



LIBRERIA Tfhka
MANUAL DE INTEGRACION

Versión 2.0 - Venezuela

Diciembre de 2023

The Factory HKA, C.A.

MANUAL DE INTEGRACION - LIBRERÍA Tfhka

VERSIÓN 2.0 - VENEZUELA

The Factory HKA

La California Norte, Callejón Gutiérrez

Edif. Riva, PB Ofic. 2-1, Caracas - Venezuela

Teléfono (212) 237.4112 • 2398176

Departamento de Soporte e Integración

integration@thefactoryhka.com

Introducción

La librería Tfhka permite la integración con sistemas administrativos desarrollados bajo la tecnología Python.

En las siguientes páginas se describen los detalles de la librería Tfhka, desde los requerimientos de hardware, la referencia a la librería de integración, los componentes adicionales tales como librerías dinámicas y finalmente la estructura de la Clase compilada que contiene los métodos y propiedades para el uso de cualquiera de las impresoras distribuidas por The Factory HKA a través de un sistema administrativo desarrollado en Python.

Requerimientos de hardware

- Procesador de 1Ghz o superior
- Memoria RAM de 2GB o superior
- Espacio libre en Disco Duro de 50MB
- Puerto USB 2.0

Requerimientos de software




- Sistema Operativo Windows 10 o superior
- Python 3.11
- Librería PySerial 3.5 (Importar desde Librería)
- IDE para manejar un desarrollo en Python 3.11

Instalación

En el SDK descargado, usted se encontrará con las tres herramientas básicas para realizar la integración del sistema administrativo con nuestra impresora fiscal; la librería de integración “TfhkaPyGD.py”, el manual de dicha librería y la librería “pyserial 3.5” requerida para la comunicación de la impresora fiscal.

Antes de comenzar a hacer uso de la librería en el desarrollo, lo primero que se debe hacer es incluir y referenciar la librería en nuestro proyecto. Esto se consigue siguiendo los siguientes pasos:

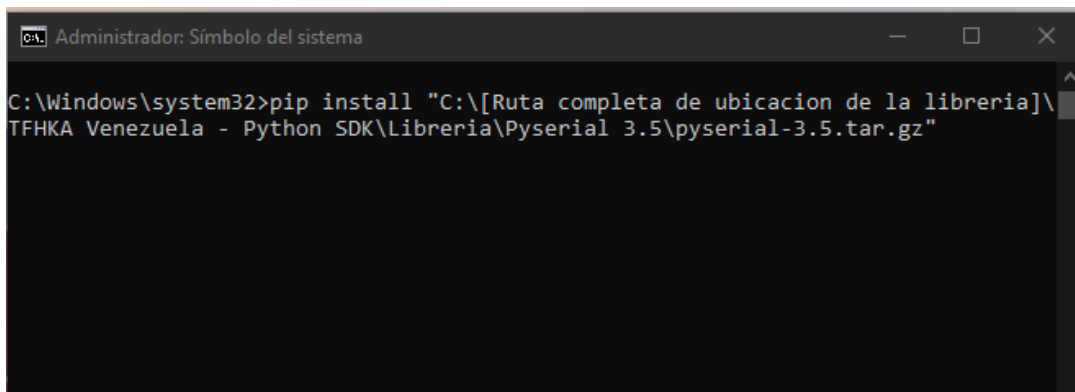
- Incluir la librería TfhkaPyGD.py y los archivos relacionados (resaltados a continuación), así como el archivo de interfaz gráfica con extensión “.ui” dentro de la carpeta que contendrá el proyecto:

	Interfaz UI Tipo: Archivo UI	Fecha de modificación: 6/11/2023 9:47 p. m. Tamaño: 42,0 KB
	Proyecto Ejemplo Tipo: Archivo de origen Python	Fecha de modificación: 7/11/2023 9:29 p. m. Tamaño: 50,5 KB
	TfhkaPyGD Tipo: Archivo de origen Python	Fecha de modificación: 7/11/2023 9:19 p. m. Tamaño: 84,6 KB

