritance

[object](#)  ← [Command](#) ← [Evaluation](#) ← [Int](#) ← [CustomInt](#)

Implements

[ICommand](#)

Inherited Members

[Evaluation.expression](#) , [Evaluation.evaluatedExpression](#) , [Evaluation.varName](#) , [Evaluation.value](#) , [Evaluation.CheckParameters\(string\[\]\)](#)  , [Evaluation.ProcessExpression\(string\)](#)  , [Evaluation.Expression](#) , [Evaluation.VarName](#) , [Evaluation.Value](#) , [Evaluation.Local](#) , [Command.program](#) , [Command.parameterList](#) , [Command.parameters](#) , [Command.paramsint](#) , [Command.Set\(StoredProgram, string\)](#)  , [Command.ProcessParameters\(string\)](#)  , [Command.ToString\(\)](#) , [Command.Program](#) , [Command.Name](#) , [Command.ParameterList](#) , [Command.Parameters](#) , [Command.Paramsint](#) , [object.Equals\(object\)](#)  , [object.Equals\(object, object\)](#)  , [object.GetHashCode\(\)](#)  , [object.GetType\(\)](#)  , [object.MemberwiseClone\(\)](#)  , [object.ReferenceEquals\(object, object\)](#) 

Constructors

CustomInt()

Initializes a new instance of the `CustomInt` class. Calls the base constructor to ensure proper initialization of the parent class.

```
public CustomInt()
```

Methods

Compile()

Overrides the `Compile` method to customize behavior during variable compilation. This can include validation, parsing, or other pre-execution logic.

```
public override void Compile()
```

Execute()

Overrides the `Execute` method to customize how variables are evaluated and stored. Includes support for evaluating arithmetic expressions.

```
public override void Execute()
```

Restrictions()

Overrides the `Restrictions` method to define or modify variable constraints. This method can be customized to include specific rules for variables.

```
public override void Restrictions()
```

Class CustomMethod


Namespace: [ASE Assignment Demo](#)

Assembly: ASE Assignment Demo.dll

The **CustomMethod** class extends the **Method** class from the BOOSE library. This class provides customization for method-related functionalities, including reducing restrictions and managing method execution and compilation behavior.

```
public class CustomMethod : Method, ICommand
```











Inheritance

[object](#)  ← [Command](#) ← [Evaluation](#) ← [Boolean](#) ← [ConditionalCommand](#) ← [CompoundCommand](#) ← [Method](#) ← [CustomMethod](#)

Implements

[ICommand](#)

Inherited Members

[Method.CheckParameters\(string\[\]\)](#) , [Method.LocalVariables](#), [Method.MethodName](#), [Method.Type](#), [CompoundCommand.ReduceRestrictions\(\)](#), [CompoundCommand.CorrespondingCommand](#), [ConditionalCommand.endLineNumber](#), [ConditionalCommand.EndLineNumber](#), [ConditionalCommand.Condition](#), [ConditionalCommand.LineNumber](#), [ConditionalCommand.CondType](#), [ConditionalCommand.ReturnLineNumber](#), [Boolean.Restrictions\(\)](#), [Boolean.BoolValue](#), [Evaluation.expression](#), [Evaluation.evaluatedExpression](#), [Evaluation.varName](#), [Evaluation.value](#), [Evaluation.ProcessExpression\(string\)](#) , [Evaluation.Expression](#), [Evaluation.VarName](#), [Evaluation.Value](#), [Evaluation.Local](#), [Command.program](#), [Command.parameterList](#), [Command.parameters](#), [Command.paramsint](#), [Command.Set\(StoredProgram, string\)](#) , [Command.ProcessParameters\(string\)](#) , [Command.ToString\(\)](#), [Command.Program](#), [Command.Name](#), [Command.ParameterList](#), [Command.Parameters](#), [Command.Paramsint](#), [object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) 

Constructors

CustomMethod()

Initializes a new instance of the **CustomMethod** class. Automatically calls the **ReduceRestrictions** method to adjust or simplify restrictions.

```
public CustomMethod()
```

Methods

Compile()

Compiles the method by invoking the base class's `Compile` method. Can be extended with additional compilation logic if required.

```
public override void Compile()
```

Execute()

Executes the method and adds it to the program's method list. This overrides the base class's `Execute` method to include custom execution logic.

```
public override void Execute()
```

Class CustomRectangle

Namespace: [ASE Assignment Demo](#)

Assembly: ASE Assignment Demo.dll

Represents a custom rectangle command that can be executed on a canvas.

```
public class CustomRectangle : CommandOneParameter, ICommand
```









Inheritance

[object](#)  ← [Command](#) ← [CanvasCommand](#) ← [CommandOneParameter](#) ← [CustomRectangle](#)

Implements

[ICommand](#)

Inherited Members

[CommandOneParameter.param1](#) , [CommandOneParameter.param1unprocessed](#) ,
[CanvasCommand.yPos](#) , [CanvasCommand.xPos](#) , [CanvasCommand.canvas](#) , [CanvasCommand.Canvas](#) ,
[Command.program](#) , [Command.parameterList](#) , [Command.parameters](#) , [Command.paramsint](#) ,
[Command.Set\(StoredProgram, string\)](#)  , [Command.Compile\(\)](#) , [Command.ProcessParameters\(string\)](#)  ,
[Command.ToString\(\)](#) , [Command.Program](#) , [Command.Name](#) , [Command.ParameterList](#) ,
[Command.Parameters](#) , [Command.Paramsint](#) , [object.Equals\(object\)](#)  , [object.Equals\(object, object\)](#)  ,
[object.GetHashCode\(\)](#)  , [object.GetType\(\)](#)  , [object.MemberwiseClone\(\)](#)  ,
[object.ReferenceEquals\(object, object\)](#) 

Constructors

CustomRectangle()

Initializes a new instance of the [CustomRectangle](#) class.

```
public CustomRectangle()
```

Methods

CheckParameters(string[])

Validates and processes the input parameters for the rectangle command.

```
public override void CheckParameters(string[] parameter)
```

Parameters

parameter [string](#)[]

Array of parameters where the first is width, the second is height, and the optional third parameter is fill status.

Exceptions

CommandException

Thrown when parameter validation fails.

Execute()

Executes the rectangle drawing command on the canvas.

```
public override void Execute()
```

Exceptions

RestrictionException

Thrown when the width or height exceeds the allowed limit.

Class CustomWhile


Namespace: [ASE Assignment Demo](#)

Assembly: ASE Assignment Demo.dll

The **CustomWhile** class inherits from the **While** class to provide customized behavior for the 'While' loop, with the added functionality of reducing restrictions as needed.

```
public class CustomWhile : While, ICommand
```











Inheritance

[object](#)  ← Command ← Evaluation ← Boolean ← ConditionalCommand ← CompoundCommand ← While ← CustomWhile

Implements

ICommand

Inherited Members

CompoundCommand.ReduceRestrictions() , [CompoundCommand.CheckParameters\(string\[\]\)](#)  , CompoundCommand.Compile() , CompoundCommand.CorrespondingCommand , ConditionalCommand.EndLineNumber , ConditionalCommand.Execute() , ConditionalCommand.EndLineNumber , ConditionalCommand.Condition , ConditionalCommand.LineNumber , ConditionalCommand.CondType , ConditionalCommand.ReturnLineNumber , Boolean.Restrictions() , Boolean.BoolValue , Evaluation.expression , Evaluation.evaluatedExpression , Evaluation.varName , Evaluation.value , [Evaluation.ProcessExpression\(string\)](#)  , Evaluation.Expression , Evaluation.VarName , Evaluation.Value , Evaluation.Local , Command.program , Command.parameterList , Command.parameters , Command.paramsint , [Command.Set\(StoredProgram, string\)](#)  , [Command.ProcessParameters\(string\)](#)  , Command.ToString() , Command.Program , Command.Name , Command.ParameterList , Command.Parameters , Command.Paramsint , [object.Equals\(object\)](#)  , [object.Equals\(object, object\)](#)  , [object.GetHashCode\(\)](#)  , [object.GetType\(\)](#)  , [object.MemberwiseClone\(\)](#)  , [object.ReferenceEquals\(object, object\)](#) 

Constructors

CustomWhile()

Initializes a new instance of the **CustomWhile** class. Calls the **ReduceRestrictions** method from the base **While** class to adjust or simplify any restrictions on the 'While' loop.

```
public CustomWhile()
```


Class Form1

Namespace: [ASE Assignment Demo](#)

Assembly: ASE Assignment Demo.dll

Main form class for the ASE Assignment Demo application.

```
public class Form1 : Form, IDropTarget, ISynchronizeInvoke, IWin32Window,
    IBindableComponent, IComponent, IDisposable, IContainerControl
```

Inheritance

[object](#) < [MarshalByRefObject](#) < [Component](#) < [Control](#) < [ScrollableControl](#) < [ContainerControl](#) < [Form](#) < Form1

Implements

[IDropTarget](#), [ISynchronizeInvoke](#), [IWin32Window](#), [IBindableComponent](#), [IComponent](#), [IDisposable](#), [IContainerControl](#)

