

Practica Python 6. Orientación a Objetos

Básicos:

1).- Cree una clase COrdenador para simular el trabajo con ordenadores. La clase COrdenador puede incluir los siguientes atributos:

CMarca: Toshiba, Mitac, Ast, etc...
CProcesador: Intel, Amd.
CPeso: 2, 2.5, 3
CEstado: Encendido o Apagado
CPantalla: Activa o inactiva

Las acciones que se pueden realizar sobre el ordenador definen el comportamiento de éste. Algunas acciones podrían ser:

Encender ordenador: Comprueba si el ordenador está ya encendido; si lo está, simplemente visualiza un mensaje que lo indica; si no lo está, enciende el ordenador y la pantalla y lo comunica mediante un mensaje.
Apagar ordenador: Comprueba si el ordenador ya está apagado: si lo está, muestra un mensaje para indicarlo; si no lo está, apaga el ordenador y lo comunica mediante un mensaje.
Desactivar la pantalla: Comprueba si la pantalla está desactivada: si lo está, muestra un mensaje para indicarlo; si no lo está, desactiva la pantalla y lo comunica mediante un mensaje.
Activar la pantalla: Comprueba si la pantalla está ya encendida; si lo está, simplemente visualiza un mensaje que lo indica; si no lo está, activa la pantalla y lo comunica mediante un mensaje.
Estado del ordenador: Muestra el valor de los atributos CMarca, CProcesador y CPeso del ordenador y muestra un mensaje indicando si la pantalla y el ordenador están encendidos.

Establecer que cuando se ejecuta el módulo directamente (no es llamado por otro módulo), realice lo siguiente:

Cree dos objetos de la clase ordenador. Ordenadorcasa y Ordenadortrabajo
Asignar valores a los atributos de cada uno de los ordenadores.
Encender el ordenador del trabajo y apagar el ordenador de casa.
Mostrar el estado de los dos ordenadores
Apagar la pantalla del ordenador del trabajo.
Visualiza el estado del ordenador del trabajo.
Apaga el ordenador del trabajo.

1.b).- Modificar el ejercicio anterior para:

- Tener en cuenta que la propiedad Cpeso es una propiedad privada y debe accederse a través de su propiedad virtual Peso. Al asignarles valor, estos procedimientos deben comprobar que al peso no se le asigna un valor < 1 .
- Sustituir el método Estado del ordenador por `__str__`

2).- Cree una clase Ccoche que represente coches. Incluya las propiedades marca, modelo y color y los métodos que simulen las acciones Arrancar motor, acelerar, subir marcha, bajar marcha, frenar, y parar motor.

Establecer que cuando se ejecuta el módulo directamente (no es llamado por otro módulo), cree un objeto coche. Arrancad el coche, acelerarlo, subir marcha, acelerar más, volver a subir marcha, frenar, bajar marcha, frenar bajar marcha y parar el motor. Se irán poniendo mensaje que muestren el cambio de estado en cada momento.

3).- Crear una clase Persona con los siguientes miembros:

```
-----Propiedades-----  
nif  
nombre  
apellidos  
edad  
-----Métodos-----  
medad  
mnombre  
mnif()  
mapellidos()  
mostrar nombreCompleto
```

En la clase persona el método constructor podrá tener valores por defecto.

Crear una clase derivada de la clase persona que se llame Alumno. Esta clase heredará los métodos de la clase base y además tendrá una propiedad curso y un método mcurso() para visualizar el curso en el que se encuentra el alumno. Hay que redefinir en la clase derivada el método mostrar() ya que además de mostrar los atributos de la clase persona debe mostrar el curso del alumno.

Avanzados:

1).- Diccionario

Modelar la Clase Diccionario. Un Diccionario dispondrá de como mínimo la siguiente información:

- Nombre del diccionario
- Editorial
- Nivel
- Palabras y cada palabra podrá tener una o varias acepciones.

Se permitirá mediante un menú:

- Crear el diccionario,
- Introducir las palabras
- Introducir acepciones a una palabra (en cualquier momento),
- consultar palabras y sus acepciones.
- sacar las palabras que empiecen por una letra ordenadas alfabéticamente

2).- La empresa informática “IPMTech” necesita llevar un registro de todos sus empleados que se encuentran en la oficina central, para eso ha creado un diagrama de clases que debe incluir lo siguiente:

A) Empleado

Atributos:

- nombre: tipo cadena (Debe ser nombre y apellido)
- edad : entero (Rango entre 18 y 45 años)
- salario: tipo numérico doble

Métodos:

- Constructor con y sin parámetros de entrada (no hay dos constructores, es poner lo de los parámetros opcionales)
- Método que permita mostrar la clasificación según la edad de acuerdo al siguiente

algoritmo:

Si edad es menor o igual a 21, Principiante

Si edad es ≥ 22 y ≤ 35 , Senior.-

- Imprimir los datos del empleado por pantalla (utilizar salto de línea \n para separar los atributos.

