

MÓDULO I

Reforzamiento N°03

1. Pedir dos números positivos mediante terminal al usuario. Mostrar como salida el número cuya sumatoria de dígitos es el mayor y los números cuya sumatoria de dígitos es menor que 10. Utilizar una o más funciones, según sea conveniente.
2. Crea una función que al ingresar dos números por parámetro mostrará todos los cuadrados de los números que hay entre ellos (Usar la función una vez y mostrar el resultado por consola). Los números serán ingresados y solicitados al usuario.
3. Crear una función que sume los dígitos del número ingresado y muestre por consola la suma de estos dígitos.
4. Pedir al usuario que ingrese una oración con un mínimo de 3 palabras la cual será usada por parámetro para una función que se creará e indicará cuantas palabras existen dentro de la oración ingresada.
5. Crear una función que aceptará por parámetro dos valores que serán ingresados por el usuario, una lista donde los valores serán llenados por el usuario también y un segundo parámetro que eliminará de la lista que fue ingresada a la función, finalmente el output de la función será la lista actualizada sin el valor que se sacará de la lista. Mostrar también la lista original y el número que fue eliminado.
6. Escribir una clase en Python que contenga un método que convierta un número entero en su cubo y contenga otro método que obtenga el cuadrado de ese resultado. El valor inicial de resultado deberá estar creado en el constructor. Considerar un método en la cual le pedirá al usuario ingresar un valor numérico.

7. Crear una clase en python que contenga un método que revierta una cadena de palabras.

Input: "Hola Pythonista, seguimos adelante"

Output: "adelante seguimos Pythonista Hola"

Lamarlo mínimo 2 veces y mostrar el resultado por consola.

8. Crear una clase que contenga dos métodos, uno que pida ingresar un nombre y apellido, un método para pedir su edad y otro método que lo imprima ambos resultados, pero estarán contenidos en un diccionario. Comprobar ambos métodos instanciando la clase respectivamente. Considerar en el constructor los valores necesarios.

9. Crear una clase llamada círculo que contenga radio en su constructor que contenga un método **área** que devuelva el área de un círculo. Aplicar excepciones en caso no se ingrese un dato tipo numérico.

Crear adicionalmente un método que devuelva el perímetro del círculo.

Instanciar la clase respectivamente para dos diferentes radios.

Habrà un método donde pedirá el radio del círculo.

Instanciar mínimo 2 veces la clase y mostrar resultados por consola.

10. Crear una clase llamada **Alumno** que tenga como atributos el nombre, edad y la nota final del alumno. Crear los métodos para inicializar sus atributos, otro para imprimirlos y un método para mostrar un mensaje con el resultado de la nota y si ha aprobado (mayor o igual a 11) o no el alumno. Instanciar la clase por lo menos 3 veces (3 alumnos)

11. Crear una clase **Persona** con los siguientes requerimientos.

La clase tendrá como atributos el nombre, edad y sueldo de una persona. Implementar los métodos necesarios para inicializar los atributos (constructor), un método para mostrar los datos e indicar si la persona es mayor de edad o no y otro método que bonificación que retornará el 20% adicional de su sueldo.

Instanciar por lo menos la clase con 2 diferentes personas.

12. Realizar una clase que administre una agenda. Se debe almacenar en un diccionario dentro de una lista para cada contacto el nombre, el teléfono, email y DNI. Deberá tener los siguientes métodos:

Añadir contacto
Mostrar contactos
Buscar contacto
(Por DNI)

13. Crear una clase Persona que contenga dos atributos: **nombre** y **edad**.
Nombre y edad se ingresarán por teclado en el constructor.
Declarar una segunda clase llamada Empleado que herede de la clase Persona y agregue un atributo sueldo y muestre si debe pagar impuestos (10% de su sueldo-encapsulamiento) (sueldo superior a 4000)
Instanciar la clase Empleado, mostrar el sueldo del empleado y cuánto debe pagar de impuesto.
14. Crear un módulo y un archivo principal (donde llamará las funciones del módulo) el módulo tendrá una función para ingresar nombres y apellidos, una función para pedir el tipo de seguro que tiene y otra función para indicar si es mayor de edad o no (pedir la edad desde consola)

Indicaciones:

- Subir su siguiente carpeta de "reforzamientos 03" a su repositorio y la estructura final en github debería verse del siguiente modo:



- Cada solución debe ser realizada en un diferente archivo Python *.py
- Correo a enviar soluciones con enlace de su repositorio:
cerseuufisi@gmail.com
- Asunto: Reforzamiento 03 - Módulo I
- Fecha máxima de entrega: sábado 10 de febrero hasta las 23:59 horas.