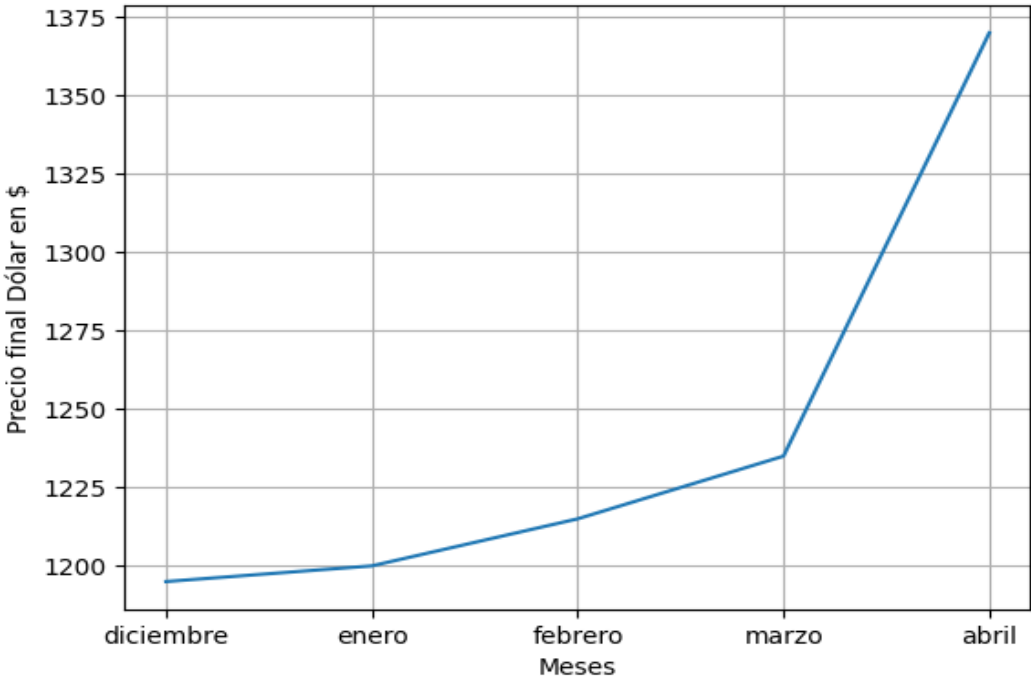


Ejercicio 1 – Tema 5			1 Pto
Cuál de los siguientes diccionarios tiene todas sus claves numéricas (son tipo int, float o complex)?			
A	a=True {a:'verano', True:False, 1.25:'invierno'}		A
B	a=1 {True:a, 1:'uno', 4:'cuatro'}		B
C	a=str(True) {a[0]:0, 2:66, 5:3.45, a:[-2,5]}		C
D	a=int(True) {2:('1', '0'), a:('3','5')}	X	D

Ejercicio 2 – Tema 5			1 Pto
¿Cuál de las siguientes instrucciones debería usar (junto a otras) para generar un gráfico como el siguiente con matplotlib? Los datos a representar son:			
<pre>precioDol = [1195, 1200, 1215, 1235, 1370] meses= ['diciembre', 'enero', 'febrero', 'marzo', 'abril']</pre>			
Evolución Precio Dólar dic/24-abr/25 			
A	ax.plot(meses,precioDol)	X	A
B	plt.set_label(x,'Precio final Dólar en \$')		B
C	ax.scatter(precioDol)		C
D	ax.bar(precioDol)		D

Ejercicio 3 – Tema 5			1 Pto																												
Dado el Dataframe <i>menu</i> :																															
<table><thead><tr><th></th><th>día</th><th>plato</th><th>calorías</th></tr></thead><tbody><tr><td>0</td><td>lunes</td><td>guiso</td><td>750.0</td></tr><tr><td>1</td><td>martes</td><td>pasta</td><td>NaN</td></tr><tr><td>2</td><td>miércoles</td><td>milanesas</td><td>560.0</td></tr><tr><td>3</td><td>jueves</td><td>pastel</td><td>NaN</td></tr><tr><td>4</td><td>viernes</td><td>guiso</td><td>800.0</td></tr><tr><td>5</td><td>sábado</td><td>asado</td><td>750.0</td></tr></tbody></table>					día	plato	calorías	0	lunes	guiso	750.0	1	martes	pasta	NaN	2	miércoles	milanesas	560.0	3	jueves	pastel	NaN	4	viernes	guiso	800.0	5	sábado	asado	750.0
	día	plato	calorías																												
0	lunes	guiso	750.0																												
1	martes	pasta	NaN																												
2	miércoles	milanesas	560.0																												
3	jueves	pastel	NaN																												
4	viernes	guiso	800.0																												
5	sábado	asado	750.0																												
¿Cuál de las siguientes instrucciones debería usar para producir la siguiente salida?																															
<table><thead><tr><th></th><th>día</th><th>plato</th><th>calorías</th></tr></thead><tbody><tr><td>0</td><td>lunes</td><td>guiso</td><td>750.0</td></tr><tr><td>4</td><td>viernes</td><td>guiso</td><td>800.0</td></tr></tbody></table>					día	plato	calorías	0	lunes	guiso	750.0	4	viernes	guiso	800.0																
	día	plato	calorías																												
0	lunes	guiso	750.0																												
4	viernes	guiso	800.0																												
A	menu[(~menu['calorías'].isnull()) & (menu['plato']=='guiso')]		X A																												
B	menu.sort_values(by=['día','calorías'])		B																												
C	menu[(menu['plato']>'p') & (menu['plato']<'q')]		C																												
D	menu[['calorías','plato']]		D																												

