

Cómo adaptar un programa Python para tus amigos no-programadores

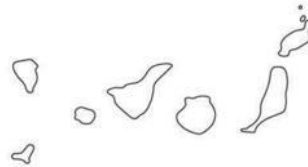
Pablo Rodríguez Pérez





Data Scientist en PiperLab

Teleco: imagen y sonido





Origen del problema





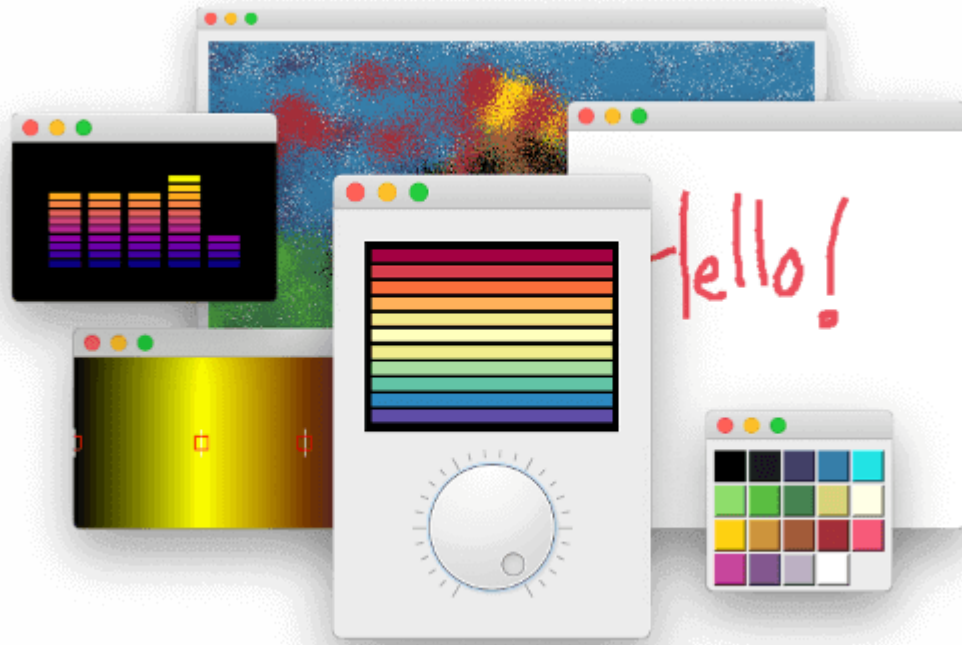
Requisitos

- BACK-END PYTHON
- INTERFAZ GRÁFICA (GUI)
- FÁCIL DE CONSTRUIR
- ARRANQUE CON DOBLE CLIC
- SIN PRERREQUISITOS NI DEPENDENCIAS
- MULTIPLATAFORMA

