Namespace BankAccountNS

Classes

BankAccount

Represents a bank account with basic functionalities for crediting and debiting.

Class BankAccount

Namespace: [BankAccountNS](#)

Assembly: Bank.dll

Represents a bank account with basic functionalities for crediting and debiting.

```
public class BankAccount
```

Inheritance

[object](#) ← BankAccount

Inherited Members

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

Constructors

BankAccount(string, double)

Initializes a new instance of the [BankAccount](#) class with a specified customer name and initial balance.

```
public BankAccount(string customerName, double balance)
```

Parameters

`customerName` [string](#)

   The name of the customer associated with the account.

`balance` [double](#)

   The initial balance of the account.

Fields

DebitAmountExceedsBalanceMessage

Message indicating that the debit amount exceeds the current balance.

```
public const string DebitAmountExceedsBalanceMessage = "Debit amount exceeds balance"
```

Field Value

[string ⧉](#)

DebitAmountLessThanZeroMessage

Message indicating that the debit amount is less than zero.

```
public const string DebitAmountLessThanZeroMessage = "Debit amount is less than zero"
```

Field Value

[string ⧉](#)

Properties

Balance

Gets the current balance of the account.

```
public double Balance { get; }
```

Property Value

[double ⧉](#)

CustomerName

Gets the name of the customer associated with this account.

```
public string CustomerName { get; }
```

Property Value

[string↗](#)

Methods

Credit(double)

Adds a specified amount to the account balance.

```
public void Credit(double amount)
```

Parameters

amount [double↗](#)

The amount to be credited to the account.

Remarks

This method validates that the credit amount is not negative. If the `amount` is less than zero, it throws an [ArgumentOutOfRangeException↗](#).

Exceptions

[ArgumentOutOfRangeException↗](#)

Thrown when the `amount` is less than zero.

Debit(double)

Deducts a specified amount from the account balance.

```
public void Debit(double amount)
```

Parameters

amount [double↗](#)

The amount to be debited from the account.

Remarks

This method checks that the debit amount is within valid limits. If the amount is greater than the balance, it throws an [ArgumentOutOfRangeException](#)⧉ with [DebitAmountExceedsBalanceMessage](#). If the amount is less than zero, it throws an [ArgumentOutOfRangeException](#)⧉ with [DebitAmountLessThanZero Message](#).

Exceptions

[ArgumentOutOfRangeException](#)⧉

Thrown when the `amount` is greater than the current balance or if the `amount` is less than zero.

Main()

Entry point for demonstrating the [BankAccount](#) class functionality.

```
public static void Main()
```

Remarks

This main method creates an instance of [BankAccount](#), performs a credit and a debit operation, and outputs the resulting balance to the console.

Namespace BankTests

Classes

[BankAccountsTests](#)

Contains unit tests for the [BankAccount](#) class to validate the Debit functionality and handle specific scenarios like invalid amounts.

Class BankAccountsTests

Namespace: BankTests

Assembly: BankTests.dll

Contains unit tests for the BankAccount class to validate the Debit functionality and handle specific scenarios like invalid amounts.

```
[TestClass]
public class BankAccountsTests
```

Inheritance

object ← BankAccountsTests

Inherited Members

object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Methods

Debit_WhenAmountIsLessThanZero_ShouldThrowArgumentOutOfRange()

Tests that attempting to debit a negative amount throws an ArgumentOutOfRangeException.

```
public void Debit_WhenAmountIsLessThanZero_ShouldThrowArgumentOutOfRange()
```

Remarks

This test checks that the Debit(double) method validates the debit amount and throws an exception if the amount is negative, ensuring that invalid transactions are not processed.

Exceptions

ArgumentOutOfRangeException

Thrown when the debit amount is less than zero.

Debit_WhenAmountIsMoreThanBalance_ShouldThrowArgumentOutOfRange()

Tests that attempting to debit an amount greater than the current balance throws an [ArgumentOutOfRangeException](#)⧉.

```
public void Debit_WhenAmountIsMoreThanBalance_ShouldThrowArgumentOutOfRange()
```

Remarks

This test verifies that the [Debit(double)](#) method does not allow the account balance to go below zero by throwing an exception if the debit amount exceeds the current balance.

Exceptions

[ArgumentOutOfRangeException](#)⧉

Thrown when the debit amount exceeds the available balance.

Debit_WithValidAmountUpdatesBalance()

Tests that a valid debit amount correctly updates the account balance.

```
public void Debit_WithValidAmountUpdatesBalance()
```

Remarks

This test verifies that when a valid amount is debited, the balance reflects the deduction. A tolerance of 0.001 is allowed in the balance check to account for floating-point precision.

