

Banjercito

Diplomado en Ciencia de Datos

Guía de Estudio

Evaluación Final

Alan Badillo Salas

Diciembre 2024

Introducción

El Diplomado en Ciencia de Datos ha llegado a su etapa final, dentro del diplomado hemos aprendido a lo largo de 5 módulos, el uso de las herramientas más importantes usadas en el campo de la ciencia de datos. En el primer módulo hemos introducido el análisis financiero a través de Excel, Power BI y Python, en el segundo módulo hemos reunido los conceptos más importantes de probabilidad y estadística, en el tercer módulo hemos trabajado más a detalle con las librerías de Numpy y Pandas y el flujo de la ciencia de datos, en el cuarto módulo hemos aprendido las tres piezas fundamentales del Machine Learning, que son la Regresión, la Clasificación y la Clusterización, así como la simulación estadística y probabilística, finalmente, en el quinto módulo hemos generalizado todos los modelos, para que una Red Neuronal pueda hacer los pronósticos generalizados a través del Deep Learning.

En esta guía repasaremos los fundamentos de cada módulo, a través de preguntas y reactivos similares a los de la evaluación final del diplomado, cubriendo los conceptos vistos en cada módulo y detallando su solución. Comprender esta guía debería ser suficiente para tener un buen desempeño como Científico de Datos y estar preparado para problemas del mercado actual y aplicaciones reales en la industria y el gobierno.

Módulo I — Introducción a Python con Finanzas

Problema 1 — Uso de Excel

Una empresa financiera posee una tabla con los montos de créditos que ha aprobado a sus clientes. La tabla posee el género de cada cliente y se desea calcular el monto total de los créditos aprobados para hombres y mujeres.

En la Figura 1 se muestra la tabla con el cliente, género, crédito y monto aprobado.

	A	B	C	D
1	Cliente	Género	Crédito	Monto
2	C001	Hombre	P0001	\$ 14,600.00
3	C002	Mujer	P0002	\$ 18,900.00
4	C003	Hombre	E0004	\$ 12,000.00
5	C004	Mujer	E0032	\$ 6,900.00
6	C005	Mujer	F0005	\$ 3,200.00
7	C006	Hombre	G0128	\$ 23,700.00

Figura 1: Créditos aprobados a clientes por género

(a) Calcule la suma de los montos para hombres y la suma de montos para mujeres usando la fórmula `=SUMA.SI(RANGO CRITERIO, CRITERIO, RANGO SUMA)`.

(b) Calcule la suma de los montos para hombres y la suma de montos para mujeres usando la fórmula `=SUMA(FILTRAR(RANGO SUMA, RANGO INCLUIR))`.

Para el inciso (a) utilizamos las fórmulas

=SUMAR.SI(B2:B7,"MUJER",D2:D7)

Figura 2: Solución al Problema 1 inciso (a)

=SUMA (FILTRAR (D2:D7, B2:B7="MUJER"))

Figura 3: Solución al Problema 1 inciso (b)

=FILTRAR(A2:D7,B2:B7="HOMBRE")

=FILTRAR(A2:D7,B2:B7="HOMBRE")

Se generará la tabla virtual con todos los créditos correspondientes a las filas de los hombres o de las mujeres.

G8 fx =FILTRAR(A2:D7,B2:B7=F3)										
	A	B	C	D	E	F	G	H	I	J
1	Cliente	Género	Crédito	Monto			USANDO CONTAR.SI Y SUMAR.SI			
2	C001	Hombre	P0001	\$ 14,600.00			Total	Suma Monto		
3	C002	Mujer	P0002	\$ 18,900.00		Hombre	3	\$ 50,300.00		
4	C003	Hombre	E0004	\$ 12,000.00		Mujer	3	\$ 29,000.00		
5	C004	Mujer	E0032	\$ 6,900.00						
6	C005	Mujer	F0005	\$ 3,200.00						
7	C006	Hombre	G0128	\$ 23,700.00			USANDO FILTRAR			
8						Hombres	C001	Hombre	P0001	\$14,600.00
9							C003	Hombre	E0004	\$12,000.00
10							C006	Hombre	G0128	\$23,700.00
11										\$50,300.00
12										

fx	=FILTRAR(A2:D7,B2:B7=F3)									
-----------	---------------------------------	--	--	--	--	--	--	--	--	--

Hombres	C001	Hombre	P0001	\$14,600.00
	C003	Hombre	E0004	\$12,000.00
	C006	Hombre	G0128	\$23,700.00
				\$50,300.00

Figura 4: Tabla virtual generada por FILTRAR (c)

Problema 2 — Uso de Tablas dinámicas en Excel

Una empresa financiera posee una tabla con los montos de créditos que ha aprobado a sus clientes y su deuda actual. La tabla posee el género de cada cliente y se desea calcular la deuda total y deuda promedio.

En la Figura 5 se muestra la tabla con el cliente, género, crédito, monto aprobado y deuda actual.









	A	B	C	D	E
1	Cliente	Género	Crédito	Monto	Deuda
2	C001	Hombre	P0001	 \$ 14,600.00	\$ 3,200.00
3	C001	Hombre	H0014	 \$ 16,300.00	\$ 2,500.00
4	C002	Mujer	P0002	 \$ 18,900.00	\$ 1,600.00
5	C003	Hombre	E0004	 \$ 12,000.00	\$ 4,800.00
6	C004	Mujer	E0032	 \$ 6,900.00	\$ 5,300.00
7	C005	Mujer	F0005	 \$ 3,200.00	\$ 2,100.00
8	C006	Hombre	G0128	 \$ 15,000.00	\$ 1,900.00
9	C005	Mujer	F0095	 \$ 12,500.00	\$ 3,500.00

Figura 5: Créditos aprobados con deuda actual a clientes por género

- Crea una tabla dinámica de A1:E9.
- Segmenta las filas por Género y luego por Cliente.
- Cuenta el número de Créditos en valores.
- Calcula la suma del Monto en valores.
- Calcula la suma de la Deuda en valores.
- Calcula el promedio del Monto en valores.
- Calcula el promedio de la Deuda en valores.
- ¿Quién es el cliente de mayor deuda?
- ¿Quién es el cliente al que se le ha prestado más en promedio?
- ¿Quién es el cliente hombre con mayor deuda promedio?
- ¿Es significativamente distinta la deuda entre hombres y mujeres?
- ¿Cuánto se ha recuperado en total de todos los créditos?

Solución al Problema 2

Para el inciso (a) creamos la tabla dinámica para el rango A1:E9 y la insertamos en una celda de la misma hoja.

The screenshot shows the Excel interface with the 'Insertar' (Insert) tab active. A PivotTable is being created from the data range A1:E9. The data is as follows:

	A	B	C	D	E
1	Cliente	Género	Crédito	Monto	Deuda
2	C001	Hombre	P0001	\$ 14,600.00	\$ 3,200.00
3	C001	Hombre	H0014	\$ 16,300.00	\$ 2,500.00
4	C002	Mujer	P0002	\$ 18,900.00	\$ 1,600.00
5	C003	Hombre	E0004	\$ 12,000.00	\$ 4,800.00
6	C004	Mujer	E0032	\$ 6,900.00	\$ 5,300.00
7	C005	Mujer	F0005	\$ 3,200.00	\$ 2,100.00
8	C006	Hombre	G0128	\$ 15,000.00	\$ 1,900.00
9	C005	Mujer	F0095	\$ 12,500.00	\$ 3,500.00

The 'Crear tabla dinámica' dialog box is open, showing the following settings:

- Selecciona los datos que desees analizar:**
 - ☒ Seleccionar una tabla o rango
 - Tabla o rango: Hoja1!\$A\$1:\$E\$9
 - ☐ Usar un origen de datos externo
 - Elegir conexión... No se recuperó ningún dato.
- Selecciona dónde deseas ubicar la tabla dinámica:**
 - ☐ Hoja de cálculo nueva
 - ☒ Hoja de cálculo existente
 - Tabla o rango: Hoja1!\$A\$11

The 'Aceptar' (Accept) button is highlighted.

Figura 6: Solución al Problema 2 inciso (a)

Para el inciso (b) arrastramos los campos del género y cliente a las filas

Campos de tabla dinámica

Formato de forma

NOMBRE DE CAMPO

Buscar en campo:

☒ Cliente
☒ Género
☒ Crédito
☒ Monto
☒ Deuda

Filtros

Columnas

: Valores

Filas

: Género
 : Cliente

Valores

: Cuenta de Crédito
 : Suma de Monto
 : Suma de Deuda
 : Promedio de Monto
 : Promedio de Deuda

Figura 7: Solución al Problema 2 inciso (b)

Para el inciso (c), (d), (e), (f), (g) arrastramos los campos como valores y configuramos cada campo si es recuento, suma o promedio

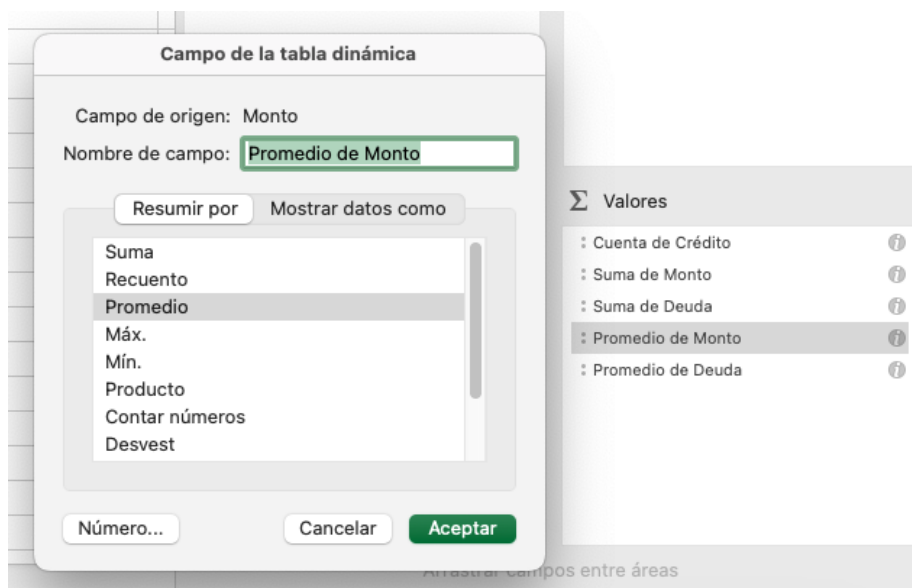


Figura 8: Solución al Problema 2 incisos (c), (d), (e), (f), (g)

Para el inciso (h), (i), (j), (k), (l) observamos los valores en la tabla dinámica

Etiquetas de fila	Cuenta de Crédito	Suma de Monto	Suma de Deuda	Promedio de Monto	Promedio de Deuda
Hombre	4	\$ 57,900.00	\$ 12,400.00	\$ 14,475.00	\$ 3,100.00
C001	2	\$ 30,900.00	\$ 5,700.00	\$ 15,450.00	\$ 2,850.00
C003	1	\$ 12,000.00	\$ 4,800.00	\$ 12,000.00	\$ 4,800.00
C006	1	\$ 15,000.00	\$ 1,900.00	\$ 15,000.00	\$ 1,900.00
Mujer	4	\$ 41,500.00	\$ 12,500.00	\$ 10,375.00	\$ 3,125.00
C002	1	\$ 18,900.00	\$ 1,600.00	\$ 18,900.00	\$ 1,600.00
C004	1	\$ 6,900.00	\$ 5,300.00	\$ 6,900.00	\$ 5,300.00
C005	2	\$ 15,700.00	\$ 5,600.00	\$ 7,850.00	\$ 2,800.00
Total general	8	\$ 99,400.00	\$ 24,900.00	\$ 12,425.00	\$ 3,112.50

Figura 9: Solución al Problema 2 incisos (h), (i), (j), (k), (l)

Respuestas: (h) C001 \$5,700, (i) C002 \$18,900, (j) C003 \$4,800, (k) No, la diferencia es de \$100 menor al 1% de diferencia, (l) $\$74,500 = \$99,400 - \$24,900$.