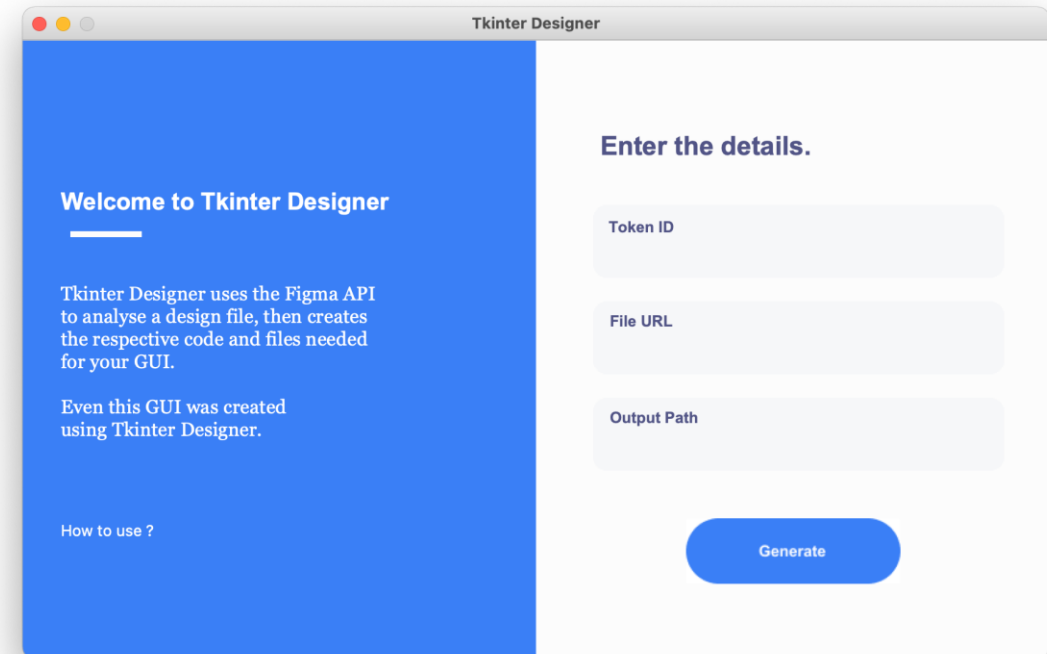
Objetivo



Crear una aplicación de escritorio con interfaz gráfica



<https://www.pythonguis.com/pyqt5-tutorial/>

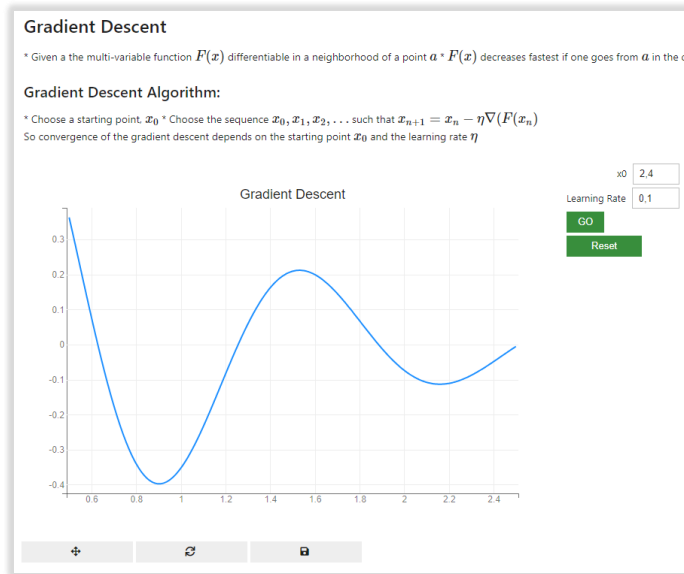


<https://github.com/ParthJadhav/Tkinter-Designer>

☑ Soluciones muy completas, con Python, sin dependencias, multiplataforma, ...