Namespace BooseTesting

Classes

[AppCanvasBase](#)
    Provides an abstract base class for a canvas with drawing methods. Includes functionality for setting color, moving the cursor, and drawing shapes and text.

[Form1](#)
    The Form1 class is responsible for providing a UI interface for Duncan's BOOSE Interpreter. It allows the user to enter drawing commands, interpret those commands, and render the output on a canvas.

[Scenario1](#)
    Scenario 1: Demonstrates drawing circles and rectangles with different colors and positions.

[Scenario2](#)
    Scenario 2: Demonstrates drawing circles and rectangles with repeated moves and color changes.

[Scenario3](#)
    Scenario 3: Displays calculated values on the canvas.

[Scenario4](#)
    Scenario 4: Demonstrates array usage by displaying text and shapes based on values.

[UnrestrictedForScenario](#)
    Unrestricted For Scenario: Demonstrates a loop to draw shapes with dynamic size and position.

[UnrestrictedIfScenario1](#)
    Unrestricted If Scenario 1: Demonstrates if-else logic with nested conditions and color changes.

[UnrestrictedMethodScenario](#)
    Unrestricted Method Scenario: Demonstrates calling a method and using its return value for drawing.

[UnrestrictedWhileScenario](#)
    Unrestricted While Scenario: Demonstrates a while loop to draw shapes with decreasing size.

[VariablesManager](#)
    Manages variables across the BooseTesting application.

Class AppCanvasBase

Namespace: [BooseTesting](#)

Assembly: BooseTesting.dll

Provides an abstract base class for a canvas with drawing methods. Includes functionality for setting color, moving the cursor, and drawing shapes and text.

```
public abstract class AppCanvasBase
```

Inheritance

[object](#) ← AppCanvasBase

Derived

[Scenario1](#), [Scenario2](#), [Scenario3](#), [Scenario4](#), [UnrestrictedForScenario](#), [UnrestrictedIfScenario1](#), [UnrestrictedMethodScenario](#), [UnrestrictedWhileScenario](#)

Inherited Members

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

Constructors

AppCanvasBase(int, int)

Initializes a new instance of the AppCanvasBase class with specified width and height.

```
public AppCanvasBase(int width, int height)
```

Parameters

`width` [int](#)

The width of the canvas.

`height` [int](#)

The height of the canvas.

Fields

CurrentPen

Pen used to draw shapes and lines.

```
protected Pen CurrentPen
```

Field Value

[Pen⬈](#)

DrawingBitmap

Bitmap object used as the drawing surface.

```
protected Bitmap DrawingBitmap
```

Field Value

[Bitmap⬈](#)

Graphics

Graphics object used to draw shapes on the bitmap.

```
protected Graphics Graphics
```

Field Value

[Graphics⬈](#)

Xpos

X-coordinate for the current drawing position.

```
protected int Xpos
```

Field Value

[int ↗](#)

Ypos

Y-coordinate for the current drawing position.

```
protected int Ypos
```

Field Value

[int ↗](#)

Methods

Circle(int)

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

```
protected void Circle(int radius)
```

Parameters

radius [int ↗](#)

   The radius of the circle.

ExecuteCommands()

Executes drawing commands specific to each scenario. Must be implemented by subclasses.

```
public abstract void ExecuteCommands()
```

GetBitmap()

Retrieves the bitmap of the current drawing.

```
public Bitmap GetBitmap()
```

Returns

[Bitmap](⧉)

The bitmap containing the drawing.

MoveTo(int, int)

Moves the drawing cursor to the specified coordinates.

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

Parameters

x [int](⧉)

The x-coordinate.

y [int](⧉)

The y-coordinate.

Rect(int, int)

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

```
protected void Rect(int width, int height)
```

Parameters

width [int](⧉)

The width of the rectangle.

height int

    The height of the rectangle.

SetColour(int, int, int)

Sets the color of the pen used for drawing.

```
public void SetColour(int r, int g, int b)
```

Parameters

r int

    Red component of the color.

g int

    Green component of the color.

b int

    Blue component of the color.

WriteText(string)

Draws text at the current position.

```
protected void WriteText(string text)
```

Parameters

text string

    The text to draw.

