

Laboratorio M3-14

Realizar los siguientes ejercicios en tu Editor de Código preferida, adjuntar tu archivo py.

Enviar laboratorio a: jpruiz@itgcorp.co

Recuerda adjuntar el número de grupo al que perteneces en el asunto del email

Ejercicio No.14 - Días de la semana

Tiempo Estimado

30 - 60 minutos

Nivel de Dificultad

Fácil / Medio

Objetivos

- Mejorar las habilidades del estudiante para definir clases desde cero.
- Definir y usar variables de instancia.
- Definir y usar métodos.

Escenario

Tu tarea es implementar una clase llamada Weeker. Sí, tus ojos no te engañan, este nombre proviene del hecho de que los objetos de esta clase podrán almacenar y manipular los días de la semana.

El constructor de la clase acepta un argumento: una cadena. La cadena representa el nombre del día de la semana y los únicos valores aceptables deben provenir del siguiente conjunto:

Lun Mar Mie Jue Vie Sab Dom

Invocar al constructor con un argumento desde fuera de este conjunto debería generar la excepción `WeekDayError` (defínela tu mismo; no te preocupes, pronto hablaremos sobre la naturaleza objetiva de las excepciones). La clase debe proporcionar las siguientes facilidades:

- Los objetos de la clase deben ser "imprimibles", es decir, deben poder convertirse implícitamente en cadenas de la misma forma que los argumentos del constructor.
- La clase debe estar equipada con métodos de un parámetro llamados **`add_days(n)`** y **`subtract_days(n)`**, siendo **`n`** un número entero que actualiza el día de la semana almacenado dentro del objeto mediante el número de días indicado, hacia adelante o hacia atrás.
- Todas las propiedades del objeto deben ser privadas.

Completa la plantilla que te proporcionamos en el editor, ejecuta su código y verifica si tu salida se ve igual que la nuestra.

Salida Esperada

Lun

Mar

Dom

Lo siento, no puedo atender tu solicitud.

Código base

```

1  class WeekDayError(Exception):
2      pass
3
4
5  class Weeker:
6      #
7      # Escribir código aquí.
8      #
9
10     def __init__(self, day):
11         #
12         # Escribir código aquí.
13         #
14
15     def __str__(self):
16         #
17         # Escribir código aquí.
18         #
19
20     def add_days(self, n):
21         #
22         # Escribir código aquí.
23         #
24
25     def subtract_days(self, n):
26         #
27         # Escribir código aquí.
28         #
29
30
31     try:
32         weekday = Weeker('Lun')
33         print(weekday)
34         weekday.add_days(15)
35         print(weekday)
36         weekday.subtract_days(23)
37         print(weekday)
38         weekday = Weeker('Lun')
39     except WeekDayError:
40         print("Lo siento, no puedo atender tu solicitud.")

```