



# PYGTK

**tutorialspoint**

SIMPLY EASY LEARNING

[www.tutorialspoint.com](http://www.tutorialspoint.com)



<https://www.facebook.com/tutorialspointindia>



<https://twitter.com/tutorialspoint>

## About the Tutorial

---

PyGTK is a set of wrappers written in Python and C for GTK + GUI library. It is part of the GNOME project. It offers comprehensive tools for building desktop applications in Python.

This tutorial discusses the basic functionalities of the different widgets found in the toolkit.

## Audience

---

This tutorial has been prepared for beginners to help them understand the basic concepts of PyGTK. Advanced programmers can also draw benefits from this tutorial.

## Prerequisites

---

Before proceeding further with this tutorial, it is recommended that you have a reasonable knowledge of Python programming language.

## Copyright & Disclaimer

---

© Copyright 2016 by Tutorials Point (I) Pvt. Ltd.

All the content and graphics published in this e-book are the property of Tutorials Point (I) Pvt. Ltd. The user of this e-book is prohibited to reuse, retain, copy, distribute or republish any contents or a part of contents of this e-book in any manner without written consent of the publisher.

We strive to update the contents of our website and tutorials as timely and as precisely as possible, however, the contents may contain inaccuracies or errors. Tutorials Point (I) Pvt. Ltd. provides no guarantee regarding the accuracy, timeliness or completeness of our website or its contents including this tutorial. If you discover any errors on our website or in this tutorial, please notify us at [contact@tutorialspoint.com](mailto:contact@tutorialspoint.com).

## Table of Contents

---

About the Tutorial .....	i
Audience.....	i
Prerequisites.....	i
Copyright & Disclaimer .....	i
Table of Contents.....	ii
<b>1. PYGTK – INTRODUCTION .....</b>	<b>1</b>
<b>2. PYGTK – ENVIRONMENT.....</b>	<b>3</b>
<b>3. PYGTK – HELLO WORLD .....</b>	<b>4</b>
<b>4. PYGTK – IMPORTANT CLASSES .....</b>	<b>6</b>
<b>5. PYGTK – WINDOW CLASS .....</b>	<b>8</b>
<b>6. PYGTK – BUTTON CLASS.....</b>	<b>10</b>
<b>7. PYGTK – LABEL CLASS .....</b>	<b>12</b>
<b>8. PYGTK – ENTRY CLASS .....</b>	<b>13</b>
<b>9. PYGTK – SIGNAL HANDLING .....</b>	<b>15</b>
<b>10. PYGTK – EVENT HANDLING .....</b>	<b>18</b>
<b>11. PYGTK – CONTAINERS .....</b>	<b>20</b>
<b>12. PYGTK – BOX CLASS.....</b>	<b>23</b>
<b>13. PYGTK – BUTTONBOX CLASS.....</b>	<b>26</b>
<b>14. PYGTK – ALIGNMENT CLASS .....</b>	<b>28</b>
<b>15. PYGTK – EVENTBOX CLASS .....</b>	<b>30</b>

16.	<b>PYGTK – LAYOUT CLASS</b> .....	<b>32</b>
17.	<b>PYGTK – COMBOBOX CLASS</b> .....	<b>34</b>
18.	<b>PYGTK – TOGGLEBUTTON CLASS</b> .....	<b>39</b>
19.	<b>PYGTK – CHECKBUTTON CLASS</b> .....	<b>42</b>
20.	<b>PYGTK – RADIOBUTTON CLASS</b> .....	<b>44</b>
21.	<b>PYGTK – MENUBAR, MENU &amp; MENUITEM</b> .....	<b>46</b>
22.	<b>PYGTK – TOOLBAR CLASS</b> .....	<b>51</b>
23.	<b>PYGTK – ADJUSTMENT CLASS</b> .....	<b>55</b>
24.	<b>PYGTK – RANGE CLASS</b> .....	<b>56</b>
25.	<b>PYGTK – SCALE CLASS</b> .....	<b>57</b>
26.	<b>PYGTK — SCROLLBAR CLASS</b> .....	<b>58</b>
27.	<b>PYGTK – DIALOG CLASS</b> .....	<b>61</b>
28.	<b>PYGTK — MESSAGEDIALOG CLASS</b> .....	<b>67</b>
29.	<b>PYGTK – ABOUTDIALOG CLASS</b> .....	<b>69</b>
30.	<b>PYGTK – FONT SELECTION DIALOG</b> .....	<b>71</b>
31.	<b>PYGTK – COLOR SELCTION DIALOG</b> .....	<b>73</b>
32.	<b>PYGTK – FILE CHOOSER DIALOG</b> .....	<b>75</b>
33.	<b>PYGTK – NOTEBOOK CLASS</b> .....	<b>77</b>
34.	<b>PYGTK – FRAME CLASS</b> .....	<b>81</b>
35.	<b>PYGTK – ASPECTFRAME CLASS</b> .....	<b>84</b>