Class Form1

Namespace: [BooseTesting](#)

Assembly: BooseTesting.dll

The Form1 class is responsible for providing a UI interface for Duncan's BOOSE Interpreter. It allows the user to enter drawing commands, interpret those commands, and render the output on a canvas.

```
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.Dispose(bool)](#) , [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()
```

Class Scenario1

Namespace: BooseTesting

Assembly: BooseTesting.dll

Scenario 1: Demonstrates drawing circles and rectangles with different colors and positions.

```
public class Scenario1 : AppCanvasBase
```

Inheritance

object ← AppCanvasBase ← Scenario1

Inherited Members

AppCanvasBase.DrawingBitmap , AppCanvasBase.Graphics , AppCanvasBase.CurrentPen , AppCanvasBase.Xpos , AppCanvasBase.Ypos , AppCanvasBase.GetBitmap() , AppCanvasBase.SetColour(int, int, int) , AppCanvasBase.MoveTo(int, int) , AppCanvasBase.Circle(int) , AppCanvasBase.Rect(int, int) , AppCanvasBase.WriteText(string) , object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() , object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Constructors

Scenario1(int, int)

```
public Scenario1(int width, int height)
```

Parameters

width int

height int

Methods

ExecuteCommands()

Executes drawing commands specific to each scenario. Must be implemented by subclasses.

```csharp
public override void ExecuteCommands()
```

Class Scenario2

Namespace: BooseTesting

Assembly: BooseTesting.dll

Scenario 2: Demonstrates drawing circles and rectangles with repeated moves and color changes.

```
public class Scenario2 : AppCanvasBase
```

Inheritance

object ← AppCanvasBase ← Scenario2

Inherited Members

AppCanvasBase.DrawingBitmap , AppCanvasBase.Graphics , AppCanvasBase.CurrentPen ,
AppCanvasBase.Xpos , AppCanvasBase.Ypos , AppCanvasBase.GetBitmap() ,
AppCanvasBase.SetColour(int, int, int) , AppCanvasBase.MoveTo(int, int) , AppCanvasBase.Circle(int) ,
AppCanvasBase.Rect(int, int) , AppCanvasBase.WriteText(string) , object.Equals(object) ,
object.Equals(object, object) , object.GetHashCode() , object.GetType() ,
object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Constructors

Scenario2(int, int)

```
public Scenario2(int width, int height)
```

Parameters

width int

height int

Methods

ExecuteCommands()

Executes drawing commands specific to each scenario. Must be implemented by subclasses.

```csharp
public override void ExecuteCommands()
```

Class Scenario3

Scenario 3: Displays calculated values on the canvas.

```
public class Scenario3 : AppCanvasBase
```

Inheritance

[object](#) ← [AppCanvasBase](#) ← Scenario3

Inherited Members

[AppCanvasBase.DrawingBitmap](#) , [AppCanvasBase.Graphics](#) , [AppCanvasBase.CurrentPen](#) , [AppCanvasBase.Xpos](#) , [AppCanvasBase.Ypos](#) , [AppCanvasBase.GetBitmap()](#) , [AppCanvasBase.SetColour(int, int, int)](#) , [AppCanvasBase.MoveTo(int, int)](#) , [AppCanvasBase.Circle(int)](#) , [AppCanvasBase.Rect(int, int)](#) , [AppCanvasBase.WriteText(string)](#) , [object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) , [object.GetType()](#) , [object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) , [object.ToString()](#)

Constructors

Scenario3(int, int)

```
public Scenario3(int width, int height)
```

Parameters

`width` [int](#)

`height` [int](#)

Methods

ExecuteCommands()

Executes drawing commands specific to each scenario. Must be implemented by subclasses.

```csharp
public override void ExecuteCommands()
```

Class Scenario4

Namespace: BooseTesting

Assembly: BooseTesting.dll

Scenario 4: Demonstrates array usage by displaying text and shapes based on values.

```
public class Scenario4 : AppCanvasBase
```

Inheritance

object ← AppCanvasBase ← Scenario4

Inherited Members

AppCanvasBase.DrawingBitmap , AppCanvasBase.Graphics , AppCanvasBase.CurrentPen ,
AppCanvasBase.Xpos , AppCanvasBase.Ypos , AppCanvasBase.GetBitmap() ,
AppCanvasBase.SetColour(int, int, int) , AppCanvasBase.MoveTo(int, int) , AppCanvasBase.Circle(int) ,
AppCanvasBase.Rect(int, int) , AppCanvasBase.WriteText(string) , object.Equals(object) ,
object.Equals(object, object) , object.GetHashCode() , object.GetType() ,
object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Constructors

Scenario4(int, int)

```
public Scenario4(int width, int height)
```

Parameters

`width` int

`height` int

Methods

ExecuteCommands()

Executes drawing commands specific to each scenario. Must be implemented by subclasses.

```csharp
public override void ExecuteCommands()
```