Inherited Members

[Form.SetVisibleCore\(bool\)](#), [Form.Activate\(\)](#), [Form.ActivateMdiChild\(Form\)](#), [Form.AddOwnedForm\(Form\)](#), [Form.AdjustFormScrollbars\(bool\)](#), [Form.Close\(\)](#), [Form.CreateAccessibilityInstance\(\)](#), [Form.CreateControlsInstance\(\)](#), [Form.CreateHandle\(\)](#), [Form.DefWndProc\(ref Message\)](#), [Form.ProcessMnemonic\(char\)](#), [Form.CenterToParent\(\)](#), [Form.CenterToScreen\(\)](#), [Form.LayoutMdi\(MdiLayout\)](#), [Form.OnActivated\(EventArgs\)](#), [Form.OnBackgroundImageChanged\(EventArgs\)](#), [Form.OnBackgroundImageLayoutChanged\(EventArgs\)](#), [Form.OnClosing\(CancelEventArgs\)](#), [Form.OnClosed\(EventArgs\)](#), [Form.OnFormClosing\(FormClosingEventArgs\)](#), [Form.OnFormClosed\(FormClosedEventArgs\)](#), [Form.OnCreateControl\(\)](#), [Form.OnDeactivate\(EventArgs\)](#), [Form.OnEnabledChanged\(EventArgs\)](#), [Form.OnEnter\(EventArgs\)](#), [Form.OnFontChanged\(EventArgs\)](#), [Form.OnGotFocus\(EventArgs\)](#), [Form.OnHandleCreated\(EventArgs\)](#), [Form.OnHandleDestroyed\(EventArgs\)](#), [Form.OnHelpButtonClicked\(CancelEventArgs\)](#), [Form.OnLayout\(LayoutEventArgs\)](#), [Form.OnLoad\(EventArgs\)](#), [Form.OnMaximizedBoundsChanged\(EventArgs\)](#), [Form.OnMaximumSizeChanged\(EventArgs\)](#), [Form.OnMinimumSizeChanged\(EventArgs\)](#), [Form.OnInputLanguageChanged\(InputLanguageChangedEventArgs\)](#), [Form.OnInputLanguageChanging\(InputLanguageChangingEventArgs\)](#), [Form.OnVisibleChanged\(EventArgs\)](#), [Form.OnMdiChildActivate\(EventArgs\)](#), [Form.OnMenuStart\(EventArgs\)](#), [Form.OnMenuComplete\(EventArgs\)](#), [Form.OnPaint\(PaintEventArgs\)](#), [Form.OnResize\(EventArgs\)](#), [Form.OnDpiChanged\(DpiChangedEventArgs\)](#), [Form.OnGetDpiScaledSize\(int, int, ref Size\)](#),

[Form.OnRightToLeftLayoutChanged\(EventArgs\)](#), [Form.OnShown\(EventArgs\)](#),
[Form.OnTextChanged\(EventArgs\)](#), [Form.ProcessCmdKey\(ref Message, Keys\)](#),
[Form.ProcessDialogKey\(Keys\)](#), [Form.ProcessDialogChar\(char\)](#),
[Form.ProcessKeyPreview\(ref Message\)](#), [Form.ProcessTabKey\(bool\)](#),
[Form.RemoveOwnedForm\(Form\)](#), [Form.Select\(bool, bool\)](#),
[Form.ScaleMinMaxSize\(float, float, bool\)](#),
[Form.GetScaledBounds\(Rectangle, SizeF, BoundsSpecified\)](#),
[Form.ScaleControl\(SizeF, BoundsSpecified\)](#), [Form.SetBoundsCore\(int, int, int, int, BoundsSpecified\)](#),
[Form.SetClientSizeCore\(int, int\)](#), [Form.SetDesktopBounds\(int, int, int, int\)](#),
[Form.SetDesktopLocation\(int, int\)](#), [Form.Show\(IWin32Window\)](#), [Form.ShowDialog\(\)](#),
[Form.ShowDialog\(IWin32Window\)](#), [Form.ToString\(\)](#), [Form.UpdateDefaultButton\(\)](#),
[Form.OnResizeBegin\(EventArgs\)](#), [Form.OnResizeEnd\(EventArgs\)](#),
[Form.OnStyleChanged\(EventArgs\)](#), [Form.ValidateChildren\(\)](#),
[Form.ValidateChildren\(ValidationConstraints\)](#), [Form.WndProc\(ref Message\)](#), [Form.AcceptButton](#),
[Form.ActiveForm](#), [Form.ActiveMdiChild](#), [Form.AllowTransparency](#), [Form.AutoScroll](#),
[Form.AutoSize](#), [Form.AutoSizeMode](#), [Form.AutoValidate](#), [Form.BackColor](#),
[Form.FormBorderStyle](#), [Form.CancelButton](#), [Form.ClientSize](#), [Form.ControlBox](#),
[Form.CreateParams](#), [Form.DefaultImeMode](#), [Form.DefaultSize](#), [Form.DesktopBounds](#),
[Form.DesktopLocation](#), [Form.DialogResult](#), [Form.HelpButton](#), [Form.Icon](#), [Form.IsMdiChild](#),
[Form.IsMdiContainer](#), [Form.IsRestrictedWindow](#), [Form.KeyPreview](#), [Form.Location](#),
[Form.MaximizedBounds](#), [Form.MaximumSize](#), [Form.MainMenuStrip](#), [Form.MinimumSize](#),
[Form.MaximizeBox](#), [Form.MdiChildren](#), [Form.MdiChildrenMinimizedAnchorBottom](#),
[Form.MdiParent](#), [Form.MinimizeBox](#), [Form.Modal](#), [Form.Opacity](#), [Form.OwnedForms](#),
[Form.Owner](#), [Form.RestoreBounds](#), [Form.RightToLeftLayout](#), [Form.ShowInTaskbar](#),
[Form.ShowIcon](#), [Form.ShowWithoutActivation](#), [Form.Size](#), [Form.SizeGripStyle](#),
[Form.StartPosition](#), [Form.Text](#), [Form.TopLevel](#), [Form.TopMost](#), [Form.TransparencyKey](#),
[Form.WindowState](#), [Form.AutoSizeChanged](#), [Form.AutoValidateChanged](#),
[Form.HelpButtonClicked](#), [Form.MaximizedBoundsChanged](#), [Form.MaximumSizeChanged](#),
[Form.MinimumSizeChanged](#), [Form.Activated](#), [Form.Deactivate](#), [Form.FormClosing](#),
[Form.FormClosed](#), [Form.Load](#), [Form.MdiChildActivate](#), [Form.MenuComplete](#),
[Form.MenuStart](#), [Form.InputLanguageChanged](#), [Form.InputLanguageChanging](#),
[Form.RightToLeftLayoutChanged](#), [Form.Shown](#), [Form.DpiChanged](#), [Form.ResizeBegin](#),
[Form.ResizeEnd](#), [ContainerControl.OnAutoValidateChanged\(EventArgs\)](#),
[ContainerControl.OnMove\(EventArgs\)](#), [ContainerControl.OnParentChanged\(EventArgs\)](#),
[ContainerControl.PerformAutoScale\(\)](#), [ContainerControl.RescaleConstantsForDpi\(int, int\)](#),
[ContainerControl.Validate\(\)](#), [ContainerControl.Validate\(bool\)](#),
[ContainerControl.AutoScaleDimensions](#), [ContainerControl.AutoScaleFactor](#),
[ContainerControl.AutoScaleMode](#), [ContainerControl.BindingContext](#),
[ContainerControl.CanEnableIme](#), [ContainerControl.ActiveControl](#),
[ContainerControl.CurrentAutoScaleDimensions](#), [ContainerControl.ParentForm](#),

[ScrollableControl.ScrollStateAutoScrolling](#) , [ScrollableControl.ScrollStateHScrollVisible](#) ,
[ScrollableControl.ScrollStateVScrollVisible](#) , [ScrollableControl.ScrollStateUserHasScrolled](#) ,
[ScrollableControl.ScrollStateFullDrag](#) , [ScrollableControl.GetScrollState\(int\)](#) ,
[ScrollableControl.OnMouseWheel\(MouseEventArgs\)](#) ,
[ScrollableControl.OnRightToLeftChanged\(EventArgs\)](#) ,
[ScrollableControl.OnPaintBackground\(PaintEventArgs\)](#) ,
[ScrollableControl.OnPaddingChanged\(EventArgs\)](#) , [ScrollableControl.SetDisplayRectLocation\(int, int\)](#) ,
[ScrollableControl.ScrollControlIntoView\(Control\)](#) , [ScrollableControl.ScrollToControl\(Control\)](#) ,
[ScrollableControl.OnScroll\(ScrollEventArgs\)](#) , [ScrollableControl.SetAutoScrollMargin\(int, int\)](#) ,
[ScrollableControl.SetScrollState\(int, bool\)](#) , [ScrollableControl.AutoScrollMargin](#) ,
[ScrollableControl.AutoScrollPosition](#) , [ScrollableControl.AutoScrollMinSize](#) ,
[ScrollableControl.DisplayRectangle](#) , [ScrollableControl.HScroll](#) , [ScrollableControl.HorizontalScroll](#) ,
[ScrollableControl.VScroll](#) , [ScrollableControl.VerticalScroll](#) , [ScrollableControl.Scroll](#) ,
[Control.GetAccessibilityObjectById\(int\)](#) , [Control.SetAutoSizeMode\(AutoSizeMode\)](#) ,
[Control.GetAutoSizeMode\(\)](#) , [Control.GetPreferredSize\(Size\)](#) ,
[Control.AccessibilityNotifyClients\(AccessibleEvents, int\)](#) ,
[Control.AccessibilityNotifyClients\(AccessibleEvents, int, int\)](#) , [Control.BeginInvoke\(Delegate\)](#) ,
[Control.BeginInvoke\(Action\)](#) , [Control.BeginInvoke\(Delegate, params object\[\]\)](#) ,
[Control.BringToFront\(\)](#) , [Control.Contains\(Control\)](#) , [Control.CreateGraphics\(\)](#) ,
[Control.CreateControl\(\)](#) , [Control.DestroyHandle\(\)](#) , [Control.DoDragDrop\(object, DragDropEffects\)](#) ,
[Control.DoDragDrop\(object, DragDropEffects, Bitmap, Point, bool\)](#) ,
[Control.DrawToBitmap\(Bitmap, Rectangle\)](#) , [Control.EndInvoke\(IAsyncResult\)](#) , [Control.FindForm\(\)](#) ,
[Control.GetTopLevel\(\)](#) , [Control.RaiseKeyEvent\(object, KeyEventArgs\)](#) ,
[Control.RaiseMouseEvent\(object, MouseEventArgs\)](#) , [Control.Focus\(\)](#) ,
[Control.FromChildHandle\(nint\)](#) , [Control.FromHandle\(nint\)](#) ,
[Control.GetChildAtPoint\(Point, GetChildAtPointSkip\)](#) , [Control.GetChildAtPoint\(Point\)](#) ,
[Control.GetContainerControl\(\)](#) , [Control.GetNextControl\(Control, bool\)](#) ,
[Control.GetStyle\(ControlStyles\)](#) , [Control.Hide\(\)](#) , [Control.InitLayout\(\)](#) , [Control.Invalidate\(Region\)](#) ,
[Control.Invalidate\(Region, bool\)](#) , [Control.Invalidate\(\)](#) , [Control.Invalidate\(bool\)](#) ,
[Control.Invalidate\(Rectangle\)](#) , [Control.Invalidate\(Rectangle, bool\)](#) , [Control.Invoke\(Action\)](#) ,
[Control.Invoke\(Delegate\)](#) , [Control.Invoke\(Delegate, params object\[\]\)](#) ,
[Control.Invoke<T>\(Func<T>\)](#) , [Control.InvokePaint\(Control, PaintEventArgs\)](#) ,
[Control.InvokePaintBackground\(Control, PaintEventArgs\)](#) , [Control.IsKeyLocked\(Keys\)](#) ,
[Control.IsInputChar\(char\)](#) , [Control.IsInputKey\(Keys\)](#) , [Control.IsMnemonic\(char, string\)](#) ,
[Control.LogicalToDeviceUnits\(int\)](#) , [Control.LogicalToDeviceUnits\(Size\)](#) ,
[Control.ScaleBitmapLogicalToDevice\(ref Bitmap\)](#) , [Control.NotifyInvalidate\(Rectangle\)](#) ,
[Control.InvokeOnClick\(Control, EventArgs\)](#) , [Control.OnAutoSizeChanged\(EventArgs\)](#) ,
[Control.OnBackColorChanged\(EventArgs\)](#) , [Control.OnBindingContextChanged\(EventArgs\)](#) ,
[Control.OnCausesValidationChanged\(EventArgs\)](#) , [Control.OnContextMenuStripChanged\(EventArgs\)](#) ,
[Control.OnCursorChanged\(EventArgs\)](#) , [Control.OnDataContextChanged\(EventArgs\)](#) ,