Problema 3 — Uso de Power BI

Una empresa financiera posee una tabla con los préstamos que ha aprobado a sus clientes indicando el monto, el estado y el valor. En cada préstamo se reporta el estado de sobre lo que se ha pagado del préstamo, lo que se debió haber cubierto actualmente (deuda) y lo que se debería cubrir en el futuro (pendiente). La financiera requiere una tabla que resuma de cada cliente sus préstamos y en cada préstamo se desglose la suma de cuánto adeuda, cuánto ha pagado y cuánto está pendiente, así como el total. También requiere un indicador que muestre la diferencia entre la suma del monto del préstamo y el valor del monto pagado, para saber cuánto se adeuda en total del préstamo, igualmente desglosado por cliente y préstamo. Finalmente la financiera requiere dos reportes visuales, un anillo que muestre el estado de los préstamos (pagado, pendiente y deuda) segmentado por cliente. Y un esquema jerárquico que permita desglosar el valor del préstamo por su estado, cliente y número de préstamo.

En la Figura 10 se muestra la tabla de préstamos, con la columna del cliente, el número de préstamo, el monto, el estado (pagado, deuda, pendiente) y el valor asociado al estado.

	ABC 123 Cliente	ABC 123 Préstamo	\$ Monto	ABC 123 Estado	\$ Valor	+
1	C001	1	15000	Pagado	7000	
2	C001	1	15000	Deuda	2000	
3	C001	1	15000	Pendiente	6000	
4	C001	2	12000	Pagado	2000	
5	C001	2	12000	Deuda	0	
6	C001	2	12000	Pendiente	10000	
7	C002	1	18000	Pagado	8000	
8	C002	1	18000	Deuda	4000	
9	C002	1	18000	Pendiente	6000	
10	C002	2	9000	Pagado	3000	
11	C002	2	9000	Deuda	6000	
12	C002	2	9000	Pendiente	0	
13	C003	1	23000	Pagado	12000	
14	C003	1	23000	Deuda	5000	
15	C003	1	23000	Pendiente	6000	
+						

Figura 10: Préstamos aprobados a clientes desglosado por cliente, número de préstamo, estado (pagado, deuda y pendiente) y valor asociado al estado

(a) Crea una matriz que reporte las filas para el cliente y número de préstamo en ese orden, el estado como columna y la suma del valor en los valores.

(b) En el modelo de datos crea una nueva medida con la fórmula

Diferencia = SUM(P103[Monto])-SUM(P103[Valor])

para reportar la diferencia entre la suma del monto menos la suma del valor asociado al estado. Donde *P103* es el nombre de la tabla creada al inicio.

(c) Crea una matriz que reporte las filas para el cliente y número de préstamo en ese orden, el estado como columna y la suma del monto, suma del valor y suma de diferencia en los valores en ese orden. Reporta solo el estado Pagado en los filtros del objeto visual.

(d) Crea una gráfica de anillo cuya leyenda este dada por el estado del préstamo, en valores reporte la suma del valor asociado al estado, y en detalles segmente al cliente.

(e) Crea un esquema jerárquico para analizar la suma del valor valor asociado al estado, explicado por el estado del préstamo, el cliente y el número de préstamo en ese orden.

(f) De la matriz del inciso (a) responde si cada desglose corresponde al monto reportado en cada préstamo.

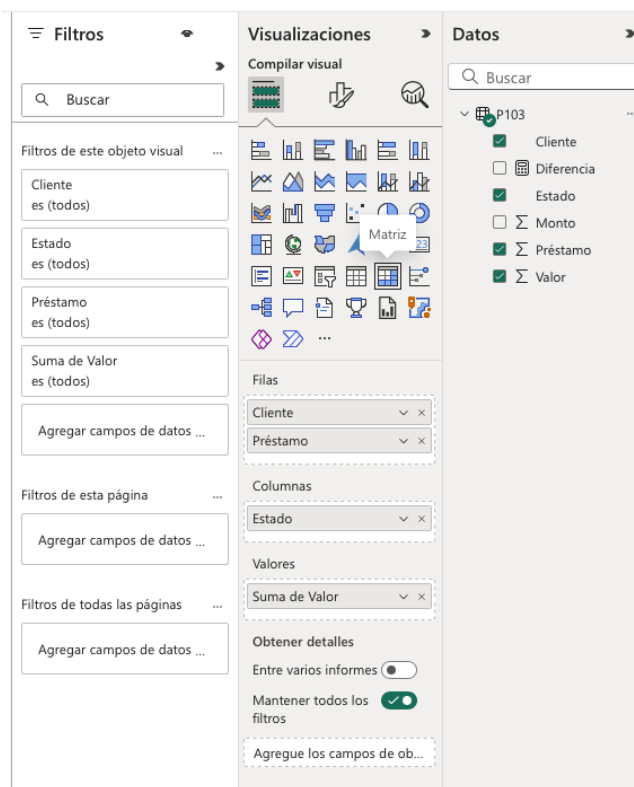
(g) De la matriz del inciso (b) responde cuál el es el préstamo con menor deuda total y usando el inciso (a) cómo es su deuda respecto al préstamo.

(h) Del anillo identifica al cliente de mayor deuda y el cliente con mayor valor pendiente.

(i) Del esquema jerárquico identifica al cliente de mayor deuda y el número de préstamo de mayor deuda para ese cliente.

Solución al Problema 3

Para el inciso (a) creamos una matriz que reporte las filas para el cliente y número de préstamo en ese orden, el estado como columna y la suma del valor en los valores.



Cliente	Deuda	Pagado	Pendiente	Total
C001	2000	9000	16000	27000
1	2000	7000	6000	15000
2	0	2000	10000	12000
C002	10000	11000	6000	27000
1	4000	8000	6000	18000
2	6000	3000	0	9000
C003	5000	12000	6000	23000
1	5000	12000	6000	23000
Total	17000	32000	28000	77000

Figura 11: Solución al Problema 3 inciso (a)

Para el inciso (b) abrimos el modelo de datos y dentro del modelo de datos creamos una nueva medida con la fórmula para calcular la diferencia.

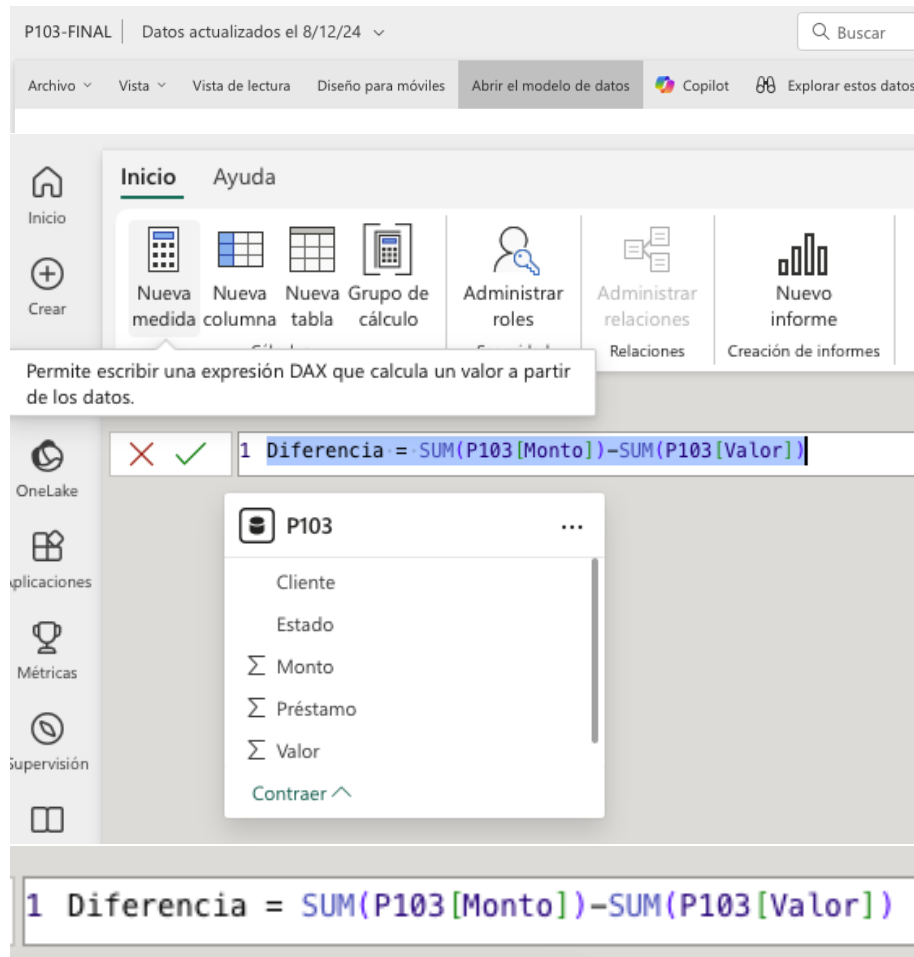


Figura 12: Solución al Problema 3 inciso (b)

Para el inciso (c) creamos una matriz que reporte las filas para el cliente y número de préstamo en ese orden, el estado como columna y la suma del monto, suma del valor y suma de diferencia en los valores en ese orden. Reporta solo el estado Pagado en los filtros del objeto visual.