Class UnrestrictedForScenario

Namespace: [BooseTesting](#)

Assembly: BooseTesting.dll

Unrestricted For Scenario: Demonstrates a loop to draw shapes with dynamic size and position.

```
public class UnrestrictedForScenario : AppCanvasBase
```

Inheritance

[object](#) ← [AppCanvasBase](#) ← UnrestrictedForScenario

Inherited Members

[AppCanvasBase.DrawingBitmap](#) , [AppCanvasBase.Graphics](#) , [AppCanvasBase.CurrentPen](#) ,
[AppCanvasBase.Xpos](#) , [AppCanvasBase.Ypos](#) , [AppCanvasBase.GetBitmap()](#) ,
[AppCanvasBase.SetColour(int, int, int)](#) , [AppCanvasBase.MoveTo(int, int)](#) , [AppCanvasBase.Circle(int)](#) ,
[AppCanvasBase.Rect(int, int)](#) , [AppCanvasBase.WriteText(string)](#) , [object.Equals(object)](#) ,
[object.Equals(object, object)](#) , [object.GetHashCode()](#) , [object.GetType()](#) ,
[object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) , [object.ToString()](#)

Constructors

UnrestrictedForScenario(int, int)

```
public UnrestrictedForScenario(int width, int height)
```

Parameters

`width` [int](#)

`height` [int](#)

Methods

ExecuteCommands()

Executes drawing commands specific to each scenario. Must be implemented by subclasses.

```csharp
public override void ExecuteCommands()
```

Class UnrestrictedIfScenario1

Namespace: BooseTesting

Assembly: BooseTesting.dll

Unrestricted If Scenario 1: Demonstrates if-else logic with nested conditions and color changes.

```
public class UnrestrictedIfScenario1 : AppCanvasBase
```

Inheritance

object ← AppCanvasBase ← UnrestrictedIfScenario1

Inherited Members

AppCanvasBase.DrawingBitmap , AppCanvasBase.Graphics , AppCanvasBase.CurrentPen ,
AppCanvasBase.Xpos , AppCanvasBase.Ypos , AppCanvasBase.GetBitmap() ,
[AppCanvasBase.SetColour(int, int, int)](AppCanvasBase.SetColour(int,int,int)) , [AppCanvasBase.MoveTo(int, int)](AppCanvasBase.MoveTo(int,int)) , AppCanvasBase.Circle(int) ,
[AppCanvasBase.Rect(int, int)](AppCanvasBase.Rect(int,int)) , AppCanvasBase.WriteText(string) , object.Equals(object) ,
[object.Equals(object, object)](object.Equals(object,object)) , object.GetHashCode() , object.GetType() ,
object.MemberwiseClone() , [object.ReferenceEquals(object, object)](object.ReferenceEquals(object,object)) , object.ToString()

Constructors

UnrestrictedIfScenario1(int, int)

```
public UnrestrictedIfScenario1(int width, int height)
```

Parameters

`width` [int](int)

`height` [int](int)

Methods

ExecuteCommands()

Executes drawing commands specific to each scenario. Must be implemented by subclasses.

```csharp
public override void ExecuteCommands()
```

Class UnrestrictedMethodScenario

Namespace: BooseTesting

Assembly: BooseTesting.dll

Unrestricted Method Scenario: Demonstrates calling a method and using its return value for drawing.

```
public class UnrestrictedMethodScenario : AppCanvasBase
```

Inheritance

object ← AppCanvasBase ← UnrestrictedMethodScenario

Inherited Members

AppCanvasBase.DrawingBitmap , AppCanvasBase.Graphics , AppCanvasBase.CurrentPen ,
AppCanvasBase.Xpos , AppCanvasBase.Ypos , AppCanvasBase.GetBitmap() ,
AppCanvasBase.SetColour(int, int, int) , AppCanvasBase.MoveTo(int, int) , AppCanvasBase.Circle(int) ,
AppCanvasBase.Rect(int, int) , AppCanvasBase.WriteText(string) , object.Equals(object) ,
object.Equals(object, object) , object.GetHashCode() , object.GetType() ,
object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Constructors

UnrestrictedMethodScenario(int, int)

```
public UnrestrictedMethodScenario(int width, int height)
```

Parameters

width int

height int

Methods

ExecuteCommands()

Executes drawing commands specific to each scenario. Must be implemented by subclasses.

```csharp
public override void ExecuteCommands()
```

Class UnrestrictedWhileScenario

Namespace: [BooseTesting](#)

