

Facultad de Ciencia y Tecnología

Ingeniería en Sistemas y Redes informáticas

Asignatura:

Programacion Computacional III

Estadística Computacional

Docente:

Willian Montes

Estudiante:

William Alfredo Irula González

(SMSS141823)

Actividad:

Lab 2

Ciclo II 2024

**Grafico de Barras:**

import pandas as pd

import matplotlib.pyplot as plt

# Cargar el dataset

df = pd.read\_csv('world\_economic\_indicators.csv') # Asegúrate de que el archivo esté en la misma carpeta o proporciona la ruta correcta

# Filtrar por año y seleccionar algunas columnas (por ejemplo, países y PIB)

df\_filtered = df[df['Year'] == 2020][['Country', 'GDP']]

# Ordenar los datos por PIB para una mejor visualización

df\_filtered = df\_filtered.sort\_values(by='GDP', ascending=False).head(10)

# Crear gráfico de barras

plt.figure(figsize=(10,6))

plt.bar(df\_filtered['Country'], df\_filtered['GDP'], color='skyblue')

plt.title('Top 10 países por PIB en 2020')

plt.xlabel('País')

plt.ylabel('PIB en miles de millones USD')

plt.xticks(rotation=45)

plt.tight\_layout()

# Mostrar el gráfico

plt.show()

**Explicación del código:**

* **Lectura del dataset**: El archivo CSV del dataset se carga utilizando pandas.
* **Filtrado de datos**: Aquí seleccionamos solo el año 2020 y tomamos las columnas de interés, en este caso, el país y el PIB.
* **Ordenación de los datos**: Los países se ordenan en función del PIB para mostrar solo los 10 primeros.
* **Creación del gráfico**: Se utiliza plt.bar para crear un gráfico de barras que muestre el PIB de cada país.

**Gráfico de Linea:**

# Filtrar los datos para un país específico, por ejemplo, Estados Unidos

df\_usa = df[df['Country'] == 'United States'][['Year', 'GDP']]

# Crear el gráfico de líneas

plt.figure(figsize=(10,6))

plt.plot(df\_usa['Year'], df\_usa['GDP'], marker='o', color='green', linestyle='-')

plt.title('Evolución del PIB de Estados Unidos (1960-2022)')

plt.xlabel('Año')

plt.ylabel('PIB en miles de millones USD')

plt.grid(True)

# Mostrar el gráfico

plt.show()

**Explicación del código del gráfico de líneas:**

* **Filtrado**: Filtramos el dataset para obtener solo los datos de Estados Unidos.
* **Gráfico de líneas**: Utilizamos plt.plot() para crear un gráfico de líneas. El eje X será el año y el eje Y el PIB.
* **Marker y línea**: marker='o' dibuja un marcador en cada punto de datos y linestyle='-' conecta los puntos con una línea.
* **Título y etiquetas**: Similar al gráfico de barras, se añaden un título y etiquetas a los ejes.
* **Grid**: plt.grid(True) añade una cuadrícula para hacer más fácil la lectura del gráfico.

**Grafico Circular:**

# Filtrar los datos relevantes (por ejemplo, tipo de tratamiento y cantidad de pacientes)

df\_treatments = pd.DataFrame({

'Treatment': ['Tratamiento A', 'Tratamiento B', 'Tratamiento C', 'Tratamiento D'],

'Patients': [500, 300, 150, 50]

})

# Crear el gráfico circular

plt.figure(figsize=(7,7))

plt.pie(df\_treatments['Patients'], labels=df\_treatments['Treatment'], autopct='%1.1f%%', colors=['#ff9999','#66b3ff','#99ff99','#ffcc99'])

plt.title('Proporción de Pacientes por Tipo de Tratamiento')

plt.axis('equal') # Para asegurar que el gráfico sea un círculo

# Mostrar el gráfico

plt.show()

**Explicación del código:**

* **Creación de los datos**: Aquí estamos simulando un DataFrame con dos columnas: Treatment para los tipos de tratamientos, y Patients para la cantidad de pacientes que recibieron cada tratamiento.
* **Gráfico circular**: Utilizamos plt.pie() para crear el gráfico. Los argumentos importantes son:
  + labels para especificar las etiquetas (tipos de tratamiento).
  + autopct='%1.1f%%' para mostrar el porcentaje de cada segmento.
  + colors para personalizar los colores del gráfico.
* **plt.axis('equal')**: Esto asegura que el gráfico sea circular, en lugar de una elipse.