



*I Wish I Knew How To ...*

*Program the Canvas Control  
with Xojo Desktop 3*

*Jan 2021 Edition (3.0)*

Written By Eugene Dakin

## Table of Contents

|                                              |    |
|----------------------------------------------|----|
| Chapter 1 - Introduction to the Canvas ..... | 12 |
| What a Canvas IS and IS NOT .....            | 13 |
| Raster vs Vector Drawing .....               | 14 |
| Canvas Fundamentals .....                    | 15 |
| Drawing a Line .....                         | 15 |
| Constructors and Destructors .....           | 16 |
| Buffers Explained.....                       | 17 |
| Single Buffer .....                          | 18 |
| Double-Buffer .....                          | 18 |
| Fundamental Shapes .....                     | 19 |
| Fill Shapes with Colour .....                | 23 |
| Colouring in Hexadecimal .....               | 26 |
| Pens .....                                   | 29 |
| Pixel .....                                  | 31 |
| Convert Colour Values.....                   | 34 |
| Graphics Object .....                        | 38 |
| Performance.....                             | 39 |
| Chapter 2 - Files .....                      | 41 |
| Loading a Graphics Picture.....              | 41 |
| Drag and Drop Picture File .....             | 44 |
| Open Xojo Picture .....                      | 47 |
| Saving A Picture Backdrop.....               | 50 |
| Saving File Formats.....                     | 53 |
| Saving Canvas Graphics Layer .....           | 54 |
| Setup Picture Graphics (Double Buffer) ..... | 54 |
| Saving Picture and Graphics Layer .....      | 58 |
| Draw Fast and Save Canvas Picture .....      | 61 |
| Chapter 3 – Text.....                        | 64 |
| Draw String.....                             | 64 |
| Rotating Text .....                          | 67 |
| Wrap Text .....                              | 71 |
| Print Text .....                             | 74 |
| Scroll Bar.....                              | 77 |
| Zooming Text (Picture) .....                 | 83 |
| Horizontal Centred Text .....                | 87 |

|                                                            |     |
|------------------------------------------------------------|-----|
| Zooming Text (Font Size) .....                             | 90  |
| Chapter 4 – Use the Mouse .....                            | 95  |
| Freehand Drawing with the Mouse .....                      | 96  |
| Move Picture with Mouse .....                              | 100 |
| Mouse Zoom .....                                           | 104 |
| Chapter 5 – Canvas Class .....                             | 107 |
| Begin a Canvas Class.....                                  | 107 |
| Things that Happen when using a Class.....                 | 111 |
| Making an Event.....                                       | 112 |
| Chapter 6 – Charts .....                                   | 113 |
| Line Chart (Introduction).....                             | 113 |
| Line Chart (Intermediate).....                             | 119 |
| Line Chart (Customize) .....                               | 126 |
| Bar Chart.....                                             | 132 |
| Pie Chart .....                                            | 140 |
| Chapter 7 – Objects.....                                   | 146 |
| First Object Program .....                                 | 147 |
| Getting Ready to use the Object .....                      | 158 |
| Mouse Moving the Objects .....                             | 160 |
| Make a Simple Game .....                                   | 169 |
| Multiple Object Types (Object 2D).....                     | 185 |
| Chapter 8 – Graphics.....                                  | 194 |
| Picture Resizing .....                                     | 194 |
| Crop a Picture (Pixel) .....                               | 197 |
| Rotate a Picture.....                                      | 202 |
| Flipping Horizontal and/or Vertical-Pixel Slower .....     | 207 |
| Flipping Horizontal and/or Vertical- Row/Column Fast ..... | 216 |
| Blur (RGBSurface – IMPORTANT) .....                        | 224 |
| Sharpening Filter .....                                    | 235 |
| Emboss Filter .....                                        | 244 |
| Edge Detection .....                                       | 250 |
| Picture with Mirror Image (Vertical) .....                 | 255 |
| Posterization .....                                        | 260 |
| Pixelation .....                                           | 265 |
| Grey Scale .....                                           | 270 |
| Merging Two Images .....                                   | 275 |
| Gaussian Blur .....                                        | 281 |
| Gaussian Sharpening .....                                  | 286 |
| Crop a Picture (DrawPicture) .....                         | 293 |