✗ Curva de aprendizaje considerable

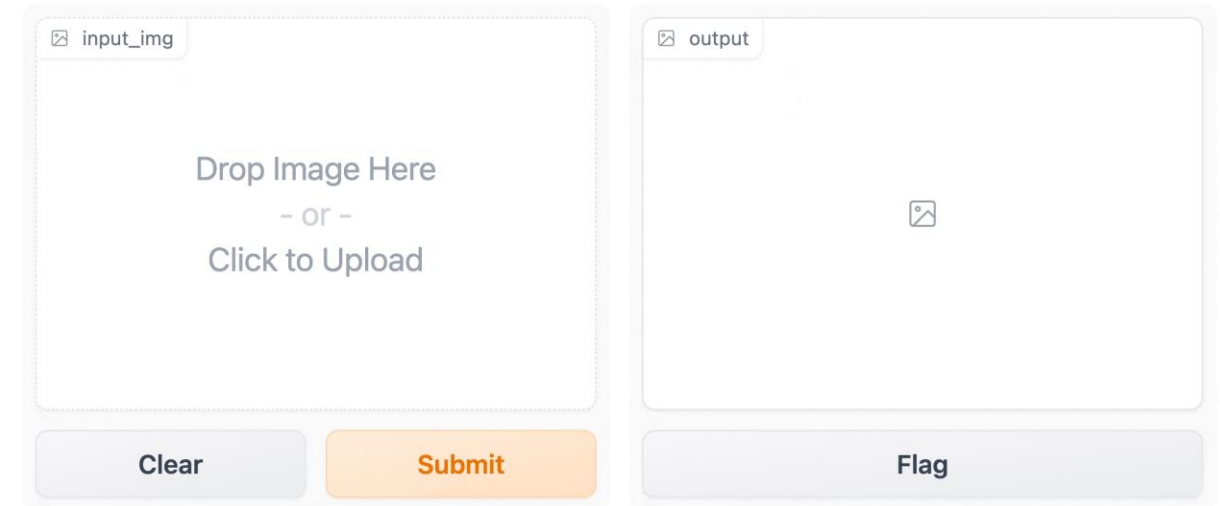
Voilà



Convierte notebooks en webapps a partir de widgets de Jupyter

