

# Namespace BOOSEapp

## Classes

### [AppCanvas](#)

Drawing Class that handles all drawing on the BOOSE interpreter according to BOOSE commands.  
AppCanvas implements methods from ICanvas interface.

### [AppCircle](#)

Class to draw a circle round the current cursor position.

### [AppCommandFactory](#)

Designed to make command objects from the string passed into it from ProgramTextBox.  
AppCommandFactory extends the BOOSE command factory and replaces BOOSE objects like 'Circle'  
with new BOOSEapp commands to get around BOOSE restrictions.

### [AppMoveTo](#)

Sets the cursor position for subsequent drawing operations, to the provide x,y position.

### [AppRect](#)

Draws a rectangle of provided width and height with the current cursor position being the top left corner.

### [BOOSEInterpreterGUI](#)

Graphical User Interface for interpreting BOOSE commands

# Class AppCanvas

Namespace: [BOOSEapp](#)

Assembly: BOOSEapp.dll

Drawing Class that handles all drawing on the BOOSE interpreter according to BOOSE commands.  
AppCanvas implements methods from ICanvas interface.

```
public class AppCanvas : ICanvas
```

Inheritance

[object](#) ← AppCanvas

Implements

ICanvas

Inherited Members

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

## Constructors

### AppCanvas(int, int)

Constructor for canvas class which creates Bitmap used to implement drawing methods based on size of the drawing area.

```
public AppCanvas(int xsize, int ysize)
```

Parameters

**xsize** [int](#)

Length of x-axis of bitmap.

**ysize** [int](#)

Length of y-axis of bitmap.

# Properties

## PenColour

Get/Set the pencolour for the next drawing operation using a native colour datatype. Casts to relevant type.

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

## Property Value

[object](#)

## Xpos

X position of next drawing program.

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

## Property Value

[int](#)

## Ypos

Y position of next drawing program.

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

## Property Value

[int](#)

# Methods

## Circle(int, bool)

Draws a circle of a set size based on the size of its radius at the cursors positon from the centre of the circle.

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

## Parameters

**radius** [int](#)

Radius of circle.

**filled** [bool](#)

Boolean value on if the shapes centre should be filled in using the brushes current colour - If True circle is drawn filled, If False outline of circle is drawn instead.

## Clear()

Fills in the bitmaps background in the default colour (DarkGrey).

```
public void Clear()
```

## DrawTo(int, int)

Draws a line using the current pen from the cursors current X and Y position to a new provided X and Y position.

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

## Parameters

**x** [int](#)

Specified X position.

**y** [int](#)

Specified Y position.

## MoveTo(int, int)

Moves the cursor to the X and Y position of the next drawing operation.

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

### Parameters

x [int ↗](#)

X position of cursor.

y [int ↗](#)

Y position of cursor.

## Rect(int, int, bool)

Draws a rectangle at the position of the cursor from the top left.

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

### Parameters

width [int ↗](#)

Width of rectangle.

height [int ↗](#)

Height of rectangle.

filled [bool ↗](#)

Boolean value on if the shapes centre should be filled in using the brushes current colour - If True rectangle is drawn filled, If False outline of rectangle is drawn instead.

## Reset()

Resets cursor position to default position (0,0) and resets Pen and Brush colour values to black.

```
public void Reset()
```

## Set(int, int)

Sets the output display size of the bitmap.

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

### Parameters

**width** [int](#)

Width of bitmap display.

**height** [int](#)

Height of bitmap display.

## SetColour(int, int, int)

Sets pen colour based on rgb values.

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

### Parameters

**red** [int](#)

32-bit denary representation of red rgb value.

**green** [int](#)

32-bit denary representation of green rgb value.

**blue** [int](#)

32-bit denary representation of blue rgb value.

## Tri(int, int)

Draws a triangle based on the x and y position of 3 points in a bounding rectangle.

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

Parameters

**width** [int](#)

Width of the bounding rectangle.

**height** [int](#)

Height of the bounding rectangle.

## WriteText(string)

Draws text on the output window at the cursors current position. Primarily used to display caught exception errors.

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

Parameters

**text** [string](#)

String to output.

## getBitmap()

Gets the drawing object of whatever native type. Returns an object so it can be cast to a native type. Use this to get native drawing type so that it can be displayed/output.

```
public object getBitmap()
```

Returns

[object](#)

CanvasBitmap

# Class AppCircle

Namespace: [BOOSEapp](#)

Assembly: BOOSEapp.dll

Class to draw a cirrcle round thecurrent cursor position.

```
public class AppCircle : CommandOneParameter, ICommand
```

## Inheritance

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

## Implements

ICommand

## Inherited Members

CommandOneParameter.param1 , CommandOneParameter.param1unprocessed ,  
CanvasCommand.yPos , CanvasCommand.xPos , CanvasCommand.canvas , CanvasCommand.Canvas ,  
Command.IsDouble , 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()

Blank constructor for factory instantiation

```
public AppCircle()
```

## AppCircle(Canvas, int)

Draws circle from cursors current position to x,y (the cursor is left at x,y

```
public AppCircle(Canvas c, int radius)
```

## Parameters

c Canvas

Canvas object

radius int ↴

Radius of circle

## Methods

### CheckParameters(string[])

Overrideen generic one parameter message to say radius not p1.