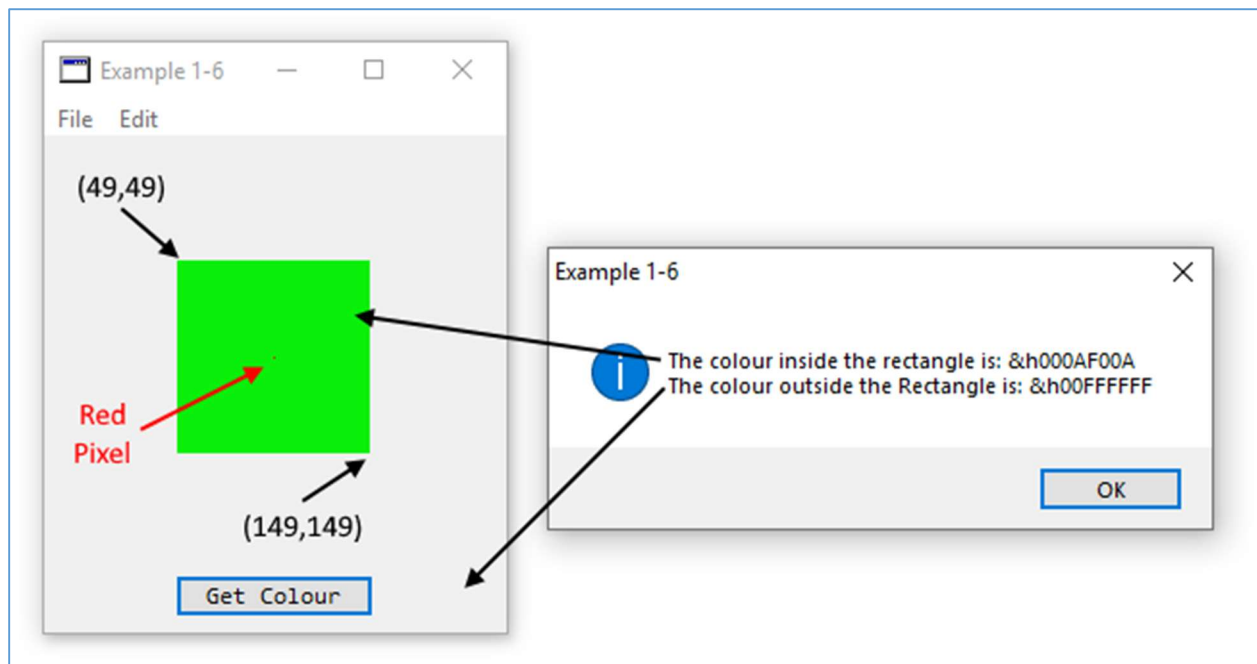
|                                             |     |
|---------------------------------------------|-----|
| Chapter 9 - Masks and Alpha Channel.....    | 298 |
| Implementing ApplyMask .....                | 299 |
| GreyScale Mask .....                        | 304 |
| Timer Fade.....                             | 310 |
| Alpha Channel .....                         | 318 |
| Chapter 10 – Building Controls .....        | 324 |
| Canvas Button – Simple.....                 | 324 |
| Canvas Button – Detail .....                | 330 |
| Canvas Switch.....                          | 340 |
| Chapter 11 – Animation .....                | 345 |
| Creating Static Template .....              | 346 |
| Moving the Picture with Keyboard Keys ..... | 350 |
| Boundary Bounce .....                       | 354 |
| Sprite Sheet Animation .....                | 359 |
| Creating/Destroying an Object .....         | 364 |
| Open Event .....                            | 366 |
| Create Objects with PushButton .....        | 369 |
| Delete Object with PushButton.....          | 370 |
| Drag and Drop Deletion.....                 | 371 |
| Chapter 12 – Games.....                     | 378 |
| Puzzle.....                                 | 378 |
| Open Event .....                            | 380 |
| Pressing Pushbutton.....                    | 385 |
| MouseDown .....                             | 387 |
| SpaceShip Shooter.....                      | 393 |
| Trigonometry .....                          | 393 |
| Ship Setup Basics (12-2a).....              | 395 |
| Ship Movement Basics.....                   | 400 |
| Setting Up Frameworks (12-2b) .....         | 403 |
| Event Handling (12-2c) .....                | 408 |
| Adding a Ship (12-2d) .....                 | 413 |
| Ship Shooting (12-2e) .....                 | 422 |
| Building Asteroids (12-2f) .....            | 428 |
| Collision (12-2g).....                      | 433 |
| Sound (12-2h) .....                         | 441 |
| Chapter 13 – HiDPI and Retina .....         | 444 |
| HiDPI/Retina Settings .....                 | 446 |
| Show Pictures .....                         | 449 |
| Chapter 14 – GraphicsPath .....             | 452 |