Ejercicio 4 – Tema 5			1 Pto
Dado los archivos vtas-suc1.txt y vtas-suc2.txt que contienen las ventas mensuales del primer semestre de cada sucursal:			
vtas-suc1.txt		vtas-suc2.txt	
ene,69.5		ene,19.5	
feb,33.0		feb,43.0	
mar,20.0		mar,0.0	
abr,10.0		abr,110.0	
may,0		may,11	
jun,0		jun,66	
¿Con cuál conjunto de modos de apertura funcionará adecuadamente el siguiente programa?			
<pre>total=0 suc1=open('vtas-suc1.txt', ...) # modo 1? todas=suc1.readlines() suc1.close() for elem in todas: lin=elem.strip('\n') lin=lin.split(',') total+=float(lin[1]) suc2=open('vtas-suc2.txt', ...) # modo 2? todas=suc2.readlines() suc2.close() for elem in todas: lin=elem.strip('\n') lin=lin.split(',') total+=float(lin[1]) linea='Total ventas semestre sucursal 1 y 2: \$'+str(total) semestre=open('vtas-semestre.txt', ...) # modo 3? semestre.write(linea) semestre.close() suc1=open('vtas-suc1.txt', ...) # modo 4? for elem in todas: suc1.write(elem) suc1.close()</pre>			
Al finalizar quedará 1 archivo adicional: vtas-semestre.txt , que contendrá una línea con el total de las ventas del semestre de las dos sucursales			
Y el archivo vtas-suc1.txt tendrá agregadas al final las líneas de vtas-suc2.txt			
A	modo 1 'r' modo 2 'r' modo 3 'w' modo 4 'a'	X	A
B	modo 1 'a' modo 2 'a' modo 3 'w' modo 4 'w'		B
C	modo 1 'r' modo 2 'r+' modo 3 'r' modo 4 'r'		C
D	modo 1 'w' modo 2 'r' modo 3 'a' modo 4 'w'		D

Ejercicio 5 – Tema 5		1 Pto	
¿Cuál programa realiza un ingreso validado de un número en el intervalo (cerrado) [1,15]?			
✓ No se debe aceptar un número que no sea mayor o igual a 1 y menor o igual a 15 ✓ El programa no debe dar fallo si se ingresa un dato que no sea número ✓ El programa debe insistir en la solicitud hasta obtener lo esperado			
A	<pre>sigue=True while sigue: try: num=float(input('Número entre 1 y 15: ')) if num<=15 and num>=1: sigue=False else: print('No está en el intervalo válido') except ValueError: print('Debe ser un número') print(num)</pre>	X	A
B	<pre>while True: try: num=float(input('Número entre 1 y 15: ')) except ValueError: print('Debe ser un número') if num>=1 and num<=15: print(num)</pre>		B
C	<pre>sigue=True while not sigue: try: num=float(input('Número entre 1 y 15: ')) if num<=15 and num>=1: print(num) else: print('No está en el intervalo válido') sigue=False except ValueError: print('Debe ser un número') print(num)</pre>		C
D	<pre>num=float(input('Número entre 1 y 15: ')) try: if num>15 and num<1: print(num) else: print('No está en el intervalo válido') except ValueError: print('Debe ser un número')</pre>		D