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

## Parameters

parameterList string ↴[]

List of parameters in command (e.g. Circle, 50)

## Exceptions

CommandException

Exception is thrown when command has more then 1 parameter as circle only takes one parameter

## Execute()

Executes the command

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

## Exceptions

### CanvasException

Exception thrown if radius is less than 1 or a double type instead of int

# Class AppCommandFactory

Namespace: [BOOSEapp](#)

Assembly: BOOSEapp.dll

Designed to make command objects from the string passed into it from ProgramTextBox.  
AppCommandFactory extends the BOOSE command factory and replaces BOOSE objects like 'Circle' with new BOOSEapp commands to get around BOOSE restrictions.

```
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\(\)](#)

## Methods

### MakeCommand(string)

Overrides original MakeCommand method and creates BOOSEapp objects to bypass the original BOOSE command restrictions. Makes a 'Command' Object from the string passed from ProgramTextBox - if the passed string does not match any of the updated commands - runs the base method of MakeCommand with the restrictions instead.

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

## Parameters

commandType [string](#)

String holding command to be created. Case is unimportant and trimmed

Returns

ICommand

ICommand object if successful

Exceptions

FactoryException

Throw if no such command

# Class AppMoveTo

Namespace: [BOOSEapp](#)

Assembly: BOOSEapp.dll

Sets the cursor position for subsequent drawing operations, to the provide x,y position.

```
public class AppMoveTo : CommandTwoParameters, ICommand
```

## Inheritance

[object](#) ← Command ← CanvasCommand ← CommandOneParameter ← CommandTwoParameters ← AppMoveTo

## Implements

ICommand

## Inherited Members

CommandTwoParameters.param2 , CommandTwoParameters.param2unprocessed ,  
CommandOneParameter.param1 , CommandOneParameter.param1unprocessed ,  
CanvasCommand.yPos , CanvasCommand.xPos , CanvasCommand.canvas , CanvasCommand.Canvas ,  
Command.IsDouble , 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

### AppMoveTo()

Blank constructor for factory instantiation.

```
public AppMoveTo()
```

## Methods

## CheckParameters(string[])

Overridden generic two parameters message to say X and Y not p1 and p2.

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

### Parameters

parameterList [string](#)[]

List of parameters in command (e.g. moveto 100,100)

### Exceptions

CommandException

Exception is thrown when command has more than 2 parameters as moveto only takes two parameters

## Execute()

Executes the MoveTo command, if successful the drawing cursor will be moved to the passed x,y position.

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

### Exceptions

CanvasException

Exception thrown if parameters are negative or double.

# Class AppRect

Namespace: [BOOSEapp](#)

Assembly: BOOSEapp.dll

Draws a rectangle of provided width and height with the current cursor position being the top left corner.

```
public class AppRect : CommandTwoParameters, ICommand
```

## Inheritance

[object](#) ← Command ← CanvasCommand ← CommandOneParameter ← CommandTwoParameters ← AppRect

## Implements

ICommand

## Inherited Members

CommandTwoParameters.param2 , CommandTwoParameters.param2unprocessed ,  
CommandOneParameter.param1 , CommandOneParameter.param1unprocessed ,  
CanvasCommand.yPos , CanvasCommand.xPos , CanvasCommand.canvas , CanvasCommand.Canvas ,  
Command.IsDouble , 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

## AppRect()

Blank constructor of factory instantiation.

```
public AppRect()
```

## AppRect(Canvas, int, int)

Draws a rectangle starting in the top left from current cursor position.

```
public AppRect(Canvas c, int width, int height)
```

### Parameters

c Canvas

Cnavas object

width int ↴

Width of rectangle

height int ↴

Height of rectangle

## Methods

### CheckParameters(string[])

Overridden generic two parameters message to say width and height not p1 and p2.

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

### Parameters

parameterList string ↴[]

List of parameters in command (e.g. rect 100,200)

### Exceptions

CommandException

Exception is thrown when command has more than 2 parameters as rect only takes two parameters

## Execute()

Executes the Rect command - if successful a rectangle will be drawn on the canvas.

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

## Exceptions

CanvasException

Exception thrown if parameters are negative or double.

# Class BOOSEInterpreterGUI

Namespace: [BOOSEapp](#)

Assembly: BOOSEapp.dll

Graphical User Interface for interpreting BOOSE commands

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

## Inheritance

```
object ↳ ← MarshalByRefObject ↳ ← Component ↳ ← Control ↳ ← ScrollableControl ↳ ←  
ContainerControl ↳ ← Form ↳ ← BOOSEInterpreterGUI
```

## 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.ScaleMinAxisSize\(float, float, bool\)](#) ,  
[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.PerformLayout\(\)](#) , [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.OnInvalidate\(EventArgs\)](#) , [Control.OnKeyDown\(KeyEventEventArgs\)](#) ,  
[Control.OnKeyPress\(KeyEventEventArgs\)](#) , [Control.OnKeyUp\(KeyEventEventArgs\)](#) ,  
[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

### BOOSEInterpreterGUI()

GUI constructor which initialises all elements of the program: It creates the bitmap canvas to be drawn on, A Command Factory to create the corresponding objects of all valid commands, a stored program to store a program of Command objects and a parser to split given commands into command types and parameters.

```
public BOOSEInterpreterGUI()
```

## 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.