**GRAPHICAL INTERFACES LIBRARIES USED IN PYTHON**

1. **KIVY**

It is an open source software library for the rapid development of applications equipped with novel user interfaces, such as multi-touch apps.

With kivy you can create apps that runs on :

* Desktop computers: macOS, linux, windows
* iOS devices: iPad,iPhone
* android devices:tablets,phones
* any other touch-enabled professional/homebrew/ devices supporting tangible user interface object

1. **DJANGO**

-Is high level python web framework that enables rapid development of sources and maintainable websites.

-It takes care of much of the hassle of web development,so ypu can focus on writing your app without need to reinvent the wheel.

-It is free and open source ,has a thriving and active community, great documentation , and many options for free and paid-for support.

Django helps you write software that is:

Complete

Django follows the "Batteries included" philosophy and provides almost everything developers might want to do "out of the box". Because everything you need is part of the one "product", it all works seamlessly together, follows consistent design principles, and has extensive and [up-to-date documentation](https://docs.djangoproject.com/en/stable/).

Versatile

Django can be (and has been) used to build almost any type of website — from content management systems and wikis, through to social networks and news sites. It can work with any client-side framework, and can deliver content in almost any format (including HTML, RSS feeds, JSON, and XML).

Internally, while it provides choices for almost any functionality you might want (e.g. several popular databases, templating engines, etc.), it can also be extended to use other components if needed.

Secure

Django helps developers avoid many common security mistakes by providing a framework that has been engineered to "do the right things" to protect the website automatically

Django enables protection against many vulnerabilities by default, including SQL injection, cross-site scripting, cross-site request forgery and [clickjacking](https://developer.mozilla.org/en-US/docs/Glossary/Clickjacking) (see [Website security](https://developer.mozilla.org/en-US/docs/Learn/Server-side/First_steps/Website_security) for more details of such attacks).

Scalable

Django uses a component-based "[shared-nothing](https://en.wikipedia.org/wiki/Shared_nothing_architecture)" architecture (each part of the architecture is independent of the others, and can hence be replaced or changed if needed). Having a clear separation between the different parts means that it can scale for increased traffic by adding hardware at any level: caching servers, database servers, or application servers. Some of the busiest sites have successfully scaled Django to meet their demands (e.g. Instagram and Disqus, to name just two).

Maintainable

Django code is written using design principles and patterns that encourage the creation of maintainable and reusable code. In particular, it makes use of the Don't Repeat Yourself (DRY) principle so there is no unnecessary duplication, reducing the amount of code. Django also promotes the grouping of related functionality into reusable "applications" and, at a lower level, groups related code into modules (along the lines of the [Model View Controller (MVC)](https://developer.mozilla.org/en-US/docs/Glossary/MVC) pattern).

Portable

Django is written in Python, which runs on many platforms. That means that you are not tied to any particular server platform, and can run your applications on many flavors of Linux, Windows, and macOS. Furthermore, Django is well-supported by many web hosting providers, who often provide specific infrastructure and documentation for hosting Django sites.

1. **WXPYTHON**

It a wrapper for the cross-platform GUC API(often referred to as a toolkit)

wxWidgets for the python programming language. It is implemented as python extension module

Applications developed with wxPython

* [Chandler](https://en.wikipedia.org/wiki/Chandler_(PIM)), a personal information manager
* [Dropbox](https://en.wikipedia.org/wiki/Dropbox_(service)), desktop client for the Dropbox cloud-based storage
* [Editra](https://en.wikipedia.org/wiki/Editra), a multi-platform text editor
* [Google Drive](https://en.wikipedia.org/wiki/Google_Drive), desktop client for the Google cloud-based storage system
* [GRASS GIS](https://en.wikipedia.org/wiki/GRASS_GIS), a free, open source geographical information system
* [Métamorphose](https://en.wikipedia.org/wiki/M%C3%A9tamorphose_(renamer)), a batch renamer
* [Phatch](https://en.wikipedia.org/wiki/Phatch), a photo batch processor
* [PlayOnLinux](https://en.wikipedia.org/wiki/PlayOnLinux) and [PlayOnMac](https://en.wikipedia.org/wiki/PlayOnMac" \o "PlayOnMac), [Wine](https://en.wikipedia.org/wiki/Wine_(software)) front-ends
* [PsychoPy](https://en.wikipedia.org/wiki/PsychoPy), experiment creation tool for neuroscience and psychology research

**4 TKINTER**

**-** is a [Python](https://en.wikipedia.org/wiki/Python_(programming_language)) [binding](https://en.wikipedia.org/wiki/Language_binding) to the [Tk](https://en.wikipedia.org/wiki/Tk_(software)" \o "Tk (software)) [GUI](https://en.wikipedia.org/wiki/Graphical_user_interface) toolkit. It is the standard Python interface to the Tk GUI toolkit, and is Python's [*de facto* standard](https://en.wikipedia.org/wiki/De_facto_standard) GUI. Tkinter is included with standard [Linux](https://en.wikipedia.org/wiki/Linux), [Microsoft Windows](https://en.wikipedia.org/wiki/Microsoft_Windows) and [macOS](https://en.wikipedia.org/wiki/MacOS" \o "MacOS) installs of Python.