<https://github.com/voila-dashboards/voila>

Gradio

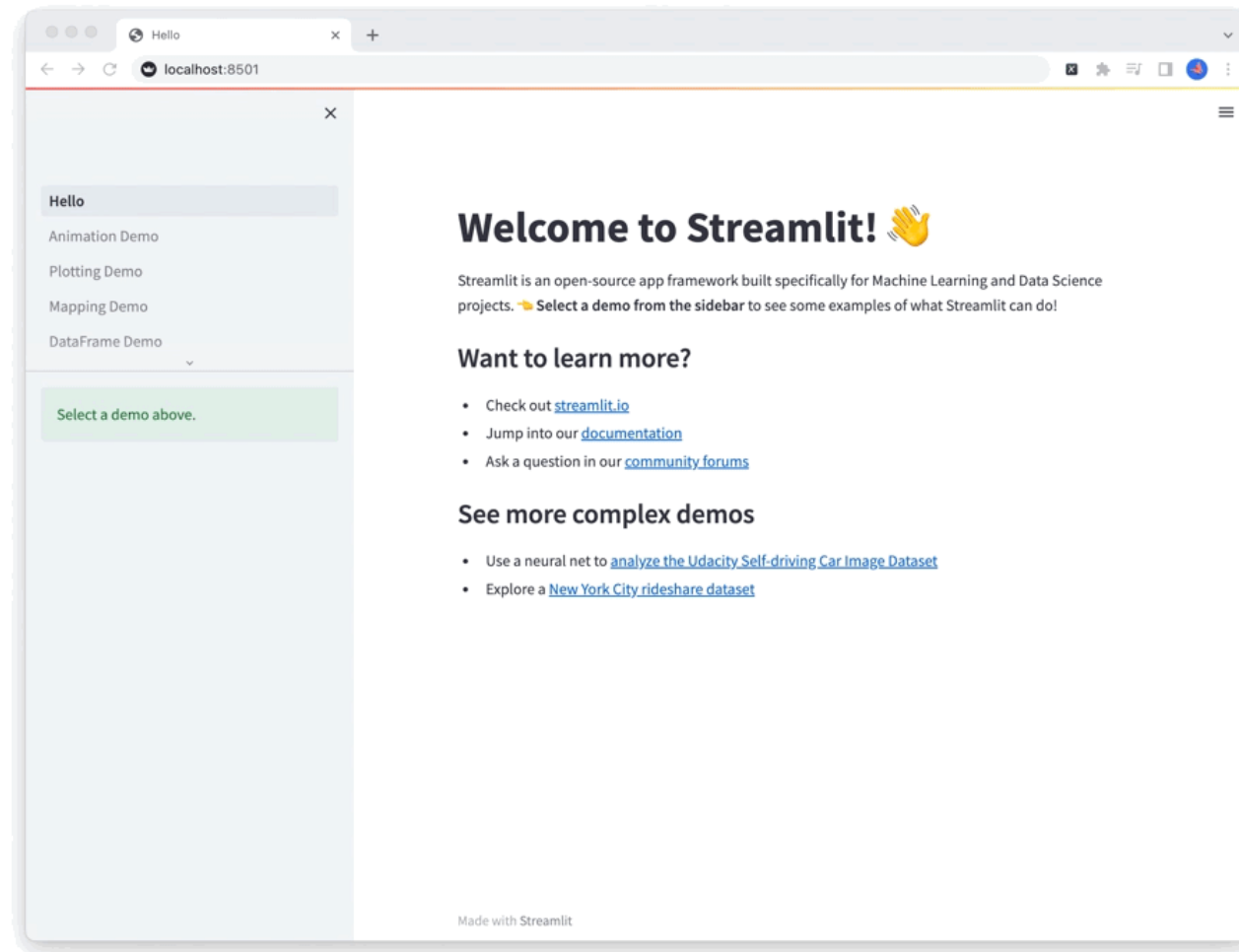


built with gradio

Crea interfaces de entrada y salida de datos fácilmente.

