

APELLIDO:	CALIFICACIÓN:
NOMBRE:	
DNI (registrado en SIU Guaraní):	
E-MAIL:	DOCENTE (nombre y apellido):
TEL:	
AULA:	

Duración del examen: 1 hora 20 minutos. Completar los datos personales con letra clara, mayúscula e imprenta. El examen consta de 13 preguntas de opción múltiple. Cada pregunta tiene una y solo una respuesta correcta. Se debe contestar marcando con una X la opción elegida.

Ejercicio 0108			
¿Cuál de las siguientes afirmaciones es correcta?			
1.	No es posible abrir más de un archivo a la vez		1.
2.	Cuando se cierra un archivo no se puede volver a editar		2.
3.	Cuando se lee un archivo se hace por bloques	X	3.
4.	No se puede borrar el contenido de un archivo entero		4.

Ejercicio 0206			
<p>Se tiene el siguiente archivo llamado 'edades.txt':</p> <p>Juan;20</p> <p>Abril;22</p> <p>Mario;21</p> <p>Paula;21</p>			
<p>¿Qué se va a imprimir luego de ejecutar el siguiente fragmento de código?</p>			
<pre>file = open('edades.txt', "w") lines = file.readlines() x = '' for line in lines:     x += line.split(';')[0] print(x) file.close()</pre>			
1.	21		1.
2.	84		2.
3.	20222121		3.
4.	Se imprime algo diferente/Hay un error en el código y no se ejecuta	X	4.

Ejercicio 0308			
<p>Dado un archivo de texto <i>arch1Par.txt</i>, que contiene apellido, nombre, celular de un contacto por línea y género (línea modelo: Álvarez Elena, 1133452218, M). El archivo se encuentra en la misma carpeta en que está el programa. ¿Cuál versión de código se debería usar si se desea eliminar todas las personas con el nombre y apellido ingresado por terminal?</p>			
1.	<pre>nom = input(f'Ingresá el nombre y apellido a eliminar: ') contactos = open('arch1Par.txt') listCont = contactos.readlines() contactos.close()  def filtrar(elem):     return elem[0] != nom  contactos = open('arch1Par.txt', 'w') contactos.writelines(list(filter(filtrar, listCont))) contactos.close()</pre>		1.
2.	<pre>nom = input(f'Ingresá el nombre y apellido a eliminar: ') contactos = open('arch1Par.txt') listCont = contactos.readlines() contactos.close()  def filtrar(elem):     return elem[0] != nom</pre>		2.

	<pre>contactos = open('arch1Par.txt', 'r+') contactos.writelines(list(filter(filtrar, listCont))) contactos.close()</pre>		
3.	<pre>nom = input(f'Ingresá el nombre y apellido a eliminar: ') contactos = open('arch1Par.txt') listCont = contactos.readlines() contactos.close()  def filtrar(elem):     return elem[0] != nom  contactos = open('arch1Par.txt', 'a') contactos.writelines(list(filter(filtrar, listCont))) contactos.close()</pre>		3.
4.	<pre>nom = input(f'Ingresá el nombre y apellido a eliminar: ') contactos = open('arch1Par.txt') listCont = contactos.readlines() contactos.close()  def filtrar(elem):     return elem.split(', ')[0] != nom  contactos = open('arch1Par.txt', 'w') contactos.writelines(list(filter(filtrar, listCont))) contactos.close()</pre>	X	4.

Ejercicio 0402			
<p>Se cuenta con un archivo de una sola línea con números separados por '/' (un archivo podría contener "100/25/30/200/81"). Además se tiene el siguiente programa:</p> <pre>file = open("numeros.txt", 'r') for elem in file.read().split('/'):     if int(elem) &lt; 80: print(elem) file.close()</pre> <p>¿Cual de las siguientes afirmaciones sobre el programa es correcta?</p>			
1.	El programa imprime la suma de todos los números menores a 80		1.
2.	El programa imprime los elementos menores a 80		2.
3.	El archivo debería ser abierto con el modo 'w'		3.
4.	El programa tiene un problema y no va a ejecutarse	X	4.

Ejercicio 0504			
<p>Respecto a la depuración de errores en el uso y manipulación de datos ¿Cuál de las siguientes afirmaciones es correcta?</p>			
1.	Hacer validaciones en el ingreso de datos es muy costoso		1.
2.	El ingreso de datos de calidad se garantiza con una buena documentación		2.
3.	Simplificar los formatos es una manera de reducir errores de ingreso	X	3.
4.	En términos de costo, da lo mismo en que etapa se identifica un error en un programa		4.