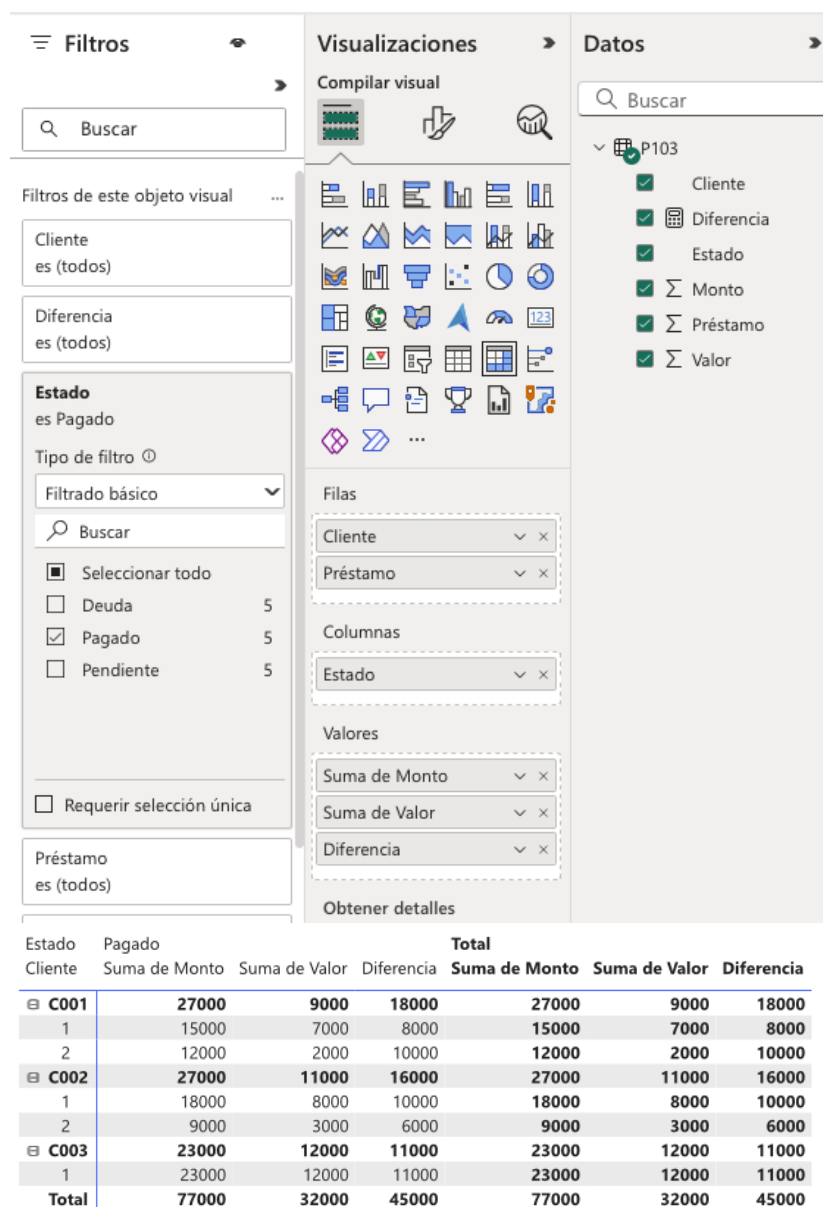
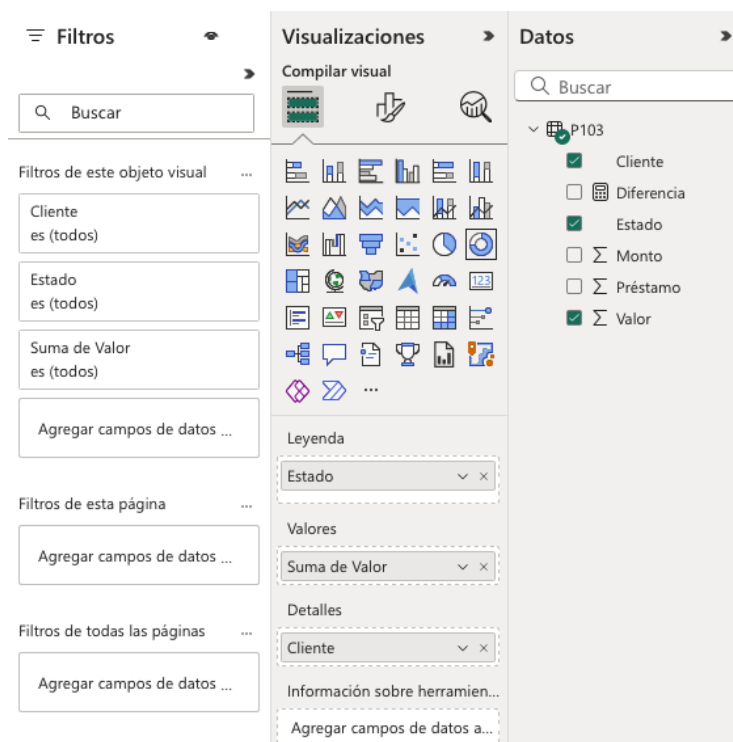


Figura 13: Solución al Problema 3 inciso (c)

Para el inciso (d) creamos una gráfica de anillo cuya leyenda este dada por el estado del préstamo, en valores reporte la suma del valor asociado al estado, y en detalles segmente al cliente.



Suma de Valor por Estado y Cliente

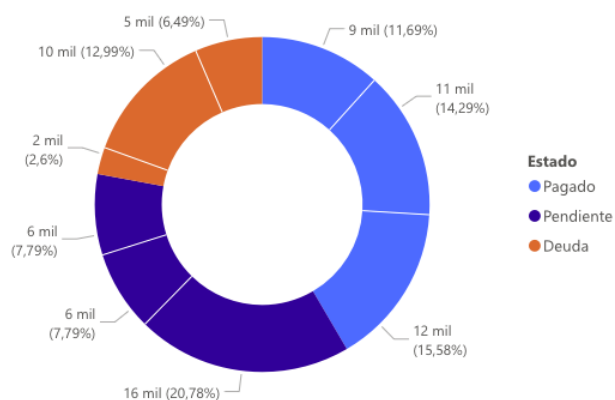


Figura 14: Solución al Problema 3 inciso (d)

Para el inciso (e) creamos un esquema jerárquico para analizar la suma del valor valor asociado al estado, explicado por el estado del préstamo, el cliente y el número de préstamo en ese orden.

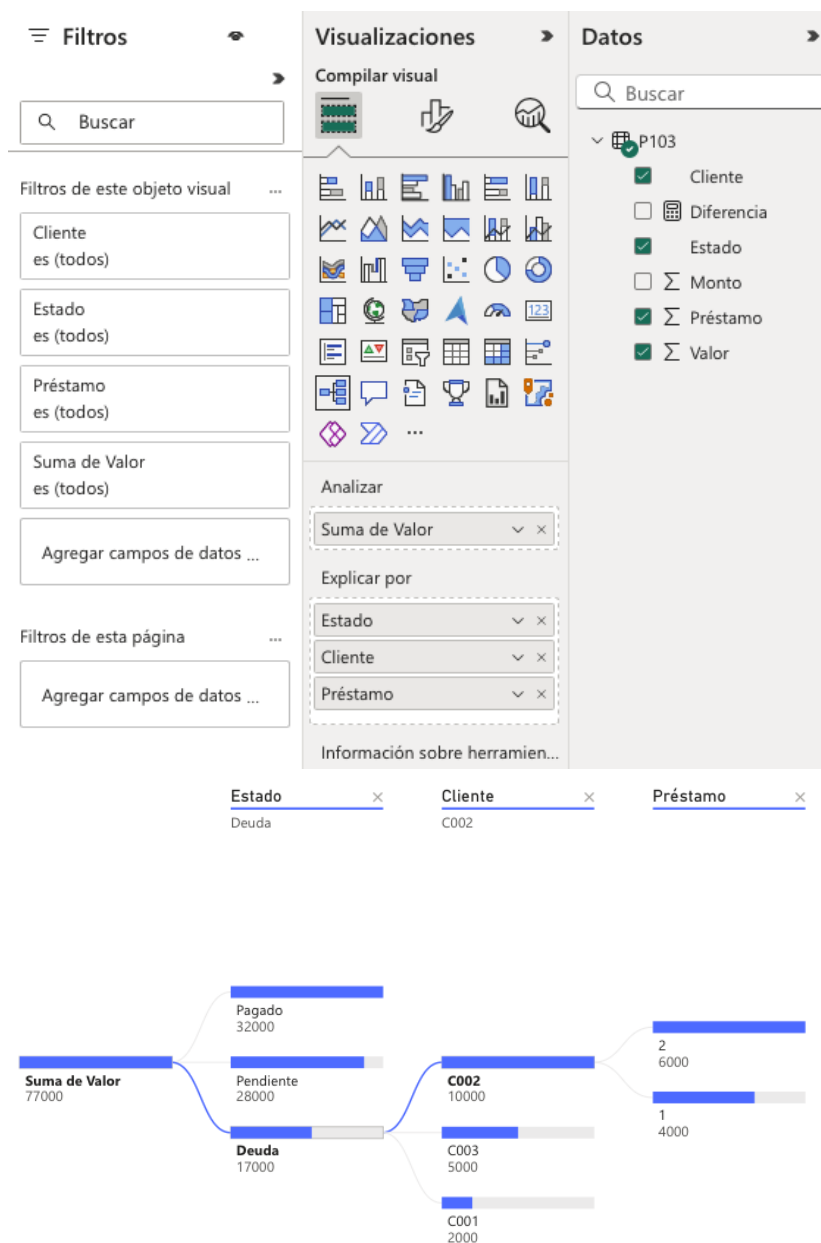


Figura 15: Solución al Problema 3 inciso (e)

Para los otros incisos las respuestas son: (f) Sí, la suma de montos en cada préstamo coincide con la suma de su deuda, lo pagado y lo pendiente. (g) Para el cliente *C002* en el préstamo número 2 la diferencia total es de **\$6,000** y es el préstamo de menor deuda, ya que lo pendiente para ese préstamo es de **\$0**, sin embargo, respecto a los otros préstamos es el de mayor deuda. (h) **C002 \$10,000 12.99%** es el cliente de mayor deuda y **C001 \$16,000 20.78%** es el cliente de mayor valor pendiente. (i) **C002 \$10,000 2 \$10,000** es el cliente de mayor deuda, con su préstamo número dos.

Problema 4 — Uso de Python

Un banco ha decidido usar Python en un ambiente controlado para poder hacer operaciones y cálculos de riesgo, para ello ha instalado un ambiente virtual cerrado en computadoras sin acceso a internet para que no se filtren los conjuntos de datos utilizados, ni los reportes generados.

El banco comenzará por automatizar una serie de funciones útiles que pueda aplicar para cálculos simples, para de esta forma crear una metodología de trabajo cuando deseen incrementar el volumen de las operaciones.