[Control.OnDockChanged\(EventArgs\)](#), [Control.OnForeColorChanged\(EventArgs\)](#),
[Control.OnNotifyMessage\(Message\)](#), [Control.OnParentBackColorChanged\(EventArgs\)](#),
[Control.OnParentBackgroundImageChanged\(EventArgs\)](#),
[Control.OnParentBindingContextChanged\(EventArgs\)](#), [Control.OnParentCursorChanged\(EventArgs\)](#),
[Control.OnParentDataContextChanged\(EventArgs\)](#), [Control.OnParentEnabledChanged\(EventArgs\)](#),
[Control.OnParentFontChanged\(EventArgs\)](#), [Control.OnParentForeColorChanged\(EventArgs\)](#),
[Control.OnParentRightToLeftChanged\(EventArgs\)](#), [Control.OnParentVisibleChanged\(EventArgs\)](#),
[Control.OnPrint\(PaintEventArgs\)](#), [Control.OnTabIndexChanged\(EventArgs\)](#),
[Control.OnTabStopChanged\(EventArgs\)](#), [Control.OnClick\(EventArgs\)](#),
[Control.OnClientSizeChanged\(EventArgs\)](#), [Control.OnControlAdded\(ControlEventArgs\)](#),
[Control.OnControlRemoved\(ControlEventArgs\)](#), [Control.OnLocationChanged\(EventArgs\)](#),
[Control.OnDoubleClick\(EventArgs\)](#), [Control.OnDragEnter\(DragEventArgs\)](#),
[Control.OnDragOver\(DragEventArgs\)](#), [Control.OnDragLeave\(EventArgs\)](#),
[Control.OnDragDrop\(DragEventArgs\)](#), [Control.OnGiveFeedback\(GiveFeedbackEventArgs\)](#),
[Control.InvokeGotFocus\(Control, EventArgs\)](#), [Control.OnHelpRequested\(HelpEventArgs\)](#),
[Control.OnInvalidated\(InvalidateEventArgs\)](#), [Control.OnKeyDown\(KeyEventArgs\)](#),
[Control.OnKeyPress\(KeyPressEventArgs\)](#), [Control.OnKeyUp\(KeyEventArgs\)](#),
[Control.OnLeave\(EventArgs\)](#), [Control.InvokeLostFocus\(Control, EventArgs\)](#),
[Control.OnLostFocus\(EventArgs\)](#), [Control.OnMarginChanged\(EventArgs\)](#),
[Control.OnMouseDoubleClick\(MouseEventArgs\)](#), [Control.OnMouseClick\(MouseEventArgs\)](#),
[Control.OnMouseCaptureChanged\(EventArgs\)](#), [Control.OnMouseDown\(MouseEventArgs\)](#),
[Control.OnMouseEnter\(EventArgs\)](#), [Control.OnMouseLeave\(EventArgs\)](#),
[Control.OnDpiChangedBeforeParent\(EventArgs\)](#), [Control.OnDpiChangedAfterParent\(EventArgs\)](#),
[Control.OnMouseHover\(EventArgs\)](#), [Control.OnMouseMove\(MouseEventArgs\)](#),
[Control.OnMouseUp\(MouseEventArgs\)](#),
[Control.OnQueryContinueDrag\(QueryContinueDragEventArgs\)](#),
[Control.OnRegionChanged\(EventArgs\)](#), [Control.OnPreviewKeyDown\(PreviewKeyDownEventArgs\)](#),
[Control.OnSizeChanged\(EventArgs\)](#), [Control.OnChangeUICues\(UICuesEventArgs\)](#),
[Control.OnSystemColorsChanged\(EventArgs\)](#), [Control.OnValidating\(CancelEventArgs\)](#),
[Control.OnValidated\(EventArgs\)](#), [Control.PerformLayout\(\)](#), [Control.PerformLayout\(Control, string\)](#),
[Control.PointToClient\(Point\)](#), [Control.PointToScreen\(Point\)](#),
[Control.PreProcessMessage\(ref Message\)](#), [Control.PreProcessControlMessage\(ref Message\)](#),
[Control.ProcessKeyEventArgs\(ref Message\)](#), [Control.ProcessKeyMessage\(ref Message\)](#),
[Control.RaiseDragEvent\(object, DragEventArgs\)](#), [Control.RaisePaintEvent\(object, PaintEventArgs\)](#),
[Control.RecreateHandle\(\)](#), [Control.RectangleToClient\(Rectangle\)](#),
[Control.RectangleToScreen\(Rectangle\)](#), [Control.ReflectMessage\(nint, ref Message\)](#),
[Control.Refresh\(\)](#), [Control.ResetMouseEventArgs\(\)](#), [Control.ResetText\(\)](#), [Control.ResumeLayout\(\)](#),
[Control.ResumeLayout\(bool\)](#), [Control.Scale\(SizeF\)](#), [Control.Select\(\)](#),
[Control.SelectNextControl\(Control, bool, bool, bool, bool\)](#), [Control.SendToBack\(\)](#),
[Control.SetBounds\(int, int, int, int\)](#), [Control.SetBounds\(int, int, int, int, BoundsSpecified\)](#),

[Control.SizeFromClientSize\(Size\)](#), [Control.SetStyle\(ControlStyles, bool\)](#), [Control.SetTopLevel\(bool\)](#), [Control.RtlTranslateAlignment\(HorizontalAlignment\)](#), [Control.RtlTranslateAlignment\(LeftRightAlignment\)](#), [Control.RtlTranslateAlignment\(ContentAlignment\)](#), [Control.RtlTranslateHorizontal\(HorizontalAlignment\)](#), [Control.RtlTranslateLeftRight\(LeftRightAlignment\)](#), [Control.RtlTranslateContent\(ContentAlignment\)](#), [Control.Show\(\)](#), [Control.SuspendLayout\(\)](#), [Control.Update\(\)](#), [Control.UpdateBounds\(\)](#), [Control.UpdateBounds\(int, int, int, int\)](#), [Control.UpdateBounds\(int, int, int, int, int, int\)](#), [Control.UpdateZOrder\(\)](#), [Control.UpdateStyles\(\)](#), [Control.OnImeModeChanged\(EventArgs\)](#), [Control.AccessibilityObject](#), [Control.AccessibleDefaultActionDescription](#), [Control.AccessibleDescription](#), [Control.AccessibleName](#), [Control.AccessibleRole](#), [Control.AllowDrop](#), [Control.Anchor](#), [Control.AutoScrollOffset](#), [Control.LayoutEngine](#), [Control.DataContext](#), [Control.BackgroundImage](#), [Control.BackgroundImageLayout](#), [Control.Bottom](#), [Control.Bounds](#), [Control.CanFocus](#), [Control.CanRaiseEvents](#), [Control.CanSelect](#), [Control.Capture](#), [Control.CausesValidation](#), [Control.CheckForIllegalCrossThreadCalls](#), [Control.ClientRectangle](#), [Control.CompanyName](#), [Control.ContainsFocus](#), [Control.ContextMenuStrip](#), [Control.Controls](#), [Control.Created](#), [Control.Cursor](#), [Control.DataBindings](#), [Control.DefaultBackColor](#), [Control.DefaultCursor](#), [Control.DefaultFont](#), [Control.DefaultForeColor](#), [Control.DefaultMargin](#), [Control.DefaultMaximumSize](#), [Control.DefaultMinimumSize](#), [Control.DefaultPadding](#), [Control.DeviceDpi](#), [Control.IsDisposed](#), [Control.Disposing](#), [Control.Dock](#), [Control.DoubleBuffered](#), [Control.Enabled](#), [Control.Focused](#), [Control.Font](#), [Control.FontHeight](#), [Control.ForeColor](#), [Control.Handle](#), [Control.HasChildren](#), [Control.Height](#), [Control.IsHandleCreated](#), [Control.InvokeRequired](#), [Control.IsAccessible](#), [Control.IsAncestorSiteInDesignMode](#), [Control.IsMirrored](#), [Control.Left](#), [Control.Margin](#), [Control.ModifierKeys](#), [Control.MouseButtons](#), [Control.MousePosition](#), [Control.Name](#), [Control.Parent](#), [Control.ProductName](#), [Control.ProductVersion](#), [Control.RecreatingHandle](#), [Control.Region](#), [Control.RenderRightToLeft](#), [Control.ResizeRedraw](#), [Control.Right](#), [Control.RightToLeft](#), [Control.ScaleChildren](#), [Control.Site](#), [Control.TabIndex](#), [Control.TabStop](#), [Control.Tag](#), [Control.Top](#), [Control.TopLevelControl](#), [Control.ShowKeyboardCues](#), [Control.ShowFocusCues](#), [Control.UseWaitCursor](#), [Control.Visible](#), [Control.Width](#), [Control.PreferredSize](#), [Control.Padding](#), [Control.ImeMode](#), [Control.ImeModeBase](#), [Control.PropagatingImeMode](#), [Control.BackColorChanged](#), [Control.BackgroundImageChanged](#), [Control.BackgroundImageLayoutChanged](#), [Control.BindingContextChanged](#), [Control.CausesValidationChanged](#), [Control.ClientSizeChanged](#), [Control.ContextMenuStripChanged](#), [Control.CursorChanged](#), [Control.DockChanged](#), [Control.EnabledChanged](#), [Control.FontChanged](#), [Control.ForeColorChanged](#), [Control.LocationChanged](#), [Control.MarginChanged](#), [Control.RegionChanged](#), [Control.RightToLeftChanged](#), [Control.SizeChanged](#), [Control.TabIndexChanged](#), [Control.TabStopChanged](#), [Control.TextChanged](#), [Control.VisibleChanged](#), [Control.Click](#),