Ejercicio 0606			
<p>¿Cuál es el código correcto para impedir un intento de acceso a una lista fuera de sus límites?</p>			
1.	<pre>misNum=[788,654,937,947,33] posi=11 try:     print('Tu número en la posición',posi,'es',misNum[posi]) except ValueError:     print('No tienes un número en la posición',posi)</pre>		1.
2.	<pre>misNum=[788,654,937,947,33]</pre>	X	2.

	<pre>posi=11 try:     print('Tu número en la posición',posi,'es',misNum[posi]) except IndexError:     print('No tienes un número en la posición',posi)</pre>		
3.	<pre>misNum=[788,654,937,947,33] posi=11 try:     print('Tu número en la posición',posi,'es',misNum[posi]) except:     print('No tienes un número en la posición',posi)</pre>		3.
4.	<pre>misNum=[788,654,937,947,33] posi=11 while True:     try:         print(' Tu número en la posición',posi,'es',misNum[posi])     except:         print('No tienes un número en la posición',posi)</pre>		4.

Ejercicio 0701			
¿Cuál es la función principal de la biblioteca Pandas en Python?			
1.	Analizar y manipular datos en bruto	X	1.
2.	Trabajar con archivos		2.
3.	Crear gráficos fácilmente		3.
4.	Modularizar nuestros programas		4.

Ejercicio 0802			
¿Cuál de los siguientes programas imprime la cantidad de líneas mayores a 10 caracteres en un archivo?			
1.	<pre>archivo = open(nombre_archivo, "r") lines = archivo.readlines() x = 0 for line in lines:     x += 1 print(x) archivo.close()</pre>		1.
2.	<pre>archivo = open(nombre_archivo, "r") lines = archivo.readlines() x = 0 for line in lines:     if line &gt; 10:         x += 1 print(x) archivo.close()</pre>		2.
3.	<pre>archivo = open(nombre_archivo, "r") lines = archivo.readlines() x = 0 for line in lines:     if len(line) &gt; 10:         x = 1 print(x) archivo.close()</pre>		3.
4.	Ninguna opción imprime correctamente la cantidad de líneas	X	4.

Ejercicio 0904
----------------

Tenemos un set de datos de Pandas que contiene personas. Sabemos que hay una columna llamada “nombre” y otra llamada “apellido” y queremos sacar del set a las personas cuyo nombre es distinto de “Juan”. ¿Cuál de los siguientes fragmentos de código hace lo pedido?

1.	<code>set_datos = set_datos[set_datos['nombre']=='Juan']</code>	X	1.
2.	<code>set_datos = set_datos['nombre']=='Juan'</code>		2.
3.	<code>set_datos = filter(set_datos, set_datos['nombre']=='Juan')</code>		3.
4.	<code>for elem in set_datos:     if(elem['nombre']!='Juan'):         set_datos.remove(elem)</code>		4.

Ejercicio 1003			
¿Cuál de las siguientes afirmaciones es verdadera?			
1.	Puedo sacar columnas del set de datos si no me interesan	X	1.
2.	La función <code>isnull()</code> borra todos los elementos nulos		2.
3.	Llamar a <code>df.iloc[:10]</code> devuelve las últimas 10 filas		3.
4.	Ninguna de las afirmaciones es verdadera		4.

Ejercicio 1101			
Se tiene el siguiente set de datos en Pandas:			
	<code>tipo_prenda</code>	<code>color</code>	<code>veces_usada</code>
0	remera	azul	3
1	pantalón	azul	6
2	remera	rojo	2
3	gorra	blanco	2
4	medias	blanco	5
¿Cuál de las siguientes opciones devuelve otro set de datos con 4 filas?			
1.	<code>set_datos[set_datos['veces_usada'] &gt; 2]</code>		1.
2.	<code>set_datos.head(3)</code>		2.
3.	<code>set_datos.iloc[4:]</code>		3.
4.	Ninguna de las opciones cumple con lo pedido	X	4.

Ejercicio 1204			
¿Cuál de los siguientes no es un tipo de gráfico en matplotlib?			
1.	Gráfico de torta		1.
2.	Gráfico de barras		2.
3.	Gráfico de dispersión		3.
4.	Todos son tipos de gráficos	X	4.

Ejercicio 1304			
Luego de ejecutar el siguiente fragmento de código: <code>x = [21,19,13,12,7,1]</code> <code>y = [0,1,2,3,4,5]</code>  <code>plt.plot(x, y)</code> <code>plt.show()</code>			
¿Cuál de las siguientes opciones describe correctamente el gráfico que se ilustrará?			
1.	En el gráfico se va a ver una línea que va de abajo a la izquierda hacia arriba a la derecha		1.
2.	En el gráfico se va a ver una línea que va de arriba a la izquierda hacia abajo a la derecha	X	2.

3.	En el gráfico se van a ver 6 puntos separados sin ninguna línea que los une		3.
4.	Ninguna de las opciones describe lo que ocurre al ejecutar el código		4.