Para ello, el banco requiere que se generen los siguientes reportes automatizados para saber que todo está funcionando correctamente.

Implementa las funciones que se detallan a continuación y resuelve los incisos siguientes. Completa el código que haga falta para que las funciones operen correctamente siguiendo las pistas de los comentarios

```
def reporteTitulo(codigo, leyenda):
    print(f"Banco Nacional - Reporte {codigo} / {leyenda}")

def reporteCorte(codigo, leyenda):
    print(f"----- {codigo} / {leyenda} -----")

def reporteFin(codigo, leyenda):
    print(f"----- {codigo} / {leyenda} -----")

def reporteBalanceDiaInicio(balance):
    reporteTitulo("BIN", "Balance al Inicio del Día")
    print(f"${balance:.2f}")
    reporteFin("BIN", "Balance al Inicio del Día")

def reporteBalanceDiaFin(balance):
    reporteTitulo("B-OUT", "Balance al Final del Día")
    print(f"${balance:.2f}")
    reporteFin("B-OUT", "Balance al Final del Día")

def reporteTotalOperacionesDia(numOperaciones):
    reporteTitulo("T-OP", "Total de Operaciones del Día")
    # Imprime el número de operaciones
    reporteFin("T-OP", "Total de Operaciones del Día")

def reporteTotalOperacionesRelativasMes(numOperacionesDia, numOperacionesMes):
    reporteTitulo("T-OR", "Tasa de Operaciones del día respecto al mes")
    tasa = 100 * numOperacionesDia / numOperacionesMes
    # Imprime el número de operaciones del día
    # luego una diagonal y el número de operaciones del mes
    # Haz un corte del reporte
    # Imprime la tasa de operaciones del día respecto al mes a un dígito
    # Ejemplo: 41/200 (20.5%)
    reporteFin("T-OR", "Tasa de Operaciones del día respecto al mes")
```

- (a) Reporta el balance de inicio del día con **\$ 185,000.32**
- (b) Reporta un total de **247** operaciones del día
- (c) Reporta un total de **247** de **192,316** operaciones relativas del mes
- (d) Reporta el balance de fin del día con **\$ 232,827.71**

Solución al Problema 4

En este problema se presetan una serie de funciones que hacen impresiones en la pantalla con los parámetros que se les mande. Las funciones son sencillas y solo reciben parámetros de texto o números para imprimir en la pantalla algunos cortes.

En la Figura 16 se muestra el código completado donde hemos colocado los `print` faltantes para que se muestren los reportes correctamente.

```
def reporteTitulo(codigo, leyenda):
    print(f"Banco Nacional - Reporte {codigo} / {leyenda}")

def reporteCorte(codigo, leyenda):
    print(f"----- {codigo} / {leyenda} -----")

def reporteFin(codigo, leyenda):
    print(f"===== {codigo} / {leyenda} =====")

def reporteBalanceDiaInicio(balance):
    reporteTitulo("BIN", "Balance al Inicio del Día")
    print(f"${balance:.2f}")
    reporteFin("BIN", "Balance al Inicio del Día")

def reporteBalanceDiaFin(balance):
    reporteTitulo("B-OUT", "Balance al Final del Día")
    print(f"${balance:.2f}")
    reporteFin("B-OUT", "Balance al Final del Día")

def reporteTotalOperacionesDia(numOperaciones):
    reporteTitulo("T-OP", "Total de Operaciones del Día")
    # Imprime el número de operaciones
    print(f"{numOperaciones}")
    reporteFin("T-OP", "Total de Operaciones del Día")

def reporteTotalOperacionesRelativasMes(numOperacionesDia, numOperacionesMes):
    reporteTitulo("T-OR", "Tasa de Operaciones del día respecto al mes")
    tasa = 100 * numOperacionesDia / numOperacionesMes
    # Imprime el número de operaciones del día
    # luego una diagonal y el número de operaciones del mes
    print(f"{numOperacionesDia} / {numOperacionesMes}")
    # Haz un corte del reporte
    reporteCorte("T-OR", "-")
    # Imprime la tasa de operaciones del día respecto al mes a un dígito
    print(f"{tasa:.1f}%")
    # Ejemplo: 41/200 (20.5%)
    reporteFin("T-OR", "Tasa de Operaciones del día respecto al mes")
```

Figura 16: Código solucionado, se agregan impresiones faltantes

En la Figura 17 se muestra el código para resolver los incisos (a), (b), (c) y (d) que únicamente solicitan llamar a las funciones con los valores establecidos en cada inciso.

```

reporteBalanceDiaInicio(185000.32)
print()
reporteTotalOperacionesDia(247)
print()
reporteTotalOperacionesRelativasMes(247, 192316)
print()
reporteBalanceDiaFin(232827.71)

```

Banco Nacional - Reporte BIN / Balance al Inicio del Día
 \$ 185000.32
 ===== BIN / Balance al Inicio del Día =====

Banco Nacional - Reporte T-OP / Total de Operaciones del Día
 247
 ===== T-OP / Total de Operaciones del Día =====

Banco Nacional - Reporte T-OR / Tasa de Operaciones del día respecto al mes
 247 / 192316
 ----- T-OR / - -----
 0.1%
 ===== T-OR / Tasa de Operaciones del día respecto al mes =====

Banco Nacional - Reporte B-OUT / Balance al Final del Día
 \$ 232827.71
 ===== B-OUT / Balance al Final del Día =====

Figura 17: Código para resolver los incisos

Problema 5 — Interés Simple

Una financiera posee montos de préstamos a proyectos y la tasa anual fija que le cobrará sus clientes. La financiera requiere calcular la suma de montos y proponer una misma tasa de interés anual única para todos los proyectos, así como encontrar la tasa anual en caso de perder el monto por algún proyecto.

En la Figura 18 se muestran los montos y la tasa anual de los montos de préstamo destinados a proyectos financiados.

	A	B
1	Monto	Tasa Anual
2	\$ 16,300.00	8.0%
3	\$ 15,200.00	13.0%
4	\$ 25,400.00	6.5%
5	\$ 184,900.00	8.3%
6	\$ 192,000.00	7.1%

Figura 18: Montos de préstamos a proyectos de una financiera y su tasa anual

- (a) Calcula el interés simple generado por la tasa anual y reporta cada interés y la suma de todos estos intereses.
- (b) Calcula la tasa anual promedio y de todas las tasas y la suma de todos los montos y calcula el interés de la suma total con la tasa anual promedio. ¿Este interés es el mismo que el de la suma del inciso (a)?
- (c) ¿Cuál debería ser la tasa anual de todos los montos para que la suma del interés sea la misma?
- (d) ¿Cuál debería ser la tasa anual de todos los montos, si no existiera el monto de \$ 25,400.00 para que la suma del interés sea la misma a las anteriores?

Solución al Problema 5

En este problema nos muestran una tabla de montos y tasas anuales y debemos calcular el interés de cada monto según su tasa y la suma de estos intereses.

En la Figura 19 se muestra el código que define los montos y tasas como listas, luego inicializa un total que sumará los intereses comenzando en cero, y luego se recorre cada pareja de monto y tasa usando la función de agrupado o embolsado llamada `zip`. Luego, dentro del iterador, calculamos los intereses del monto, según su tasa y los acumulamos en el total, finalmente se hacen las impresiones necesarias para reportar el monto, su tasa, el interés y al final la suma de todos los intereses.

```
montos = [16300, 15200, 25400, 184900, 192000]
tasas = [8, 13, 6.5, 8.3, 7.1]

total = 0

print("Monto          Tasa A.   Interés Simple")

print("-" * 40)

# Recorrer montos y tasas por fusión zip (embolsado)
for monto, tasa in zip(montos, tasas):
    interes = monto * tasa/100
    total = total + interes
    print(f"${monto:10.2f}    ({tasa:4.1f}%)    ${interes:9.2f}")

print("-" * 40)

print(" " * 15 + f"Total:    " + f"${total:9.2f}")
```

Monto	Tasa A.	Interés Simple
\$ 16300.00	(8.0%)	\$ 1304.00
\$ 15200.00	(13.0%)	\$ 1976.00
\$ 25400.00	(6.5%)	\$ 1651.00
\$ 184900.00	(8.3%)	\$ 15346.70
\$ 192000.00	(7.1%)	\$ 13632.00
Total:		\$ 33909.70

Figura 19: Código para reportar el interés de montos y tasas con su total final, resultado de la suma de intereses

En la Figura 20 se muestra el código que resuelve los incisos (b), (c) y (d). Observa que al promediar la suma de intereses no es igual y que podemos calcular la tasa promedio analíticamente.

```
tasa_prom = sum(tasas) / len(tasas)
```

```
tasa_prom
```

```
8.58
```

```
montos_suma = sum(montos)
```

```
montos_suma
```

```
433800
```

```
montos_suma * tasa_prom/100
```

```
37220.04
```

```
# Tasa fija que generen el mismo interés  
# para la suma de todos los montos
```

```
#  $m_1 * x + m_2 * x + \dots + m_5 * x = 33909.70$ 
```

```
x = 33909.70 / sum(montos)
```

```
x * 100
```

```
7.8168971876440745
```

```
# Tasa fija que generen el mismo interés  
# para la suma de todos los montos (modificados)
```

```
montos = [16300, 15200, 184900, 192000]
```

```
100 * 33909.70 / sum(montos)
```

```
8.303060724779627
```

Figura 20: Código para resolver los incisos

Problema 6 — Interés Compuesto

Una financiera desea calcular el interés compuesto que generaría un monto de \$ **4,312,939.83** con una tasa anual de **8%**, si se tuviera que pagar mensualmente a una tasa compuesta de **0.67%** y si tuviera que pagar a tasa compuesta de **0.34%**.

- (a) Calcula el interés compuesto a 12 meses con una tasa mensual de **0.67%**.
- (b) Calcula el interés compuesto a 24 quincenas con una tasa quincenal de **0.34%**

Solución al Problema 6

En este problema nos pide calcular el interés compuesto al 12 meses y 24 quincenas para compararlo con el interés simple. El interés calcula los intereses periodo por periodo y los va acumulando en un nuevo monto sobre el que se calculará el nuevo interés. Este procedimiento es importante para algunas financieras que contemplan incrementos al monto por pagos tardíos o penalizaciones haciendo que se recupere el riesgo ocurrido en cada periodo sin tener que fijar un único interés sobre todo el el monto inicial.