36.	PYGTK – TREEVIEW CLASS.....	87
37.	PYGTK – PANED CLASS .....	95
38.	PYGTK – STATUSBAR CLASS.....	98
39.	PYGTK – PROGRESSBAR CLASS.....	101
40.	PYGTK – VIEWPORT CLASS .....	104
41.	PYGTK – SCROLLEDWINDOW CLASS.....	105
42.	PYGTK – ARROW CLASS.....	108
43.	PYGTK – IMAGE CLASS .....	111
44.	PYGTK – DRAWINGAREA CLASS .....	113
45.	PYGTK – SPINBUTTON CLASS.....	116
46.	PYGTK – CALENDAR CLASS.....	120
47.	PYGTK – CLIPBOARD CLASS.....	125
48.	PYGTK – RULER CLASS.....	128
49.	PYGTK – TIMEOUT .....	131
50.	PYGTK — DRAG AND DROP .....	134

# 1. PyGTK – Introduction

PyGTK is a set of wrappers written in Python and C for GTK + GUI library. It is part of the GNOME project. It offers comprehensive tools for building desktop applications in Python. Python bindings for other popular GUI libraries are also available.

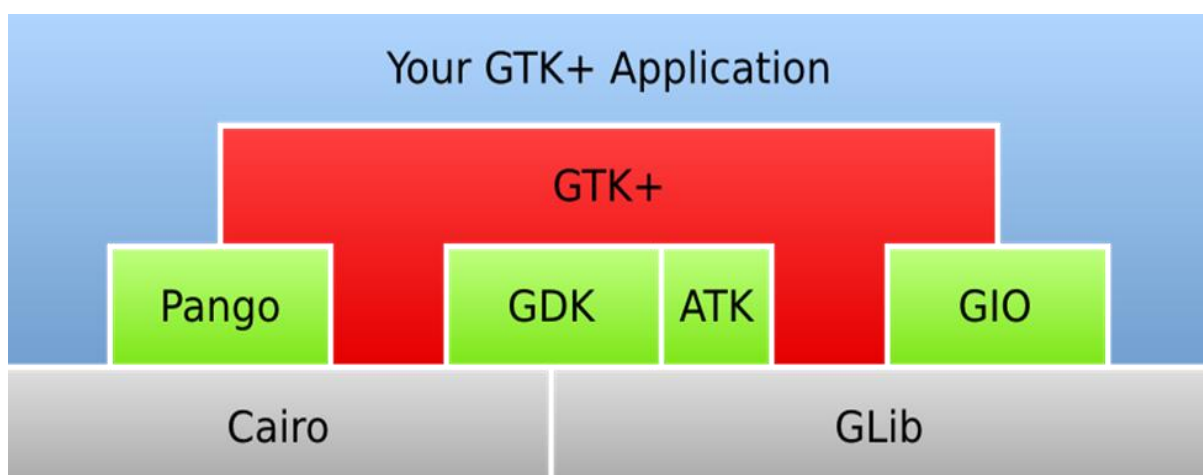
**PyQt** is a Python port of QT library. Our PyQt tutorial can be found [here](#). Similarly, wxPython toolkit is Python binding for wxWidgets, another popular cross-platform GUI library. Our wxPython tutorial is available [here](#).

**GTK+**, or the GIMP Toolkit, is a multi-platform toolkit for creating graphical user interfaces. Offering a complete set of widgets, GTK+ is suitable for projects ranging from small one-off tools to complete application suites.

GTK+ has been designed from the ground up to support a wide range of languages. PyGTK is a Python wrapper for GTK+.

GTK+ is built around the following four libraries:

- **Glib** — A low-level core library that forms the basis of GTK+. It provides data structure handling for C.
- **Pango** — A library for layout and rendering of text with an emphasis on internationalization.
- **Cairo** — A library for 2D graphics with support for multiple output devices (including the X Window System, Win32)
- **ATK** — A library for a set of interfaces providing accessibility tools such as screen readers, magnifiers, and alternative input devices.



PyGTK eases the process and helps you create programs with a graphical user interface using the Python programming language. The underlying GTK+ library provides all kinds of visual elements and utilities for it to develop full-featured applications for the GNOME Desktop. PyGTK is a cross-platform library. It is a free software distributed under the LGPL license.

PyGTK is built around GTK + 2.x. In order to build applications for GTK +3, PyGObject bindings are also available.

