I Wish I Knew How To …

Program the Canvas Control with Xojo Desktop

July 2014 Edition (1.0)

Written By Eugene Dakin
Contributions by Alain Bailleul

About the Author/Contributor

Eugene Dakin is an author, Professional Chemist, oilfield consultant and programmer in Canada. When he is in-between courses and oilfield projects, he and his wife spend valuable time raising their wonderful children. He has been hobby programming on computer's for over 25 years. He has created over 185 technical and industrial simulation programs.

Eugene can be contacted via:
E-mail: eugenedakin@hotmail.com
Website: http://www.scispec.com

Dedication

Alain Bailleul is a contributor to this book and would like to thank his parents Gilbert and Irene. At a very young age, they gave him his first computer and stimulated him to push his boundaries ever since.
Website: http://alwaysbusycorner.wordpress.com/

Eugene dedicates this book to his patient family. Without their support this would not have been written.

Assumptions

To be able to use the code in this book, you need a version of Xojo. These test programs were created and tested on Windows 8.1 and Maverick. Code for this book was written in my licensed version of Xojo Pro 2014 R2.

Notice

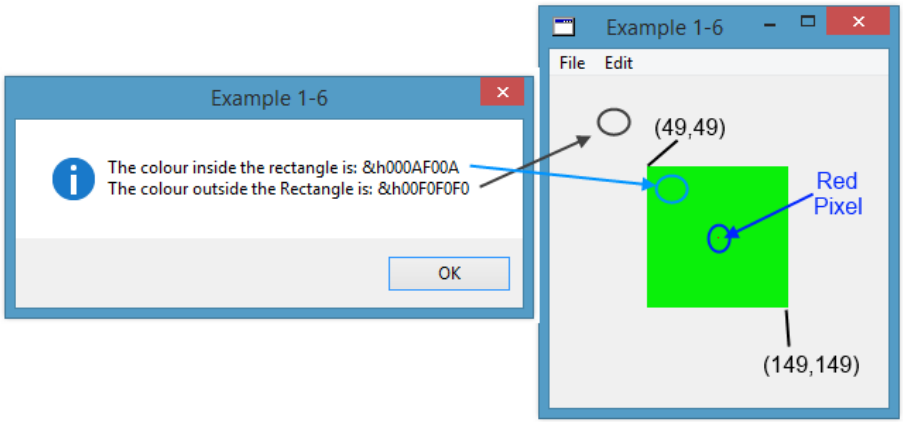
This information is intended for Xojo programmers to learn how to reasonably use the Canvas control. This is not intended to be a complete reference for Xojo or the Canvas control.

Table of Contents

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 200th 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 23. Example 1-6: Pixel Code

```
'Draw a green filled rectangle
g.ForeColor = RGB(10, 240, 10) 'Green
g.FillRect(49, 49, 100, 100) 'Filled Rectangle

'Colour the centre pixel red
g.Pixel (99, 99) = RGB(240, 10, 10)

'Get the pixel values outside and inside the Rectangle
MsgBox "The colour inside the rectangle is: " + CStr(g.Pixel(60,60)) + chr(13) +_
"The colour outside the Rectangle is: " + CStr(g.Pixel(20,20))
```

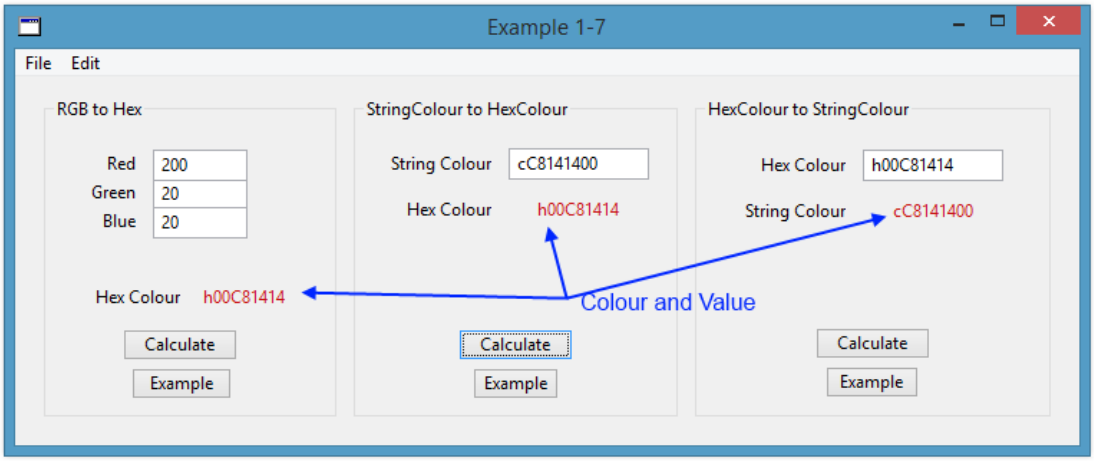
The top layer of the *Canvas* (ForeColor) is set to a green colour with the RGB values of 10 red, 240 green, and 10 blue. A filled rectangle is drawn with the upper-left hand coordinates of 49 and 49, with a width and height of 100 x 100. The filled rectangle is drawn from the pixel set 49,49 to 149,149. 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.

Convert Colour Values

The previous example gets and uses the hexadecimal value of the string. There are three beginning ways to have a colour represented, RGB, String Colour, and Hex Colour. Hexadecimal colour was introduced in the previous example and will be added in this example for thoroughness of the topic.

Figure 10. Example 1-7: Converting Colours



This example has three sections: 1) RGB to Hex, 2) String Colour to Hex Colour, and 3) Hex Colour to String Colour. Programming with the Canvas can create a need to understand this conversion fundamental.

The first example converts the RGB value to hexadecimal colour and changes the text of the Hex Colour label to the RGB colour. In this case the h00C81414 value is the colour red and the text colour is changed to this shade of red. The hexadecimal colour is the standardized way in which textcolors are changed in Xojo. When the RGB value of red, green, and blue are converted to a value, the value is hexadecimal and this value is entered in the textcolor for LblHex.

Index

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!

This book "I Wish I Knew How to … Program Xojo's Canvas Control" 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 link libraries (dll), COM, or Active X parts to add. This code was tested with Xojo 2014 R2 on Windows, OS X, and Ubuntu machines.

There are 12 chapters and over 400 pages with over 60 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 Great White Software. This book can be purchased at http://great-white-software.com/rblibrary/ where many great Xojo resources are available.

Happy programming!

Eugene

Alain Bailleul, is a contributor to this book and is the Senior Software Architect at One-Two. He likes to experiment with new technologies like the Kinect, Computer Vision, and Artificial Intelligence.

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

ISBN: 978-1-927924-06-8