Muy usado en Hugging Face.

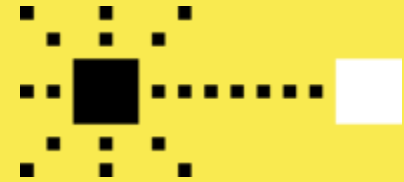
<https://github.com/gradio-app/gradio>



Objetivo



Crear una aplicación de escritorio con interfaz gráfica **que funcione haciendo doble clic**



01.

**Instalar
Python**

02.

**Instalar
dependencias**

03.

Usar terminal



- **Pyinstaller:**
 - Solo crea ejecutables para el mismo sistema operativo
 - Windows, macOS y Linux
- **Auto-py-to-exe:**
 - Interfaz sencilla
 - Solo Windows
 - Problemas con la importación de algunas dependencias como OpenCV
- **Pynsist:**
 - Solo Windows
 - Guarda todo lo necesario para una instalación offline
 - El resultado puede pesar fácilmente cientos de MB o GB
- **Constructor**
 - Crear un instalador a partir de un entorno de conda
 - Solo crea ejecutables para el mismo sistema operativo

<https://pyinstaller.org/en/stable/>

<https://github.com/brentvollebrecht/auto-py-to-exe>

<https://github.com/takluyver/pynsist>

<https://conda.github.io/constructor/>

Objetivo

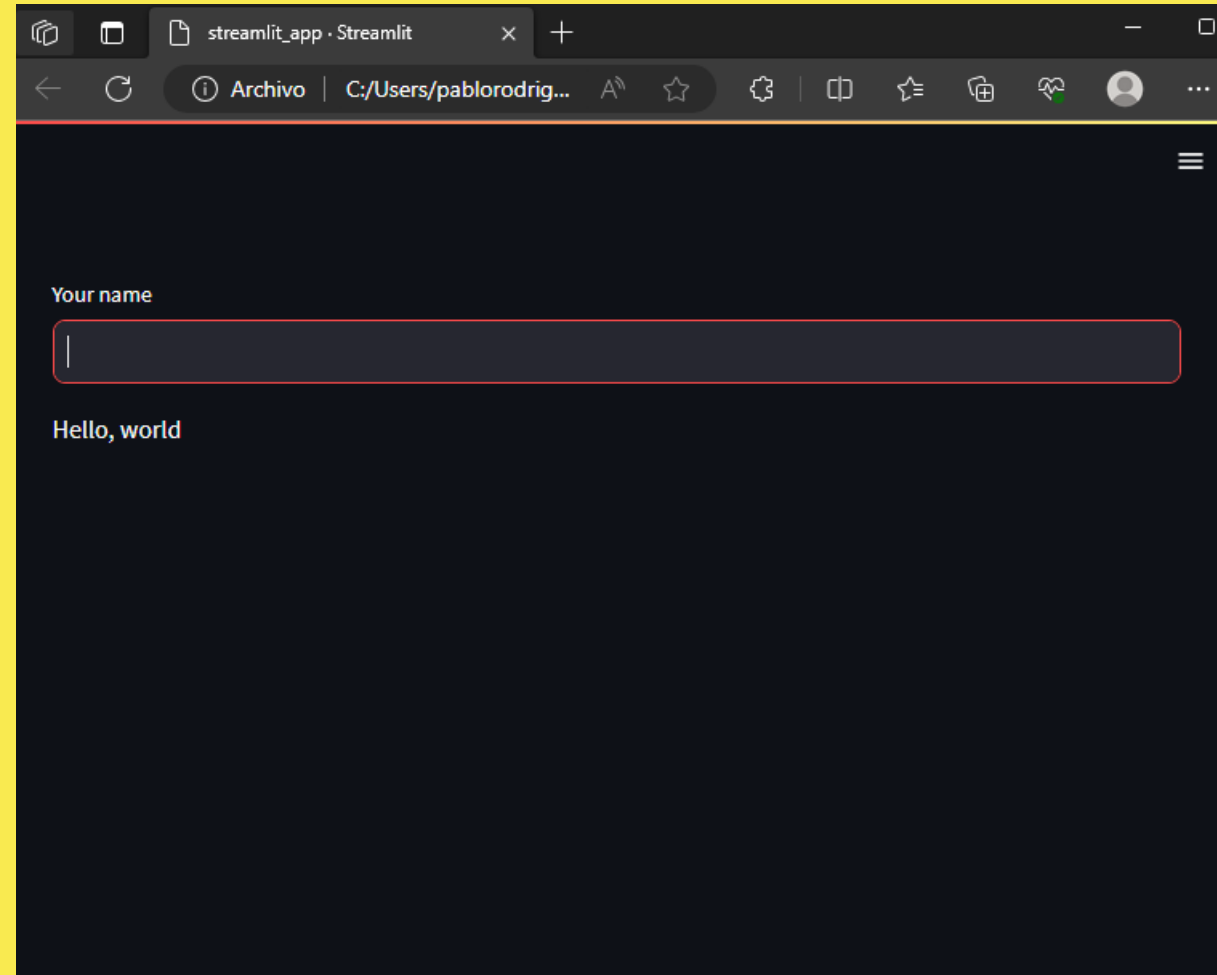