- Un método que permita aumentar el salario en un porcentaje que sería pasado como parámetro al método.

B) Programador (Especialización de Empleado). Clase que hereda de Empleado todos los atributos y métodos.

- Atributos:

- lineasDeCodigoPorHora : tipo entero
- lenguajeDominante: tipo cadena

- Metodos:

- Constructor con y sin parámetros de entrada. (no hay dos constructores, es poner lo de los parámetros opcionales)

Implementar mediante una lista de empleados y un menú la funcionalidad necesaria para dar crear empleados de los diferentes tipos y poder aumentar su sueldo.

3).- Crear una aplicacion para gestionar un videoclub. El videoclub cuenta con varios tipos de producto

- Todos los productos tienen:
Referencia (Titulo, tipo(pelicula/videojuego), precio alquiler, plazo alquiler (días), alquilado (si/no)
- Pelicula
Genero (accion, fantastica, drama, aventuras, puzzle, infantil), año, director, interpretes.
- Videojuego
Estilo (accion, deportes, aventuras, puzzle, infantil), plataforma (Xbox, playstation, wii)
- Se mantiene un listado de clientes
Nº cliente, nombre, direccion, telefono, productos alquilado
- Se guarda un listado de registros de alquiler
 - Cliente, producto, fecha alquiler, fecha devolucion, importe

Crear una aplicacion de consola con el siguiente menú:

- Lista productos
- Añadir producto
- Ficha producto
- Lista clientes
- Añadir cliente
- Ficha cliente
- Alquiler producto

4).- Este ejercicio trata de modelar un banco.

- El banco consiste de una lista de cuentas y una lista de clientes.
- De los **clientes** tenemos su DNI, nombre, apellidos, dirección y teléfono.
- Sobre un banco podemos efectuar una revisión de todas las cuentas del mismo.
- En cada cuenta podemos ingresar dinero, sacar dinero, consultar saldo, cambiar el cliente, efectuar una revisión mensual.
- La revisión mensual de la cuenta hace lo siguiente:
$$\text{Saldo} = \text{saldo} + \text{interes_cuenta} - \text{comision}$$
- Todas las cuentas tienen una comisión de 0,6€ mensuales que puede ser modificada.

Hay tres tipos de cuenta: CuentaCorriente (CC), CuentaVivienda (CV) y FondoInversion (FI).

La CC tiene un interés del 0.1, la CV del 0.2 y el FI de un 0.34.

De la CC no se puede sacar más dinero del que hay, del FI se pueden sacar hasta 500€ de números rojos (si se supera la cuenta se bloquea) y de la CV no se puede sacar dinero.

La revisión mensual de la CV no paga comisión bancaria.

La cuenta de puntos, cada vez que se ingresan 6€ se suma un punto. Se pueden ver cuántos puntos lleva la cuenta.

Hay que implementar el modelo del banco, y crear una la funcionalidad de las clases.

5).-

- A) Implementar la clase **Asignatura** que represente el nombre de una asignatura y la nota correspondiente obtenida.

Las operaciones son:

Constructor que acepte como parámetros el nombre de la asignatura y la nota obtenida.

Métodos para modificar la nota (setNota) y para consultar la nota (getNota).

Método que nos devuelva “Aprobado” si la nota es mayor o igual a 60 o “Reprobado” si la nota es menor que 60.

Método para consultar el nombre de la asignatura.

- B) Implementar la clase **Alumno** que incluya una colección de Asignaturas a las que el alumno ha asistido. Además de incluir los atributos nombre y edad. Las operaciones disponibles sobre el alumno son:

Constructor que acepte como parámetro el nombre del alumno y edad.

Métodos para modificar el nombre (setNombre) y para consultarlo (getNombre).

Métodos para modificar y consultar la edad.

Método que nos devuelva el promedio del alumno.

Método para agregar una Asignatura a su plan de estudio; verificar que la asignatura no exista previamente en el arreglo de este Alumno.

- C) Realizar las siguientes operaciones sobre las clases Alumno y Asignatura.

1. Crear una lista de alumnos (por ejemplo tres instancias de la clase Alumno) con sus respectivos nombre y edad.
2. Para cada alumno establecer sus asignaturas y la nota obtenida.
3. Imprimir en pantalla:
 1. Nombre del alumno.
 2. Edad.
 3. Asignaturas que cursó: Nombre de la asignatura. Nota obtenida. Si es una asignatura aprobada o no.
 4. Promedio del alumno