|                                              |     |
|----------------------------------------------|-----|
| AddLineToPoint .....                         | 454 |
| AddCurveToPoint .....                        | 456 |
| AddArc .....                                 | 458 |
| AddQuadraticCurveToPoint .....               | 460 |
| AddRectangle .....                           | 462 |
| AddRoundRectangle.....                       | 464 |
| Appendix A – Common Settings and Terms ..... | 466 |
| Index.....                                   | 467 |

## Pixel

A *Canvas* is made of many individual dots or pixels, and when a canvas has a width of 200 by 200, that means that there are 200 pixels in the X direction, and 200 pixels in the Y direction. Because *Canvas* pixels are zero-based, this means that the first of 200 pixels starts at zero (0), and the 200<sup>th</sup> pixel ends at 199. The pixel method in the graphics layer of the canvas can get or set the colour of the pixel.

Figure 9. Example 1-6: Screen Grab



The above screen grab shows the results of the following code.

**Code 22. Example 1-6: Draw Rectangle and Pixel Code**

```

Sub Paint(g As Graphics, areas() As REALbasic.Rect) Handles Paint
    #Pragma Unused areas
    //Draw a green filled rectangle
    g.DrawingColor = RGB(10, 240, 10) //Green
    g.FillRectangle(49.0, 49.0, 100.0, 100.0) //Filled Rectangle

    //Colour the centre pixel red
    Var p as New Picture(1,1,32)
    p.RGBSurface.Pixel(0,0) = RGB(240, 10, 10)
    g.DrawPicture(p, 99.0, 99.0)
End Sub

```

The top layer of the *Canvas* (*DrawingColor*) is set to a green colour with the RGB values of 10 red, 240 green, and 10 blue. A filled rectangle is drawn with the upper-left hand coordinates of 49 and 49, with a width and height of 100 x 100. The filled rectangle is drawn from the pixel set 49,49 to 149,149. To draw the pixel at the very centre of the rectangle a red colour, the pixel set 99,99 is chosen (reminder: a 200x200 rectangle is zero-based and has dimensions from 0-199). The pixel is coloured to red with the RGB value of 240-Red, 10-Green, and 10-Blue.

The next code shows the hexadecimal colour value for a pixel which is inside the rectangle and has a green colour with the value &h000AF00A and is shown in a message box. The colour outside the rectangle is &h00F0F0F0.

**Code 23. Example 1-6: PushButton1 Action Event**

```

Sub Action() Handles Action
    Var p as new Picture(Canvas1.Width, Canvas1.Height, 32)
    Canvas1.DrawInto(p.Graphics, 0, 0)
    //Get the pixel values outside and inside the Rectangle
    MessageBox.Show "The colour inside the rectangle is: " + CStr(p.RGBSurface.Pixel(60,60))
    + chr(13) + _
    "The colour outside the Rectangle is: " + CStr(p.RGBSurface.Pixel(20,20))
End Sub

```

Getting a colour can't be performed in the Paint event, and an action event in a pushbutton has been added to retrieve the different colours.

