



Universidad Autónoma de Nuevo León
Facultad de Ingeniería Mecánica y Eléctrica



LENGUAJES DE PROGRAMACIÓN Y LAB

A9-Ej. Lenguaje orientado a objetos

Docente: Ing. Karla Patricia Uribe Sierra

Alumno: Bernardo Gutiérrez López

Matrícula:2222904

Grupo:005

Día y Hora: M5

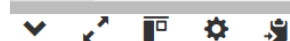
Fecha de elaboración: 13/05/2025

Introducción

La programación orientada a objetos (POO) es un paradigma que organiza el código en torno a clases y objetos, permitiendo representar conceptos del mundo real de manera estructurada. Este paradigma facilita el mantenimiento, la reutilización del código y la escalabilidad del software. En este reporte se explicarán los conceptos de tipos de datos, objetos y encapsulación, utilizando un único programa escrito en Python que simula el manejo de información de un alumno.

Programa

```
1 class Alumno:
2     def __init__(self, nombre: str, calificacion: int):
3         # Atributos privados (encapsulados)
4         self.__nombre = nombre          # Tipo de dato: str
5         self.__calificacion = calificacion # Tipo de dato: int
6
7     # Getter para nombre
8     def get_nombre(self) -> str:
9         return self.__nombre
10
11    # Setter para nombre
12    def set_nombre(self, nombre: str):
13        self.__nombre = nombre
14
15    # Getter para calificación
16    def get_calificacion(self) -> int:
17        return self.__calificacion
18
19    # Setter para calificación
20    def set_calificacion(self, calificacion: int):
21        self.__calificacion = calificacion
22
23    # Método que evalúa si está aprobado
24    def esta_aprobado(self) -> bool:
25        return self.__calificacion >= 70 # Tipo de dato: bool
26
27    # Crear un objeto de la clase Alumno
28    alumno1 = Alumno("Lucía", 82)
29
30    # Mostrar información usando getters
31    print("Nombre:", alumno1.get_nombre())          # Tipo de dato: str
32    print("Calificación:", alumno1.get_calificacion()) # Tipo de dato: int
33
34    # Evaluar aprobación
35    if alumno1.esta_aprobado():                      # Tipo de dato: bool
36        print("Estado: Aprobado")
37    else:
38        print("Estado: Reprobado")
39
```



input

Nombre: Lucía

Calificación: 82

Estado: Aprobado

Tipos de datos utilizados

Elemento	Tipo de dato	Descripción
nombre	str	Cadena de texto que almacena el nombre del alumno.
calificación	int	Número entero que representa la nota.
esta_aprobado()	bool	Devuelve True o False según la nota.

Objeto

- El objeto alumno1 es una instancia de la clase Alumno.
- Tiene estado (nombre, calificacion) y comportamiento (esta_aprobado())

Encapsulación

- Los atributos __nombre y __calificacion están encapsulados (privados).
- Se accede a ellos mediante métodos públicos: get_ y set_, lo cual:
 - >Protege la integridad de los datos.
 - >Permite aplicar reglas (como validar rangos, si se deseara).

Conclusión

Este programa muestra cómo un solo ejemplo puede reflejar los tres conceptos clave de la programación orientada a objetos:

- Tipos de datos, para manejar información estructurada.
- Objetos, como instancias que encapsulan atributos y métodos.
- Encapsulación, para proteger los datos internos y permitir su acceso controlado.

Gracias al paradigma (estilo de programación) POO, este programa es fácil de mantener, reutilizar y ampliar, siendo una base sólida para desarrollar aplicaciones más complejas con buena estructura y claridad.