

PROGRAMACIÓN AVANZADA CON PYTHON

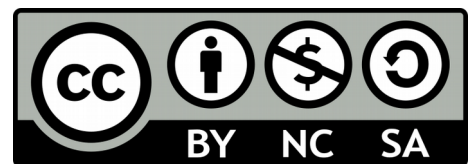
(CEFIRE CTEM)



POO en Python (parte 2) Ejercicio obligatorio

Esta obra está sujeta a la licencia Reconocimiento-NoComercial-CompartirIgual 4.0 Internacional de Creative Commons. Para ver una copia de esta licencia, visitad

<http://creativecommons.org/licenses/by-nc-sa/4.0/>.



Autora: María Paz Segura Valero (segura_marval@gva.es)

CONTENIDO

1. Introducción.....	2
2. Enunciado.....	2

1. Introducción

En este documento puedes encontrar el **ejercicio obligatorio** de esta unidad. Es imprescindible entregarlo en tiempo y forma para superar esta parte del curso.

Tendrás la oportunidad de realizar la entrega de varias versiones del ejercicio hasta que consigas superarlo y la profesora te indicará en cada corrección las mejoras necesarias.

En cualquier momento puedes lanzar preguntas al **foro** del curso o realizar una **entrega parcial** del ejercicio acompañada de una **lista de dudas** para que la profesora pueda orientarte en su resolución.

2. Enunciado

Vamos a crear un programa que ayude a una prestigiosa Chef a gestionar las recetas de su restaurante. Para ello, mostrará un menú que permitirá confeccionar las recetas a partir de un conjunto de *ingredientes* y la descripción de los pasos a seguir (*elaboración*).

El programa mostrará el siguiente menú:

```
=====
                        COCINANDO
=====
1 - Añadir ingrediente
2 - Escribir elaboración
3 - Mostrar receta
0 - SALIR
-----
Dame la opción:
```

Deberás crear las siguientes clases:

- Clase `Ingrediente`:
 - Esta clase es derivada de la clase `Articulo` del modulo `clases_base`.
 - Atributos: `codigo`, `modelo` y `cantidad`. Recuerda que los dos primeros son heredados de la clase base `Articulo`.
 - Métodos: `__init__()` para construir un objeto de la clase y `__str__()` para construir una cadena de texto con los valores de los atributos del objeto.
 - En los dos métodos, utiliza el método correspondiente de la *clase base* y luego añade el código necesario de la *clase derivada*.

- Clase Receta:
 - **Atributos:** ingredientes y elaboracion. El primer atributo es una lista de objetos Ingrediente y el segundo es un texto descriptivo.
 - **Métodos:**
 - `__init__()` para construir un objeto de la clase.
 - `__str__()` para construir una cadena de texto con los valores de los atributos del objeto.
 - `anyadir_ingrediente()` que recibirá como parámetro un objeto de la clase Ingrediente y lo añadirá al final de la lista ingredientes de la receta.

Retomando el menú del programa, cada opción seleccionada ejecutará una nueva función que responderá a la tarea correspondiente.

Opción	Función ejecutada
1	<p>elegir_ingrediente()</p> <p>Esta función pedirá al usuario los datos necesarios para crear un nuevo objeto de la clase Ingrediente y lo añadirá a la lista de ingredientes pasada por parámetro (utiliza el método <code>anyadir_ingrediente()</code>). Además, imprimirá la nueva lista de ingredientes.</p> <p><u>Ejemplo:</u></p> <pre> AÑADIR INGREDIENTE: --> Dame el código: 3 --> Dame el modelo: tomate --> Dame la cantidad: 100 Ingredientes: (1, aceite) --> 10 (2, arroz) --> 250 (3, tomate) --> 100 </pre> <p>A la función se le pasará como parámetro de entrada el objeto Receta del programa principal y devolverá el mismo objeto con la lista de ingredientes modificada.</p>
2	<p>escribir_elaboracion()</p> <p>Esta función pedirá el texto descriptivo de la elaboración de la receta.</p> <p><u>Ejemplo:</u></p> <pre> ESCRIBIR ELABORACION: --> Dame el texto: Echar el aceite a la sartén y cuando esté caliente... </pre> <p>A la función se le pasará como parámetro de entrada el objeto Receta del programa principal y devolverá el mismo objeto con la elaboración modificada.</p>

3	<p>mostrar_receta()</p> <p>Esta función mostrará por pantalla la información de la receta creada con las opciones anteriores.</p> <p><u>Ejemplo 1:</u> No se ha completado la receta.</p> <pre>RECETA: --> Ingredientes: (no seleccionados) ---> Elaboración: (no facilitada)</pre> <p><u>Ejemplo 2:</u> Se han elegido los ingredientes y se ha escrito la elaboración.</p> <pre>RECETA: --> Ingredientes: (1, aceite) --> 10 (2, arroz) --> 250 (3, tomate) --> 100 ---> Elaboración: Echar el aceite a la sartén y cuando esté caliente...</pre> <p>A la función se le pasará como parámetro de entrada el objeto <code>Receta</code> y no devolverá ningún parámetro de salida.</p> <p>Si tienes bien construido el método <code>__str__()</code> solo hará falta que hagas <code>print(objeto_receta)</code> para que funcione.</p>
0	El programa acabará.

Cuando se acabe de ejecutar una acción, el programa volverá a mostrar el menú al usuario, excepto si se ha seleccionado la *opción 0*.

Te recomiendo que al principio del programa crees un objeto `Receta` vacío que sea el que pases como parámetro a las funciones. Puedes crearlo así:

```
r = Receta([], "")
```

Donde `[]` es la lista de ingredientes vacío y `''` es una cadena de texto vacía.

No es necesario que contemples el encapsulamiento de objetos.

Sube al aula virtual el ejercicio en el fichero **ud4_ejercicio_obligatorio.py**.