This example draws a rectangle and pixel and then is able to read the colours and show the hexadecimal values to the user.

## Index

- &, 37
- Abs, 401
- Absolute, 401
- AddArc, 459
- AddCurveToPoint, 456
- AddLineToPoint, 454
- AddQuadraticCurveToPoint, 461
- AllowLiveScrolling, 79
- Alpha Channel, 298, 318
- Animation, 345
  - Create Object, 364
  - Destroy Object, 364
- Animation Sprite Sheet, 359
- Animation Template, 346
- ArcShape, 185
- Backdrop Layer, 13
- Bar Chart, 132
- Begin Canvas Class, 107
- Bezier, 39, 192
- Bicycle Movement, 350
- Blur, 224
- Blur Gaussian, 281
- Blurring, 224
- Bounce Boundary, 354
- Boundary Bounce, 354
- Buffers
  - Double Buffer, 18
  - Explained, 17
  - Single Buffer, 18
  - Triple Buffer, 18
- Building Controls, 324
- ByRef, 241
- Byte, 37
- Canny Edge Detection, 250
- Canvas Button - Detail, 330
- Canvas Button - Simple, 324
- Canvas Class, 107
- Canvas Coordinates, 12
- Canvas Fundamentals, 15
- Canvas Switch, 340
- Centre Text, 87
- Channel
  - Alpha, 318
  - Blue, 318
  - Green, 318
  - Red, 318
- Chart Bar, 132
- Chart Line, 113
- Chart Pie, 140
- Charts, 113
- Class Canvas, 107
- Class Event Create, 112
- Class Events, 111
- Colour Channels, 36
- Colour Conversion, 34
- Colouring in Hexadecimal, 26
- Constructor, 16
- Controls Building, 324
- Convert Colours, 34
- Coordinates, 12
- Cos, 395
- Create Class Event, 112
- Crop Picture, 197, 293
- CurveShape, 185
- Customized Line Chart, 126
- DCI, 444
- Destructor, 16
- Detect Edge, 250
- Digital Camera Initiatives, 444
- Double Buffer, 18, 54
- Drag and Drop Picture File, 44