Assembly: BooseTesting.dll

Unrestricted While Scenario: Demonstrates a while loop to draw shapes with decreasing size.

```
public class UnrestrictedWhileScenario : AppCanvasBase
```

Inheritance

[object](#) ← [AppCanvasBase](#) ← UnrestrictedWhileScenario

Inherited Members

[AppCanvasBase.DrawingBitmap](#) , [AppCanvasBase.Graphics](#) , [AppCanvasBase.CurrentPen](#) , [AppCanvasBase.Xpos](#) , [AppCanvasBase.Ypos](#) , [AppCanvasBase.GetBitmap()](#) , [AppCanvasBase.SetColour(int, int, int)](#) , [AppCanvasBase.MoveTo(int, int)](#) , [AppCanvasBase.Circle(int)](#) , [AppCanvasBase.Rect(int, int)](#) , [AppCanvasBase.WriteText(string)](#) , [object.Equals(object)](#) , [object.Equals(object, object)](#) , [object.GetHashCode()](#) , [object.GetType()](#) , [object.MemberwiseClone()](#) , [object.ReferenceEquals(object, object)](#) , [object.ToString()](#)

Constructors

UnrestrictedWhileScenario(int, int)

```
public UnrestrictedWhileScenario(int width, int height)
```

Parameters

`width` [int](#)

`height` [int](#)

Methods

ExecuteCommands()

Executes drawing commands specific to each scenario. Must be implemented by subclasses.

```csharp
public override void ExecuteCommands()
```

Class VariablesManager

Namespace:

Assembly: BooseTesting.dll

Manages variables across the BooseTesting application.

```
public static class VariablesManager
```

Inheritance

object ← VariablesManager

Inherited Members

object.Equals(object) , object.Equals(object, object) , object.GetHashCode() , object.GetType() ,
object.MemberwiseClone() , object.ReferenceEquals(object, object) , object.ToString()

Methods

ClearVariables()

Clears all variables.

```
public static void ClearVariables()
```

GetAllVariables()

Retrieves all variables.

```
public static Dictionary<string, double> GetAllVariables()
```

Returns

Dictionary <string, double >

SetVariable(string, double)

Adds or updates a variable.

```
public static void SetVariable(string name, double value)
```

Parameters

name string⬀

value double⬀

TryGetVariable(string, out double)

Retrieves the value of a variable.

```
public static bool TryGetVariable(string name, out double value)
```

Parameters

name string⬀

value double⬀

Returns

bool⬀

Namespace Shubz_ASE

Classes

[AppCanvas](#)

    Provides a customizable canvas control for drawing, managing, and saving shapes, images, and text in a Windows Forms application. The [AppCanvas](#) class includes functionality for setting pen properties, drawing various shapes, and hosting an HTTP API server to allow remote control of the canvas.

[Form1](#)

    The main form for the BOOSE Drawing Application, providing an interactive interface for drawing shapes, setting colors, adjusting brush thickness, and evaluating mathematical expressions. The [Form1](#) class enables users to work with an [AppCanvas](#) for drawing, and includes text-based input fields for color and mathematical expression evaluation.

Class AppCanvas

Namespace: Shubz_ASE

Assembly: Shubz ASE.dll

Provides a customizable canvas control for drawing, managing, and saving shapes, images, and text in a Windows Forms application. The AppCanvas class includes functionality for setting pen properties, drawing various shapes, and hosting an HTTP API server to allow remote control of the canvas.

```
public class AppCanvas : UserControl, IDropTarget, ISynchronizeInvoke, IWin32Window,
IBindableComponent, IComponent, IDisposable, IContainerControl
```

Inheritance

object ← MarshalByRefObject ← Component ← Control ← ScrollableControl ←
ContainerControl ← UserControl ← AppCanvas

Implements

IDropTarget, ISynchronizeInvoke, IWin32Window, IBindableComponent, IComponent,
IDisposable, IContainerControl

