



Semana 12

Métodos

Programación

2020-2021

Ejercicio 1. Crear una clase `Persona` que contenga los siguientes atributos: `nombre`, `edad`, `DNI`, `letraDNI`, `sexo` (hombre, mujer), `peso` y `altura`. Crear un método `init` dando valor adecuado a cada atributo (el `sexo` será mujer por defecto). El campo `letraDNI` se debe calcular automáticamente (se debe buscar información en internet sobre cómo calcular este campo sabiendo el número de DNI). Crear como propiedades los siguientes atributos:

- `edad: int`: Asigna el valor de `edad` al atributo `edad` de la persona.
- `sexo: str`: Asigna el valor de `sexo` al atributo `sexo` de la persona
- `dni: int`: Asigna el valor de `dni` al atributo `dni` de la persona. Deberá comprobar que el número introducido tiene la longitud adecuada (8 como máximo y 1 como mínimo). Además invocará al método `calcularLetraDni()` que se definirá más adelante.
- `peso: float`: Asigna el valor de `peso` al atributo `peso` de la persona, comprobando que el valor recibido es correcto.
- `altura: float`: Asigna el valor de `altura` al atributo `altura` de la persona, comprobando que el valor recibido es correcto.

Crear además los siguientes métodos:

- `__calcularLetraDni()`: Calcula la letra del DNI en base al valor del atributo `DNI` de la persona, y lo almacena en el atributo `letraDNI`.
- `__str__()`: Devuelve un `String` para imprimir por pantalla los atributos de la persona de acuerdo al siguiente formato:

Información personal:

Nombre: Pepe

Sexo: Hombre

Edad: 16 años

DNI: 821946-H

Peso: 80.0 kg

Altura: 177.0 cm

- Crear un programa que cree una tupla con 3 objetos de la clase `Persona` en base a las siguientes reglas:
 - El primer objeto se creará con valores de los campos elegidos por el programador.
 - El segundo objeto se creará con valores que se recibirán por el teclado.
 - El tercer objeto obtendrá por teclado todos los valores de los campos menos el `peso`, la `altura` y el `sexo`, a los que se asignarán los valores 90, 185 y el sexo por defecto respectivamente.
- Por último se mostrará la información de cada objeto por pantalla utilizando el método `__str__` definido con anterioridad.

Ejercicio 2. Crear una clase `Rectángulo` con tres atributos: `base`, `altura` y `cuadrado` (que especifica si además de rectángulo es un cuadrado o no)

- Crear propiedades para cambiar y leer el valor de los atributos (decidir cuáles se deben poder especificar externamente y cuáles se pueden calcular automáticamente. Decidir también los valores válidos para cada uno de ellos).
- Crear un método `init`. Si solo se recibe un parámetro será la base y creará un cuadrado. Reutilizar las propiedades y métodos creados anteriormente.
- Crear un método `str` que devuelve un `String` del tipo: “un <rectángulo/cuadrado> de base <base> y altura <altura>” donde los elementos entre <> deben cambiarse por los valores adecuados.
- Crear un método `eq` que compara dos rectángulos y devuelve `true` si ambos tienen la misma base y la misma altura, considerando que un rectángulo de (`base`, `altura`) es igual que uno de (`altura`, `base`).
- Crear un método `perimetro` que devuelve el perímetro del Rectángulo.
- Crear un método `area` que devuelve el área del Rectángulo.
- Crear un método que devuelve el valor del lado mayor del rectángulo (la altura o la base)
- Crear un método que convierte un rectángulo en un cuadrado. Deberá convertir la base o la altura al valor mayor de ambos. Devolverá `True` si lo que se cambió fue la base y `False` si lo que se cambió fue la altura.
- En una case principal crear una lista de Rectángulos con un número aleatorio de entre 10 y 1000 elementos. Crear los rectángulos de la lista con dimensiones enteras entre 1 y 10. Imprimir los que sean cuadrados, los de mayor área, los de mayor perímetro y los de lado más grande. Convertir los de mayor lado a cuadrados, especificando si lo que se ha cambiado es la base o la altura.

Normas de entrega

Los ejercicios se deben subir a Aula Global antes de las 7:00 de la mañana del próximo día con clase de laboratorio). Se deberá subir un fichero comprimido **zip** con un fichero para cada programa, nombrados `ejercicio1.py`, `ejercicio2.py`, etc. El nombre del fichero será “s12-iniciales-del-alumno.zip” (por ejemplo Lucía Pérez Gómez subirá un archivo llamado `s12-lpg.zip`).