- Draw String, 64
- Drawing a Line, 15
- Drawing Shapes, 19
- DrawLine, 20
- DrawNotelcon, 22
- DrawOval, 20
- DrawPolygon, 21
- DrawRect, 21
- DrawRoundRect, 21
- DrawStoplcon, 22
- DrawText, 22
- Edge Detection, 250
- Embossed Filter, 244
- Examples
  - 01-01 Draw a Line, 16
  - 01-01a Constructor, 17
  - 01-01b Destructor, 17
  - 01-02 Fundamental Shapes, 19
  - 01-03 Filling Shapes, 23
  - 01-04 Open, 27
  - 01-04 Paint, 27
  - 01-05 Pen Width and Height, 29
  - 01-06 Pixel, 32
  - 01-07 Hex to String Colour Calculate, 37
  - 01-07 RGB to Hex, 35
  - 01-07 RGB to Hex Calculate, 35
  - 01-07 String to Hex Calculate, 36
  - 01-08 Bezier Curve, 39
  - 02-01 Load File Pushbutton, 43
  - 02-02 Drag Drop Picture, 46
  - 02-03 IDE File Load, 49
  - 02-04 Save Event, 52
  - 02-05 Save Picture Graphics, 56
  - 02-06 Picture and Graphics, 60
  - 02-07 Save Canvas Drawing, 63
  - 03-01 Draw a String, 65
  - 03-02 Rotate Text, 68
  - 03-03 Wrap Text, 72
  - 03-04 Print Text, 75
  - 03-05 Draw Text Scroll Bar, 81
  - 03-06 Paint Magnification, 85
  - 03-07 Centre Text Pushbutton, 88
  - 03-08 DrawTheText Method, 92
  - 04-01 Clear Canvas, 97
  - 04-01 MouseDown Drawing, 96
  - 04-01 MouseDrag Drawing, 96
  - 04-02 MouseDown Move, 101
  - 04-02 MouseDrag Move, 101
  - 04-03 MouseWheel Event, 104
  - 05-01 Canvas Class Open, 109
  - 05-01 Canvas Class Paint, 109
  - 06-01 Chart Intro Class Open, 115
  - 06-01 DrawLineChart Open, 116
  - 06-02 DrawLineChart Method, 123
  - 06-03 Draw Chart Action Event, 131
  - 06-04 DrawBarChart Method, 135
  - 06-05 DrawPieChart Method, 143
  - 07-01 DrawObjects, 153
  - 07-02 BringToFront Method, 164
  - 07-02 CustomCanvas MouseDown Method, 165
  - 07-03 LoadNineCards Method, 173
  - 07-03 MouseUp Snap Method, 182
  - 07-04 PixMapShape Method, 192
  - 07-04 ShowArcShape Method, 191
  - 07-04 ShowCurveShape Method, 192
  - 07-04 ShowFigureShape Method, 191
  - 07-04 ShowOvalShape Method, 190
  - 07-04 ShowRectShape Method, 189
  - 07-04 ShowRoundRectShape Method, 190
  - 07-04 ShowString Method, 193
  - 08-01 DrawPicture Resize, 196
  - 08-02 Picture Crop Pixel Method, 201
  - 08-03 Rotate Picture Method, 206
  - 08-04 FlipHor Pixel Method, 211
  - 08-04 FlipHorVert Pixel Method, 215
  - 08-04 FlipVert Pixel Method, 213
  - 08-05 FlipHor Width Method, 218
  - 08-05 FlipHorVert Row/Column Method, 223
  - 08-05 FlipVert Height Method, 221
  - 08-06 Resize Blur Method, 230
  - 08-06 RGBSurface Blur Method, 232

08-07 PixelSharpen Method, 241  
08-07 Sharpen Method, 239  
08-08 Emboss Method, 247  
08-09 Edge Detect Method, 253  
08-10 VMirror Method, 258  
08-11 Posterize Method, 262  
08-12 Pixelated Method, 268  
08-13 GreyScale Method, 273  
08-14 MergeTwo Method, 279  
08-15 Gaussian Blur Fast Method, 284  
08-16 Gaussian Sharpen Fast Method,  
291  
08-17 Picture Crop DrawPicture Method,  
297  
09-01 AddMask Method, 303  
09-02 DrawGreyScaleMask Method, 308  
09-03 Timer1 Action Fade, 315  
09-04 DrawAlphaSquares Method, 322  
10-01 CustomCanvas Simple Button, 326  
10-02 CustomCanvas DrawABackground,  
335  
10-02 CustomCanvas DrawButtonWords,  
335  
10-03 CanvasSwitch DrawSwitch Method,  
343  
11-01 CanvasAnimation Template Open  
Event, 348  
11-02 CanvasAnimation KeyDown Event,  
350  
11-03 CanvasAnimation StartBouncing  
Event, 357  
11-04 CanvasAnimation LoadSprites  
Method, 361  
11-04 CanvasAnimation ShowSprites  
Method, 363  
11-05 AddObject, 369  
11-05 PushButton Remove Object, 370  
12-01 Puzzle SaveImageAs9Pieces, 382  
12-01 Puzzle ShufflePieces, 385  
12-01 ReversePieceWithBlank, 390  
12-02a DrawShip Method, 397  
12-02a Ship KeyDown Method, 400  
12-2b FrameWork Thread, 405  
12-2c KeyDown Event, 410  
12-2d DrawShip Method, 416  
12-2e DrawShots Method, 425  
12-2f DrawAsteroids Method, 430  
12-2g CheckCollisions Method, 435  
12-2h Add Sound Effects, 442  
13-01 Check HiDPI Settings, 447  
13-02 Show HiDPI Pictures, 451  
14-01 GraphicsPath AddArc, 459  
14-01 GraphicsPath AddLineToPoint, 454  
14-02 GraphicsPath AddCurveToPoint,  
456  
14-04 GraphicsPath  
AddQuadraticCurveToPoint, 461  
14-05 GraphicsPath AddRectangle, 463  
14-06 GraphicsPath AddRoundRectangle,  
465  
Fast Flip Picture, 216  
FHD, 444  
FigureShape, 185  
Files, 41  
Filling Shapes, 23  
FillOval, 24  
FillPolygon, 25  
FillRectangle, 25  
FillRoundRectangle, 25  
Flip Picture Fast, 216  
Flip Picture Pixels, 207  
Font Size Zoom, 90  
fps, 355  
FPS, 40, 394  
frames per second, 355  
Frames Per Second, 40, 394  
Freehand Mouse, 96  
Full High Definition, 444  
Games, 378  
Gaussian Blur, 281  
Gaussian Sharpening, 286  
Graphics, 194