Crear una aplicación **portable** de escritorio con interfaz gráfica que funcione haciendo doble clic



Port de Streamlit a WebAssembly

<https://github.com/whitphx/stlite>

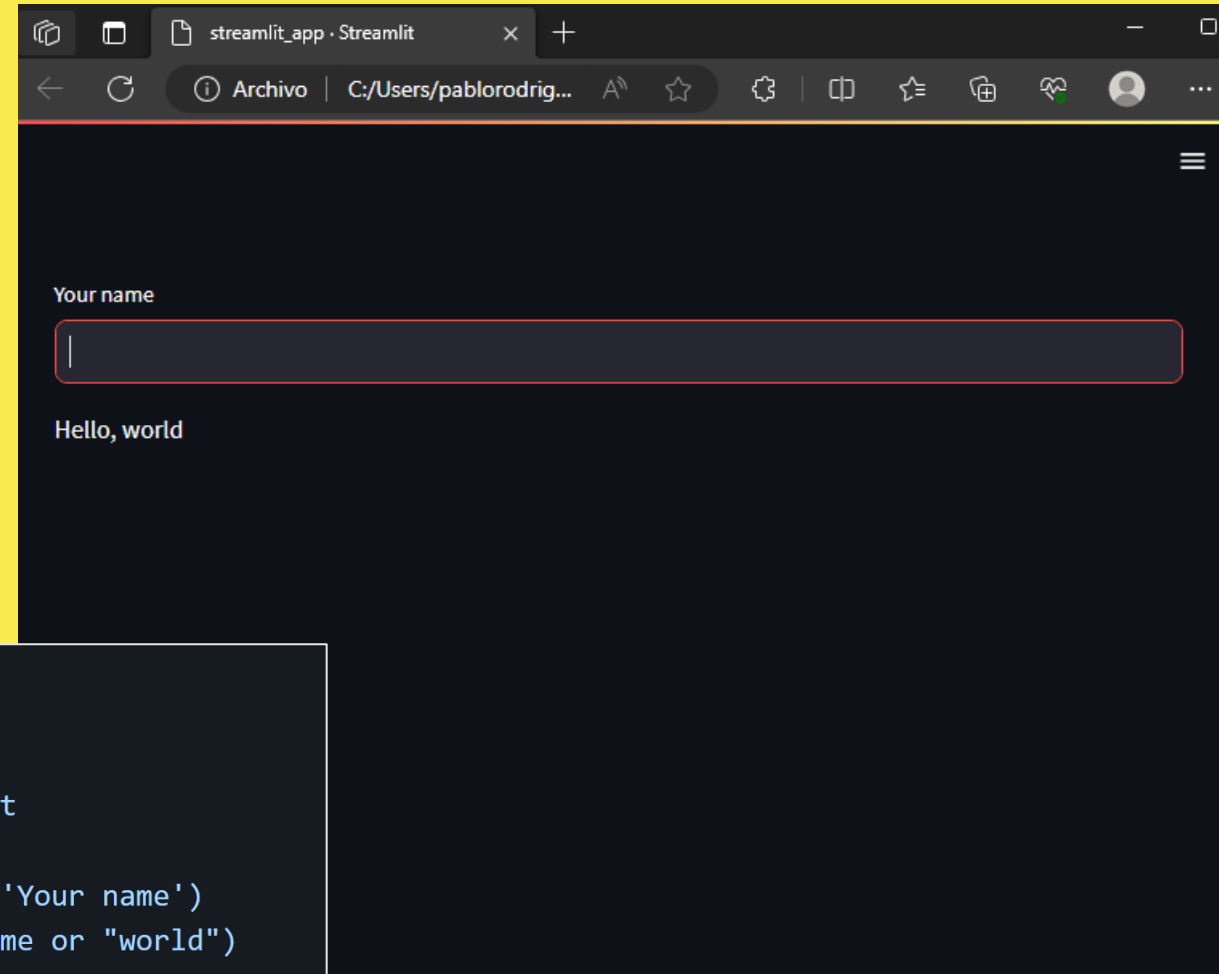


```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta
      name="viewport"
      content="width=device-width, initial-scale=1, shrink-to-fit=no"
    />
    <title>stlite app</title>
    <link
      rel="stylesheet"
      href="https://cdn.jsdelivr.net/npm/@stlite/mountable@0.34.0/build/stlite.css"
    />
  </head>
  <body>
    <div id="root"></div>
    <script src="https://cdn.jsdelivr.net/npm/@stlite/mountable@0.34.0/build/stlite.js"></script>
    <script>
      stlite.mount(
        `
import streamlit as st

name = st.text_input('Your name')
st.write("Hello,", name or "world")
`,
        document.getElementById("root")
      );
    </script>
  </body>
</html>
```

```
<script>
  stlite.mount(
    `
import streamlit as st

name = st.text_input('Your name')
st.write("Hello,", name or "world")
`,
    document.getElementById("root")
  );
</script>
```



