

15 DE ENERO DE 2026



TAREA 4.3 - DISEÑO Y ELABORACIÓN DE GRÁFICOS DE VENTAS

Nombre: Daniel García Méndez

Curso: 2ºDAM

Módulo: Desarrollo de interfaces

Unidad: Tema 4

Contenido

TAREA 4.3 - Diseño y elaboración de gráficos de ventas	3
Ejercicio 1: Elección del gráfico adecuado.....	3
Ejercicio 2: Práctica: elaboración del informe con Datapane.....	4
Webgrafía.....	8

TAREA 4.3 - Diseño y elaboración de gráficos de ventas

Contexto

Nos han encargado el diseño de un conjunto de gráficos para incluir en los informes de ventas de una cadena de tiendas de deporte.

Los datos de ventas se encuentran en el fichero:

A partir de estos datos, se deberán analizar las ventas desde distintos puntos de vista, eligiendo el tipo de gráfico más adecuado en cada caso y generando posteriormente un informe Datapane en formato HTML.

La tarea consta de dos partes:

1. **Un ejercicio de análisis.**
2. **Una práctica de programación con Pandas y Datapane.**

Ejercicio 1: Elección del gráfico adecuado

Antes de programar, es necesario reflexionar sobre qué tipo de gráfico es más adecuado para cada situación, desde el punto de vista del diseño de interfaces.

Enunciado

Tenemos que representar la siguiente información:

1. La distribución de las ventas entre los distintos tipos de productos que vende la empresa, para ver cuáles tienen más importancia.

Gráfico adecuado: Gráfico de barras

Justificación: Permite comparar de forma clara el volumen de ventas entre cada tipo de producto. La diferencia de altura entre barras hace muy evidente qué productos tienen más importancia en términos de ventas.

2. La evolución de las ventas totales de la empresa en los últimos 2 años, para comprobar cómo ha evolucionado el negocio.

Gráfico adecuado: Gráfico de líneas

Justificación: Es el más adecuado para representar evolución temporal.

Muestra de forma clara si las ventas suben, bajan o se mantienen entre un año y otro.

3. Las ventas totales en cada uno de los países donde la empresa tiene tiendas, para analizar qué mercados son más importantes.

Gráfico adecuado: Gráfico de barras

Justificación: Facilita la comparación entre países. Cada barra representa un país y permite ver rápidamente qué mercados son más fuertes y cuáles tienen menos ventas.

Tarea

Para cada uno de los casos anteriores, indica:

- **El tipo de gráfico más adecuado de los vistos en clase** (líneas, barras o sectores).
- **Una breve justificación (1–2 frases)** explicando el motivo de la elección.

Ejercicio 2: Práctica: elaboración del informe con Datapane

A partir del fichero **DI_U05_A02_PP_E_01.csv**, deberás crear un programa en Python que genere un informe Datapane con los gráficos solicitados.

El informe se generará como un archivo HTML y deberá abrirse correctamente en el navegador.

Gráficos que debe incluir el informe

El informe debe contener los siguientes gráficos, en este orden:

1. Gráfico de sectores (tarta)

- **Objetivo:** representar la distribución de las ventas totales entre los distintos tipos de producto.
- Los datos deben agruparse por tipo de producto.
- El gráfico debe permitir identificar fácilmente qué productos tienen mayor peso en el total.

Lo que hice en este código fue agrupar por tipo de producto y sumar las ventas de cada tipo con la instrucción señalada en la imagen.

Como el enunciado no pide ninguna etiqueta y leyenda, puse el ylabel vacío con "" y legend = False y por último puse un texto como título del gráfico.

Uso el plt.close() para poder pasar de un gráfico a otro cerrando el actual.

```
# 1. GRÁFICO DE TARTA
ventas_por_tipo = df.groupby("Tipo de producto")["Ventas"].sum()

grafico_tarta = ventas_por_tipo.plot.pie(
    ylabel="",
    legend=False,
    title="Distribución de ventas por tipo de producto"
)

grafico_datapane = dp.Plot(grafico_tarta)
plt.close()
```

2. Gráfico de líneas

- **Objetivo:** mostrar la evolución de las ventas totales de la empresa en los últimos 2 años.
- **Eje horizontal:** años.
- **Eje vertical:** ventas totales por año.