- Graphics Layer, 13
- Graphics Object, 38
- GreyScale, 270
- GreyScale Mask, 304
- HD, 444
- HiDPI, 444
- High Definition, 444
- High Dots Per Inch, 444
- Image Merge, 275
- Implement ApplyMask, 299
- Infinite Loop, 410
- Inspector Behavior, 331, 332
- Intermediate Chart, 119
- Introduction, 12
- Invalidate(False), 39
- JPEG
  - QualityDefault, 53
  - QualityHigh, 53
  - QualityLow, 53
  - QualityMax, 53
  - QualityMedium, 53
  - QualityMin, 53
- Key
  - Chr, 347, 350
  - Down Arrow, 350
  - Left Arrow, 350
  - Right Arrow, 350
  - Up Arrow, 350
- Line Chart, 113
- Load Picture, 41
- Loop Infinite, 410
- Low-Pass Filter, 224
- Mask Fade, 310
- Mask Fundamentals, 298
- Mask Generate, 304
- Mask GreyScale, 304
- Mask Timer, 310
- Masks, 298
- Math Trigonometry, 393
- Merge Two Images, 275
- Middle, 36, 37
- Mirror Image Vertical, 255
- Module, 395
- Mouse, 95
  - MouseDown, 95
  - MouseDown, 95
  - MouseEnter, 95
  - MouseExit, 95
  - MouseMove, 95
  - MouseUp, 95
  - MouseWheel, 95
- Mouse Move Picture, 100
- Mouse Zoom, 104
- Move Bicycle, 350
- Moving Objects, 160
- Object Motion, 350
- Object Program, 147
- Object2D, 185
  - ArcShape, 185
  - CurveShape, 185
  - FigureShape, 185
  - OvalShape, 185
  - PixmapShape, 185
  - RectShape, 185
  - RoundRectShape, 185
  - TextShape, 185
- Objects, 146
  - X and O's, 169
- Objects Moving, 160
- Opacity, 310
- Open Xojo Picture, 47
- OvalShape, 185
- Pens, 29
- Performance, 39
- Picture
  - SaveAsGIF, 53
  - SaveAsJPEG, 53
  - SaveAsPNG, 53
  - SaveAsTIFF, 53

