Namespace ASE_SaruAcharya

Classes

AppCanvas

Represents a canvas for drawing shapes, lines, and text.

AppCircle

Represents a command for drawing a circle on a canvas.

AppCommandFactory

A custom implementation of the BOOSE.CommandFactory class that creates commands specific to the application.

AppRectangle

Represents a command for drawing a rectangle on a canvas.

AppTriangle

Represents a command to draw a triangle on a canvas.

AppWrite

Represents a command for writing text on a canvas.

Form1

Represents the main form of the application, which serves as the user interface for drawing and running commands on a canvas.

Class AppCanvas

Namespace: ASE_SaruAcharya

Assembly: ASE_SaruAcharya.dll

Represents a canvas for drawing shapes, lines, and text.

```
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 current pen color.

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

Property Value

object

Xpos

Gets or sets the current X-coordinate of the pen.

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

Property Value

int

Ypos

Gets or sets the current Y-coordinate of the pen.

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

Property Value

int

Methods

Circle(int, bool)

Draws a circle with the specified radius and fill option.

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

Parameters

`radius` int

The radius of the circle.

`filled` bool

If true, the circle is filled; otherwise, it is outlined.

Exceptions

CanvasException

Thrown if the radius is invalid.

Clear()

Clears the canvas with a gray background.

```
public void Clear()
```

DrawTo(int, int)

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

```
public void DrawTo(int toX, int toY)
```

Parameters

`toX` int

The X-coordinate of the end point.

`toY` int

The Y-coordinate of the end point.

Exceptions

CanvasException

Thrown if the position is out of bounds.

MoveTo(int, int)

Moves the pen to the specified position without drawing.

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

Parameters

x int

The X-coordinate of the position.

y int

The Y-coordinate of the position.

Exceptions

CanvasException

Thrown if the position is out of bounds.

Rect(int, int, bool)

Draws a rectangle with the specified dimensions and fill option.

```
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

If true, the rectangle is filled; otherwise, it is outlined.

Exceptions

CanvasException

Thrown if the dimensions are invalid.

Reset()

Resets the pen position to the origin (0, 0).

```
public void Reset()
```

Set(int, int)

Resizes the canvas to the specified dimensions.

```
public void Set(int xsize, int ysize)
```

Parameters

`xsize` int

The width of the canvas.

`ysize` 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 (0-255).

`green` int

The green component (0-255).

`blue` int

The blue component (0-255).

Exceptions

CanvasException

Thrown if the RGB values are invalid.

Tri(int, int)

Draws a triangle with the specified base 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.

Exceptions

CanvasException

Thrown if the text is null or empty.

getBitmap()

Gets the current bitmap of the canvas.

```
public object getBitmap()
```

Returns

object

The current bitmap.

Class AppCircle

Namespace: ASE_SaruAcharya

Assembly: ASE_SaruAcharya.dll

Represents a command for drawing a circle on a canvas.

```
public class AppCircle : CommandTwoParameters, ICommand
```

Inheritance

object ← Command ← CanvasCommand ← CommandOneParameter ← CommandTwoParameters ← AppCircle

Implements

ICommand

Inherited Members

CommandTwoParameters.param2 , CommandTwoParameters.param2unprocessed , 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

AppCircle()

Initializes a new instance of the AppCircle class.

```
public AppCircle()
```

AppCircle(Canvas, bool)

Initializes a new instance of the AppCircle class with the specified canvas and fill option.

```
public AppCircle(Canvas c, bool filled)
```

Parameters

`c` Canvas

The canvas on which the circle will be drawn.

`filled` bool

Determines whether the circle is filled or outlined.

Methods

CheckParameters(string[])

Validates the parameters for the circle command.

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

Parameters

`parameterList` string[]

The list of parameters to validate.

Exceptions

CommandException

Thrown if the parameter list is null, contains an incorrect number of parameters, or if the filled parameter is invalid.

Execute()

Executes the circle drawing command. Parses the radius from the command parameters and draws the circle on the canvas.

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

Exceptions

CommandException

Thrown if the parameters are invalid or missing.

Class AppCommandFactory

Namespace: ASE_SaruAcharya

Assembly: ASE_SaruAcharya.dll

A custom implementation of the BOOSE.CommandFactory class that creates commands specific to 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()

Initializes a new instance of the AppCommandFactory class.

```
public 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 command to create (e.g., "tri", "write", "circle", "rect").

Returns

ICommand

An instance of the corresponding command if the command type matches a known command; otherwise, the base factory's command creation method is used.

Exceptions

CommandException

Thrown if the base factory fails to create a valid command for an unknown command type.

Class AppRectangle

Namespace: ASE_SaruAcharya

Assembly: ASE_SaruAcharya.dll

Represents a command for drawing a rectangle on a canvas.

```
public class AppRectangle : CommandThreeParameters, ICommand
```

Inheritance

object ← Command ← CanvasCommand ← CommandOneParameter ← CommandTwoParameters ← CommandThreeParameters ← AppRectangle

Implements

ICommand

Inherited Members

CommandThreeParameters.param3 , CommandThreeParameters.param3unprocessed , CommandTwoParameters.param2 , CommandTwoParameters.param2unprocessed , 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

AppRectangle()

Initializes a new instance of the AppRectangle class.

```
public AppRectangle()
```

AppRectangle(Canvas, bool)

Initializes a new instance of the AppRectangle class with the specified canvas and fill option.