En la Figura 21 se muestra el código para calcular el interés simple del monto respecto a la tasa anual, y luego el código para ir calculando mes por mes el interés compuesto. El código se puede adaptar fácilmente para calcular los intereses quincenales en lugar de los mensuales, considerando ahora 24 periodos. Observa como se define una variable que guarde el monto acumulado y como se calcula el interés mes por mes para ese monto acumulado. También se usa `mes + 1` porque `range(12)` genera 12 iteraciones desde 0 hasta 11.

```
monto = 4312939.83

interesSimpleAnual = monto * 8/100

interesSimpleAnual

345035.1864

tasaMensual = 0.67/100
interesCompuestoMensual = 0
montoAcumulado = monto

for mes in range(12):
    interesMes = montoAcumulado * tasaMensual
    print(f"Mes: {mes + 1:2} I:$ {montoAcumulado:10.2f}      ($ {interesMes:8.2f})")
    montoAcumulado = montoAcumulado + interesMes
    print(f"      F:$ {montoAcumulado:10.2f}")
    print("-" * 40)

interesFinal = montoAcumulado - monto

print(f"Interés final: $ {interesFinal:8.2f}")
```

Figura 21: Código para generar los intereses mensuales compuestos

En la Figura 22 se muestra el reporte generado por el código, mostrando cual es el monto acumulado al inicio y al final y el interes que se calcula. Además se muestra al final el último interés calculado que es el interés compuesto final.

Mes: 1	I:\$ 4312939.83	(\$ 28896.70)
	F:\$ 4341836.53	
Mes: 2	I:\$ 4341836.53	(\$ 29090.30)
	F:\$ 4370926.83	
Mes: 3	I:\$ 4370926.83	(\$ 29285.21)
	F:\$ 4400212.04	
Mes: 4	I:\$ 4400212.04	(\$ 29481.42)
	F:\$ 4429693.46	
Mes: 5	I:\$ 4429693.46	(\$ 29678.95)
	F:\$ 4459372.41	
Mes: 6	I:\$ 4459372.41	(\$ 29877.80)
	F:\$ 4489250.20	
Mes: 7	I:\$ 4489250.20	(\$ 30077.98)
	F:\$ 4519328.18	
Mes: 8	I:\$ 4519328.18	(\$ 30279.50)
	F:\$ 4549607.68	
Mes: 9	I:\$ 4549607.68	(\$ 30482.37)
	F:\$ 4580090.05	
Mes: 10	I:\$ 4580090.05	(\$ 30686.60)
	F:\$ 4610776.65	
Mes: 11	I:\$ 4610776.65	(\$ 30892.20)
	F:\$ 4641668.86	
Mes: 12	I:\$ 4641668.86	(\$ 31099.18)
	F:\$ 4672768.04	
Interés final: \$ 359828.21		

Figura 22: Reporte de intereses acumulados en 12 meses

En la Figura 23 se muestra el código para generar de forma analítica el interés compuesto a 12 meses. Luego se muestra la solución al inciso (b) ya sin generar la tabla, solo reportar el interés compuesto para las 24 quincenas.

```
monto = 4312939.83
tasaMensual = 0.67/100
periodos = 12

monto * (1 + tasaMensual)**periodos - monto

359828.2082369188
```

```
monto = 4312939.83
tasaQuincenal = 0.34/100
periodos = 24

monto * (1 + tasaQuincenal)**periodos - monto

366045.8919148287
```

Figura 23: Intereses acumulados en 12 meses y en 24 quincenas

Problema 7 — Tasa de Interés Variable

Una financiera desea calcular el interés que generaría un monto de **\$ 2,617,800.01** con una tasa mensual variable que comienza en **6%** el primer mes y se actualiza a **5%** para el segundo mes, luego **4%** al tercer mes, **3%** para el cuarto mes y luego **2%** y **1%** para el quinto y sexto mes, luego de ahí se aplica el **0.5%** en los meses restantes.

- (a) Calcula el interés compuesto para los 12 meses usando el interés variable.
- (b) Si se usara un interés anual fijo, ¿De cuánto sería?

Solución al Problema 7

En este problema nos pide calcular el interés variable donde mes con más el interés varía según alguna regla establecida, por ejemplo, para que los primeros meses el interés sea alto y luego se regule. Esto ayuda a financieras y prestamistas para que recuperen rápidamente el capital invertido en los primeros meses y luego se arriegen a que el cliente que ya no pague no genere una pérdida fuerte para la empresa tras quedarse el bien adquirido a meses.

En la Figura 24 se muestra el código para calcular el interés variable que se va acumulando, muy similar al problema del interés acumulado, pero ahora considerando la regla de variación del interés dada por el contador y la condición especial para que a partir del mes 7 equivalente a que el contador llegue a cero, de que el interés se regule.

```
monto = 217800.01

interesCompuestoMensual = 0
montoAcumulado = monto

contador = 6

for mes in range(12):
    tasaMensual = contador / 100

    # Simulamos una sanción de $3,000 en el mes 3
    #if mes == 2:
    # montoAcumulado = montoAcumulado + 3000

    interesMes = montoAcumulado * tasaMensual
    print(f"Mes: {mes + 1:2} I:$ {montoAcumulado:10.2f}    ($ {interesMes:8.2f} / {tasaMensual * 100:.1f}%)")
    montoAcumulado = montoAcumulado + interesMes
    print(f"      F:$ {montoAcumulado:10.2f}")
    print("-" * 40)

    contador = contador - 1

    if contador < 1:
        contador = 0.5

interesFinal = montoAcumulado - monto

print(f"Interés final: $ {interesFinal:8.2f} / ({interesFinal/monto * 100:.1f}%")
```

Figura 24: Código para generar los intereses variables compuestos mensuales

En la Figura 25 se muestra el reporte generado por el código, mostrando el monto acumulado y el interés que va variando mes con mes hasta regularizarse en los últimos meses como lo pide el problema.

Mes:	1	I:\$	217800.01	(\$ 13068.00 / 6.0%)
		F:\$	230868.01	
<hr/>				
Mes:	2	I:\$	230868.01	(\$ 11543.40 / 5.0%)
		F:\$	242411.41	
<hr/>				
Mes:	3	I:\$	242411.41	(\$ 9696.46 / 4.0%)
		F:\$	252107.87	
<hr/>				
Mes:	4	I:\$	252107.87	(\$ 7563.24 / 3.0%)
		F:\$	259671.10	
<hr/>				
Mes:	5	I:\$	259671.10	(\$ 5193.42 / 2.0%)
		F:\$	264864.53	
<hr/>				
Mes:	6	I:\$	264864.53	(\$ 2648.65 / 1.0%)
		F:\$	267513.17	
<hr/>				
Mes:	7	I:\$	267513.17	(\$ 1337.57 / 0.5%)
		F:\$	268850.74	
<hr/>				
Mes:	8	I:\$	268850.74	(\$ 1344.25 / 0.5%)
		F:\$	270194.99	
<hr/>				
Mes:	9	I:\$	270194.99	(\$ 1350.97 / 0.5%)
		F:\$	271545.97	
<hr/>				
Mes:	10	I:\$	271545.97	(\$ 1357.73 / 0.5%)
		F:\$	272903.70	
<hr/>				
Mes:	11	I:\$	272903.70	(\$ 1364.52 / 0.5%)
		F:\$	274268.21	
<hr/>				
Mes:	12	I:\$	274268.21	(\$ 1371.34 / 0.5%)
		F:\$	275639.55	
<hr/>				
Interés final:			\$ 57839.54 / (26.6%)	

Figura 25: Reporte de intereses variables acumulados en 12 meses

Problema 8 — Tabla de Amortización Simple

Una financiera desea calcular una tabla de amortización para un monto financiado de **\$ 6,500.00**, con un enganche inicial de **\$ 2,300.00**, el cual pagará los intereses amortizados. Los intereses están diseñados para que bajen linealmente durante 48 semanas y recuperen **80%** del monto financiado. Es decir, se necesitan recuperar intereses por el **80%** de los **\$ 6,500.00**, diseñando intereses que se reduzcan linealmente en 48 pagos y luego se considere el enganche inicial como descuento inmediato a los primeros intereses.

- (a) Calcula la tabla de amortización sin considerar el enganche.
- (b) ¿Cuántos pagos iniciales cubre el enganche?
- (c) Determina un abono fijo para terminar de pagar los intereses más el monto financiado en 48 semanas.

Solución al Problema 8

En este problema nos pide construir una tabla de amortización indicando el abono fijo y la parte de intereses.

En la Figura 26 se muestra el código para describir el problema, especificar el monto y calcular los intereses esperados por el cliente y el total que terminará pagando, así como reporte generado.

```
monto = 6500
periodos = 48
interesObjetivo = monto * 80/100

# En realidad lo que pagará el cliente es:
montoMasInteres = monto + interesObjetivo

print(f"Monto:           $ {monto:8.2f}")
print(f"Financiamiento: $ {interesObjetivo:8.2f}")
print(f"Total a pagar:    $ {montoMasInteres:7.2f}")
print("-" * 30)
print(f"Plazos:           48 semanas")
print()
```

```
Monto:           $ 6500.00
Financiamiento: $ 5200.00
Total a pagar:   $ 11700.00
-----
Plazos:           48 semanas
```

Figura 26: Código para describir el problema de la amortización

En la Figura 27 se muestra el código que calcula el abono fijo sobre el monto que se distribuirá en los 48 meses, luego el monto acumulado, seguido de un valor de amortización, este valor puede ser cualquiera, y se debe ir ajustando hasta que el monto acumulado sea el mismo al total esperado que se desea alcanzar al mes 48, en este caso se partió de un valor inicial de **\$1,000**, el cual se pasaba y se fue ajustando hasta encontrar el valor de **\$285.233** que daba una solución exacta a dos dígitos para alcanzar los **\$11,700.00** esperados. Se describe también el factor de amortización, que será el que determine cuánto se reducirán los intereses mes con mes, en este caso el mes siguiente tendrá intereses por el **95%** del mes anterior. Dentro del iterador se calcula el pago que deberá realizar el cliente en cada mes, el monto que se va acumulando y el interés amortizado actualizado.

```
abonoFijo = monto / periodos

montoAcumulado = 0
amortizacion = 284.233
interesAmortizado = amortizacion
factorAmortizacion = 0.95

for i in range(periodos):
    pago = interesAmortizado + abonoFijo
    montoAcumulado = montoAcumulado + pago
    print(f"[{i + 1:2}] ${montoAcumulado:8.2f} | ${interesAmortizado:7.2f} | ${abonoFijo:7.2f} | ${pago:7.2f}")
    interesAmortizado = interesAmortizado * factorAmortizacion
```