[Control.ControlAdded](#) , [Control.ControlRemoved](#) , [Control.DataContextChanged](#) ,
[Control.DragDrop](#) , [Control.DragEnter](#) , [Control.DragOver](#) , [Control.DragLeave](#) ,
[Control.GiveFeedback](#) , [Control.HandleCreated](#) , [Control.HandleDestroyed](#) ,
[Control.HelpRequested](#) , [Control.Invalidated](#) , [Control.PaddingChanged](#) , [Control.Paint](#) ,
[Control.QueryContinueDrag](#) , [Control.QueryAccessibilityHelp](#) , [Control.DoubleClick](#) ,
[Control.Enter](#) , [Control.GotFocus](#) , [Control.KeyDown](#) , [Control.KeyPress](#) , [Control.KeyUp](#) ,
[Control.Layout](#) , [Control.Leave](#) , [Control.LostFocus](#) , [Control.MouseClick](#) ,
[Control.MouseDoubleClick](#) , [Control.MouseCaptureChanged](#) , [Control.MouseDown](#) ,
[Control.MouseEnter](#) , [Control.MouseLeave](#) , [Control.DpiChangedBeforeParent](#) ,
[Control.DpiChangedAfterParent](#) , [Control.MouseHover](#) , [Control.MouseMove](#) , [Control.MouseUp](#) ,
[Control.MouseWheel](#) , [Control.Move](#) , [Control.PreviewKeyDown](#) , [Control.Resize](#) ,
[Control.ChangeUICues](#) , [Control.StyleChanged](#) , [Control.SystemColorsChanged](#) ,
[Control.Validating](#) , [Control.Validated](#) , [Control.ParentChanged](#) , [Control.ImeModeChanged](#) ,
[Component.Dispose\(\)](#) , [Component.GetService\(Type\)](#) , [Component.Container](#) ,
[Component.DesignMode](#) , [Component.Events](#) , [Component.Disposed](#) ,
[MarshalByRefObject.GetLifetimeService\(\)](#) , [MarshalByRefObject.InitializeLifetimeService\(\)](#) ,
[MarshalByRefObject.MemberwiseClone\(bool\)](#) , [object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) ,
[object.GetHashCode\(\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) ,
[object.ReferenceEquals\(object, object\)](#)

Constructors

Form1()

Initializes a new instance of the [Form1](#) class.

```
public Form1()
```

Methods

Dispose(bool)

Clean up any resources being used.

```
protected override void Dispose(bool disposing)
```

Parameters

disposing [bool](#)

true if managed resources should be disposed; otherwise, false.

Namespace AppCanvasWeb.Controllers

Classes

[CanvasController](#)

[HomeController](#)

Class CanvasController

Namespace: [AppCanvasWeb.Controllers](#)

Assembly: AppCanvasWeb.dll

```
public class CanvasController : Controller, IActionFilter, IAsyncActionFilter,
IFilterMetadata, IDisposable
```

Inheritance

[object](#) ← [ControllerBase](#) ← [Controller](#) ← CanvasController

Implements

[IActionFilter](#), [IAsyncActionFilter](#), [IFilterMetadata](#), [IDisposable](#)

Inherited Members

[Controller.View\(\)](#), [Controller.View\(string\)](#), [Controller.View\(object\)](#),
[Controller.View\(string, object\)](#), [Controller.PartialView\(\)](#), [Controller.PartialView\(string\)](#),
[Controller.PartialView\(object\)](#), [Controller.PartialView\(string, object\)](#),
[Controller.ViewComponent\(string\)](#), [Controller.ViewComponent\(Type\)](#),
[Controller.ViewComponent\(string, object\)](#), [Controller.ViewComponent\(Type, object\)](#),
[Controller.Json\(object\)](#), [Controller.Json\(object, object\)](#),
[Controller.OnActionExecuting\(ActionExecutingContext\)](#),
[Controller.OnActionExecuted\(ActionExecutedContext\)](#),
[Controller.OnActionExecutionAsync\(ActionExecutingContext, ActionExecutionDelegate\)](#),
[Controller.Dispose\(\)](#), [Controller.Dispose\(bool\)](#), [Controller.ViewData](#), [Controller.TempData](#),
[Controller.ViewBag](#), [ControllerBase.StatusCode\(int\)](#), [ControllerBase.StatusCode\(int, object\)](#),
[ControllerBase.Content\(string\)](#), [ControllerBase.Content\(string, string\)](#),
[ControllerBase.Content\(string, string, Encoding\)](#),
[ControllerBase.Content\(string, MediaTypeHeaderValue\)](#), [ControllerBase.NoContent\(\)](#),
[ControllerBase.Ok\(\)](#), [ControllerBase.Ok\(object\)](#), [ControllerBase.Redirect\(string\)](#),
[ControllerBase.RedirectPermanent\(string\)](#), [ControllerBase.RedirectPreserveMethod\(string\)](#),
[ControllerBase.RedirectPermanentPreserveMethod\(string\)](#), [ControllerBase.LocalRedirect\(string\)](#),
[ControllerBase.LocalRedirectPermanent\(string\)](#), [ControllerBase.LocalRedirectPreserveMethod\(string\)](#),
[ControllerBase.LocalRedirectPermanentPreserveMethod\(string\)](#), [ControllerBase.RedirectToAction\(\)](#),
[ControllerBase.RedirectToAction\(string\)](#), [ControllerBase.RedirectToAction\(string, object\)](#),
[ControllerBase.RedirectToAction\(string, string\)](#),
[ControllerBase.RedirectToAction\(string, string, object\)](#),
[ControllerBase.RedirectToAction\(string, string, string\)](#),
[ControllerBase.RedirectToAction\(string, string, object, string\)](#),

[ControllerBase.RedirectToActionPreserveMethod\(string, string, object, string\)](#) ,
[ControllerBase.RedirectToActionPermanent\(string\)](#) ,
[ControllerBase.RedirectToActionPermanent\(string, object\)](#) ,
[ControllerBase.RedirectToActionPermanent\(string, string\)](#) ,
[ControllerBase.RedirectToActionPermanent\(string, string, string\)](#) ,
[ControllerBase.RedirectToActionPermanent\(string, string, object\)](#) ,
[ControllerBase.RedirectToActionPermanent\(string, string, object, string\)](#) ,
[ControllerBase.RedirectToActionPermanentPreserveMethod\(string, string, object, string\)](#) ,
[ControllerBase.RedirectToRoute\(string\)](#) , [ControllerBase.RedirectToRoute\(object\)](#) ,
[ControllerBase.RedirectToRoute\(string, object\)](#) , [ControllerBase.RedirectToRoute\(string, string\)](#) ,
[ControllerBase.RedirectToRoute\(string, object, string\)](#) ,
[ControllerBase.RedirectToRoutePreserveMethod\(string, object, string\)](#) ,
[ControllerBase.RedirectToRoutePermanent\(string\)](#) ,
[ControllerBase.RedirectToRoutePermanent\(object\)](#) ,
[ControllerBase.RedirectToRoutePermanent\(string, object\)](#) ,
[ControllerBase.RedirectToRoutePermanent\(string, string\)](#) ,
[ControllerBase.RedirectToRoutePermanent\(string, object, string\)](#) ,
[ControllerBase.RedirectToRoutePermanentPreserveMethod\(string, object, string\)](#) ,
[ControllerBase.RedirectToPage\(string\)](#) , [ControllerBase.RedirectToPage\(string, object\)](#) ,
[ControllerBase.RedirectToPage\(string, string\)](#) , [ControllerBase.RedirectToPage\(string, string, object\)](#) ,
[ControllerBase.RedirectToPage\(string, string, string\)](#) ,
[ControllerBase.RedirectToPage\(string, string, object, string\)](#) ,
[ControllerBase.RedirectToPagePermanent\(string\)](#) ,
[ControllerBase.RedirectToPagePermanent\(string, object\)](#) ,
[ControllerBase.RedirectToPagePermanent\(string, string\)](#) ,
[ControllerBase.RedirectToPagePermanent\(string, string, string\)](#) ,
[ControllerBase.RedirectToPagePermanent\(string, string, object, string\)](#) ,
[ControllerBase.RedirectToPagePreserveMethod\(string, string, object, string\)](#) ,
[ControllerBase.RedirectToPagePermanentPreserveMethod\(string, string, object, string\)](#) ,
[ControllerBase.File\(byte\[\], string\)](#) , [ControllerBase.File\(byte\[\], string, bool\)](#) ,
[ControllerBase.File\(byte\[\], string, string\)](#) , [ControllerBase.File\(byte\[\], string, string, bool\)](#) ,
[ControllerBase.File\(byte\[\], string, DateTimeOffset?, EntityTagHeaderValue\)](#) ,
[ControllerBase.File\(byte\[\], string, DateTimeOffset?, EntityTagHeaderValue, bool\)](#) ,
[ControllerBase.File\(byte\[\], string, string, DateTimeOffset?, EntityTagHeaderValue\)](#) ,
[ControllerBase.File\(byte\[\], string, string, DateTimeOffset?, EntityTagHeaderValue, bool\)](#) ,
[ControllerBase.File\(Stream, string\)](#) , [ControllerBase.File\(Stream, string, bool\)](#) ,
[ControllerBase.File\(Stream, string, string\)](#) , [ControllerBase.File\(Stream, string, string, bool\)](#) ,
[ControllerBase.File\(Stream, string, DateTimeOffset?, EntityTagHeaderValue\)](#) ,
[ControllerBase.File\(Stream, string, DateTimeOffset?, EntityTagHeaderValue, bool\)](#) ,
[ControllerBase.File\(Stream, string, string, DateTimeOffset?, EntityTagHeaderValue\)](#) ,