Usé la instrucción groupby señalada en la imagen para agrupar por Año y sumar las ventas de cada año, después usé el .tail(2) para filtrar por los años 2021 y 2020.

Para hacer el gráfico de líneas usé .plot.line para indicar que es un gráfico de líneas y luego puse y, xlabel para insertar un texto en los diferentes ejes seguido de un título encima (title).

Usé también el plt.close() para cerrar el gráfico actual y poder ver el otro.

```
# 2. GRÁFICO DE LÍNEAS (últimos 2 años)
ventas_anuales = df.groupby("Año")["Ventas"].sum()
ventas_de_dos_anios = ventas_anuales.tail(2)

grafico_lineas = ventas_de_dos_anios.plot.line(
    ylabel="Ventas totales",
    xlabel="Año",
    title="Evolución de ventas totales (últimos 2 años)"
)

grafico_lineas_datapane = dp.Plot(grafico_lineas)
plt.close()
```

3. Gráfico de barras

- Objetivo:** representar las ventas totales por país.
- Cada barra debe corresponder a un país.
- El gráfico debe facilitar la comparación entre mercados.

Usé la instrucción groupby para agrupar los datos por Región y sumar las ventas totales de cada una. Después utilicé .plot.bar para indicar que quería un gráfico de barras, y le pasé el parámetro y="Región" para definir el eje vertical, aunque en este caso no es necesario porque ya estás agrupando por región.

Luego envolví el gráfico con dp.Plot() para que Datapane lo pudiera mostrar dentro del informe. También añadí plt.close() para cerrar el gráfico actual y evitar que se mezcle con otros.

Por último, creé un título personalizado usando dp.HTML(...), donde definí el estilo con etiquetas HTML y CSS para que el encabezado del informe se viera centrado, con fondo oscuro y letras blancas.

```
# 3. GRÁFICO DE BARRAS POR REGIÓN
ventas_totales_por_region = df.groupby("Región")["Ventas"].sum()
grafico_matplotlib_Barras = ventas_totales_por_region.plot.bar(y= "Región")

graficoData = dp.Plot(grafico_matplotlib_Barras)
plt.close()

✓ titulo = dp.HTML(
    '<h1 style="font-size:30px; text-align:center; color:#ffffff; background-color:#4d4d4d;">'
    'Informe de ventas '
    '</h1>'
)
```

Explicación del código de informe

Primero usé dp.HTML para crear un título personalizado con estilo, usando etiquetas HTML y CSS para que se vea centrado, con fondo oscuro y letras blancas. Ese título lo guardé en una variable llamada título.

Después monté el informe completo con dp.Report, donde fui añadiendo cada parte por orden:

- El título principal (título)
- Un texto explicativo con dp.Text para la distribución de ventas por tipo de producto
- El gráfico de tarta (grafico_datapane)
- Otro texto para la evolución de ventas en los últimos 2 años
- El gráfico de líneas (grafico_lineas_datapane)
- Un tercer texto para mostrar las ventas por regiones
- Y el gráfico de barras (graficoData)

Al final usé reporte.savepara guardar el informe como archivo HTML y que se abra automáticamente al terminar.

```
# 3. INFORME
reporte = dp.Report(
    titulo,
    dp.Text("# Distribución de ventas por tipo de producto"),
    grafico_datapane,
    dp.Text("# Evolución de ventas en los últimos 2 años"),
    grafico_lineas_datapane,
    dp.Text("# Evolución de ventas en las diferentes regiones"),
    graficoData
)

reporte.save("informe_ventas_por_tipo.html", open=True)
```

Webgrafía

<https://experts-denry-b9a.craft.me/5EVzmnvFZAgHAP>

<https://docs.datapane.com/>