Inherited Members

UserControl.ValidateChildren() , UserControl.ValidateChildren(ValidationConstraints) ,
UserControl.OnCreateControl() , UserControl.OnLoad(EventArgs) ,
UserControl.OnResize(EventArgs) , UserControl.OnMouseDown(MouseEventArgs) ,
UserControl.WndProc(ref Message) , UserControl.AutoSize , UserControl.AutoSizeMode ,
UserControl.AutoValidate , UserControl.BorderStyle , UserControl.CreateParams ,
UserControl.DefaultSize , UserControl.AutoSizeChanged , UserControl.AutoValidateChanged ,
UserControl.Load , ContainerControl.AdjustFormScrollbars(bool) , ContainerControl.Dispose(bool) ,
ContainerControl.OnAutoValidateChanged(EventArgs) ,
ContainerControl.OnFontChanged(EventArgs) , ContainerControl.OnLayout(LayoutEventArgs) ,
ContainerControl.OnMove(EventArgs) , ContainerControl.OnParentChanged(EventArgs) ,
ContainerControl.PerformAutoScale() , ContainerControl.ScaleMinMaxSize(float, float, bool) ,
ContainerControl.ProcessDialogChar(char) , ContainerControl.ProcessDialogKey(Keys) ,
ContainerControl.ProcessCmdKey(ref Message, Keys) , ContainerControl.ProcessMnemonic(char) ,
ContainerControl.ProcessTabKey(bool) , ContainerControl.RescaleConstantsForDpi(int, int) ,
ContainerControl.Select(bool, bool) , ContainerControl.UpdateDefaultButton() ,
ContainerControl.Validate() , ContainerControl.Validate(bool) ,
ContainerControl.AutoScaleDimensions , ContainerControl.AutoScaleFactor ,
ContainerControl.AutoScaleMode , ContainerControl.BindingContext ,
ContainerControl.CanEnableIme , ContainerControl.ActiveControl ,

Control.LogicalToDeviceUnits(int) , Control.LogicalToDeviceUnits(Size) ,
Control.ScaleBitmapLogicalToDevice(ref Bitmap) , Control.NotifyInvalidate(Rectangle) ,
Control.InvokeOnClick(Control, EventArgs) , Control.OnAutoSizeChanged(EventArgs) ,
Control.OnBackColorChanged(EventArgs) , Control.OnBackgroundImageChanged(EventArgs) ,
Control.OnBackgroundImageLayoutChanged(EventArgs) ,
Control.OnBindingContextChanged(EventArgs) , Control.OnCausesValidationChanged(EventArgs) ,
Control.OnContextMenuStripChanged(EventArgs) , Control.OnCursorChanged(EventArgs) ,
Control.OnDataContextChanged(EventArgs) , Control.OnDockChanged(EventArgs) ,
Control.OnEnabledChanged(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.OnTextChanged(EventArgs) ,
Control.OnClick(EventArgs) , Control.OnClientSizeChanged(EventArgs) ,
Control.OnControlAdded(ControlEventArgs) , Control.OnControlRemoved(ControlEventArgs) ,
Control.OnHandleCreated(EventArgs) , Control.OnLocationChanged(EventArgs) ,
Control.OnHandleDestroyed(EventArgs) , Control.OnDoubleClick(EventArgs) ,
Control.OnDragEnter(DragEventArgs) , Control.OnDragOver(DragEventArgs) ,
Control.OnDragLeave(EventArgs) , Control.OnDragDrop(DragEventArgs) ,
Control.OnGiveFeedback(GiveFeedbackEventArgs) , Control.OnEnter(EventArgs) ,
Control.InvokeGotFocus(Control, EventArgs) , Control.OnGotFocus(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.OnMouseEnter(EventArgs) , Control.OnMouseLeave(EventArgs) ,
Control.OnDpiChangedBeforeParent(EventArgs) , Control.OnDpiChangedAfterParent(EventArgs) ,
Control.OnMouseHover(EventArgs) , Control.OnMouseMove(MouseEventArgs) ,
Control.OnMouseUp(MouseEventArgs) , Control.OnPaint(PaintEventArgs) ,
Control.OnQueryContinueDrag(QueryContinueDragEventArgs) ,
Control.OnRegionChanged(EventArgs) , Control.OnPreviewKeyDown(PreviewKeyDownEventArgs) ,
Control.OnSizeChanged(EventArgs) , Control.OnChangeUICues(UICuesEventArgs) ,
Control.OnStyleChanged(EventArgs) , 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.ProcessKeyPreview(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.SetBoundsCore(int, int, int, int, BoundsSpecified)☑ , Control.SetClientSizeCore(int, int)☑ ,
Control.SizeFromClientSize(Size)☑ , Control.SetStyle(ControlStyles, bool)☑ , Control.SetTopLevel(bool)☑ ,
Control.SetVisibleCore(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.BackColor☑ , Control.BackgroundImage☑ ,
Control.BackgroundImageLayout☑ , Control.Bottom☑ , Control.Bounds☑ , Control.CanFocus☑ ,
Control.CanRaiseEvents☑ , Control.CanSelect☑ , Control.Capture☑ , Control.CausesValidation☑ ,
Control.CheckForIllegalCrossThreadCalls☑ , Control.ClientRectangle☑ , Control.ClientSize☑ ,
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.Location☑ ,
Control.Margin☑ , Control.MaximumSize☑ , Control.MinimumSize☑ , 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.Size , Control.TabIndex , Control.TabStop ,
Control.Tag , Control.Text , Control.Top , Control.TopLevelControl , Control.ShowKeyboardCues ,
Control.ShowFocusCues , Control.UseWaitCursor , Control.Visible , Control.Width ,
Control.PreferredSize , Control.Padding , Control.DefaultImeMode , 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.ToString() ,
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