[ControllerBase.File\(Stream, string, string, DateTimeOffset?, EntityTagHeaderValue, bool\)](#),
[ControllerBase.File\(string, string\)](#), [ControllerBase.File\(string, string, bool\)](#),
[ControllerBase.File\(string, string, string\)](#), [ControllerBase.File\(string, string, string, bool\)](#),
[ControllerBase.File\(string, string, DateTimeOffset?, EntityTagHeaderValue\)](#),
[ControllerBase.File\(string, string, DateTimeOffset?, EntityTagHeaderValue, bool\)](#),
[ControllerBase.File\(string, string, string, DateTimeOffset?, EntityTagHeaderValue\)](#),
[ControllerBase.File\(string, string, string, DateTimeOffset?, EntityTagHeaderValue, bool\)](#),
[ControllerBase.PhysicalFile\(string, string\)](#), [ControllerBase.PhysicalFile\(string, string, bool\)](#),
[ControllerBase.PhysicalFile\(string, string, string\)](#),
[ControllerBase.PhysicalFile\(string, string, string, bool\)](#),
[ControllerBase.PhysicalFile\(string, string, DateTimeOffset?, EntityTagHeaderValue\)](#),
[ControllerBase.PhysicalFile\(string, string, DateTimeOffset?, EntityTagHeaderValue, bool\)](#),
[ControllerBase.PhysicalFile\(string, string, string, DateTimeOffset?, EntityTagHeaderValue\)](#),
[ControllerBase.PhysicalFile\(string, string, string, DateTimeOffset?, EntityTagHeaderValue, bool\)](#),
[ControllerBase.Unauthorized\(\)](#), [ControllerBase.Unauthorized\(object\)](#), [ControllerBase.NotFound\(\)](#),
[ControllerBase.NotFound\(object\)](#), [ControllerBase.BadRequest\(\)](#),
[ControllerBase.BadRequest\(object\)](#), [ControllerBase.BadRequest\(ModelStateDictionary\)](#),
[ControllerBase.UnprocessableEntity\(\)](#), [ControllerBase.UnprocessableEntity\(object\)](#),
[ControllerBase.UnprocessableEntity\(ModelStateDictionary\)](#), [ControllerBase.Conflict\(\)](#),
[ControllerBase.Conflict\(object\)](#), [ControllerBase.Conflict\(ModelStateDictionary\)](#),
[ControllerBase.Problem\(string, string, int?, string, string\)](#),
[ControllerBase.ValidationProblem\(ValidationProblemDetails\)](#),
[ControllerBase.ValidationProblem\(ModelStateDictionary\)](#), [ControllerBase.ValidationProblem\(\)](#),
[ControllerBase.ValidationProblem\(string, string, int?, string, string, ModelStateDictionary\)](#),
[ControllerBase.Created\(\)](#), [ControllerBase.Created\(string, object\)](#),
[ControllerBase.Created\(Uri, object\)](#), [ControllerBase.CreatedAtAction\(string, object\)](#),
[ControllerBase.CreatedAtAction\(string, object, object\)](#),
[ControllerBase.CreatedAtAction\(string, string, object, object\)](#),
[ControllerBase.CreatedAtRoute\(string, object\)](#), [ControllerBase.CreatedAtRoute\(object, object\)](#),
[ControllerBase.CreatedAtRoute\(string, object, object\)](#), [ControllerBase.Accepted\(\)](#),
[ControllerBase.Accepted\(object\)](#), [ControllerBase.Accepted\(Uri\)](#), [ControllerBase.Accepted\(string\)](#),
[ControllerBase.Accepted\(string, object\)](#), [ControllerBase.Accepted\(Uri, object\)](#),
[ControllerBase.AcceptedAtAction\(string\)](#), [ControllerBase.AcceptedAtAction\(string, string\)](#),
[ControllerBase.AcceptedAtAction\(string, object\)](#),
[ControllerBase.AcceptedAtAction\(string, string, object\)](#),
[ControllerBase.AcceptedAtAction\(string, object, object\)](#),
[ControllerBase.AcceptedAtAction\(string, string, object, object\)](#),
[ControllerBase.AcceptedAtRoute\(object\)](#), [ControllerBase.AcceptedAtRoute\(string\)](#),
[ControllerBase.AcceptedAtRoute\(string, object\)](#), [ControllerBase.AcceptedAtRoute\(object, object\)](#),
[ControllerBase.AcceptedAtRoute\(string, object, object\)](#), [ControllerBase.Challenge\(\)](#),

[ControllerBase.Challenge\(params string\[\]\).](#), [ControllerBase.Challenge\(AuthenticationProperties\).](#), [ControllerBase.Challenge\(AuthenticationProperties, params string\[\]\).](#), [ControllerBase.Forbid\(\).](#), [ControllerBase.Forbid\(params string\[\]\).](#), [ControllerBase.Forbid\(AuthenticationProperties\).](#), [ControllerBase.Forbid\(AuthenticationProperties, params string\[\]\).](#), [ControllerBase.SignIn\(ClaimsPrincipal\).](#), [ControllerBase.SignIn\(ClaimsPrincipal, string\).](#), [ControllerBase.SignIn\(ClaimsPrincipal, AuthenticationProperties\).](#), [ControllerBase.SignIn\(ClaimsPrincipal, AuthenticationProperties, string\).](#), [ControllerBase.SignOut\(\).](#), [ControllerBase.SignOut\(AuthenticationProperties\).](#), [ControllerBase.SignOut\(params string\[\]\).](#), [ControllerBase.SignOut\(AuthenticationProperties, params string\[\]\).](#), [ControllerBase.TryUpdateModelAsync<TModel>\(TModel\).](#), [ControllerBase.TryUpdateModelAsync<TModel>\(TModel, string\).](#), [ControllerBase.TryUpdateModelAsync<TModel>\(TModel, string, IValueProvider\).](#), [ControllerBase.TryUpdateModelAsync<TModel>\(TModel, string, params Expression<Func<TModel, object>>\[\]\).](#), [ControllerBase.TryUpdateModelAsync<TModel>\(TModel, string, Func<ModelMetadata, bool>\).](#), [ControllerBase.TryUpdateModelAsync<TModel>\(TModel, string, IValueProvider, params Expression<Func<TModel, object>>\[\]\).](#), [ControllerBase.TryUpdateModelAsync<TModel>\(TModel, string, IValueProvider, Func<ModelMetadata, bool>\).](#), [ControllerBase.TryUpdateModelAsync\(object, Type, string\).](#), [ControllerBase.TryUpdateModelAsync\(object, Type, string, IValueProvider, Func<ModelMetadata, bool>\).](#), [ControllerBase.TryValidateModel\(object\).](#), [ControllerBase.TryValidateModel\(object, string\).](#), [ControllerBase.HttpContext.](#), [ControllerBase.Request.](#), [ControllerBase.Response.](#), [ControllerBase.RouteData.](#), [ControllerBase.ModelState.](#), [ControllerBase.ControllerContext.](#), [ControllerBase.MetadataProvider.](#), [ControllerBase.ModelBinderFactory.](#), [ControllerBase.Url.](#), [ControllerBase.ObjectValidator.](#), [ControllerBase.ProblemDetailsFactory.](#), [ControllerBase.User.](#), [ControllerBase.Empty.](#), [object.Equals\(object\).](#), [object.Equals\(object, object\).](#), [object.GetHashCode\(\).](#), [object.GetType\(\).](#), [object.MemberwiseClone\(\).](#), [object.ReferenceEquals\(object, object\).](#), [object.ToString\(\).](#)

Constructors

CanvasController()

Initializes the CanvasController with necessary dependencies.

```
public CanvasController()
```

Methods

CanvasView()

Handles the view for the canvas.

```
public ActionResult CanvasView()
```

Returns

[ActionResult](#)

ClearCanvas()

Clears the canvas and resets the program.

```
[HttpPost]  
public JsonResult ClearCanvas()
```

Returns

[JsonResult](#)

A success response after clearing the canvas.

RunProgram(JsonElement)

Runs the program entered by the user and updates the canvas.

```
[HttpPost]  
public JsonResult RunProgram(JsonElement payload)
```

Parameters

payload [JsonElement](#)

Returns

[JsonResult](#) 

The updated canvas as an image in base64 format.