```
public AppRectangle(Canvas c, bool filled)
```

Parameters

`c` Canvas

The canvas on which the rectangle will be drawn.

`filled` bool

Determines whether the rectangle is filled or outlined.

Methods

CheckParameters(string[])

Validates the parameters for the rectangle command.

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

Parameters

`parameterList` string[]

The list of parameters to validate.

Exceptions

CommandException

Thrown if the parameter list is null, contains an incorrect number of parameters, or if the filled parameter is invalid.

Execute()

Executes the rectangle drawing command. Parses the width and height from the command parameters and draws the rectangle on the canvas.

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

Exceptions

CommandException

Thrown if the parameters are invalid or missing.

Class AppTriangle

Namespace: ASE_SaruAcharya

Assembly: ASE_SaruAcharya.dll

Represents a command to draw a triangle on a canvas.

```
public class AppTriangle : CommandTwoParameters, ICommand
```

Inheritance

object ← Command ← CanvasCommand ← CommandOneParameter ← CommandTwoParameters ← AppTriangle

Implements

ICommand

Inherited Members

CommandTwoParameters.param2 , CommandTwoParameters.param2unprocessed , 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

AppTriangle()

Initializes a new instance of the AppTriangle class.

```
public AppTriangle()
```

AppTriangle(Canvas, int, int)

Initializes a new instance of the AppTriangle class with the specified canvas, width, and height.

```
public AppTriangle(Canvas canvas, int width, int height)
```

Parameters

`canvas` Canvas

The canvas on which the triangle will be drawn.

`width` int

The width of the triangle.

`height` int

The height of the triangle.

Methods

CheckParameters(string[])

Checks the validity of the parameters provided to the triangle command.

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

Parameters

`parameterList` string[]

An array of parameters to validate.

Exceptions

CommandException

Thrown if the number of parameters is incorrect or if width/height are invalid.

Execute()

Executes the triangle drawing command by parsing parameters and drawing the triangle on the canvas.

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

Exceptions

CommandException

Thrown if the parameters are invalid.

Class AppWrite

Namespace: ASE_SaruAcharya

Assembly: ASE_SaruAcharya.dll

Represents a command for writing text on a canvas.

```
public class AppWrite : CommandOneParameter, ICommand
```

Inheritance

object ← Command ← CanvasCommand ← CommandOneParameter ← AppWrite

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

AppWrite()

Initializes a new instance of the AppWrite class.

```
public AppWrite()
```

AppWrite(Canvas, string)

Initializes a new instance of the AppWrite class with the specified canvas and text.

```
public AppWrite(Canvas c, string text)
```

Parameters

c Canvas

The canvas on which the text will be written.

text string

The text to write on the canvas.

Methods

CheckParameters(string[])

Validates the parameters for the write text command.

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

Parameters

parameterList string[]

The list of parameters to validate.

Exceptions

CommandException

Thrown if the parameter list does not contain exactly one parameter.

Execute()

Executes the write text command. Retrieves the text from the command parameters and writes it on the canvas.

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

Exceptions

CommandException

Thrown if the parameters are invalid.

Class Form1

Namespace: ASE_SaruAcharya

Assembly: ASE_SaruAcharya.dll

Represents the main form of the application, which serves as the user interface for drawing and running commands 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.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. Sets up the canvas, command factory, program storage, and parser.

```
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 ASE_SaruAcharya.Tests

Classes

AppCanvasTests

Unit tests for the AppCanvas class to verify its behavior and ensure robustness.

Class AppCanvasTests

Namespace: ASE_SaruAcharya.Tests

Assembly: UnitTesting.dll

Unit tests for the AppCanvas class to verify its behavior and ensure robustness.

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

Inheritance

object ← AppCanvasTests

Inherited Members

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

Methods

Circle_InvalidRadius_ThrowsException()

Tests the Circle(int, bool) method with an invalid radius and expects an exception.

```
[TestMethod]
[ExpectedException(typeof(CanvasException))]
public void Circle_InvalidRadius_ThrowsException()
```

Circle_ValidRadius_DrawsCircle()

Tests the Circle(int, bool) method with a valid radius.

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

Clear_CanvasCleared_Successfully()

Tests the Clear() method to ensure the canvas is cleared successfully.

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

MoveTo_InvalidPosition_ThrowsException()

Tests the MoveTo(int, int) method with invalid coordinates and expects an exception.

```
[TestMethod]
[ExpectedException(typeof(CanvasException))]
public void MoveTo_InvalidPosition_ThrowsException()
```

MoveTo_ValidPosition_UpdatesPosition()

Tests the MoveTo(int, int) method with valid coordinates.

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

Rect_InvalidDimensions_ThrowsException()

Tests the Rect(int, int, bool) method with invalid dimensions and expects an exception.

```
[TestMethod]
[ExpectedException(typeof(CanvasException))]
public void Rect_InvalidDimensions_ThrowsException()
```

Rect_ValidDimensions_DrawsRectangle()

Tests the Rect(int, int, bool) method with valid dimensions.

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

SetColour_InvalidValues_ThrowsException()

Tests the SetColour(int, int, int) method with invalid color values and expects an exception.

```
[TestMethod]
[ExpectedException(typeof(CanvasException))]
public void SetColour_InvalidValues_ThrowsException()
```

SetColour_ValidValues_UpdatesPenColour()

Tests the SetColour(int, int, int) method with valid color values.

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

Setup()

Initializes resources before each test is executed.

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

Tri_ValidDimensions_DrawsTriangle()

Tests the Tri(int, int) method with valid dimensions.

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

WriteText_NullOrEmptyText_ThrowsException()

Tests the WriteText(string) method with null or empty text input and expects an exception.

```
[TestMethod]
[ExpectedException(typeof(CanvasException))]
```

```
public void WriteText_NullOrEmptyText_ThrowsException()
```

WriteText_ValidText_DrawsText()

Tests the WriteText(string) method with valid text input.

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