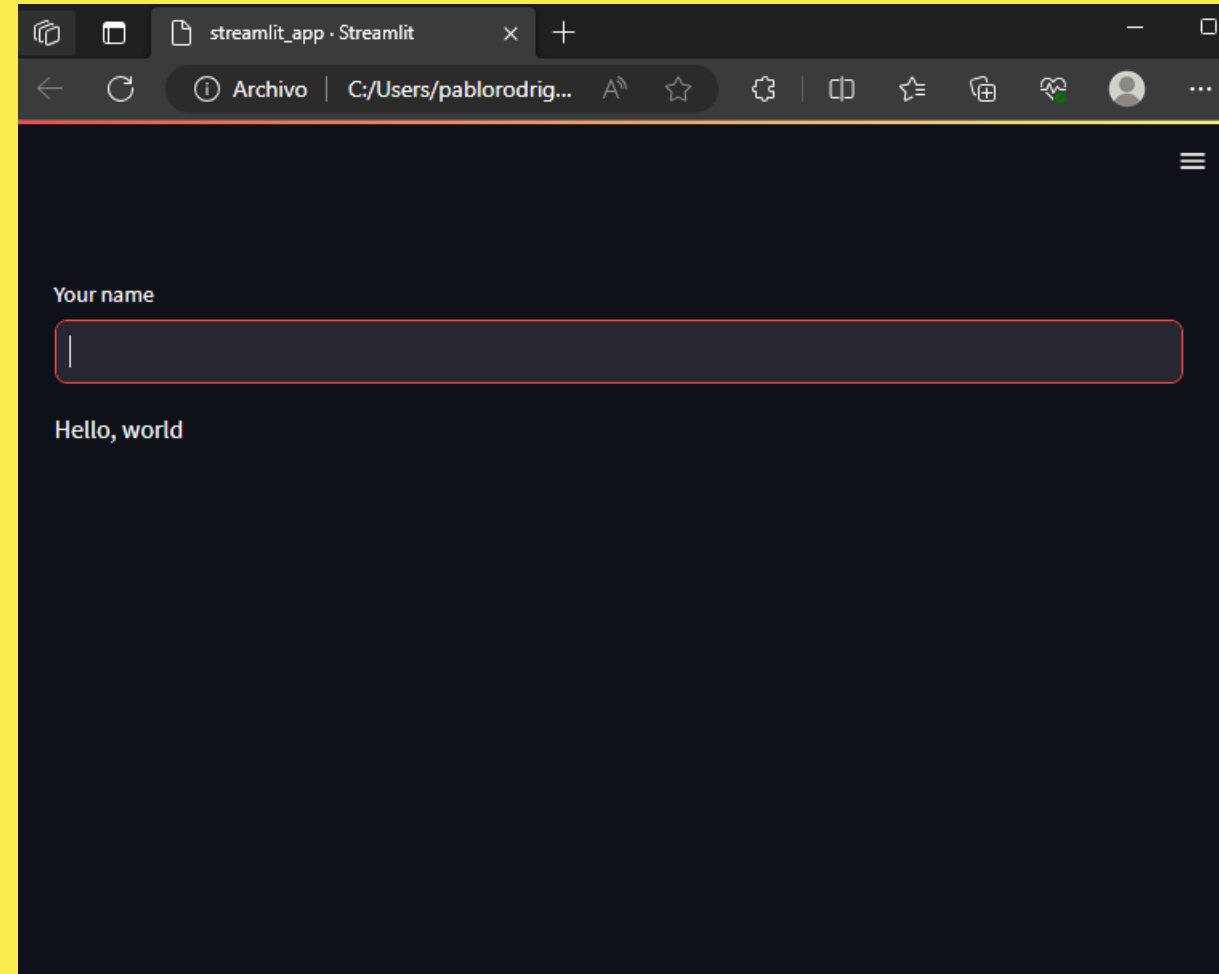
DEMO TIME

<https://github.com/whitphx/stlite>





- Tiempo de arranque alto
- Código legible por el receptor
- Problemas con paquetes basados en C
 - Numpy, pandas, ...
- Complica el uso de varios scripts .py





- Tiempo de arranque alto
- Código legible por el receptor
- Problemas con paquetes basados en C
 - Numpy, pandas, Tensorflow, ...
- Complica el uso de varios scripts .py
 - <https://github.com/pablorodriper/streamlit-to-stlite>

```
stlite.mount(  
  {  
    entrypoint: "🔥_Hello.py",  
    files: {  
      "🔥_Hello.py": `  
import streamlit as st  
  
st.set_page_config(page_title="Hello")  
st.title("Main page")  
`  
    },  
    "pages/1 🌟 Page1.py": `  
import streamlit as st  
  
st.set_page_config(page_title="Page1")  
st.title("Page 1")  
`  
    },  
    "pages/2 🚫 Page2.py": `  
import streamlit as st  
  
st.set_page_config(page_title="Page2")  
st.title("Page 2")  
`  
    },  
  },  
  document.getElementById("root")  
);
```

GUIs

- Tkinter
- PyQt
- Voilà
- Solara
- Gradio
- Streamlit

Ejecutables

- Pyinstaller
- Auto-py-to-exe
- Pynsist
- Constructor
- Briefcase

Web

- ReactPy
 - Crear componentes como React
 - <https://github.com/reactive-python/reactpy>
- EEL
 - Electron-like offline HTML/JS GUI
 - <https://github.com/python-eel/Eel>

Portable





Blog técnico

<https://www.modeldifferently.com/2023/09/crear-una-aplicacion-de-escritorio-con-python-parte1/>

Blog divulgativo

<https://piperlab.es/more-than-data/>

Contacto

<https://www.linkedin.com/in/pablorodriguezperez/>

<https://sigmoid.social/@pabloRPBello>

<https://twitter.com/pabloRPBello>

¡Gracias!



PIPERLAB
Business Data Science. Differently.