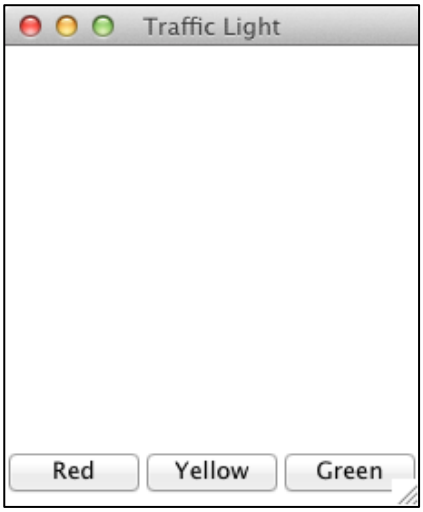
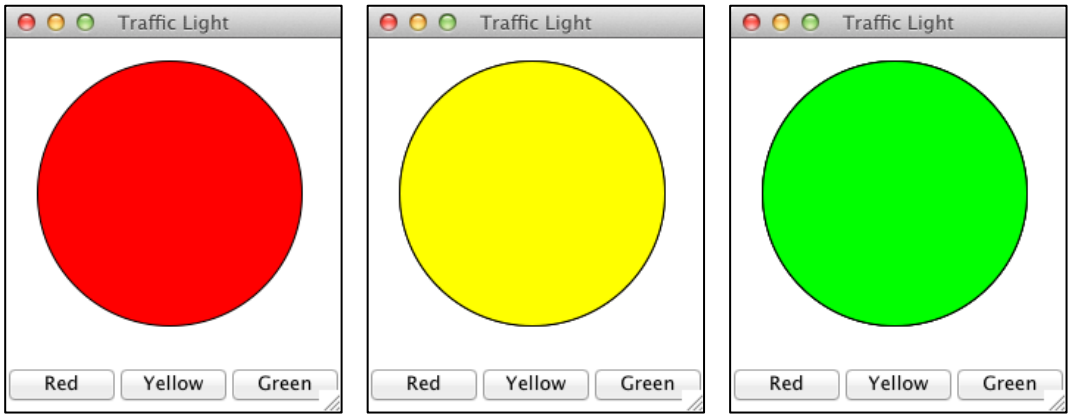
IFB104 GUI Workshop Exercise: "Traffic Light"

The tkinter Application Programming Interface allows us to create windows full of widgets, as an alternative to purely textual user interfaces. In this easy exercise you will use tkinter's `Tk` class to create a window, its `Button` class to create some buttons, and the `Canvas` class to create a drawing surface. You will then link the buttons to the canvas, so that when the buttons are pushed the drawing changes.

When started your program should pop up a window something like the following. (Here the program was executed under Mac OS X. The user interface may look different under other operating systems.)



This window has a drawing canvas, initially empty, and three buttons labelled 'Red', 'Yellow' and 'Green'. Each time one of the buttons is pushed, a circle of the corresponding colour is drawn on the canvas, in imitation of a traffic light.



As an additional feature, you could restrict the buttons so that they only change the drawing in the usual green-yellow-red-green sequence. (There is no yellow between red and green in real traffic lights!).

Apart from this the only other behaviour possible is to minimise, maximise or close the window using the standard operating system buttons (which happen to be red, yellow and green in the case of Mac OS X!).

To do this you will need to:

1. Create the window.
2. Create the `Button` and `Canvas` widgets.
3. Define three functions that change the image drawn on the `Canvas` widget. To draw a circle on the canvas you need to use its `create_oval` function with an appropriate `fill` colour parameter. The first four parameters to this function are the *x* and *y* coordinates of the oval's top-left and bottom-right boundaries.
4. Link the buttons to the corresponding function by making the functions the `command` executed when the buttons are pressed.
5. Place the `Button` and `Label` widgets on the screen. To achieve the neat layout shown on the previous page we used the 'grid' geometry manager, but you may find it easier just to 'pack' them.
6. Start the window's 'event loop'.