[1]	\$ 419.65	\$ 284.23	\$ 135.42	\$ 419.65
[2]	\$ 825.09	\$ 270.02	\$ 135.42	\$ 405.44
[3]	\$ 1217.02	\$ 256.52	\$ 135.42	\$ 391.94
[4]	\$ 1596.14	\$ 243.69	\$ 135.42	\$ 379.11
[5]	\$ 1963.06	\$ 231.51	\$ 135.42	\$ 366.93
[6]	\$ 2318.41	\$ 219.93	\$ 135.42	\$ 355.35
[7]	\$ 2662.77	\$ 208.94	\$ 135.42	\$ 344.35
[8]	\$ 2996.67	\$ 198.49	\$ 135.42	\$ 333.91
[9]	\$ 3320.66	\$ 188.57	\$ 135.42	\$ 323.98
[10]	\$ 3635.21	\$ 179.14	\$ 135.42	\$ 314.55
[11]	\$ 3940.81	\$ 170.18	\$ 135.42	\$ 305.60
[12]	\$ 4237.90	\$ 161.67	\$ 135.42	\$ 297.09
[13]	\$ 4526.90	\$ 153.59	\$ 135.42	\$ 289.00
[14]	\$ 4808.23	\$ 145.91	\$ 135.42	\$ 281.33
[15]	\$ 5082.26	\$ 138.61	\$ 135.42	\$ 274.03
[16]	\$ 5349.36	\$ 131.68	\$ 135.42	\$ 267.10
[17]	\$ 5609.87	\$ 125.10	\$ 135.42	\$ 260.52
[18]	\$ 5864.13	\$ 118.84	\$ 135.42	\$ 254.26
[19]	\$ 6112.45	\$ 112.90	\$ 135.42	\$ 248.32
[20]	\$ 6355.12	\$ 107.26	\$ 135.42	\$ 242.67
[21]	\$ 6592.43	\$ 101.89	\$ 135.42	\$ 237.31
[22]	\$ 6824.65	\$ 96.80	\$ 135.42	\$ 232.22
[23]	\$ 7052.02	\$ 91.96	\$ 135.42	\$ 227.38
[24]	\$ 7274.80	\$ 87.36	\$ 135.42	\$ 222.78

Figura 27: Código para generar la tabla de amortización con sus primeros resultados

En la Figura 28 se muestra la otra parte del reporte para los últimos meses, observando que efectivamente al mes 48 el cliente ha pagado un acumulado de \$11,700.00.

[25]	\$ 7493.21		\$ 82.99		\$ 135.42		\$ 218.41
[26]	\$ 7707.47		\$ 78.84		\$ 135.42		\$ 214.26
[27]	\$ 7917.79		\$ 74.90		\$ 135.42		\$ 210.32
[28]	\$ 8124.36		\$ 71.16		\$ 135.42		\$ 206.57
[29]	\$ 8327.38		\$ 67.60		\$ 135.42		\$ 203.01
[30]	\$ 8527.01		\$ 64.22		\$ 135.42		\$ 199.64
[31]	\$ 8723.44		\$ 61.01		\$ 135.42		\$ 196.42
[32]	\$ 8916.81		\$ 57.96		\$ 135.42		\$ 193.37
[33]	\$ 9107.29		\$ 55.06		\$ 135.42		\$ 190.48
[34]	\$ 9295.01		\$ 52.31		\$ 135.42		\$ 187.72
[35]	\$ 9480.12		\$ 49.69		\$ 135.42		\$ 185.11
[36]	\$ 9662.74		\$ 47.21		\$ 135.42		\$ 182.62
[37]	\$ 9843.00		\$ 44.85		\$ 135.42		\$ 180.26
[38]	\$10021.02		\$ 42.60		\$ 135.42		\$ 178.02
[39]	\$10196.91		\$ 40.47		\$ 135.42		\$ 175.89
[40]	\$10370.78		\$ 38.45		\$ 135.42		\$ 173.87
[41]	\$10542.72		\$ 36.53		\$ 135.42		\$ 171.94
[42]	\$10712.84		\$ 34.70		\$ 135.42		\$ 170.12
[43]	\$10881.22		\$ 32.97		\$ 135.42		\$ 168.38
[44]	\$11047.96		\$ 31.32		\$ 135.42		\$ 166.73
[45]	\$11213.13		\$ 29.75		\$ 135.42		\$ 165.17
[46]	\$11376.81		\$ 28.26		\$ 135.42		\$ 163.68
[47]	\$11539.07		\$ 26.85		\$ 135.42		\$ 162.27
[48]	\$11700.00		\$ 25.51		\$ 135.42		\$ 160.93

Figura 28: Reporte de amortización para los últimos meses

Problema 9 — Interés Simple Futuro

Un banco financió un proyecto durante 36 meses por **\$ 8,496,500.00**, con una tasa anual fija y simple del **12%** y desea calcular los intereses que se generarán al mes 25 y al mes 30. También desea saber en qué mes los intereses superarán el valor de los **\$ 40,000.00**.

- (a) Calcula los intereses para el mes 25 y 30.
- (b) Determina en qué mes los intereses superarán los **\$ 40,000.00**.
- (c) Calcula el valor de las cuotas fijas para cubrir el préstamos en los 36 meses.
- (d) Calcula el valor de las cuotas fijas para cubrir el préstamos en los 36 meses, si se da un adelanto del **10%** del valor del proyecto (los intereses deberían ser los mismos).

Solución al Problema 9

En este problema nos pide calcular el interés futuro para el mes 25 y 30 usando el interés simple. Esto lo podemos hacer proyectando los meses futuros entre 12 meses que tiene el año ya que la tasa es anual.

En la Figura 29 se muestra el código que calcula los intereses anuales y los futuros para el total de meses. Observa que con una proyección de los periodos entre 12 se consigue.

```
monto = 8496500.00
periodos = 36
tasaAnual = 12/100

interesesAnuales = monto * tasaAnual

interesesFuturos = monto * (periodos / 12) * tasaAnual

interesesAnuales, interesesFuturos

(1019580.0, 3058740.0)

interesesFuturos = monto * (25 / 12) * tasaAnual

interesesFuturos

2124125.0

interesesFuturos = monto * (30 / 12) * tasaAnual

interesesFuturos

2548950.0
```

Figura 29: Código para calcular los intereses futuros

En la Figura 30 se muestra el código que calcula todos los intereses futuros, mes con mes, para determinar que los intereses superarán los **\$ 40,000.00** a partir del quinto mes. También se calcula la cuota fija que considera los intereses futuros y el descuento inicial al monto del **10%**.

```
cuotaFija = (monto + monto * (36 / 12) * tasaAnual - monto * 0.1) / 36

for mes in range(periodos):
    interesesFuturosMes = monto * ((mes + 1) / 12) * tasaAnual
    print(f"[{mes + 1:2}] ${interesesFuturosMes:11.2f} ${cuotaFija:10.2f}")
```

[1]	\$	84965.00	\$	297377.50
[2]	\$	169930.00	\$	297377.50
[3]	\$	254895.00	\$	297377.50
[4]	\$	339860.00	\$	297377.50
[5]	\$	424825.00	\$	297377.50
[6]	\$	509790.00	\$	297377.50
[7]	\$	594755.00	\$	297377.50
[8]	\$	679720.00	\$	297377.50
[9]	\$	764685.00	\$	297377.50
[10]	\$	849650.00	\$	297377.50
[11]	\$	934615.00	\$	297377.50
[12]	\$	1019580.00	\$	297377.50
[13]	\$	1104545.00	\$	297377.50
[14]	\$	1189510.00	\$	297377.50
[15]	\$	1274475.00	\$	297377.50
[16]	\$	1359440.00	\$	297377.50
[17]	\$	1444405.00	\$	297377.50
[18]	\$	1529370.00	\$	297377.50
[19]	\$	1614335.00	\$	297377.50
[20]	\$	1699300.00	\$	297377.50
[21]	\$	1784265.00	\$	297377.50
[22]	\$	1869230.00	\$	297377.50
[23]	\$	1954195.00	\$	297377.50
[24]	\$	2039160.00	\$	297377.50
[25]	\$	2124125.00	\$	297377.50
[26]	\$	2209090.00	\$	297377.50
[27]	\$	2294055.00	\$	297377.50
[28]	\$	2379020.00	\$	297377.50
[29]	\$	2463985.00	\$	297377.50
[30]	\$	2548950.00	\$	297377.50
[31]	\$	2633915.00	\$	297377.50
[32]	\$	2718880.00	\$	297377.50
[33]	\$	2803845.00	\$	297377.50
[34]	\$	2888810.00	\$	297377.50
[35]	\$	2973775.00	\$	297377.50
[36]	\$	3058740.00	\$	297377.50

Figura 30: Código para generar la tabla de intereses futuros mes por mes

Problema 10 — Punto de Liquidación

Un banco financió un proyecto durante 48 meses por **\$ 6,232,800.00**, con una tasa anual fija y simple del **12%** y desea calcular el valor de liquidación que considere pagar los intereses futuros y la deuda total, la liquidación se llevará a cabo en el mes 28.

- (a) Calcula los intereses totales.
- (b) Calcula los intereses faltantes a partir del mes 28.
- (c) Calcula la deuda de liquidación para el mes 28.

Solución al Problema 10

En este problema nos pide calcular el punto de liquidación, que consiste en calcular los intereses futuros y la suma de cuotas pendientes a partir de cierto mes. En este caso queremos sumar los intereses futuros a partir del mes 29 hasta el mes 48 y las cuotas pendientes para hacer una liquidación.

En la Figura 31 se muestra el código que define el monto, periodos, tasa, mes de corte y resta de los meses faltantes, luego reporta el monto, el abono mensual, lo que se lleva pagado hasta el mes 28 y lo que se debe del monto hasta el mes 48 sin considerar los intereses. Observa que se calculan los montos pagados y la diferencia al monto total.

```
monto = 6232520
periodos = 48
tasa = 12/100

corte = 28

resta = periodos - corte

print(f"Financiamiento de {periodos} meses")
print(f"con tasa anual fija del {tasa * 100:.1f}%")
print("-" * 52)
print()

print(f"Monto total      ${monto:11.2f} (sin intereses)")
print()

abonoFijo = monto/periodos

print(f"Abono mensual     ${abonoFijo:11.2f} (sin intereses)")
print()

montoPagado = abonoFijo * corte

print(f"Monto pagado      ${montoPagado:11.2f} ({corte} meses de abono)")

montoPendiente = monto - montoPagado

print(f"Monto pendiente    ${montoPendiente:11.2f} ({resta} meses de abono)")
print()

print("-" * 52)
print()
```

Figura 31: Código para reportar los montos futuros

En la Figura 32 se muestra el código que reporta ahora los intereses totales, los intereses mensuales, el interés pagado hasta el mes de corte y la diferencia entre el total y al corte que será el interés pendiente. Finalmente se calcula el monto de liquidación como lo pendiente entre el monto más los interés pendientes.

```
interesTotal = monto * (periodos / 12) * tasa

print(f"Interés total      ${interesTotal:11.2f} ({periodos} meses)")
print()

interesMensual = interesTotal/periodos

print(f"Interés mensual    ${interesMensual:11.2f} ({corte} meses)")
print()

interesPagado = monto * (corte / 12) * tasa

print(f"Interés pagado      ${interesPagado:11.2f} ({corte} meses)")

interesPendiente = interesTotal - interesPagado

print(f"Interés pendiente   ${interesPendiente:11.2f} ({resta} meses)")
print()

print("-" * 52)
print()

liquidacion = montoPendiente + interesPendiente

print(f"Liquidación          ${liquidacion:11.2f} (mes {corte} / {periodos})")
print()
```

Figura 32: Código para reportar los intereses futuros

En la Figura 33 se muestra el reporte generado que explica los montos e intereses hasta llegar al monto de liquidación.

