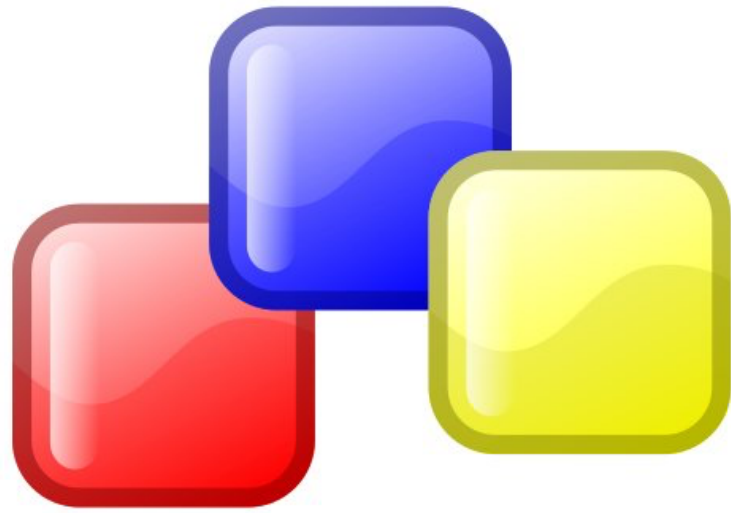


wxWidgets + wxPython

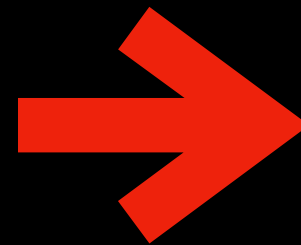
wxWidgets



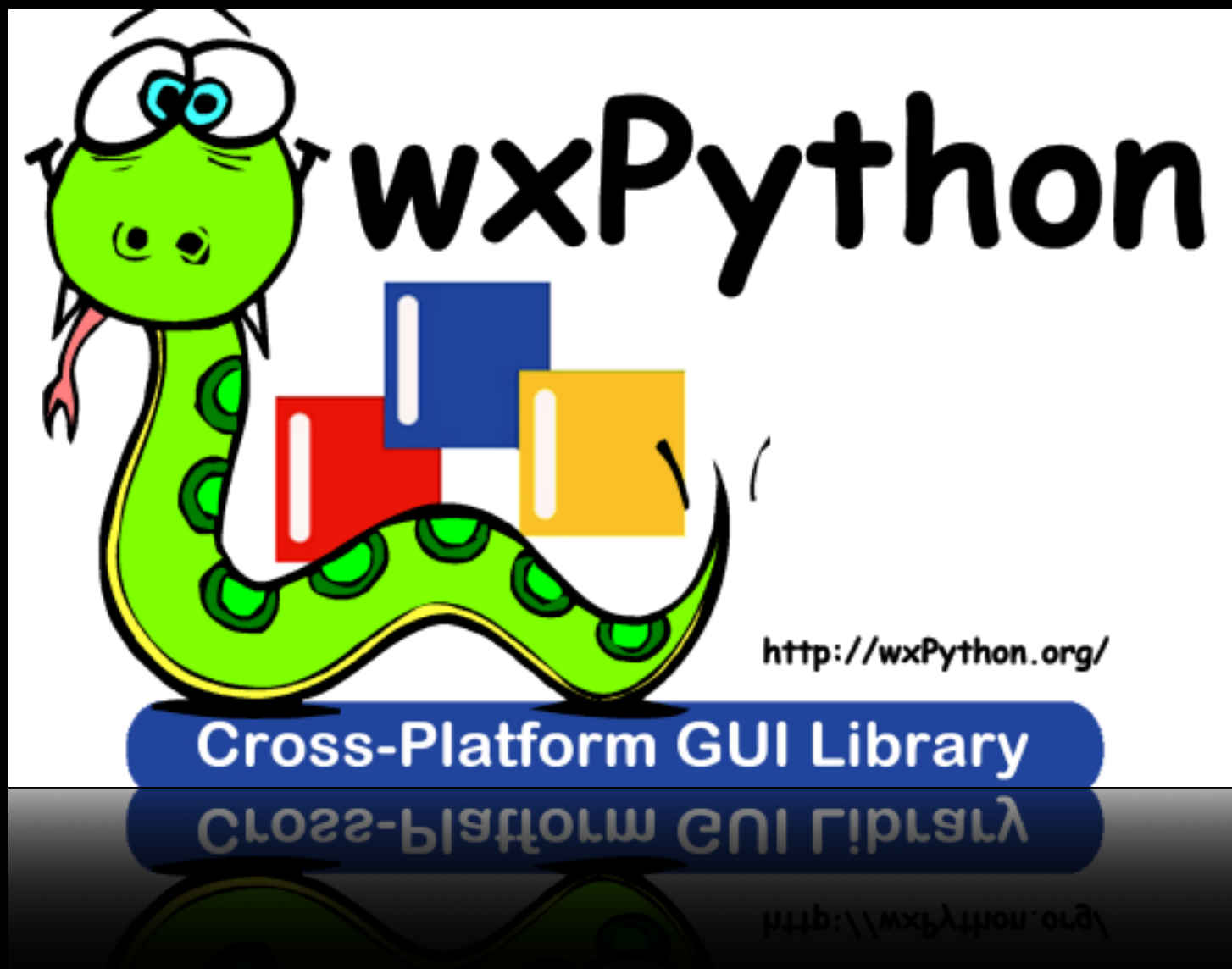
wxWidgets
Cross-Platform GUI Toolkit

<http://www.wxwidgets.org>

Нативный Look&Feel



wxPython



<https://wxpython.org>

Пример приложения на wxPython



Жизненный цикл приложения

```
# создаем приложение  
app = wx.App()
```

```
# создаем форму  
form = CounterForm(None, "wxCounter")
```

```
# показываем форму  
form.Show(True)
```

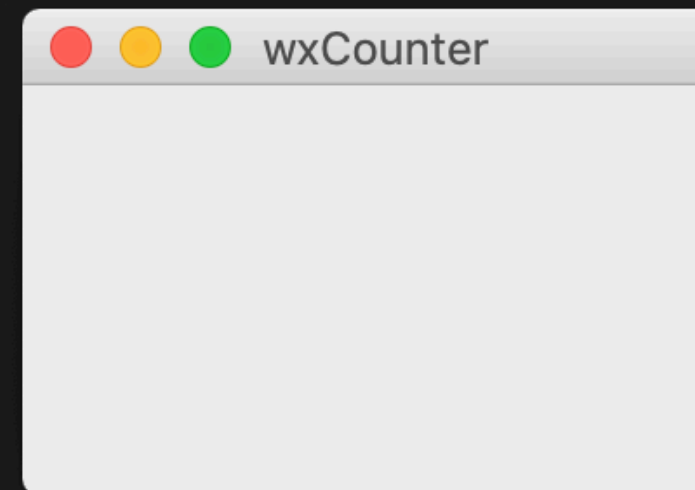
```
# запускаем главный цикл приложения  
app.MainLoop()
```



Забирает управление, и возвращает при закрытии приложения

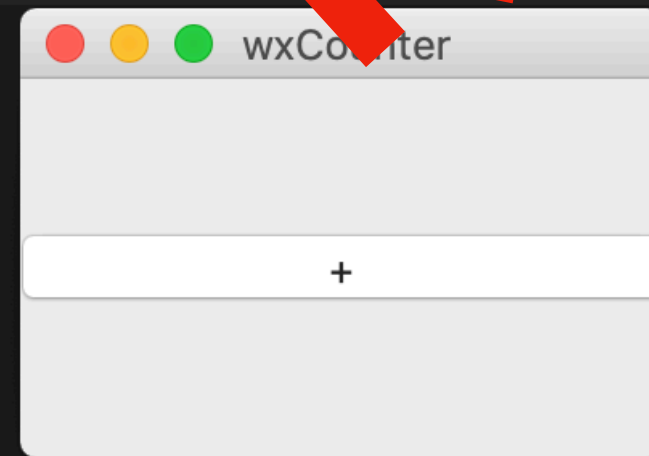
Простейшее окно

```
1  import wx
2
3  class CounterForm(wx.Frame):
4      def __init__(self, parent, title):
5          # создаем фрейм, и настраиваем параметры
6          wx.Frame.__init__(self, parent,
7                             title = title,
8                             size = (200, 140),
9                             style = wx.MINIMIZE_BOX | wx.CAPTION | wx.CLOSE_BOX | wx.MAXIMIZE_BOX
10                             )
11
12
13  # создаем приложение
14  app = wx.App()
15
16  # создаем форму
17  form = CounterForm(None, "wxCounter")
18
19  # показываем форму
20  form.Show(True)
21
22  # запускаем главный цикл приложения
23  app.MainLoop()
```



Создание элементов управления

```
1 import wx
2
3 class CounterForm(wx.Frame):
4     def __init__(self, parent, title):
5         # создаем фрейм, и настраиваем параметры
6         wx.Frame.__init__(self, parent,
7                             title = title,
8                             size = (200, 140),
9                             style = wx.MINIMIZE_BOX | wx.CAPTION | wx.CLOSE_BOX | wx.MAXIMIZE_BOX
10        )
11
12        # создаем элементы управления
13        self.plusButton = wx.Button(self, label = "+")
14
15 # создаем приложение
16 app = wx.App()
17
18 # создаем форму
19 form = CounterForm(None, "wxCounter")
20
21 # показываем форму
22 form.Show(True)
23
24 # запускаем главный цикл приложения
25 app.MainLoop()
```