## 2. PyGTK – Environment

### PyGTK for Microsoft Windows

The installation of PyGTK for Microsoft Windows involves the following steps:

- **Step 1:** Install a 32-bit Python interpreter (latest Python 2.7 distribution)
- **Step 2:** Download and install GTK+ runtime.
- **Step 3:** Download and install PyGTK from the following URL:  
(<http://ftp.gnome.org/pub/GNOME/binaries/win32/pygtk/>)
- **Step 4:** It is also recommended that you download PyCairo and PyGobject modules from the following URLs:  
(<http://ftp.gnome.org/pub/GNOME/binaries/win32/pycairo/>)  
(<http://ftp.gnome.org/pub/GNOME/binaries/win32/pygobject/>)
- **Step 5:** For convenience, all-in-one installer which handles all of the PyGTK dependencies is also available. Download and install the latest all-in-one installer for Windows from the following URL:  
(<http://ftp.gnome.org/pub/GNOME/binaries/win32/pygtk/2.24/pygtk-2.24.0.win32-py2.7.msi>)

### PyGTK for Linux

PyGTK is included in most Linux distributions (including Debian, Fedora, Ubuntu, RedHat etc); the source code can also be downloaded and compiled from the following URL

<http://ftp.gnome.org/pub/GNOME/sources/pygtk/2.24/>



## 3. PyGTK – Hello World

Creating a window using PyGTK is very simple. To proceed, we first need to import the gtk module in our code.

```
import gtk
```

The gtk module contains the gtk.Window class. Its object constructs a toplevel window. We derive a class from gtk.Window.

```
class PyApp(gtk.Window):
```

Define the constructor and call the **show\_all()** method of the gtk.window class.

```
def __init__(self):  
    super(PyApp, self).__init__()  
    self.show_all()
```

We now have to declare the object of this class and start an event loop by calling its main() method.

```
PyApp()  
gtk.main()
```

It is recommended we add a label "**Hello World**" in the parent window.

```
label = gtk.Label("Hello World")  
self.add(label)
```

The following is a complete code to display "**Hello World**":

```
import gtk  
  
class PyApp(gtk.Window):  
    def __init__(self):  
        super(PyApp, self).__init__()  
        self.set_default_size(300,200)  
        self.set_title("Hello World in PyGTK")
```

```
label = gtk.Label("Hello World")
self.add(label)
self.show_all()
PyApp()
gtk.main()
```

The implementation of the above code will yield the following output:



## 4. PyGTK – Important Classes

The PyGTK module contains various widgets. `gtk.Object` class acts as the base class for most of the widgets as well as for some non-widget classes. The toplevel window for desktop applications using PyGTK is provided by `gtk.Window` class. The following table lists the important widgets and their functions:

<code>gtk.Widget</code>	This is a <code>gtk</code> .base class for all PyGTK widgets. <code>gtk.Widget</code> provides a common set of methods and signals for the widgets.
<code>gtk.Window</code>	This is a toplevel window that holds one child widget. <code>gtk.Window</code> is a display area decorated with a title bar, and items to allow the user to close, resize and move the window.
<code>gtk.Button</code>	This is a pushbutton widget that issues a signal when clicked. <code>gtk.Button</code> is usually displayed as a pushbutton with a text label and is generally used to attach a callback function.
<code>gtk.Entry</code>	This is a single line text entry widget.
<code>gtk.Label</code>	This widget displays a limited amount of read-only text.
<code>gtk.ButtonBox</code>	This is a base class for widgets that contains multiple buttons.
<code>gtk.HBox</code>	This is a container that organizes its child widgets into a single horizontal row.
<code>gtk.VBox</code>	This is a container that organizes its child widgets into a single column.
<code>gtk.Fixed</code>	This is a container that can place child widgets at fixed positions and with fixed sizes, given in pixels.
<code>gtk.Layout</code>	This provides infinite scrollable area containing child widgets and custom drawing.
<code>gtk.MenuItem</code>	This widget implements the appearance and behavior of menu items. The derived widget subclasses of the <code>gtk.MenuItem</code> are the only valid children of menus. When selected by a user, they can display a popup menu or invoke an associated function or method
<code>gtk.Menu</code>	This is a dropdown menu consisting of a list of <code>_MenuItem</code> objects which can be navigated and activated by the user to perform application functions.
<code>gtk.MenuBar</code>	This displays the menu items horizontally in an application window or dialog.
<code>gtk.ComboBox</code>	This widget is used to choose from a list of items.
<code>gtk.Scale</code>	This is a horizontal or vertical slider control to select a numeric value.

gtk.Scrollbar	This displays a horizontal or vertical scrollbar.
gtk.ProgressBar	This is used to display the progress of a long running operation.
gtk.Dialog	This displays a popup window for user information and action.
gtk.Notebook	This widget is a container whose children are overlapping pages that can be switched between using tab labels.
gtk.Paned	This is a base class for widgets with two panes, arranged either horizontally or vertically. Child widgets are added to the panes of the widget. The division between the two children can be adjusted by the user.
gtk.TextView	This widget displays the contents of a TextBuffer object.
gtk.Toolbar	This container holds and manages a set of buttons and widgets in a horizontal or vertical bar.
gtk.TreeView	This widget displays the contents of standard TreeModel (ListStore, TreeStore, TreeModelSort)
gtk.DrawingArea	This widget helps in creating custom user interface elements. gtk.DrawingArea is essentially a blank widget containing a window that you can draw on.
gtk.Calendar	This widget displays a calendar and allows the user to select a date.
gtk.Viewport	This widget displays a portion of a larger widget.

End of ebook preview

If you liked what you saw...

Buy it from our store @ <https://store.tutorialspoint.com>