## Description

As with most other modern Tk bindings, Tkinter is implemented as a Python wrapper around a complete [Tcl](https://en.wikipedia.org/wiki/Tcl" \o "Tcl) [interpreter](https://en.wikipedia.org/wiki/Interpreter_(computing)) embedded in the Python [interpreter](https://en.wikipedia.org/wiki/Language_interpretation). Tkinter calls are translated into Tcl commands, which are fed to this embedded interpreter, thus making it possible to mix Python and Tcl in a single application.

**5. PyQt**

- Is a [Python](https://en.wikipedia.org/wiki/Python_(programming_language)) [binding](https://en.wikipedia.org/wiki/Language_binding) of the [cross-platform](https://en.wikipedia.org/wiki/Cross-platform) [GUI](https://en.wikipedia.org/wiki/GUI) toolkit [Qt](https://en.wikipedia.org/wiki/Qt_(toolkit)" \o "Qt (toolkit)), implemented as a Python [plug-in](https://en.wikipedia.org/wiki/Plug-in_(computing)). PyQt is [free software](https://en.wikipedia.org/wiki/Free_software) developed by the [British](https://en.wikipedia.org/wiki/United_Kingdom) firm Riverbank Computing. It is available under similar terms to Qt versions older than ; this means a variety of licenses including [GNU General Public License](https://en.wikipedia.org/wiki/GNU_General_Public_License) (GPL) and commercial license, but not the [GNU Lesser General Public License](https://en.wikipedia.org/wiki/GNU_Lesser_General_Public_License) (LGPL).

PyQt supports [Microsoft Windows](https://en.wikipedia.org/wiki/Microsoft_Windows) as well as various flavours of [UNIX](https://en.wikipedia.org/wiki/Unix), including [Linux](https://en.wikipedia.org/wiki/Linux) and [MacOS (or Darwin)](https://en.wikipedia.org/wiki/MacOS" \o "MacOS).

PyQt implements around 440 classes and over 6,000 functions and methodsincluding:

* a substantial set of [GUI widgets](https://en.wikipedia.org/wiki/GUI_widget)
* [classes](https://en.wikipedia.org/wiki/Class_(computer_science)) for accessing [SQL](https://en.wikipedia.org/wiki/SQL) [databases](https://en.wikipedia.org/wiki/Database) ([ODBC](https://en.wikipedia.org/wiki/ODBC), [MySQL](https://en.wikipedia.org/wiki/MySQL), [PostgreSQL](https://en.wikipedia.org/wiki/PostgreSQL), [Oracle](https://en.wikipedia.org/wiki/Oracle_database), [SQLite](https://en.wikipedia.org/wiki/SQLite))[[6]](https://en.wikipedia.org/wiki/PyQt#cite_note-6)
* QScintilla, [Scintilla](https://en.wikipedia.org/wiki/Scintilla_(editing_component))-based rich text editor widget
* data aware widgets that are automatically populated from a database
* an [XML](https://en.wikipedia.org/wiki/XML) parser
* [SVG](https://en.wikipedia.org/wiki/Scalable_Vector_Graphics) support
* classes for embedding [ActiveX](https://en.wikipedia.org/wiki/ActiveX) controls on Windows (only in commercial version)
  1. **PySide**

Is a [Python](https://en.wikipedia.org/wiki/Python_(programming_language)) [binding](https://en.wikipedia.org/wiki/Language_binding) of the [cross-platform](https://en.wikipedia.org/wiki/Cross-platform) [GUI](https://en.wikipedia.org/wiki/Graphical_user_interface) toolkit [Qt](https://en.wikipedia.org/wiki/Qt_(framework)" \o "Qt (framework)) developed by [The Qt Company](https://en.wikipedia.org/wiki/The_Qt_Company), as part of the **Qt for Python** project. It is one of the [alternatives](https://en.wikipedia.org/wiki/List_of_widget_toolkits) to the standard library package [Tkinter](https://en.wikipedia.org/wiki/Tkinter" \o "Tkinter). Like Qt, PySide is [free software](https://en.wikipedia.org/wiki/Free_software). PySide supports [Linux](https://en.wikipedia.org/wiki/Linux)/[X11](https://en.wikipedia.org/wiki/X_Window_System), [macOS](https://en.wikipedia.org/wiki/MacOS" \o "MacOS), and [Microsoft Windows](https://en.wikipedia.org/wiki/Microsoft_Windows).