Абсолютное позиционирование не нужно

```
# создаем главный, горизонтальный сизер
# [   |   |   ]
self.horzSizer = wx.BoxSizer(wx.HORIZONTAL)
self.SetSizer(self.horzSizer)

# создаем элементы управления
self.plusButton = wx.Button(self, label = "+")
self.minusButton = wx.Button(self, label = "-")
self.counterText = wx.TextCtrl(self, value = "XXX", style = wx.TE_CENTER | wx.TE_READONLY)

# записываем элементы управления в сизер
self.horzSizer.AddStretchSpacer(1)
self.horzSizer.Add(self.minusButton, flag = wx.ALIGN_CENTER)
self.horzSizer.Add(self.counterText, flag = wx.ALIGN_CENTER | wx.LEFT | wx.RIGHT, border = 2)
self.horzSizer.Add(self.plusButton, flag = wx.ALIGN_CENTER)
self.horzSizer.AddStretchSpacer(1)
```

<https://jenyay.net/Programming/WxSizer>

Для реагирования на событие надо подписаться на него (Bind)

```
# привязываем события  
self.plusButton.Bind(wx.EVT_BUTTON, self.OnPlusClick)
```



```
# нажатие на кнопку +  
def OnPlusClick(self, event):  
    self.count = self.count + 1
```

DEMO

```
import wx

class CounterForm(wx.Frame):
    def __init__(self, parent, title):
        # создаем фрейм, и настраиваем параметры
        wx.Frame.__init__(self, parent,
            title = title,
            size = (200, 140),
            style = wx.MINIMIZE_BOX | wx.CAPTION | wx.CLOSE_BOX | wx.MAXIMIZE_BOX
        )

        # ui
        # создаем главный, горизонтальный сизер
        # [   |   |   ]
        self.horzSizer = wx.BoxSizer(wx.HORIZONTAL)
        self.SetSizer(self.horzSizer)

        # создаем элементы управления
        self.plusButton = wx.Button(self, label = "+")
        self.minusButton = wx.Button(self, label = "-")
        self.counterText = wx.TextCtrl(self, value = "XXX", style = wx.TE_CENTER | wx.TE_READONLY)

        # настраиваем их параметры
        self.plusButton.SetMinSize((50, self.plusButton.MinWidth))
        self.minusButton.SetMinSize((50, self.minusButton.MinWidth))
        self.counterText.SetMinSize((60, self.counterText.MinWidth))

        # записываем элементы управления в сизер
        self.horzSizer.AddStretchSpacer(1)
        self.horzSizer.Add(self.minusButton, flag = wx.ALIGN_CENTER)
        self.horzSizer.Add(self.counterText, flag = wx.ALIGN_CENTER | wx.LEFT | wx.RIGHT, border = 2)
        self.horzSizer.Add(self.plusButton, flag = wx.ALIGN_CENTER)
        self.horzSizer.AddStretchSpacer(1)

        # привязываем события
        self.Bind(wx.EVT_SIZE, self.OnSize)
        self.plusButton.Bind(wx.EVT_BUTTON, self.OnPlusClick)
        self.minusButton.Bind(wx.EVT_BUTTON, self.OnMinusClick)

        # инициализация
        self.count = 0

        # свойство count
        @property
        def count(self):
            return self._count
        @count.setter
        def count(self, value):
            self._count = max(0, value)
            self.counterText.Value = str(self.count)

        # изменение размеров окна
        def OnSize(self, event):
            self.Size = (max(event.Size[0], 180), max(event.Size[1], 60))
            self.Layout()

        # нажатие на кнопку +
        def OnPlusClick(self, event):
            self.count = self.count + 1

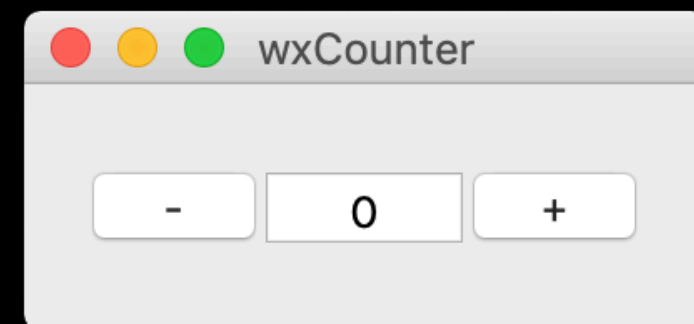
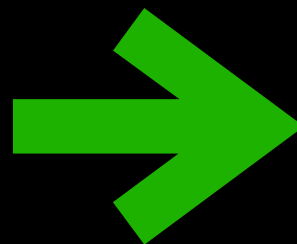
        # нажатие на кнопку -
        def OnMinusClick(self, event):
            self.count = self.count - 1

# создаем приложение
app = wx.App()

# создаем форму
form = CounterForm(None, "wxCounter")

# показываем форму
form.Show(True)

# запускаем главный цикл приложения
app.MainLoop()
```



Загрузить себе: <https://github.com/AnotherStudent/wxPython>