**Patrón de Creación Object Pool**  
  
Jarvic Balanta Santacruz  
Universidad Cooperativa de Colombia  
Facultad de Ingeniería  
Patrones de Diseño Orientado a Objetos

# Resumen

El patrón **Object Pool** es un patrón de diseño creacional que permite la reutilización de objetos en lugar de crear nuevas instancias cada vez que se necesitan. Este patrón es útil en escenarios donde la creación de objetos es costosa en términos de rendimiento y recursos, como en conexiones a bases de datos, hilos de ejecución o conexiones de red

# Introducción

En el desarrollo de software, la gestión eficiente de los recursos es crucial para garantizar un rendimiento óptimo. **Object Pool** es un patrón de diseño creacional que permite administrar un conjunto de objetos ya inicializados y disponibles para su reutilización, evitando la sobrecarga de creación de nuevas instancias. Este patrón se aplica en sistemas de alto rendimiento donde se requiere optimizar el uso de objetos costosos y controlar el consumo de memoria.

Este documento explora su definición, cuándo aplicarlo, sus ventajas y desventajas, y ejemplos de implementación en distintos sectores, proporcionando un enfoque práctico para su adopción en escenarios del mundo real.

**¿Cuándo utilizar Object Pool?**

* Cuando la creación de objetos es costosa en términos de tiempo y recursos.
* En aplicaciones donde los objetos se reutilizan frecuentemente.
* Cuando es necesario controlar la cantidad de instancias activas para evitar sobrecarga en la memoria.

**¿Cuándo NO utilizar Object Pool?**

* Si los objetos son livianos y su creación no afecta el rendimiento.
* Cuando cada objeto debe mantener un estado único y personalizado.
* En sistemas con alto nivel de concurrencia sin sincronización adecuada.

# Resultados

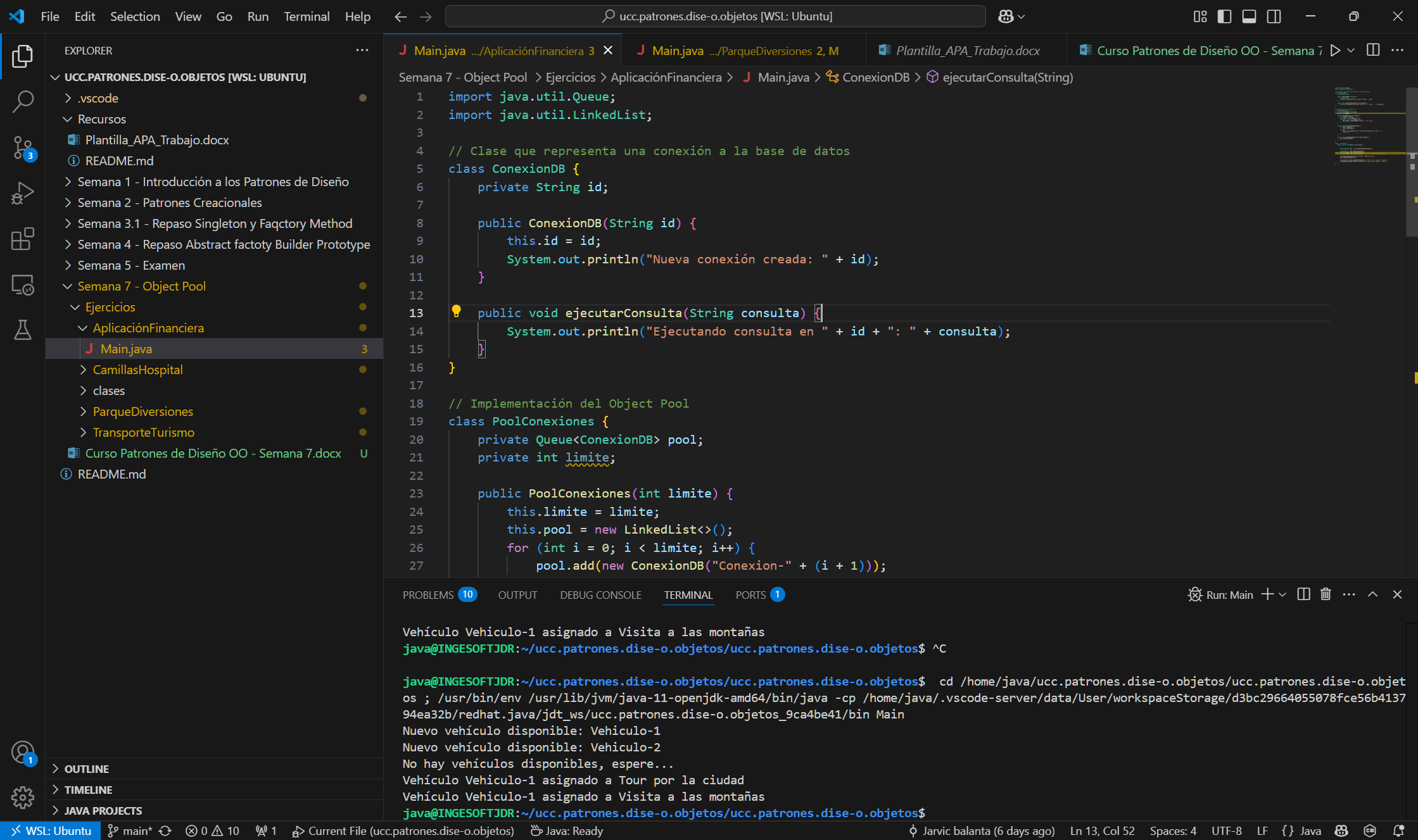


Ilustración 1. Resultados Implementacióm.

# Conclusiones

El patrón **Object Pool** es una solución efectiva para optimizar el rendimiento en sistemas que requieren la reutilización de objetos costosos. Sin embargo, su implementación debe considerar factores como la administración eficiente del pool, la sincronización en entornos concurrentes y la correcta liberación de los objetos para evitar fuga de memoria.

En los ejercicios prácticos, se observó su utilidad en sectores financieros, turísticos y médicos, demostrando cómo puede mejorar el rendimiento y la escalabilidad de las aplicaciones cuando se usa correctamente.

# Referencias

Bolaños Rodríguez, H. A. (2025). Curso Patrones de Diseño OO - Semana 7. Universidad Cooperativa de Colombia.

.