Class HomeController

Namespace: [AppCanvasWeb.Controllers](#)

Assembly: AppCanvasWeb.dll

```
public class HomeController : Controller, IActionFilter, IAsyncActionFilter,
IFilterMetadata, IDisposable
```

Inheritance

[object](#) ← [ControllerBase](#) ← [Controller](#) ← HomeController

Implements

[IActionFilter](#), [IAsyncActionFilter](#), [IFilterMetadata](#), [IDisposable](#)

Inherited Members

[Controller.View\(\)](#), [Controller.View\(string\)](#), [Controller.View\(object\)](#),
[Controller.View\(string, object\)](#), [Controller.PartialView\(\)](#), [Controller.PartialView\(string\)](#),
[Controller.PartialView\(object\)](#), [Controller.PartialView\(string, object\)](#),
[Controller.ViewComponent\(string\)](#), [Controller.ViewComponent\(Type\)](#),
[Controller.ViewComponent\(string, object\)](#), [Controller.ViewComponent\(Type, object\)](#),
[Controller.Json\(object\)](#), [Controller.Json\(object, object\)](#),
[Controller.OnActionExecuting\(ActionExecutingContext\)](#),
[Controller.OnActionExecuted\(ActionExecutedContext\)](#),
[Controller.OnActionExecutionAsync\(ActionExecutingContext, ActionExecutionDelegate\)](#),
[Controller.Dispose\(\)](#), [Controller.Dispose\(bool\)](#), [Controller.ViewData](#), [Controller.TempData](#),
[Controller.ViewBag](#), [ControllerBase.StatusCode\(int\)](#), [ControllerBase.StatusCode\(int, object\)](#),
[ControllerBase.Content\(string\)](#), [ControllerBase.Content\(string, string\)](#),
[ControllerBase.Content\(string, string, Encoding\)](#),
[ControllerBase.Content\(string, MediaTypeHeaderValue\)](#), [ControllerBase.NoContent\(\)](#),
[ControllerBase.Ok\(\)](#), [ControllerBase.Ok\(object\)](#), [ControllerBase.Redirect\(string\)](#),
[ControllerBase.RedirectPermanent\(string\)](#), [ControllerBase.RedirectPreserveMethod\(string\)](#),
[ControllerBase.RedirectPermanentPreserveMethod\(string\)](#), [ControllerBase.LocalRedirect\(string\)](#),
[ControllerBase.LocalRedirectPermanent\(string\)](#), [ControllerBase.LocalRedirectPreserveMethod\(string\)](#),
[ControllerBase.LocalRedirectPermanentPreserveMethod\(string\)](#), [ControllerBase.RedirectToAction\(\)](#),
[ControllerBase.RedirectToAction\(string\)](#), [ControllerBase.RedirectToAction\(string, object\)](#),
[ControllerBase.RedirectToAction\(string, string\)](#),
[ControllerBase.RedirectToAction\(string, string, object\)](#),
[ControllerBase.RedirectToAction\(string, string, string\)](#),
[ControllerBase.RedirectToAction\(string, string, object, string\)](#),

[ControllerBase.RedirectToActionPreserveMethod\(string, string, object, string\)](#),
[ControllerBase.RedirectToActionPermanent\(string\)](#),
[ControllerBase.RedirectToActionPermanent\(string, object\)](#),
[ControllerBase.RedirectToActionPermanent\(string, string\)](#),
[ControllerBase.RedirectToActionPermanent\(string, string, string\)](#),
[ControllerBase.RedirectToActionPermanent\(string, string, object\)](#),
[ControllerBase.RedirectToActionPermanent\(string, string, object, string\)](#),
[ControllerBase.RedirectToActionPermanentPreserveMethod\(string, string, object, string\)](#),
[ControllerBase.RedirectToRoute\(string\)](#), [ControllerBase.RedirectToRoute\(object\)](#),
[ControllerBase.RedirectToRoute\(string, object\)](#), [ControllerBase.RedirectToRoute\(string, string\)](#),
[ControllerBase.RedirectToRoute\(string, object, string\)](#),
[ControllerBase.RedirectToRoutePreserveMethod\(string, object, string\)](#),
[ControllerBase.RedirectToRoutePermanent\(string\)](#),
[ControllerBase.RedirectToRoutePermanent\(object\)](#),
[ControllerBase.RedirectToRoutePermanent\(string, object\)](#),
[ControllerBase.RedirectToRoutePermanent\(string, string\)](#),
[ControllerBase.RedirectToRoutePermanent\(string, object, string\)](#),
[ControllerBase.RedirectToRoutePermanentPreserveMethod\(string, object, string\)](#),
[ControllerBase.RedirectToPage\(string\)](#), [ControllerBase.RedirectToPage\(string, object\)](#),
[ControllerBase.RedirectToPage\(string, string\)](#), [ControllerBase.RedirectToPage\(string, string, object\)](#),
[ControllerBase.RedirectToPage\(string, string, string\)](#),
[ControllerBase.RedirectToPage\(string, string, object, string\)](#),
[ControllerBase.RedirectToPagePermanent\(string\)](#),
[ControllerBase.RedirectToPagePermanent\(string, object\)](#),
[ControllerBase.RedirectToPagePermanent\(string, string\)](#),
[ControllerBase.RedirectToPagePermanent\(string, string, string\)](#),
[ControllerBase.RedirectToPagePermanent\(string, string, object, string\)](#),
[ControllerBase.RedirectToPagePreserveMethod\(string, string, object, string\)](#),
[ControllerBase.RedirectToPagePermanentPreserveMethod\(string, string, object, string\)](#),
[ControllerBase.File\(byte\[\], string\)](#), [ControllerBase.File\(byte\[\], string, bool\)](#),
[ControllerBase.File\(byte\[\], string, string\)](#), [ControllerBase.File\(byte\[\], string, string, bool\)](#),
[ControllerBase.File\(byte\[\], string, DateTimeOffset?, EntityTagHeaderValue\)](#),
[ControllerBase.File\(byte\[\], string, DateTimeOffset?, EntityTagHeaderValue, bool\)](#),
[ControllerBase.File\(byte\[\], string, string, DateTimeOffset?, EntityTagHeaderValue\)](#),
[ControllerBase.File\(byte\[\], string, string, DateTimeOffset?, EntityTagHeaderValue, bool\)](#),
[ControllerBase.File\(Stream, string\)](#), [ControllerBase.File\(Stream, string, bool\)](#),
[ControllerBase.File\(Stream, string, string\)](#), [ControllerBase.File\(Stream, string, string, bool\)](#),
[ControllerBase.File\(Stream, string, DateTimeOffset?, EntityTagHeaderValue\)](#),
[ControllerBase.File\(Stream, string, DateTimeOffset?, EntityTagHeaderValue, bool\)](#),
[ControllerBase.File\(Stream, string, string, DateTimeOffset?, EntityTagHeaderValue\)](#),

[ControllerBase.File\(Stream, string, string, DateTimeOffset?, EntityTagHeaderValue, bool\)](#),
[ControllerBase.File\(string, string\)](#), [ControllerBase.File\(string, string, bool\)](#),
[ControllerBase.File\(string, string, string\)](#), [ControllerBase.File\(string, string, string, bool\)](#),
[ControllerBase.File\(string, string, DateTimeOffset?, EntityTagHeaderValue\)](#),
[ControllerBase.File\(string, string, DateTimeOffset?, EntityTagHeaderValue, bool\)](#),
[ControllerBase.File\(string, string, string, DateTimeOffset?, EntityTagHeaderValue\)](#),
[ControllerBase.File\(string, string, string, DateTimeOffset?, EntityTagHeaderValue, bool\)](#),
[ControllerBase.PhysicalFile\(string, string\)](#), [ControllerBase.PhysicalFile\(string, string, bool\)](#),
[ControllerBase.PhysicalFile\(string, string, string\)](#),
[ControllerBase.PhysicalFile\(string, string, string, bool\)](#),
[ControllerBase.PhysicalFile\(string, string, DateTimeOffset?, EntityTagHeaderValue\)](#),
[ControllerBase.PhysicalFile\(string, string, DateTimeOffset?, EntityTagHeaderValue, bool\)](#),
[ControllerBase.PhysicalFile\(string, string, string, DateTimeOffset?, EntityTagHeaderValue\)](#),
[ControllerBase.PhysicalFile\(string, string, string, DateTimeOffset?, EntityTagHeaderValue, bool\)](#),
[ControllerBase.Unauthorized\(\)](#), [ControllerBase.Unauthorized\(object\)](#), [ControllerBase.NotFound\(\)](#),
[ControllerBase.NotFound\(object\)](#), [ControllerBase.BadRequest\(\)](#),
[ControllerBase.BadRequest\(object\)](#), [ControllerBase.BadRequest\(ModelStateDictionary\)](#),
[ControllerBase.UnprocessableEntity\(\)](#), [ControllerBase.UnprocessableEntity\(object\)](#),
[ControllerBase.UnprocessableEntity\(ModelStateDictionary\)](#), [ControllerBase.Conflict\(\)](#),
[ControllerBase.Conflict\(object\)](#), [ControllerBase.Conflict\(ModelStateDictionary\)](#),
[ControllerBase.Problem\(string, string, int?, string, string\)](#),
[ControllerBase.ValidationProblem\(ValidationProblemDetails\)](#),
[ControllerBase.ValidationProblem\(ModelStateDictionary\)](#), [ControllerBase.ValidationProblem\(\)](#),
[ControllerBase.ValidationProblem\(string, string, int?, string, string, ModelStateDictionary\)](#),
[ControllerBase.Created\(\)](#), [ControllerBase.Created\(string, object\)](#),
[ControllerBase.Created\(Uri, object\)](#), [ControllerBase.CreatedAtAction\(string, object\)](#),
[ControllerBase.CreatedAtAction\(string, object, object\)](#),
[ControllerBase.CreatedAtAction\(string, string, object, object\)](#),
[ControllerBase.CreatedAtRoute\(string, object\)](#), [ControllerBase.CreatedAtRoute\(object, object\)](#),
[ControllerBase.CreatedAtRoute\(string, object, object\)](#), [ControllerBase.Accepted\(\)](#),
[ControllerBase.Accepted\(object\)](#), [ControllerBase.Accepted\(Uri\)](#), [ControllerBase.Accepted\(string\)](#),
[ControllerBase.Accepted\(string, object\)](#), [ControllerBase.Accepted\(Uri, object\)](#),
[ControllerBase.AcceptedAtAction\(string\)](#), [ControllerBase.AcceptedAtAction\(string, string\)](#),
[ControllerBase.AcceptedAtAction\(string, object\)](#),
[ControllerBase.AcceptedAtAction\(string, string, object\)](#),
[ControllerBase.AcceptedAtAction\(string, object, object\)](#),
[ControllerBase.AcceptedAtAction\(string, string, object, object\)](#),
[ControllerBase.AcceptedAtRoute\(object\)](#), [ControllerBase.AcceptedAtRoute\(string\)](#),
[ControllerBase.AcceptedAtRoute\(string, object\)](#), [ControllerBase.AcceptedAtRoute\(object, object\)](#),
[ControllerBase.AcceptedAtRoute\(string, object, object\)](#), [ControllerBase.Challenge\(\)](#),

