

TUP – PROGRAMACIÓN II

Guía de Ejercicios Prácticos Ciclo Lectivo 2023

Trabajo Práctico N° 1

Tema: Clases

1) Clase: Estudiante

Construye una clase llamada Estudiante con los siguientes atributos:

- **nombre:** El nombre del estudiante.
- **edad:** La edad del estudiante.
- **promedio:** El promedio de calificaciones del estudiante.
- **carrera:** La carrera que está estudiando.

La clase debe tener los siguientes métodos:

1. **verificar_aprobacion(self):** Este método debe verificar si el estudiante ha aprobado. Si el promedio es mayor o igual a 70, debe retornar "Aprobado". En caso contrario, debe retornar "Reprobado".
2. **es_mayor_de_edad(self):** Este método debe verificar si el estudiante es mayor de edad. Si la edad es mayor o igual a 18, debe retornar True. Si es menor, debe retornar False.

2) Clase: Libro

Crea una clase llamada Libro con los siguientes atributos:

- **título:** El título del libro.
- **autor:** El autor del libro.
- **año_publicación:** El año en que se publicó el libro.
- **disponible:** Un booleano que indica si el libro está disponible para préstamo.

La clase debe tener los siguientes métodos:

1. **solicitar_prestamo(self):** Este método debe cambiar el estado del libro a no disponible y retornar un mensaje indicando que el libro ha sido prestado, siempre y cuando esté disponible. Si no está disponible, debe retornar un mensaje indicando que el libro no puede ser prestado en ese momento.
2. **devolver_libro(self, días):** Este método recibe la cantidad de días que el libro ha sido prestado y calcula una posible multa si se ha excedido el tiempo de préstamo. Si no hay multa, debe retornar un mensaje indicando que el libro ha sido devuelto.

3) Clase: TiendaOnline

Escribe una clase llamada TiendaOnline con los siguientes atributos:

- **nombre_tienda**: El nombre de la tienda en línea.
- **productos_disponibles**: Una lista de productos disponibles en la tienda.
- **ubicación**: La ubicación de la tienda en línea.
- **ventas_mensuales**: El total de ventas mensuales de la tienda.

La clase debe tener los siguientes métodos:

1. **buscar_producto(self, producto)**: Este método debe buscar un producto en la lista de productos disponibles y retornar su información si se encuentra. Si no se encuentra, debe retornar un mensaje indicando que el producto no está disponible en este momento.
2. **calcular_descuento(self, monto_compra)**: Este método debe calcular un descuento en base al monto de compra. Si el monto es mayor o igual a \$100, debe aplicar un descuento del 10%. Si es mayor o igual a \$200, debe aplicar un descuento del 20%. Debe retornar el monto total con el descuento aplicado.

4) Clase: Coche

Modela una clase Coche que represente un vehículo. Define los atributos necesarios para almacenar la marca, modelo, año y velocidad actual del coche. Implementa métodos para acelerar y frenar el coche, actualizando su velocidad en función de los valores proporcionados.

5) Clase: Banco

Crea una clase Banco que simule un banco. Define atributos para el nombre del banco, el saldo total, las tasas de interés y una lista de clientes. Implementa métodos para abrir una cuenta bancaria para un cliente y calcular los intereses acumulados en base a las tasas de interés.

6) Clase: ProductoElectrónico

Diseña una clase ProductoElectrónico que represente un producto electrónico. Define los atributos necesarios para el nombre, marca, precio y disponibilidad del producto. Crea métodos para simular la compra del producto, cambiando su estado de disponibilidad, y para verificar la garantía en caso de defectos.

7) Clase: Película

Modela una clase Película que almacene información sobre películas. Define atributos para el título, director, duración y género de la película. Implementa métodos para reproducir la película, mostrando su título y duración, y para recomendar la película a un usuario en base a su género preferido.

8) Clase: Restaurante

Crea una clase Restaurante para representar un lugar de comida. Define atributos para el nombre del restaurante, tipo de cocina, cantidad de mesas disponibles y valoración. Implementa métodos para permitir a los clientes reservar mesas y calificar el restaurante, actualizando su valoración.

9) Clase: Empleado

Diseña una clase Empleado que almacene información sobre los empleados de una empresa. Define atributos para el nombre, edad, salario y cargo del empleado. Crea métodos para calcular y aplicar aumentos salariales en base a un porcentaje y para cambiar el cargo del empleado.

10) Clase: AnimalDoméstico

Modela una clase AnimalDoméstico que represente a un animal que puede ser una mascota. Define atributos para el nombre, especie, edad y dueño del animal. Implementa métodos para hacer un sonido característico de la especie y para verificar si el animal es joven, adulto o anciano en función de su edad y especie.