

Práctico 3

Entrada Estándar

1. Escriba un script que solicite el nombre y al apellido del usuario usando la función input. El script debe finalizar volcando a pantalla los datos ingresados.
2. Modifique el script anterior para solicitar la edad del usuario y imprimir en pantalla la edad en meses.

Manejo de Archivos

3. Modifique el script anterior para guardar los datos del usuario en un archivo de texto.
4. Genere un script que permita volcar a pantalla el contenido del archivo generado en el ejercicio anterior.
5. Escriba un script que añada al archivo generado: "Fin del archivo" 6. Escriba otro archivo que elimina las líneas impares.
7. Modifique el script anterior para usar la sentencia with.
8. ¿Qué hace el siguiente script ?
with open("archivo.txt", "r+") as file:
 print ("Nombre del archivo: ", file.name)
 print (file.read()) file.seek(0,
0) linea = file.readline() print
(f"Primera línea: {linea}")

Programación Orientada a Objeto

9. ¿Cuáles son las características principales de la programación orientada a objetos? ¿Python las soporta ?
10. Escriba una clase que contenga solo un par de variables. Genere una instancia para acceder y modificar los valores.
11. Escriba una clase "Persona" con los atributos: nombre, apellido, edad. Use el constructor para inicializar los valores. También defina el método "misdatos" para volcar a pantalla un resumen de los datos personales. Genere un par de instancias e invoque el método misdatos.
12. ¿Una instancia de una clase es un objeto?
13. ¿Se pueden añadir o borrar variables a un objeto ? ¿Cómo?
14. ¿Cuales son las variables de la clase ? ¿Cómo se acceden ?
15. ¿Cuales son las variables de la instancia ? ¿Cómo se acceden ?
16. ¿Qué indica un guión al principio de una variable?
17. ¿Cómo se accede a una variable que empieza con dos guiones ?
18. ¿Qué retorna el siguiente código?
class phone:
 number = 12345
 def call (self):
 print ("calling ...") class
 smartphone(phone):

```

    number = 56789    def call(self):
print ("calling from smartphone...")    def
message (self):        print ("messaging...")

```

```

myphone1 = smartphone()
myphone1.call()
myphone1.message()

```

19. ¿Qué retorna el siguiente código?

```

class phone:    __number = 1234    def
call(self):        print ("calling...",
phone.__number)

```

```

myphone1 = phone ()
myphone1.call ()
print ("calling...", phone.__number)

```

20. Escriba una clase de Python que tenga dos métodos `get_String` e `print_String`. `get_String` acepta una cadena del usuario e `print_String` imprime la cadena en mayúsculas.
21. Escriba una clase de Python llamada Rectángulo construida por un largo y ancho y un método que calcule el área de un rectángulo.

Manejo de Excepciones

22. ¿Cuál es el resultado del siguiente código?

```

a = [1, 2, 3] try:
    print (f"Primmer element = {a[0]}")    print (f"Tercer
element = {a[3]}") except IndexError:    print ("Error
accediendo a los elementos de la lista")

```

23. Escriba un script que solicite al usuario la edad (valor entero) y el peso (valor real) y verifique que el primero represente un número int válido y que el segundo represente un número flotante válido. Haga comentarios útiles si no lo son.
24. Escriba un script que intente leer un archivo. Proporcione comentarios útiles si el archivo no existe o si algo sale mal al leer el archivo.
25. ¿Cómo funciona el `else` en el siguiente script?

```

def dividir(a , b):
try:
    c = ((a+b) / (a-b))
except ZeroDivisionError:
    print("a-b es 0")
else:
    print(c)

```

```

dividir(2.0, 3.0) dividir(3.0,
3.0)

```

26. Analice el resultado del siguiente script:try:
- ```

 raise NameError("Lanzando un excepción") except
NameError:
 print ("Una excepción")
raise

```