

```
## Cristian Eduardo Gonzalez Primero      65RR012      01-10-22
# Crear una clase llamada Persona. Sus atributos son: nombre, edad y DNI.
# Construye los siguientes métodos para la clase:
# • Un constructor, donde los datos pueden estar vacíos.
# • Los setters y getters para cada uno de los atributos. Hay que validar las en
# (Edad > 0, DNI Limite de 10 caracteres y opcional que valide que solo sean núme
# • mostrar(): Muestra los datos de la persona.
# • esMayorDeEdad(): Devuelve un valor lógico indicando si es mayor de edad.
```

```
class Persona:
```

```
    def __init__(self, nombre = '', edad = int(), DNI = ''):
        self.nombre = nombre
        self.edad = self._is_valid_age(edad)
        self.dni = self._is_valid_dni(DNI)
```

```
    # getter method
```

```
    @property
```

```
    def get_nombre(self):
        return self.nombre
```

```
    @get_nombre.setter
```

```
    def set_nombre(self, nuevoNombre):
        self.nombre = nuevoNombre
```

```
    # getter method
```

```
    @property
```

```
    def get_edad(self):
        return self.edad
```

```
    @get_edad.setter
```

```
    def set_edad(self, nuevaEdad):
        self.edad = self._is_valid_age(nuevaEdad)
```

```
    # getter method
```

```
    @property
```

```
    def get_dni(self):
        return self.dni
```

```
    @get_dni.setter
```

```
    def set_dni(self, c):
        self.dni = self._is_valid_dni(c)
```

```
    def mostrar(self):
```

```
        print(self.nombre)
        print(self.edad)
        print(self.dni)
```

```
    def _is_valid_age(self, age):
```

```
        if age < 0:
            raise ValueError("Edad negativa no aceptada")
        return age
```

```
    def _is_valid_dni(self, _dni):
```

```
        if len(_dni) > 10:
            raise ValueError("DNI no puede exceder más de 10 caracteres.")
        return _dni
```

```
IDLE Shell 3.10.7
File Edit Shell Debug Options Window Help

Python 3.10.7 (tags/v3.10.7:6cc6b13, Sep 5 2022, 14:08:36) [MSC v.1933 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.

>>>
===== RESTART: C:\Users\PILARES\Desktop\01-10-22\Practica7-1.py =====
>>> cristian = Persona('Cristian', -5, 'GOPC951223HDF')
Traceback (most recent call last):
  File "<pyshell#0>", line 1, in <module>
    cristian = Persona('Cristian', -5, 'GOPC951223HDF')
  File "C:\Users\PILARES\Desktop\01-10-22\Practica7-1.py", line 13, in __init__
    self.edad = self._is_valid_age(edad)
  File "C:\Users\PILARES\Desktop\01-10-22\Practica7-1.py", line 57, in _is_valid_age
    raise ValueError("Edad negativa no aceptada")
ValueError: Edad negativa no aceptada
>>> cristian = Persona('Cristian', 26, 'GOPC951223HDF')
Traceback (most recent call last):
  File "<pyshell#1>", line 1, in <module>
    cristian = Persona('Cristian', 26, 'GOPC951223HDF')
  File "C:\Users\PILARES\Desktop\01-10-22\Practica7-1.py", line 14, in __init__
    self.dni = self._is_valid_dni(DNI)
  File "C:\Users\PILARES\Desktop\01-10-22\Practica7-1.py", line 62, in _is_valid_dni
    raise ValueError("DNI no puede exceder más de 10 caracteres.")
ValueError: DNI no puede exceder más de 10 caracteres.
>>> cristian = Persona('Cristian', 26, 'GOPC951223')
>>> cristian.mostrar()
Cristian
26
GOPC951223
>>> cristian.esMayorDeEdad()
Es mayor de edad.
True
>>> |
```

Ln: 31 Col: 0

```
Practica7-2.py - C:\Users\PILARES\Desktop\01-10-22\Practica7-2.py (3.10.7)
File Edit Format Run Options Window Help

## Cristian Eduardo Gonzalez Primero 65RR012 01-10-22

##Crea una clase llamada Cuenta que tendrá los siguientes atributos: titular (que es una per.
# El titular será obligatorio y la cantidad es opcional. Construye los siguientes métodos pa.
# • Un constructor, donde los datos pueden estar vacíos.
# • Los setters y getters para cada uno de los atributos. El atributo no se puede
# modificar directamente, sólo ingresando o retirando dinero.
# • mostrar(): Muestra los datos de la cuenta.
# • ingresar(cantidad): se ingresa una cantidad a la cuenta, si la cantidad
# introducida es negativa, no se hará nada.
# • retirar(cantidad): se retira una cantidad a la cuenta. La cuenta puede estar
# en números rojos.

class Cuenta:
    def __init__(self, titular, cantidad=float()):
        self.titular = titular
        self.cantidad = cantidad

    @property
    def get_titular(self):
        return self.titular

    @get_titular.setter
    def set_titular(self, nuevoTitular):
        self.titular = nuevoTitular

    @property
    def get_cantidad(self):
        return self.cantidad

    def mostrar(self):
        print('Titular: ',self.titular)
        print('Cantidad: ',self.cantidad)

    def ingresar(self, ingreso):
        if ingreso >= 0:
            self.cantidad += ingreso
            return self.cantidad
        else:
            return self.cantidad

    def retirar(self, retiro):
        self.cantidad -= retiro
        return self.cantidad
```

Ln: 7 Col: 61

```
IDLE Shell 3.10.7
File Edit Shell Debug Options Window Help
Python 3.10.7 (tags/v3.10.7:6cc6b13, Sep  5 2022, 14:08:36) [MSC v.1933 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\Users\PILARES\Desktop\01-10-22\Practica7-2.py =====
>>> daniel = Cuenta('Daniel')
>>> daniel.mostrar()
Titular: Daniel
Cantidad: 0.0
>>> daniel.ingresar(500)
500.0
>>> daniel.retirar(49.90)
450.1
>>> daniel.mostrar()
Titular: Daniel
Cantidad: 450.1
>>> |
```

Ln: 16 Col: 0