Ejercicio 6 – Tema 5				2 Ptos	
Dado el Dataframe <i>menu</i> :					
	día	plato	calorías		
0	lunes	guiso	750.0		
1	martes	pasta	NaN		
2	miércoles	milanesas	560.0		
3	jueves	pastel	NaN		
4	viernes	guiso	800.0		
5	sábado	asado	750.0		
¿Qué salida produce la siguiente instrucción?					
<pre>print(menu[(menu['calorías'].isnull())])</pre>					
A	calorías				A
	750.0 2				
	560.0 1				
	800.0 1				
B	día	plato	calorías		B
	0 lunes	guiso	750.0		
	1 martes	pasta	NaN		
C	día	plato	calorías	X	C
	1 martes	pasta	NaN		
	3 jueves	pastel	NaN		
D	día	plato	calorías		D
	0 lunes	guiso	750.0		
	4 viernes	guiso	800.0		
	5 sábado	asado	750.0		

Ejercicio 7 – Tema 5			2 Ptos
Dado el siguiente programa:			
<pre>def calc1(num): return num%2!=0 def calc2(num): return num*2 def calc3(num): return (num,num%2) numeros=[36,14007,33,20] # línea de código faltante print(nuevaLista)</pre>			
¿Cuál debería ser la línea de código faltante para que la salida sea la siguiente?			
<pre>[(36, 0), (14007, 1), (33, 1), (20, 0)]</pre>			
A	<code>nuevaLista.sort(map(numeros,calc3))</code>		A
B	<code>nuevaLista=list(map(calc3,numeros))</code>	X	B
C	<code>nuevaLista=list(map(calc2,numeros))</code>		C
D	<code>nuevaLista.append(map(calc1,numeros))</code>		D

Ejercicio 8 – Tema 5		2 Ptos	
Dado el siguiente programa:			
<pre>flores=... #?? proveedores=['','Don Pedro','Cartagena Rosas','Escobar II', 'El Brocal','Vivero Sarrundis'] tipo=input('Qué tipo de flor necesita? ') print('Lista de Proveedores de',tipo) for clave in flores: if tipo.lower() in flores[clave]: print(proveedores[clave])</pre>			
¿Cuál de las opciones de la estructura flores funcionará correctamente para producir la siguiente salida?			
<pre>Qué tipo de flor necesita? liliu<u>m</u> Lista de Proveedores de liliu<u>m</u> Vivero Sarrundis Don Pedro Escobar II</pre>			
A	<pre>flores={'DON PEDRO':['rosa monocolor','rosa bicolor', 'rosa rococó','gerbera'], 'EL BROCAL':['gerbera','jazmín','liliu<u>m</u>', 'margarita natural'], 'VIVERO SARRUNDIS':['liliu<u>m</u>','lisisianto','gisófila', 'rosa rococó'], 'ESCOBAR II':['rosa bicolor','clavel bicolor'], 'CARTAGENA ROSAS':['liliu<u>m</u>','margarita natural', 'margarita teñida']}</pre>		A
B	<pre>flores=['rosa monocolor','rosa bicolor', 'rosa rococó','gerbera','gerbera', 'jazmín','liliu<u>m</u>','margarita natural', 'liliu<u>m</u>','lisisianto','gisófila','rosa rococó', 'rosa bicolor','clavel bicolor','liliu<u>m</u>', 'margarita natural','margarita teñida']</pre>		B
C	<pre>flores={2:['rosa monocolor','rosa bicolor', 'rosa rococó','gerbera'], 5:['gerbera','jazmín','liliu<u>m</u>', 'margarita natural'], 1:['liliu<u>m</u>','lisisianto','gisófila','rosa rococó'], 4:['rosa bicolor','clavel bicolor'], 3:['liliu<u>m</u>','margarita natural', 'margarita teñida']}</pre>	X	C
D	<pre>flores={'CR':['rosa monocolor','rosa bicolor', 'rosa rococó','gerbera'], 'VS':['gerbera','jazmín','liliu<u>m</u>', 'margarita natural'], 'DP':['liliu<u>m</u>','lisisianto','gisófila', 'rosa rococó'], 'EB':['rosa bicolor','clavel bicolor'], 'E2':['liliu<u>m</u>','margarita natural', 'margarita teñida']}</pre>		D

Ejercicio 9 – Tema 5		2 Ptos	
¿Cuál es la salida del siguiente programa?			
<pre>def edita (txt): return txt[2:].lower() def patron(txt): cant=0 for voc in 'aeiou': if voc in txt: cant+=1 return cant>1 frutas=['CIRUELA', 'FRUTILLA', 'MANZANA', 'NARANJA'] salida=list(map(edita,frutas)) mostrar=list(filter(patron,salida)) print(mostrar)</pre>			
A	['ciruela', 'frutilla', 'naranja']		A
B	['MANZANA', 'NARANJA']		B
C	['manzan', 'naranj']		C
D	['ruela', 'utilla']	X	D