

4. Desarrollo de aplicaciones web con Python (Streamlit)

Existen muchos paquetes orientados al desarrollo de aplicaciones web usando Python.

El paquete *Streamlit* es una de las opciones más sencillas para desarrollar aplicaciones web usando exclusivamente código Python.

Instalación de Streamlit

Para instalar streamlit la mejor opción es crear un entorno virtual de Python y no instalar el paquete de forma global.

Para crear un entorno virtual desde la terminal, creamos un directorio por ejemplo en el escritorio llamado "streamlit", y ejecutamos los siguientes comandos desde la terminal:

Windows :

```
cd C:\Users\TuUsuario\Desktop\streamlit
python -m venv .venv
.venv\Scripts\activate
pip install streamlit
```

Macos :

```
cd ~/Desktop/streamlit
python3 -m venv .venv
source .venv/bin/activate
pip install streamlit
```

Estas ordenes sirven para movernos al directorio streamlit, crear un entorno virtual llamado ".venv", activar el entorno e instalar la librería streamlit en ese entorno.

Posteriormente creamos un archivo llamado "prueba_streamlit.py" con el siguiente código:

```
import streamlit as st

st.write("Probando Streamlit !!")
```

para poder mostrar el archivo en el explorador debemos ejecutar desde el terminal:

```
streamlit run prueba_streamlit.py
```

4.1. Principales funciones de Streamlit

La mejor forma de aprender las diferentes funciones que incluye la librería es con la hoja resumen que podemos encontrar en la documentación.

[Streamlit cheat sheet](#)

Flujo de ejecución:

Streamlit ejecuta todo el código en el arranque y cada vez que uno de los valores obtenidos por pantalla cambien se vuelve a ejecutar el código entero.

4.2. Ejemplo de aplicación a desarrollar

Vamos desarrollar un visor de datos meteorológicos por día.

La aplicación debe cargar un dataframe con los datos del archivo "sevilla_met.csv", pedir al usuario el mes y el día que debe mostrar, filtrar el dataframe y mostrarlo para ese día y mes.

Añadir también una gráfica con líneas que muestre una variable previamente seleccionada para el día elegido.

Para ello copiamos el archivo "sevilla_met.csv" en nuestro directorio streamlit y modificamos el contenido de "prueba_streamlit.py" con el siguiente código:

```

import streamlit as st
import pandas as pd

st.set_page_config(page_title="Meteo Streamlit ejemplo", layout="wide")
meteo_df = pd.read_csv("sevilla_met.csv")

with st.sidebar:
    st.title("Meteo Streamlit ejemplo")
    st.header("Configuración")
    mes = st.number_input("Número del mes", min_value=1, max_value=12, value=1,
step=1)
    dia = st.number_input("Número de día", min_value=1, max_value=31, value=1,
step=1)
    variable = st.selectbox("Variable", meteo_df.columns, index=3)

meteo_df_dia = meteo_df[(meteo_df["mes"] == mes) & (meteo_df["día"] == dia)]
st.write(f"***Gráfica de la {variable} para el {dia}:{mes}***")
st.line_chart(meteo_df_dia,x="hora",y=[variable])

# DataFrame display
with st.expander('Ver el DataFrame de valores'):
    st.dataframe(meteo_df_dia)

```

Para poder mostrar el archivo en el explorador debemos ejecutar desde el terminal:

```
streamlit run prueba_streamlit.py
```

Ejercicio propuesto:

Hacer un aplicación web que cargando los datos del archivo Andalucía.xlsx en un DataFrame muestre la población, área y número de municipios totales de Andalucía y una gráfica de cada una de las variables anteriores seleccionadas previamente.