AppCanvas()

```
public AppCanvas()
```

AppCanvas(int, int)

Initializes a new instance of the [AppCanvas](#) class with specified dimensions.

```
public AppCanvas(int v, int v1)
```

Parameters

v [int](#)

    The width of the canvas in pixels.

v1 [int](#)

    The height of the canvas in pixels.

Properties

BrushThickness

Gets or sets the thickness of the brush.

```
public float BrushThickness { get; set; }
```

Property Value

[float](#)

Exceptions

[ArgumentException](#)

    Thrown when the thickness is set to a value of zero or less.

PenColour

Gets or sets the color of the pen used for drawing.

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

Property Value

[Color](#)⬀

Xpos

Gets or sets the X-coordinate position of the pen on the canvas.

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

Property Value

[int](#)⬀

Ypos

Gets or sets the Y-coordinate position of the pen on the canvas.

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

Property Value

[int](#)⬀

Methods

Circle(int, bool)

Draws a circle at the center of the canvas with a specified radius.

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

Parameters

radius int↗

    The radius of the circle.

filled bool↗

    Indicates whether the circle is filled with color or just an outline.

Clear()

Clears the entire canvas, resetting it to a blank 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 to draw to.

y int↗

    The Y-coordinate to draw to.

GetPixel(int, int)

Retrieves the color of a specific pixel at the given coordinates on the canvas.

```
public Color GetPixel(int x, int y)
```

Parameters

x int↗

    The X-coordinate of the pixel.

y int↗

    The Y-coordinate of the pixel.

Returns

[Color↗](#)

    The color of the pixel at the specified coordinates.

Exceptions

[ArgumentOutOfRangeException↗](#)

    Thrown when the coordinates are outside the canvas bounds.

MoveTo(int, int)

Moves the pen to the specified coordinates on the canvas.

```
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 centered on the canvas with specified dimensions.

```
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 is filled with color or just an outline.

Reset()

Resets the canvas by clearing all drawings and returning pen settings to default values.

```
public void Reset()
```

SaveCanvas(string)

Saves the current state of the canvas to an image file.

```
public void SaveCanvas(string fileName)
```

Parameters

fileName string↗

   The file name or path where the canvas will be saved.

Set(int, int)

Sets the size of the canvas.

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

Parameters

width int

The new width of the canvas in pixels.

height int

The new height of the canvas in pixels.

SetBrushThickness(float)

```
public void SetBrushThickness(float thickness)
```

Parameters

thickness float

SetColour(int, int, int)

Sets the pen color using specified RGB values.

```
public void SetColour(int r, int g, int b)
```

Parameters

r int

The red component (0-255).

g int

The green component (0-255).

b int

The blue component (0-255).

StartApiServer()

Starts an HTTP API server that allows remote control of the canvas. The server listens for requests to perform drawing actions or set pen properties.

```
public void StartApiServer()
```

Tri(int, int)

Draws a triangle centered on the canvas with specified dimensions.

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

Parameters

width int⧉

 The width of the triangle.

height int⧉

 The height of the triangle.

WriteText(string)

Writes a specified text at the center of the canvas.

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

Parameters

text string⧉

 The text to write on the canvas.

getBitmap()

Retrieves the current canvas as a [Bitmap](#)⧉ object.

```
public object getBitmap()
```

Returns

[object](#)⧉

The [Bitmap](#)⧉ representing the current state of the canvas.

Class Form1

Namespace: [Shubz_ASE](#)

Assembly: Shubz ASE.dll

The main form for the BOOSE Drawing Application, providing an interactive interface for drawing shapes, setting colors, adjusting brush thickness, and evaluating mathematical expressions. The [Form1](#) class enables users to work with an [AppCanvas](#) for drawing, and includes text-based input fields for color and mathematical expression evaluation.

```
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.Dispose(bool)](#) , [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)](#) ,

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)

Remarks

This form is divided into the following components:

- AppCanvas for drawing various shapes with adjustable colors and brush thickness.
- RGB color input to set the pen color for drawing.
- Mathematical expression evaluator with support for basic and trigonometric functions.
- Buttons for drawing shapes, clearing, and resetting the canvas.

Constructors

Form1()

Default constructor for Form1. Initializes all primary components, including the canvas, drawing buttons, and mathematical input controls.