Financiamiento de 48 meses	
con tasa anual fija del 12.0%	

Monto total	\$ 6232520.00 (sin intereses)
Abono mensual	\$ 129844.17 (sin intereses)
Monto pagado	\$ 3635636.67 (28 meses de abono)
Monto pendiente	\$ 2596883.33 (20 meses de abono)

Interés total	\$ 2991609.60 (48 meses)
Interés mensual	\$ 62325.20 (28 meses)
Interés pagado	\$ 1745105.60 (28 meses)
Interés pendiente	\$ 1246504.00 (20 meses)

Liquidación	\$ 3843387.33 (mes 28 / 48)

Figura 33: Reporte de los montos e intereses y el punto de liquidación

Módulo II — Estadística y Probabilidad con Python

Problema 1 — Probabilidad de un Evento Simple

Una financiera posee una tabla con el género de sus clientes, si tiene trabajo, si está casado y si tiene alguna deuda.

En la Figura 34 se muestra la tabla con el género, trabajo, casado y deuda de los clientes.

	A	B	C	D
1	Género	Trabajo	Casado	Deuda
2	HOMBRE	SI	NO	SI
3	MUJER	NO	SI	NO
4	MUJER	SI	SI	NO
5	HOMBRE	NO	SI	SI
6	MUJER	SI	NO	SI
7	HOMBRE	SI	SI	NO
8	HOMBRE	NO	NO	SI
9	MUJER	SI	SI	SI
10	HOMBRE	SI	SI	NO

Figura 34: Tabla de deuda por género, trabajo y casado

- (a) Calcula la probabilidad de ser hombre
- (b) Calcula la probabilidad de ser mujer
- (c) Calcula la probabilidad de tener trabajo
- (d) Calcula la probabilidad de no tener trabajo
- (e) Calcula la probabilidad de estar casado
- (f) Calcula la probabilidad de no estar casado
- (g) Calcula la probabilidad de tener deuda
- (h) Calcula la probabilidad de no tener deuda

Problema 2 — Tabla conjunta de Dos Eventos

Un restaurante está inspeccionando la bebidas asociadas al pago con tarjeta de sus clientes, porque sospecha que los clientes que pagan con tarjeta de crédito consumen más cerveza que aquellos que pagan con tarjeta de débito.

En la Figura 35 se muestra la tabla con los datos de los clientes si utilizaron tarjeta de crédito o débito y el tipo de bebida que consumieron.

	A	B	C
1	Cliente	Tarjeta	Bebida
2	C001	Débito	Refresco
3	C002	Crédito	Cerveza
4	C003	Débito	Refresco
5	C004	Crédito	Cerveza
6	C005	Crédito	Refresco
7	C006	Débito	Cerveza
8	C007	Débito	Refresco
9	C008	Débito	Refresco
10	C009	Crédito	Cerveza
11	C010	Débito	Refresco
12	C011	Crédito	Cerveza
13	C012	Crédito	Refresco
14	C013	Débito	Refresco
15	C014	Débito	Cerveza
16	C015	Débito	Cerveza

Figura 35: Tabla de tipo bebida consumida asociada al tipo de tarjeta

- (a) Genera la tabla conjunta donde las filas sean el tipo de tarjeta (Débito, Crédito) y las columnas el tipo de bebida (Refresco, Cerveza). En cada celda pon el número total de registros que cumplen la fila y la columna, por ejemplo, para la fila Débito y la columna Refresco, hay 6 registros.
- (b) Agrega una columna adicional con los totales marginales para la fila, sumando todas las columnas.
- (c) Agrega una fila adicional para los totales marginales para la columna, sumando todas las filas.
- (d) Indica el total de registros en la intersección de la columna marginal y la fila marginal.
- (e) Genera la tabla de probabilidad conjunta, que es la misma que la tabla conjunta, solo que cada valor está dividido sobre el total de registros.
- (f) Calcula la probabilidad de tener una tarjeta de débito
- (g) Calcula la probabilidad de tener una tarjeta de crédito
- (h) Calcula la probabilidad de haber consumido refresco
- (i) Calcula la probabilidad de tener consumido cerveza

Problema 3 — Probabilidad Conjunta de Dos Eventos

Una empresa ha recuperado la tabla conjunta del personal que fue promovido a un mejor puesto y los que no a lo largo de 3 años. La tabla divide los hombres y mujeres contra si fueron o no promovidos, también cuenta con sus totales marginales. La empresa necesita encontrar la probabilidad conjunta de que los hombres hayan sido promovidos o no.

En la Figura 36 se muestra la tabla conjunta de promoción del personal por género.

	A	B	C	D
1	Personal	Promovido	No Promovido	
2	Hombre	172	85	257
3	Mujer	140	67	207
4		312	152	464

Figura 36: Tabla conjunta de promoción del personal por género

- (a) Divide cada valor de la tabla conjunta por el total marginal de 464.
- (b) Indica cual es la probabilidad total de que un hombre haya sido promovido.
- (c) Indica cual es la probabilidad total de que un hombre no haya sido promovido.
- (d) Indica cual es la probabilidad total de que una mujer haya sido promovida.
- (e) Indica cual es la probabilidad total de que una mujer no haya sido promovida.
- (f) Indica cual es la probabilidad total de que alguien sea hombre.
- (g) Indica cual es la probabilidad total de que alguien sea mujer.
- (h) Indica cual es la probabilidad total de que alguien sea promovido.
- (i) Indica cual es la probabilidad total de que alguien no sea promovido.

Problema 4 — Probabilidad Condicional de Dos Eventos

Una fábrica de cosméticos está diseñando un nuevo programa para que sus empleados trabajen dos horas menos de lo convencional. Para determinar si su modelo funciona, ha contratado empleados temporales, a los cuales se les pregunta si quieren ganar \$ 2,800 por 6 horas diarias durante los 6 días de jornada de la semana, o ganar \$ 4,080 por 8 horas. Al cabo de la semana, el empleado temporal tiene la oportunidad de tener un contrato de 3 meses o renunciar.

Después de construir la tabla conjunta con el número de empleados que trabajaron 6 horas u 8 horas y que renunciaron o no renunciaron, la empresa decide generar la tabla de probabilidad conjunta para no exponer los números netos sobre los empleados que contrató. Ahora requiere calcular la probabilidad de haber trabajado 6 horas dado que renunció y haber trabajado 8 horas dado que renunció, para poder entender cuál era turno que eligió una persona que decidió renunciar al final.

Además la empresa necesita saber cuál es la probabilidad de renunciar, dado que se trabaja 6 horas y la probabilidad de renunciar dado que se trabaja 8 horas, para entender quién tiene mayor probabilidad de renunciar y decidir si vale la pena mantener un turno de 8 horas en el futuro.

En la Figura 37 se muestra la tabla de probabilidad de los empleados que deciden renunciar o no en turnos de 6 y 8 horas.

	A	B	C	D
1		6 horas	8 horas	
2	Renuncia	0.15	0.34	0.49
3	No Renuncia	0.24	0.27	0.51
4		0.39	0.61	1

Figura 37: Tabla de probabilidad de renuncia por turno

- (a) Indica cual es la probabilidad que alguien renuncie $P(Renuncia)$
- (b) Indica la probabilidad de que alguien trabaje 6 horas $P(6h)$
- (c) Indica la probabilidad de que alguien trabaje 8 horas $P(8h)$
- (d) Indica la probabilidad de que alguien trabaje 6 horas y renuncie $P(6h \cap Renuncia)$
- (e) Indica la probabilidad de que alguien trabaje 8 horas y renuncie $P(8h \cap Renuncia)$
- (f) Calcula la probabilidad de trabajar 6 horas dado que se renuncia

$$P(6h|Renuncia) = \frac{P(6h \cap Renuncia)}{P(Renuncia)}.$$
- (g) Calcula la probabilidad de trabajar 8 horas dado que se renuncia

$$P(8h|Renuncia) = \frac{P(8h \cap Renuncia)}{P(Renuncia)}.$$
- (h) Calcula la probabilidad de renunciar dado que se trabajó 6 horas

$$P(Renuncia|6h) = \frac{P(6h \cap Renuncia)}{P(6h)}.$$
- (i) Calcula la probabilidad de renunciar dado que se trabajó 8 horas

$$P(Renuncia|8h) = \frac{P(8h \cap Renuncia)}{P(8h)}.$$

- Problema 5 — Probabilidad Conjunta de Tres Eventos
- Problema 6 — Probabilidad Condicional de Tres Eventos
- Problema 7 — Estadísticos Principales de Un Eje
- Problema 8 — Estadísticos Principales de Dos Ejes
- Problema 9 — Matriz de Correlación entre Tres Ejes
- Problema 10 — Prueba F entre Dos Hipótesis de Un Eje

Módulo III — Ciencia de Datos

Problema 1 — Estructuración de un Dataset en Excel

Crea un archivo Excel que represente la siguiente información:

País	Año	Población (millones)	PIB per cápita (USD)
México	2022	126	10000
Estados Unidos	2022	331	63000
Canadá	2022	38	48000
Brasil	2022	213	7500
Japón	2022	126	40000

Table 1: Datos para estructurar

Guarda el archivo con el nombre `indicadores.xlsx`.