**7. Pygame**

-Is a [cross-platform](https://en.wikipedia.org/wiki/Cross-platform) set of [Python](https://en.wikipedia.org/wiki/Python_(programming_language)) modules designed for writing [video games](https://en.wikipedia.org/wiki/Video_game). It includes [computer graphics](https://en.wikipedia.org/wiki/Computer_graphics) and sound [libraries](https://en.wikipedia.org/wiki/Library_(computing)) designed to be used with the Python [programming language](https://en.wikipedia.org/wiki/Programming_language)

## History

Pygame was originally written by Pete Shinners to replace PySDL after its development stalled. It has been a [community](https://en.wikipedia.org/wiki/Free_software_community) project since 2000 and is released under the [free software](https://en.wikipedia.org/wiki/Free_software) [GNU Lesser General Public License](https://en.wikipedia.org/wiki/GNU_Lesser_General_Public_License) (which "provides for Pygame to be distributed with [open source](https://en.wikipedia.org/wiki/Open-source_software) and commercial 7software").

**8. Piglet**

**-** Is a [library](https://en.wikipedia.org/wiki/Library_(computing)) for the [Python](https://en.wikipedia.org/wiki/Python_(programming_language)) [programming language](https://en.wikipedia.org/wiki/Programming_language) that provides an [object-oriented](https://en.wikipedia.org/wiki/Object-oriented_programming) [application programming interface](https://en.wikipedia.org/wiki/Application_programming_interface) for the creation of games and other multimedia [applications](https://en.wikipedia.org/wiki/Application_software). pyglet runs on [Microsoft Windows](https://en.wikipedia.org/wiki/Microsoft_Windows), [macOS](https://en.wikipedia.org/wiki/MacOS" \o "MacOS), and [Linux](https://en.wikipedia.org/wiki/Linux); it is released under the [BSD Licence](https://en.wikipedia.org/wiki/BSD_Licence). pyglet was first created by Alex Holkner.

It supports windowed and full-screen operations as well as multiple monitors. Images, video, and sound files in a range of formats can be done natively, with more additional capabilities supplied by the optional AVbin plugin, which uses the [Libav](https://en.wikipedia.org/wiki/Libav" \o "Libav) package to provide support for audio formats including [MP3](https://en.wikipedia.org/wiki/MP3), [Ogg](https://en.wikipedia.org/wiki/Ogg" \o "Ogg)/Vorbis, and [Windows Media Audio](https://en.wikipedia.org/wiki/Windows_Media_Audio), and video formats such as [DivX](https://en.wikipedia.org/wiki/DivX), [MPEG-2](https://en.wikipedia.org/wiki/MPEG-2), [H.264](https://en.wikipedia.org/wiki/H.264), [WMV](https://en.wikipedia.org/wiki/WMV), and [XviD](https://en.wikipedia.org/wiki/XviD" \o "XviD).

An advantage of pyglet over many other libraries is that it requires no external dependencies, and uses the [ctypes](https://en.wikipedia.org/wiki/Ctypes_(language_binding)" \o "Ctypes (language binding)) library, a pure-Python C compiler. It builds on [OpenGL](https://en.wikipedia.org/wiki/OpenGL).

**9. PyGTK**

-Is a set of [Python](https://en.wikipedia.org/wiki/Python_(programming_language)) [wrappers](https://en.wikipedia.org/wiki/Adapter_pattern) for the [GTK](https://en.wikipedia.org/wiki/GTK) [graphical user interface](https://en.wikipedia.org/wiki/Graphical_user_interface) [library](https://en.wikipedia.org/wiki/Library_(computer_science)). PyGTK is [free software](https://en.wikipedia.org/wiki/Free_software) and licensed under the [LGPL](https://en.wikipedia.org/wiki/LGPL). It is analogous to [PyQt](https://en.wikipedia.org/wiki/PyQt" \o "PyQt)/[PySide](https://en.wikipedia.org/wiki/PySide" \o "PySide) and [wxPython](https://en.wikipedia.org/wiki/WxPython" \o "WxPython), the Python wrappers for [Qt](https://en.wikipedia.org/wiki/Qt_(framework)" \o "Qt (framework)) and [wxWidgets](https://en.wikipedia.org/wiki/WxWidgets" \o "WxWidgets), respectively. Its original author is [GNOME](https://en.wikipedia.org/wiki/GNOME) developer James Henstridge. There are six people in the core development team, with various other people who have submitted patches and bug reports. PyGTK has been selected as the environment of choice for applications running on [One Laptop Per Child](https://en.wikipedia.org/wiki/One_Laptop_Per_Child) systems.

PyGTK will be phased out with the transition to GTK version 3 and be replaced with PyGObject, which uses [GObject Introspection](https://en.wikipedia.org/wiki/GObject_Introspection) to generate bindings for Python and other languages on the fly. This is expected to eliminate the delay between GTK updates and corresponding [language binding](https://en.wikipedia.org/wiki/Language_binding) updates, as well as reduce maintenance burden on the developers.