[ControllerBase.Challenge\(params string\[\]\)](#), [ControllerBase.Challenge\(AuthenticationProperties\)](#), [ControllerBase.Challenge\(AuthenticationProperties, params string\[\]\)](#), [ControllerBase.Forbid\(\)](#), [ControllerBase.Forbid\(params string\[\]\)](#), [ControllerBase.Forbid\(AuthenticationProperties\)](#), [ControllerBase.Forbid\(AuthenticationProperties, params string\[\]\)](#), [ControllerBase.SignIn\(ClaimsPrincipal\)](#), [ControllerBase.SignIn\(ClaimsPrincipal, string\)](#), [ControllerBase.SignIn\(ClaimsPrincipal, AuthenticationProperties\)](#), [ControllerBase.SignIn\(ClaimsPrincipal, AuthenticationProperties, string\)](#), [ControllerBase.SignOut\(\)](#), [ControllerBase.SignOut\(AuthenticationProperties\)](#), [ControllerBase.SignOut\(params string\[\]\)](#), [ControllerBase.SignOut\(AuthenticationProperties, params string\[\]\)](#), [ControllerBase.TryUpdateModelAsync<TModel>\(TModel\)](#), [ControllerBase.TryUpdateModelAsync<TModel>\(TModel, string\)](#), [ControllerBase.TryUpdateModelAsync<TModel>\(TModel, string, IValueProvider\)](#), [ControllerBase.TryUpdateModelAsync<TModel>\(TModel, string, params Expression<Func<TModel, object>>\[\]\)](#), [ControllerBase.TryUpdateModelAsync<TModel>\(TModel, string, Func<ModelMetadata, bool>\)](#), [ControllerBase.TryUpdateModelAsync<TModel>\(TModel, string, IValueProvider, params Expression<Func<TModel, object>>\[\]\)](#), [ControllerBase.TryUpdateModelAsync<TModel>\(TModel, string, IValueProvider, Func<ModelMetadata, bool>\)](#), [ControllerBase.TryUpdateModelAsync\(object, Type, string\)](#), [ControllerBase.TryUpdateModelAsync\(object, Type, string, IValueProvider, Func<ModelMetadata, bool>\)](#), [ControllerBase.TryValidateModel\(object\)](#), [ControllerBase.TryValidateModel\(object, string\)](#), [ControllerBase.HttpContext](#), [ControllerBase.Request](#), [ControllerBase.Response](#), [ControllerBase.RouteData](#), [ControllerBase.ModelState](#), [ControllerBase.ControllerContext](#), [ControllerBase.MetadataProvider](#), [ControllerBase.ModelBinderFactory](#), [ControllerBase.Url](#), [ControllerBase.ObjectValidator](#), [ControllerBase.ProblemDetailsFactory](#), [ControllerBase.User](#), [ControllerBase.Empty](#), [object.Equals\(object\)](#), [object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#), [object.GetType\(\)](#), [object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#), [object.ToString\(\)](#)

Constructors

HomeController(ILogger<HomeController>)

```
public HomeController(ILogger<HomeController> logger)
```

Parameters

logger [ILogger](#) <[HomeController](#)>

Methods

Error()

```
[ResponseCache(Duration = 0, Location = ResponseCacheLocation.None, NoStore = true)]  
public IActionResult Error()
```

Returns

[IActionResult](#)

Index()

```
public IActionResult Index()
```

Returns

[IActionResult](#)

Privacy()

```
public IActionResult Privacy()
```

Returns

[IActionResult](#)

Namespace AppCanvasWeb.Models

Classes

[ErrorViewModel](#)

Class ErrorViewModel

Namespace: [AppCanvasWeb.Models](#)

Assembly: AppCanvasWeb.dll

```
public class ErrorViewModel
```

Inheritance

[object](#) ← ErrorViewModel

Inherited Members

[object.Equals\(object\)](#), [object.Equals\(object, object\)](#), [object.GetHashCode\(\)](#), [object.GetType\(\)](#), [object.MemberwiseClone\(\)](#), [object.ReferenceEquals\(object, object\)](#), [object.ToString\(\)](#)

Properties

RequestId

```
public string? RequestId { get; set; }
```

Property Value

[string](#)

ShowRequestId

```
public bool ShowRequestId { get; }
```

Property Value

[bool](#)

Namespace AssignTests

Classes

[AppCanvasTest](#)

Contains unit tests for verifying core functionality of BOOSE drawing commands and their integration with the CustomCanvas.

Class AppCanvasTest

Namespace: [AssignTests](#)

Assembly: TestMethods.dll

Contains unit tests for verifying core functionality of BOOSE drawing commands and their integration with the CustomCanvas.

```
[TestClass]
public sealed class AppCanvasTest
```

Inheritance

[object](#)  ← AppCanvasTest

Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,
[object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#) 

Methods

Constructor_ShouldExecuteWithoutExceptions_ForCustomArray(
)

```
[TestMethod]
public void Constructor_ShouldExecuteWithoutExceptions_ForCustomArray()
```

Constructor_ShouldExecuteWithoutExceptions_ForCustomMethod(
)

```
[TestMethod]
public void Constructor_ShouldExecuteWithoutExceptions_ForCustomMethod()
```

Constructor_ShouldExecuteWithoutExceptions_ForCustomReal()

```
[TestMethod]  
public void Constructor_ShouldExecuteWithoutExceptions_ForCustomReal()
```

Constructor_ShouldExecuteWithoutExceptions_ForCustomWhile()

```
[TestMethod]  
public void Constructor_ShouldExecuteWithoutExceptions_ForCustomWhile()
```

DrawTo_ShouldUpdatePenPosition()

Ensures that calling CustomCanvas.DrawTo(int, int) updates the pen position after drawing a line.

```
[TestMethod]  
public void DrawTo_ShouldUpdatePenPosition()
```

Moveto_ShouldUpdatePenPosition()

Verifies that calling CustomCanvas.MoveTo(int, int) updates the pen position correctly.

```
[TestMethod]  
public void Moveto_ShouldUpdatePenPosition()
```

MultilineScript_ShouldExecuteCommandsSequentially()

Tests a multiline script containing multiple commands to ensure that all commands are parsed and executed correctly in sequence.

```
[TestMethod]  
public void MultilineScript_ShouldExecuteCommandsSequentially()
```

Restrictions_ShouldExecuteWithoutExceptionsInt()


```
[TestMethod]  
public void Restrictions_ShouldExecuteWithoutExceptionsInt()
```

Restrictions_ShouldExecuteWithoutExceptions_ForCustomFor()

```
[TestMethod]  
public void Restrictions_ShouldExecuteWithoutExceptions_ForCustomFor()
```

Namespace BOOSE

Classes

[CustomArray](#)

CustomArray class that extends the functionality of Array to provide custom behavior.

[CustomReal](#)

The `CustomReal` class inherits from the `Real` class to provide custom behavior for handling real (floating-point) numbers with customizable restrictions and compilation logic.

[CustomStoredProgram](#)

The `CustomStoredProgram` class extends the `StoredProgram` class to provide custom behavior for executing stored programs, with the added functionality of managing the stack and handling exceptions.

Class CustomArray


Namespace: [BOOSE](#)

Assembly: ASE Assignment Demo.dll

CustomArray class that extends the functionality of Array to provide custom behavior.

```
public class CustomArray : Array, ICommand
```

















Inheritance

[object](#)  ← Command ← Evaluation ← Array ← CustomArray

Implements

ICommand

Inherited Members

Array.PEEK , Array.POKE , Array.type , Array.rows , Array.columns , Array.valueInt , Array.valueReal , Array.intArray , Array.realArray , Array.pokeValue , Array.peekVar , Array.rowS , Array.columnS , Array.row , Array.column , Array.ArrayRestrictions() , Array.ReduceRestrictionCounter() , Array.Compile() , [Array.CheckParameters\(string\[\]\)](#)  , Array.Execute() , [Array.ProcessArrayParametersCompile\(bool\)](#)  , [Array.ProcessArrayParametersExecute\(bool\)](#)  , [Array.SetIntArray\(int, int, int\)](#)  , [Array.SetRealArray\(double, int, int\)](#)  , [Array.GetIntArray\(int, int\)](#)  , [Array.GetRealArray\(int, int\)](#)  , Array.Rows , Array.Columns , Evaluation.expression , Evaluation.evaluatedExpression , Evaluation.varName , Evaluation.value , [Evaluation.ProcessExpression\(string\)](#)  , Evaluation.Expression , Evaluation.VarName , Evaluation.Value , Evaluation.Local , Command.program , Command.parameterList , Command.parameters , Command.paramsint , [Command.Set\(StoredProgram, string\)](#)  , [Command.ProcessParameters\(string\)](#)  , Command.ToString() , Command.Program , Command.Name , Command.ParameterList , Command.Parameters , Command.Paramsint , [object.Equals\(object\)](#)  , [object.Equals\(object, object\)](#)  , [object.GetHashCode\(\)](#)  , [object.GetType\(\)](#)  , [object.MemberwiseClone\(\)](#)  , [object.ReferenceEquals\(object, object\)](#) 

