IFB104 GUI Workshop Exercise: "Countdown"

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 a button, and the `Label` class to create some text in the window. You will then link the button to the label, so that when the button is pushed the text 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 single button labelled 'Push' and a label whose text is the string representation of the number 10. Each time the button is pushed, the number's value should be decreased by one. For instance, in our solution the window has the following appearance after the button has been pushed six times.

As a final refinement, you can change the label's text to some appropriate message when the countdown reaches zero.

Apart from this the only other behaviour possible is to minimise, maximise or close the window using the standard operating system buttons (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 `Label` widgets.
3. Define a function that changes the `text` parameter of the `Label` widget to decrement the number displayed, and make this function the command called when the `Button` widget is pushed. At first you could just change the label to a fixed value when the button is pushed, but to get the 'count down' effect you will need to use a global Python variable to keep track of the current number.
4. 'Pack' the `Button` and `Label` widgets on the screen.
5. Start the window's 'event loop'.