**Carpeta: PYxFlask**

**APRENDE FLASK EN 2 DÍAS**

**🎯 Objetivo General**

Que comprendas los fundamentos de Flask, desarrolles una aplicación web básica con rutas, formularios y procesamiento lógico, y que al final puedas crear una pequeña herramienta funcional que puedas integrar a procesos reales de la intranet.

**🗓️ Día 1: Fundamentos + Lógica Web**

**📌 Meta del día:** Comprender la arquitectura de Flask, servir rutas y procesar datos básicos desde formularios.

**🛠️ Paso 1: Instalación y entorno**

Configura el entorno con ‘**py -m venv .venv**’

Activa el entorno con ‘**source .venv\Scripts\activate’**

Instala Flask: ‘**pip install flask’**

Dentro de **PYxFlask**, crea un archivo **app.py**

**📘 Temas del día:**

* ¿Qué es Flask?
* Cómo funciona una app web con rutas.
* Templates con Jinja2.
* Procesamiento básico con **POST** y **GET**.

**🧠 Ejercicio 1: Tu primera ruta**

Crea una ruta / que devuelva 'Hola, Juan. Flask está funcionando.'

from flask import Flask

app = Flask(\_\_name\_\_)

@app.route("/")

def inicio():

return "Hola, Arias. Flask está funcionando."

if \_\_name\_\_ == "\_\_main\_\_":

app.run(debug=True)

**🧠 Ejercicio 2: Rutas dinámicas**

* Agrega una ruta **/saludo/<nombre>**
* Usa lógica básica para saludar diferente según la hora del día.

**🧠 Ejercicio 3: Formulario HTML**

* Crea una ruta **/formulario** con método **GET** y **POST**
* Muestra un formulario para ingresar nombre y número de celular.
* Si se envía correctamente, muestra una respuesta personalizada con los datos procesados.

**💡 Mini reto del día:**

* Construye un formulario que permita ingresar un número celular y, al enviarlo, lo limpie y lo muestre en pantalla en formato válido.
* Este será la base del ejercicio real del Día 2.

**🗓️ Día 2: Aplicación práctica real**

**📌 Meta del día:** Conectar la lógica con datos reales y generar algo útil para VTM intranet.

**📘 Temas del día:**

* Formularios reales.
* Validación de datos.
* Lectura y manipulación de diccionarios.
* Mostrar datos en tablas con HTML y Jinja2.
* Pensamiento en flujos de entrada → procesamiento → salida.

**🧠 Ejercicio 4: Carga de contactos**

* Simula una base de datos como un diccionario en Python con datos de contactos. (usa la del ejercicio **limpiar\_contactos\_dict** en PYxVanessa)
* Muestra esos contactos en pantalla.
* Crea una vista **/contactos** que los muestre en tabla HTML con Jinja2.

**🧠 Ejercicio 5: Limpieza de teléfonos**

* Usa la lógica del ejercicio anterior (limpiar números).
* Crea un botón en **/contactos** que diga “Limpiar teléfonos”.
* Al presionarlo, muestra los contactos válidos y sus teléfonos ya corregidos.

**🧠 TIPS PARA EL APRENDIZAJE:**

* Toma notas con ejemplos propios.
* Escribe el flujo en tu cuaderno antes de codificar cualquier cosa
* Usa prints para entender qué se envía y qué se recibe.
* Piensa siempre en “entrada → proceso → salida”.
* Comenta cada bloque que no entiendas bien.
* **PREGUNTEME** todo lo que no entienda después de documentarse.