Problema 2 — Dataset de indicadores en CSV

Convierte el archivo `indicadores.xlsx` del Problema 1 en un archivo CSV llamado `indicadores.csv`. Usa el siguiente formato de columnas:

```
País,Año,Población,PIB_per_capita
México,2022,126,10000
Estados Unidos,2022,331,63000
Canadá,2022,38,48000
Brasil,2022,213,7500
Japón,2022,126,40000
```

Problema 3 — Adquisición de un CSV en Python

Escribe un script en Python que lea el archivo `indicadores.csv` del Problema 2 y lo muestre en la consola. Completa el siguiente código:

```
import pandas as pd

# Ruta del archivo
file_path = "...".

# Cargar el CSV
df = pd.read_csv(file_path)

# Mostrar el DataFrame
print(...)
```

Problema 4 — Indicadores en Series de Pandas

Usando el DataFrame cargado en el Problema 3, crea una Serie con los valores del PIB per cápita. Completa el siguiente código:

```
# Crear la Serie
pib_per_capita = ...

# Mostrar la Serie
print(pib_per_capita)
```

Problema 5 — Indicadores en DataFrames de Pandas

Agrega una nueva columna llamada `PIB_total` que calcule el PIB total como el producto de `Población` y `PIB per cápita`. Completa el código:

```
# Agregar la columna
df['PIB_total'] = ...

# Mostrar el DataFrame
print(df)
```

Problema 6 — Filtrar columnas en DataFrames

Filtra únicamente las columnas `País` y `PIB_total`. Completa el código:

```
# Filtrar columnas
filtered_df = df[...]

# Mostrar las columnas filtradas
print(filtered_df)
```

Problema 7 — Filtrar filas en DataFrames

Filtra las filas donde el PIB total sea mayor a 1 billón de USD (**1e12**). Completa el código:

```
# Filtrar filas
high_pib = df[...]  
  
# Mostrar las filas filtradas  
print(high_pib)
```

Problema 8 — Agrupación de datos en DataFrames

Agrupar los datos por año y calcula el promedio de PIB per cápita. Completa el código:

```
# Agrupar y calcular promedio
average_pib = df.groupby(...)[...].mean()

# Mostrar el resultado
print(average_pib)
```

Problema 9 — Combinación de datos en DataFrames

Combina el DataFrame actual con otro que contenga información adicional, como índice de desarrollo humano (IDH). Completa el código:

```
# Crear otro DataFrame
idh_data = {
    "País": ["Mexico", "Estados Unidos", "Canada", "Brasil", "Japon"],
    "IDH": [0.758, 0.926, 0.929, 0.765, 0.919]
}
idh_df = pd.DataFrame(idh_data)

# Combinar DataFrames
combined_df = pd.merge(df, idh_df, on=...)

# Mostrar el resultado
print(combined_df)
```


Problema 10 — Visualización de datos

Crea un gráfico de barras que muestre el PIB total por país usando la librería `matplotlib`. Completa el código:

```
import matplotlib.pyplot as plt

# Crear grafico de barras
plt.bar(..., ...)

# Configurar etiquetas
plt.xlabel("País")
plt.ylabel("PIB Total (USD)")
plt.title("PIB Total por País")

# Mostrar grafico
plt.show()
```

Módulo IV — Machine Learning

Problema 1 — Aprendizaje Supervisado

Dado el siguiente conjunto de datos en formato CSV:

```
Edad,Salario,Compra
25,50000,No
35,60000,Sí
45,80000,Sí
30,70000,No
40,85000,Sí
```

Carga estos datos en un `DataFrame` de `pandas` y utiliza la librería `sklearn` para entrenar un modelo de clasificación que prediga si una persona comprará un producto basado en su edad y salario.

```
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier
import pandas as pd

# Cargar los datos
data = ...

# Separar variables independientes y dependientes
X = ...
y = ...

# Dividir los datos en conjuntos de entrenamiento y prueba
X_train, X_test, y_train, y_test = train_test_split(...)

# Entrenar el modelo
model = DecisionTreeClassifier()
model.fit(...)

# Evaluar el modelo
accuracy = model.score(...)
print(f"Accuracy: {accuracy}")
```

Problema 2 — Regresión Simple

Usa el siguiente conjunto de datos para predecir el precio de una casa basado en su tamaño en pies cuadrados:

Tamaño	Precio
1000	150000
1500	200000
2000	250000
2500	300000
3000	350000

Crea un modelo de regresión lineal y muestra la ecuación de la recta ajustada.

```
from sklearn.linear_model import LinearRegression

# Datos
X = ...
y = ...

# Crear modelo
model = LinearRegression()
model.fit(...)

# Mostrar ecuacion de la recta
print(f"Coeficiente: {model.coef_}")
print(f"Intercepto: {model.intercept_}")
```

Problema 3 — Regresión Múltiple

Amplía los datos del Problema 2 agregando una columna de número de habitaciones:

```
Tamaño,Habitaciones,Precio
1000,2,150000
1500,3,200000
2000,4,250000
2500,4,300000
3000,5,350000
```

Entrena un modelo de regresión múltiple para predecir el precio de una casa.

Problema 4 — Clasificación Simple

Utilizando el conjunto de datos del Problema 1, usa una regresión logística para predecir la compra de un producto.

```
from sklearn.linear_model import LogisticRegression

# Crear modelo
model = LogisticRegression()
model.fit(...)

# Predicción
pred = model.predict(...)
print(f"Predicciones: {pred}")
```

Problema 5 — Clasificación Múltiple

Dado el siguiente conjunto de datos, crea un modelo de clasificación que prediga la categoría del cliente (A, B o C) basado en el ingreso y la edad:

Edad	Ingreso	Categoría
25	30000	A
35	50000	B
45	70000	C
30	40000	A
40	60000	B

Problema 6 — Aprendizaje No Supervisado

Usa el siguiente conjunto de datos para aplicar el algoritmo de K-Means y agrupar los puntos en 2 clústeres:

X,Y
1,2
2,1
3,4
5,6
6,5

```
from sklearn.cluster import KMeans

# Datos
X = ...

# Crear modelo
kmeans = KMeans(n_clusters=2)
kmeans.fit(...)

# Mostrar centroides
print(f"Centroides: {kmeans.cluster_centers_}")
```

Problema 7 — Reducción de Dimensionalidad

Aplica PCA (Análisis de Componentes Principales) a un dataset con 3 dimensiones y reduce el espacio a 2 dimensiones. Usa datos sintéticos generados con `numpy`.

Problema 8 — Matriz de Adyacencia

Crea una matriz de adyacencia que represente el siguiente grafo:

$$A \leftrightarrow B, A \leftrightarrow C, B \leftrightarrow D, C \leftrightarrow D$$

```
import numpy as np

# Matriz de adyacencia
adj_matrix = np.array([
    [0, 1, 1, 0],
    [1, 0, 0, 1],
    [1, 0, 0, 1],
    [0, 1, 1, 0]
])

print(adj_matrix)
```


Problema 9 — Matriz de Similitud

Dado un conjunto de puntos en un espacio 2D, calcula una matriz de similitud basada en la distancia euclidiana inversa.

Problema 10 — Clusterización

Aplica el algoritmo DBSCAN para encontrar clústeres en el siguiente conjunto de datos:

X,Y
1,2
2,2
3,3
8,8
8,9
25,30

```
from sklearn.cluster import DBSCAN

# Datos
X = ...

# Aplicar DBSCAN
dbscan = DBSCAN(eps=2, min_samples=2)
dbscan.fit(...)

# Mostrar etiquetas de cluster
print(f"Etiquetas: {dbscan.labels_}")
```

Módulo IV — Deep Learning

Problema 1 — Modelo Generalizado

Construye un modelo generalizado con Keras que tenga:

- Una capa de entrada con 10 neuronas.
- Una capa oculta con 16 neuronas y activación ReLU.
- Una capa de salida con una sola neurona y activación sigmoidal.

```
from keras.models import Sequential
from keras.layers import Dense

# Crear el modelo
model = Sequential([
    Dense(16, activation='relu', input_shape=(10,)),
    ...
])

# Compilar el modelo
model.compile(optimizer='adam', loss='binary_crossentropy', metrics=['accuracy'])
```

Problema 2 — Perceptrón Simple

Implementa un perceptrón simple para resolver el problema lógico AND:

Input: [0,0], [0,1], [1,0], [1,1]

Output: [0, 0, 0, 1]

```
import numpy as np
from keras.models import Sequential
from keras.layers import Dense

# Datos
X = np.array([[0,0], [0,1], [1,0], [1,1]])
y = np.array([0], [0], [0], [1])

# Modelo
model = Sequential([
    Dense(1, activation='sigmoid', input_shape=(2,))
])

model.compile(optimizer='sgd', loss='binary_crossentropy', metrics=['accuracy'])

# Entrenar
model.fit(X, y, epochs=100, verbose=1)
```

Problema 3 — Capa de Perceptrones Múltiples

Crea un modelo con tres capas densas: la primera de 8 neuronas, la segunda de 4 neuronas, y la tercera de 2 neuronas, todas con activación ReLU.

```
model = Sequential([
    Dense(8, activation='relu', input_shape=(10,)),
    Dense(4, activation='relu'),
    Dense(2, activation='relu')
])
```

Problema 4 — Modelo Secuencial Simple

Entrena un modelo secuencial para predecir el precio de una casa basada en el tamaño:

Tamaño (m²): [50, 60, 70, 80, 90]

Precio (USD): [100, 120, 140, 160, 180]

```
X = np.array([[50], [60], [70], [80], [90]])
y = np.array([[100], [120], [140], [160], [180]])

model = Sequential([
    Dense(1, activation='linear', input_shape=(1,))
])

model.compile(optimizer='sgd', loss='mean_squared_error')
model.fit(X, y, epochs=100, verbose=1)
```

Problema 5 — Capas Densas Ocultas

Modifica el modelo del problema 4 para agregar una capa oculta con 10 neuronas y activación ReLU.

Problema 6 — Regresión Generalizada

Entrena un modelo para realizar una regresión en el siguiente conjunto de datos:

x: [1, 2, 3, 4, 5]

y: [2.1, 4.2, 6.3, 8.4, 10.5]

Ajusta la salida usando una función de pérdida `mean_absolute_error`.

Problema 7 — Clasificación Generalizada

Usa un modelo de clasificación para identificar si un punto en el plano pertenece al primer cuadrante. Los datos son:

x: $[(1,1), (-1,1), (1,-1), (-1,-1)]$

y: $[1, 0, 0, 0]$

Entrena el modelo usando activación sigmoideal y optimizador Adam.

Problema 8 — Exportación de una Red Neuronal

Guarda el modelo creado en el Problema 7 en un archivo llamado `modelo.h5`.

```
model.save('modelo.h5')
```

Problema 9 — Importación de una Red Neuronal

Carga el modelo guardado en el Problema 8 y realiza predicciones en los datos:

```
from keras.models import load_model

# Cargar modelo
model = load_model('modelo.h5')

# Predicciones
pred = model.predict([[1,1]])
print(pred)
```

Problema 10 — Predicción y Evaluación

Evalúa el modelo del Problema 9 usando los siguientes datos de prueba:

X_test: [(1,1), (0,0), (-1,-1)]
y_test: [1, 0, 0]

```
# Evaluar modelo  
loss, accuracy = model.evaluate(X_test, y_test)  
print(f"Loss: {loss}, Accuracy: {accuracy}")
```