Constructors

CustomArray()

Constructor that increments the array count and checks restrictions.

```
public CustomArray()
```


Class CustomReal


Namespace: [BOOSE](#)

Assembly: ASE Assignment Demo.dll

The `CustomReal` class inherits from the `Real` class to provide custom behavior for handling real (floating-point) numbers with customizable restrictions and compilation logic.

```
public class CustomReal : Real, ICommand
```











Inheritance

[object](#)  ← [Command](#) ← [Evaluation](#) ← [Real](#) ← [CustomReal](#)

Implements

[ICommand](#)

Inherited Members

[Real.Execute\(\)](#) , [Real.Value](#) , [Evaluation.expression](#) , [Evaluation.evaluatedExpression](#) , [Evaluation.varName](#) , [Evaluation.value](#) , [Evaluation.CheckParameters\(string\[\]\)](#)  , [Evaluation.ProcessExpression\(string\)](#)  , [Evaluation.Expression](#) , [Evaluation.VarName](#) , [Evaluation.Local](#) , [Command.program](#) , [Command.parameterList](#) , [Command.parameters](#) , [Command.paramsint](#) , [Command.Set\(StoredProgram, string\)](#)  , [Command.ProcessParameters\(string\)](#)  , [Command.ToString\(\)](#) , [Command.Program](#) , [Command.Name](#) , [Command.ParameterList](#) , [Command.Parameters](#) , [Command.Paramsint](#) , [object.Equals\(object\)](#)  , [object.Equals\(object, object\)](#)  , [object.GetHashCode\(\)](#)  , [object.GetType\(\)](#)  , [object.MemberwiseClone\(\)](#)  , [object.ReferenceEquals\(object, object\)](#) 

Constructors

CustomReal()

Initializes a new instance of the `CustomReal` class. Calls the base constructor to ensure proper initialization of the parent class.

```
public CustomReal()
```

Methods

Compile()

Overrides the `Compile` method to handle custom compilation logic for real variables. Calls the base class's `Compile` method and allows additional compilation logic.

```
public override void Compile()
```

Restrictions()

Overrides the `Restrictions` method to define or modify restrictions for real variables. This method can be customized to include specific rules for real number limits or constraints.

```
public override void Restrictions()
```

Class CustomStoredProgram

Namespace: [BOOSE](#)

Assembly: ASE Assignment Demo.dll

The **CustomStoredProgram** class extends the **StoredProgram** class to provide custom behavior for executing stored programs, with the added functionality of managing the stack and handling exceptions.

```
public class CustomStoredProgram : StoredProgram, IList, ICollection, IEnumerable,
    ICloneable, IStoredProgram
```



Inheritance

[object](#)  ← [ArrayList](#)  ← StoredProgram ← CustomStoredProgram

Implements

[IList](#) , [ICollection](#) , [IEnumerable](#) , [ICloneable](#) , IStoredProgram

Inherited Members

StoredProgram.SyntaxOk , StoredProgram.AddMethod(Method) , [StoredProgram.GetMethod\(string\)](#)  ,
StoredProgram.AddVariable(Evaluation) , [StoredProgram.GetVariable\(string\)](#)  ,
[StoredProgram.GetVariable\(int\)](#)  , StoredProgram.FindVariable(Evaluation) ,
[StoredProgram.FindVariable\(string\)](#)  , [StoredProgram.VariableExists\(string\)](#)  ,
[StoredProgram.GetVarValue\(string\)](#)  , [StoredProgram.UpdateVariable\(string, int\)](#)  ,
[StoredProgram.UpdateVariable\(string, double\)](#)  , [StoredProgram.UpdateVariable\(string, bool\)](#)  ,
[StoredProgram.DeleteVariable\(string\)](#)  , [StoredProgram.IsExpression\(string\)](#)  ,
[StoredProgram.EvaluateExpressionWithString\(string\)](#)  , [StoredProgram.EvaluateExpression\(string\)](#)  ,
StoredProgram.Push(ConditionalCommand) , StoredProgram.Pop() , StoredProgram.Add(Command) ,
StoredProgram.NextCommand() , StoredProgram.ResetProgram() , StoredProgram.CommandsLeft() ,
StoredProgram.PC , [ArrayList.Adapter\(IList\)](#)  , [ArrayList.Add\(object\)](#)  ,
[ArrayList.AddRange\(ICollection\)](#)  , [ArrayList.BinarySearch\(int, int, object, IComparer\)](#)  ,
[ArrayList.BinarySearch\(object\)](#)  , [ArrayList.BinarySearch\(object, IComparer\)](#)  , [ArrayList.Clear\(\)](#)  ,
[ArrayList.Clone\(\)](#)  , [ArrayList.Contains\(object\)](#)  , [ArrayList.CopyTo\(Array\)](#)  ,
[ArrayList.CopyTo\(Array, int\)](#)  , [ArrayList.CopyTo\(int, Array, int, int\)](#)  , [ArrayList.FixedSize\(ArrayList\)](#)  ,
[ArrayList.FixedSize\(IList\)](#)  , [ArrayList.GetEnumerator\(\)](#)  , [ArrayList.GetEnumerator\(int, int\)](#)  ,
[ArrayList.GetRange\(int, int\)](#)  , [ArrayList.IndexOf\(object\)](#)  , [ArrayList.IndexOf\(object, int\)](#)  ,
[ArrayList.IndexOf\(object, int, int\)](#)  , [ArrayList.Insert\(int, object\)](#)  ,
[ArrayList.InsertRange\(int, ICollection\)](#)  , [ArrayList.LastIndexOf\(object\)](#)  ,
[ArrayList.LastIndexOf\(object, int\)](#)  , [ArrayList.LastIndexOf\(object, int, int\)](#)  ,
[ArrayList.ReadOnly\(ArrayList\)](#)  , [ArrayList.ReadOnly\(IList\)](#)  , [ArrayList.Remove\(object\)](#)  ,
[ArrayList.RemoveAt\(int\)](#)  , [ArrayList.RemoveRange\(int, int\)](#)  , [ArrayList.Repeat\(object, int\)](#)  ,

[ArrayList.Reverse\(\)](#) , [ArrayList.Reverse\(int, int\)](#) , [ArrayList.SetRange\(int, ICollection\)](#) , [ArrayList.Sort\(\)](#) , [ArrayList.Sort\(IComparer\)](#) , [ArrayList.Sort\(int, int, IComparer\)](#) , [ArrayList.Synchronized\(ArrayList\)](#) , [ArrayList.Synchronized\(IList\)](#) , [ArrayList.ToArray\(\)](#) , [ArrayList.ToArray\(Type\)](#) , [ArrayList.TrimToSize\(\)](#) , [ArrayList.Capacity](#) , [ArrayList.Count](#) , [ArrayList.IsFixedSize](#) , [ArrayList.IsReadOnly](#) , [ArrayList.IsSynchronized](#) , [ArrayList.this\[int\]](#) , [ArrayList.SyncRoot](#) , [object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#)

Constructors

CustomStoredProgram(ICanvas)

Initializes a new instance of the `CustomStoredProgram` class. The constructor accesses the private `Stack` field from the base class using reflection, and initializes it if necessary.

```
public CustomStoredProgram(ICanvas canvas)
```

Parameters

`canvas` `ICanvas`

The `ICanvas` instance used for visual representation of the program.

Methods

Run()

Runs the stored program by executing each command in the sequence. The program continues to run while commands are available, and the stack is managed.

```
public override void Run()
```


Namespace ase_assessment

Classes

[CustomWrite](#)

Represents write commandx This is the write class which Implements Evaluation class to perform command execution and parameter checking.

Class CustomWrite

Namespace: [ase assessment](#)

Assembly: ASE Assignment Demo.dll

Represents write command. This is the write class which implements Evaluation class to perform command execution and parameter checking.

```
public class CustomWrite : Evaluation, ICommand
```










Inheritance

[object](#)  ← Command ← Evaluation ← CustomWrite

Implements

ICommand

Inherited Members

Evaluation.expression , Evaluation.evaluatedExpression , Evaluation.varName , Evaluation.value , Evaluation.Compile() , [Evaluation.ProcessExpression\(string\)](#)  , Evaluation.Expression , Evaluation.VarName , Evaluation.Value , Evaluation.Local , Command.program , Command.parameterList , Command.parameters , Command.paramsint , [Command.Set\(StoredProgram, string\)](#)  , [Command.ProcessParameters\(string\)](#)  , Command.ToString() , Command.Program , Command.Name , Command.ParameterList , Command.Parameters , Command.Paramsint , [object.Equals\(object\)](#)  , [object.Equals\(object, object\)](#)  , [object.GetHashCode\(\)](#)  , [object.GetType\(\)](#)  , [object.MemberwiseClone\(\)](#)  , [object.ReferenceEquals\(object, object\)](#) 



Properties

OutputCallback

Static output callback handler (a delegate) for the output. This delegate is called with the evaluated expression if it is not null.

```
public static Action<string>? OutputCallback { get; set; }
```

Property Value

[Action](#)  <[string](#)  >

Methods

CheckParameters(string[])

Checks if the parameters is passed to the 'Write' command are verified. Gurantees atleast one parameter is passes to the write command.

```
public override void CheckParameters(string[] parameter)
```

Parameters

parameter [string](#)[]

The parameter which needs to be checked.

Exceptions

CommandException

Throws error when parameter length is less than 1.

Execute()

Executes the 'Write' command by evaluating the first parameters. if not null then passess to the output callback.

```
public override void Execute()
```

Exceptions

CommandException

Throws error when parameter is null.