

# Tarea: Modelos de Programación y Bases de Datos

MARIANA MORENO SANTOYO

February 2, 2024

## 1 Modelo Orientado a Objetos

### 1.1 Descripción

El modelo orientado a objetos (OO) es un paradigma de programación basado en el concepto de "objetos", que son instancias de "clases". Estos objetos pueden contener datos, en forma de campos, y código, en forma de métodos.

### 1.2 Ventajas

- **Reutilización de código y mantenibilidad:** Facilita la reutilización de código y la modularidad, lo que mejora la mantenibilidad.
- **Abstracción y encapsulación:** Permite una representación simplificada de la realidad y oculta los detalles de implementación.
- **Herencia y polimorfismo:** Ofrece flexibilidad y extensibilidad en el diseño.

### 1.3 Desventajas

- **Curva de aprendizaje:** Puede ser complejo para principiantes.
- **Consumo de recursos:** Requiere más memoria y procesamiento.
- **Complejidad en depuración:** Los errores pueden ser difíciles de rastrear debido a la interacción entre objetos.

### 1.4 Casos de Uso

Utilizado ampliamente en el desarrollo de software, especialmente en aplicaciones de escritorio, web y móviles, así como en sistemas de gestión de bases de datos y videojuegos.

## 2 Modelos NoSQL

### 2.1 Descripción General

Los modelos NoSQL representan un enfoque alternativo a las bases de datos relacionales tradicionales. Están diseñados para manejar grandes volúmenes de datos distribuidos y proporcionan flexibilidad en términos de esquemas y estructuras de datos. NoSQL se destaca por su capacidad para escalar horizontalmente y manejar una variedad de estructuras de datos, como clave-valor, documentales y grafos.

### 2.2 Clave-Valor

**Descripción:** Almacena datos en pares de clave-valor.

**Ventajas:** Alta velocidad y escalabilidad.

**Desventajas:** Limitado para consultas complejas.

**Casos de Uso:** Almacenamiento de sesiones, sistemas de caché.

### 2.3 Documentales

**Descripción:** Organiza datos en documentos (formatos JSON/XML).

**Ventajas:** Flexibilidad para datos semi-estructurados.

**Desventajas:** Menos eficiente en relaciones complejas.

**Casos de Uso:** Sistemas de gestión de contenido, comercio electrónico.

### 2.4 Grafos

**Descripción:** Utiliza nodos y aristas para representar relaciones complejas.

**Ventajas:** Eficiente en mapeo de relaciones complejas.

**Desventajas:** Mayor complejidad en diseño.

**Casos de Uso:** Redes sociales, sistemas de recomendación.

## 3 Referencias

1. El Blog Python, "Modelado orientado a objetos: clave en el desarrollo de software," Disponible en: [elblogpython.com](http://elblogpython.com)
2. UNAM, "Modelo Orientado a Objetos," Disponible en: [programas.cuaed.unam.mx](http://programas.cuaed.unam.mx)
3. El Mundo Infinito, "Programación orientada a objetos - ventajas y desventajas," Disponible en: [elmundoinfinito.com](http://elmundoinfinito.com)
4. Proscont, "Ventajas y desventajas de programación orientada a objetos," Disponible en: [www.proscont.com](http://www.proscont.com)
5. Ventajas y Desventajas Top, "Bases de Datos Orientadas a Objetos: Un Análisis de sus Ventajas y Desventajas," Disponible en: [ventajasydesventajastop.com](http://ventajasydesventajastop.com)

6. El Blog Python, "Ventajas y aplicaciones de las bases de datos orientadas a objetos," Disponible en: [elblogpython.com](http://elblogpython.com)
7. Stackscale, "Bases de datos NoSQL: características y tipos," [En línea]. Disponible en: [www.stackscale.com](http://www.stackscale.com)
8. The Data Schools, "NoSQL: Modelos de base de datos no relacionales," [En línea]. Disponible en: [thedataschools.com](http://thedataschools.com)
9. EDteam, "Tipos de bases de datos NoSQL," [En línea]. Disponible en: [ed.team](http://ed.team)