Picture Load, 41  
Picture Mouse Movement, 100  
Picture Objects, 146  
Picture Resizing, 194  
Pie Chart, 140  
Pixel, 31  
Pixel Flip Picture, 207  
Pixelation, 265  
PixmapShape, 185  
PNG, 41, 319  
Portable Network Graphics, 319  
Portable Networks Graphics, 41  
Posterization, 260  
Print Text, 74  
Puzzle, 378  
QualityDefault, 53  
QualityHigh, 53  
QualityLow, 53  
QualityMax, 53  
QualityMedium, 53  
QualityMin, 53  
Radian, 67, 394  
Raster, 14  
RectShape, 185  
Refresh, 66  
Refresh(False), 39  
Retina, 444  
RGBSurface, 232  
Rotate Picture, 202  
Rotating Text, 67  
RoundRectShape, 185  
Save a Backdrop Picture, 50  
Save Picture and Graphics, 58  
SaveAsGIF, 53  
SaveAsJPEG, 53  
SaveAsPNG, 53  
SaveAsTIFF, 53  
Saving Canvas Graphics Layer, 54  
Saving Picture Graphics, 54  
Scroll Bar, 77  
ScrollBar  
    AllowLiveScrolling, 79  
    Horizontal, 79  
    Vertical, 79  
Shape Colour, 23  
Shapes  
    DrawLine, 20  
    DrawNotelcon, 22  
    DrawOval, 20  
    DrawPolygon, 21  
    DrawRect, 21  
    DrawRoundRect, 21  
    DrawStopIcon, 22  
    DrawText, 22  
    FillOval, 24  
    FillPolygon, 25  
    FillRectangle, 25  
    FillRoundRectangle, 25  
Sharpening Filter, 235  
Sharpening Gaussian, 286  
Ship Movement Basics, 400  
Ship Setup Basics, 395  
Simple Canvas Button, 324  
Sin, 395  
Single Buffer, 18  
Sound, 441  
SpaceShip  
    Asteroids, 428  
    Event Handling, 408  
    Framework, 403  
    Ship, 413  
    Shooting, 422  
SpaceShip Shooter, 393  
Sprite, 345  
Sprite Sheet, 359  
Static Template, 346  
Switch Canvas, 340

Text, 64  
Text Centre, 87  
Text Rotate, 67  
Text Zoom, 83  
TextShape, 67, 185  
    Bold, 70  
    BorderOpacity, 70  
    FontSize, 70  
    Underline, 70  
Thread, 404  
Timer, 355, 404  
    Period, 355  
    RunMode, 355  
    Timer1, 315  
Timer Fade, 310  
Timer Mask, 310  
Transparency, 318  
Transparent, 298  
Trigonometry, 393  
Triple Buffer, 18  
UHD, 444  
Ultra High Definition, 444  
Use the Mouse, 95  
Vector, 14  
Wrap Text, 71  
Zoom Mouse, 104  
Zooming Text, 83  
Zooming Text (Font Size), 90

The 'I Wish I Knew' series contains technical data and advice that makes sense and contains practical and numerous examples with explanations to allow you to ease into the steep programming curve. You can create custom canvas Desktop applications today!

Version 3 of this Canvas book is a large update to the successful previous Canvas books. Much work has been made to keep the examples similar to previous versions, which assists in converting the many changes in code from API1 to API2. A new chapter has been added with the drawing method called GraphicsPath. Thousands of changes have been made in this book with code changes to updates and explanations. This book "I Wish I Knew How to ... Program Xojo's Canvas Control with Xojo Desktop 3" starts with the fundamentals of the Canvas on the Desktop and builds your knowledge in each chapter. It is assumed that the reader has a good fundamental understanding of programming with Xojo before reading this book, as this starts at an intermediate level. The basics of many topics are discussed to give an overview of fundamental concepts. Each of these topics can then be pieced together to create your own gaming, picture editing, or animation Michaelangelo masterpiece!

The book is written as a guide and reference to Xojo programmers who program Desktop Applications in Windows, Mac, and Linux (Ubuntu). There are no dynamic libraries (dll, dylib, so), COM, or Active X parts to add. This code was tested with Xojo 2020 R2.1.

There are 14 chapters and over 450 pages with over 65 example programs.

Examples include two games, animation, cropping, blurs, edge detection, pixilation and more. Many screenshots have been added to show the results of the code with an index to help find topics quickly.

This is one of many books at Xojo Library. This book can be purchased at <http://xojolibrary.com/> where many great Xojo resources are available.

Happy programming!

Eugene

---

**Eugene Dakin MBA, Ph.D., P.Chem.,** is an author of Xojo and other programming language reference materials and has many years of experience in the programming industry.

ISBN: 978-1-927924-28-0