



Estructura básica de una aplicación PyQt

Este código abre una ventana sin contenido. A partir de aquí, construiremos la aplicación. ([Archivo a01](#))

```
from PyQt6.QtWidgets import QApplication, QWidget

# Sólo es necesario para acceder a los argumentos de la línea de comando
import sys

# Necesita una (¡y solo una!) instancia de QApplication por aplicación.
# Pase sys.argv para permitir argumentos de línea de comando para su aplicación.
# Si sabe que no utilizará argumentos de línea de comando, sustitúyalo por una lista vacía
# de esta manera: QApplication([])
app = QApplication(sys.argv)

# Crea un widget Qt, que será nuestra ventana.
ventana = QWidget()
ventana.show() #!!!IMPORTANTE!!! Las ventanas están ocultas de forma predeterminada.

# Iniciar el ciclo de eventos.
app.exec()

# La aplicación no llegará aquí hasta que salga y el evento # bucle se ha detenido.
print("Has cerrado la ventana")
```

Mediante definición de objetos

Para crear una aplicación básica con PyQt utilizando clases, debemos seguir los siguientes pasos y estructura: [\(Archivo a02\)](#)

2. Importar las Librerías Necesarias

Además de `QApplication` y `QWidget`, añadimos un layout a modo de bastidor en el que podemos "colgar" elementos con `QVBoxLayout` o `QHBoxLayout`. También agregamos un botón que no hace nada.

```
from PyQt6.QtWidgets import QApplication, QWidget, QVBoxLayout, QPushButton
```

3. Crear la Clase de la Ventana Principal

La **VentanaPrincipal** de nuestra aplicación hereda de `QWidget` para tener su funcionalidad. Aquí es donde definiremos el layout y los widgets de tu ventana.

```
class VentanaPrincipal(QWidget):
    def __init__(self):
        super().__init__()
        self.initUI()

    def initUI(self):
        # Configuración de la ventana
        self.setWindowTitle('Mi Aplicación PyQt')
        self.setGeometry(300, 300, 300, 200) # x, y, ancho, alto

        # Crear un layout vertical
        layout = QVBoxLayout()

        # Añadir widgets al layout
        boton = QPushButton('Haz clic aquí')
        layout.addWidget(boton)

        # Establecer el layout de la ventana
        self.setLayout(layout)
```

4. Crear la Función `main`

Esta función creará una instancia de `QApplication`, instanciará la ventana principal y ejecutará el ciclo de eventos de la aplicación.

```
def main():
    app = QApplication([])
    ventana = VentanaPrincipal()
    ventana.show()
    app.exec()

if __name__ == '__main__':
    main()
```

5. Ejecutar la Aplicación

Verás aparecer la ventana que has definido.