```
public Form1()
```

Namespace UnitTesting

Classes

[AppCanvasTests](#)

Contains unit tests for the [AppCanvasUnitTest](#) class to validate various drawing functionalities such as moving, setting colors, and drawing shapes.

Interfaces

[IAppCanvas](#)

[IDrawingCommand](#)

Class AppCanvasTests

Namespace: UnitTesting

Assembly: UnitTesting.dll

Contains unit tests for the AppCanvasUnitTest class to validate various drawing functionalities such as moving, setting colors, and drawing shapes.

```
[TestClass]
public class AppCanvasTests
```

Inheritance

object ← AppCanvasTests

Inherited Members

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

Methods

Circle_ShouldDrawCircleOnCanvas()

Tests that [Circle(int, bool, Color)](Circle(int,_bool,_Color)) correctly draws a filled circle.

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

Clear_ShouldResetCanvas()

Tests that Clear() resets the canvas to a default color.

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

DrawElephant_ShouldNotThrowError()

Tests that the DrawElephant() method does not throw an exception.

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

DrawFish_ShouldNotThrowError()

Tests that the DrawFish() method does not throw an exception.

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

DrawLion_ShouldNotThrowError()

Tests that the DrawLion() method does not throw an exception.

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

DrawStar_ShouldNotThrowError()

Tests that DrawStar(int) with a specified size does not throw an exception.

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

DrawTo_ShouldUpdatePosition()

Tests that DrawTo(int, int) updates the position correctly when drawing.

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

DrawWhale_ShouldNotThrowError()

Tests that the DrawWhale() method does not throw an exception.

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

MoveTo_ShouldUpdatePosition()

Tests that the MoveTo(int, int) method correctly updates the position.

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

Rect_ShouldDrawRectangleOnCanvas()

Tests that Rect(int, int, bool) correctly draws a filled rectangle on the canvas.

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

Reset_ShouldResetPositionAndSettings()

Tests that Reset() resets the canvas position, color, and settings.

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

SetBrushThickness_ShouldChangeThickness()

Tests that SetBrushThickness(float) changes the brush thickness correctly.

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

SetColour_ShouldUpdatePenColour()

Tests that [SetColour(int, int, int)](#) changes the pen color as expected.

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

Setup()

Initializes the test by setting up a new instance of [AppCanvasUnitTest](#).

```
[TestInitialize]
public void Setup()
```

Tri_ShouldDrawTriangleOnCanvas()

Tests that [Tri(int, int)](#) correctly draws a triangle.

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

UndoCommand_ShouldUndoLastDrawingAction()

Tests that [UndoLastCommand()](#) undoes the last drawing action.

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

VerifyPixelColor()

Verifies the color of a pixel after drawing on a cleared canvas.

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

WriteText_ShouldNotThrowError()

Tests that [WriteText(string)](#) does not throw an exception when writing text.

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

Interface IAppCanvas

Namespace: UnitTesting

Assembly: UnitTesting.dll

```
public interface IAppCanvas
```

Properties

Xpos

```
int Xpos { get; }
```

Property Value

[int](int)

Ypos

```
int Ypos { get; }
```

Property Value

[int](int)

Methods

Clear()

```
void Clear()
```

Clear_ShouldResetCanvas()

```
void Clear_ShouldResetCanvas()
```

DrawTo(int, int)

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

Parameters

x int↗

y int↗

DrawTo_ShouldUpdatePosition()

```
void DrawTo_ShouldUpdatePosition()
```

MoveTo(int, int)

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

Parameters

x int↗

y int↗

MoveTo_ShouldUpdatePosition()

```
void MoveTo_ShouldUpdatePosition()
```

Reset()

```
void Reset()
```

Reset_ShouldResetPositionAndSettings()

```
void Reset_ShouldResetPositionAndSettings()
```

SetColour(int, int, int)

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

Parameters

red int↗

green int↗

blue int↗

SetColour_ShouldUpdateColour()

```
void SetColour_ShouldUpdateColour()
```

Setup()

```
void Setup()
```

WriteText(string)

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

Parameters

`text` [string](#)

WriteText_ShouldNotThrowError()

```
void WriteText_ShouldNotThrowError()
```

Interface IDrawingCommand

Namespace: UnitTesting

Assembly: UnitTesting.dll

```
public interface IDrawingCommand
```

Methods

Execute()

```
void Execute()
```

Undo()

```
void Undo()
```