- Creamos un nuevo proyecto e importamos la librería de integración TfhkaPyGD y las librerías nativas de Python señaladas a continuación:

```
10 import sys
11 import TfhkaPyGD
12 import os
13 import time
```

- La librería TfhkaPyGD hace uso de la librería “PySerial”, se debe instalar la versión proporcionada en la carpeta de la librería, dentro de la carpeta “Pyserial 3.5” se encuentra el archivo con extensión “.tar.gz” para ser instalado. Esto debe ser instalado con el terminal CMD a través del comando pip especificando la ruta de instalación con esta estructura:



- Le mostrara los siguientes mensajes indicando que ha sido exitoso:

```
Processing c:\users\ecamacho\desktop\tfhka venezuela - python sdk\libreria\pyserial 3.5\pyserial-3.5.tar.gz
Installing build dependencies ... done
Getting requirements to build wheel ... done
Preparing metadata (pyproject.toml) ... done
Building wheels for collected packages: pyserial
Building wheel for pyserial (pyproject.toml) ... done
Created wheel for pyserial: filename=pyserial-3.5-py2.py3-none-any.whl size=90541 sha256=0f467d8f408c5a5d97c779034cbb7aa9348902e5bfc589d9c1fc1743a3144b2b
Stored in directory: c:\users\ecamacho\appdata\local\pip\cache\wheels\d0\52\91\879a8784e3f2d7660f71abd650ae768ccea98ffb18c4cff12a
Successfully built pyserial
Installing collected packages: pyserial
Successfully installed pyserial-3.5
```

- Finalmente, se crea el objeto de tipo Tfhka que es el que maneja los distintos métodos y funciones a utilizar para interactuar con la impresora fiscal.

```
20 class Principal(QMainWindow):
21
22     def __init__(self):
23         super().__init__()
24         QMainWindow.__init__(self)
25         ui_file_path = os.path.join(os.path.dirname(__file__), 'DemoPython.ui')
26         uic.loadUi(ui_file_path, self)
27
28         self.printer = TfhkaPyGD.Tfhka()
29
```

Acto seguido, nos encontramos listos para empezar a utilizar los métodos y funciones que contiene la clase Tfhka.

Importaciones y declaraciones

Una vez importada la librería TfhkaPyGD al proyecto se puede empezar a trabajar con la clase Tfhka, con sus atributos y métodos públicos.

Implementación

```
import TfhkaPyGD
```

Declaración de un objeto tipo Tfhka

```
<Nombre del Objeto> = TfhkaPyGD.Tfhka()
```

Ejemplo:

```
self.printer = TfhkaPyGD.Tfhka()
```

Funciones de la clase Tfhka

OpenFpctrl

Permite realizar la apertura del puerto de comunicaciones por el cual se establecerá comunicación con la impresora.

OpenFpctrl (String Puerto)

- **Parámetros:**

String Puerto: Nombre del puerto COM a abrir.

- **Retorno:**

True: Puerto Abierto.

False: Falla en apertura.

CloseFpctrl

Permite cerrar el puerto COM abierto anteriormente:

CloseFpctrl ()

ReadFpStatus

Permite leer las variables de estado y error de la impresora. Al ejecutar este método se establece el valor de la variable tipo String Estado. (Ver Anexos 1 y 2)

ReadFpStatus ()

- **Retorno:**

Cadena de caracteres que contienen el estado y error de la impresora.

SendCmd

Permite realizar el envío de comandos hacia la impresora, en forma de tramas de caracteres ASCII, tal como es descrito en los manuales de integración de las respectivas impresoras y en el Manual de Protocolos y Comandos del protocolo TFHKA.

SendCmd (String Cmd)

- **Parámetros:**

String Cmd: Trama en ASCII.

- **Retorno:**

True: Método ejecutado exitosamente.

False: Error en ejecución del método.

SendCmdFile

Permite realizar el envío de un archivo de lotes de comando hacia la impresora.

SendCmdFile (String path)

- **Parámetros:**

String path: ruta en la cual se encuentra el archivo de comandos por lote que se enviará a la impresora.

- **Retorno:**

True: Método ejecutado exitosamente.

False: Error en ejecución del método.