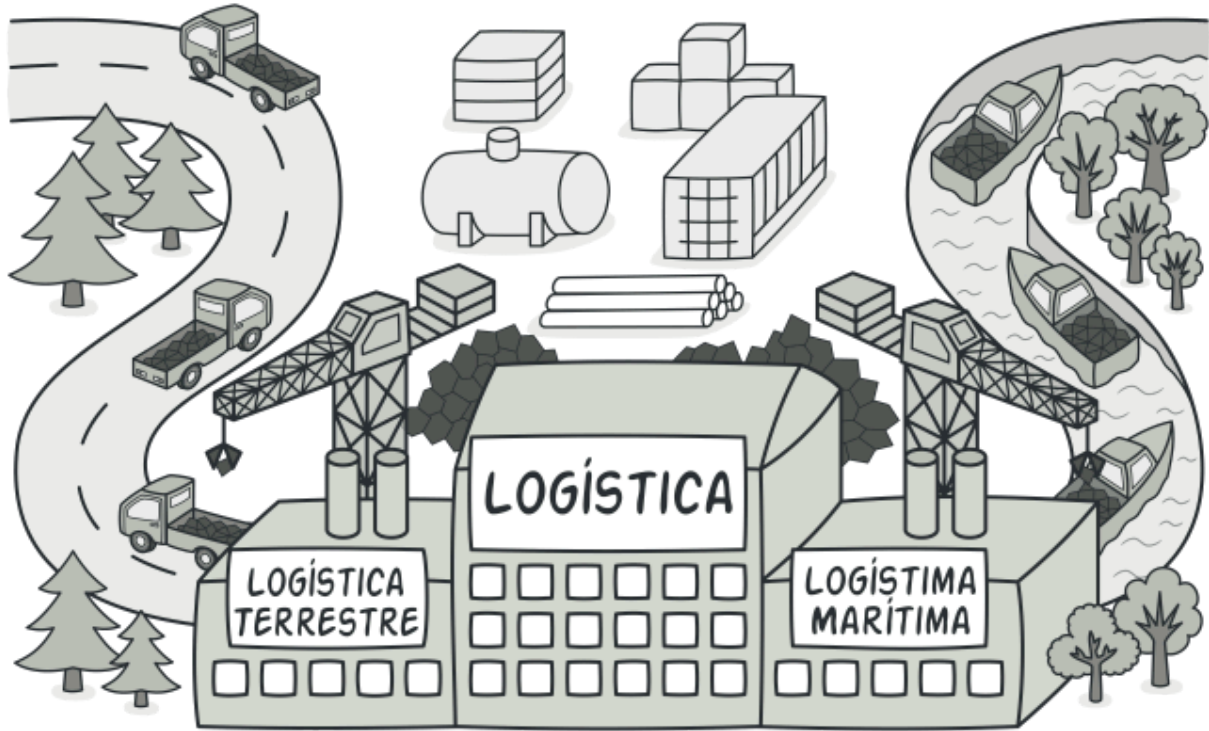


Ejemplo de diseño *Patrón Factory*



Autora: Rashi Chugani Narwani

Planteamiento del Problema

Para abordar el desarrollo de esta aplicación, se requiere la capacidad de crear y gestionar diversos tipos de vehículos, como coches y motos, con los siguientes criterios: la aplicación debe generar instancias de vehículos sin necesidad de especificar clases concretas en el código, permitiendo así una flexibilidad en la creación; debe ser sencillo añadir nuevos tipos de vehículos sin alterar el código existente que controla la creación de estos; además, se busca centralizar la creación de vehículos en una única clase para mantener la organización del código y facilitar su mantenimiento.

Solución con el Patrón de Diseño *Factory*

Para cumplir con estos requisitos, optaremos por la implementación del patrón de diseño 'Factory'. Este enfoque nos permitirá encapsular la lógica de creación de objetos, garantizando así que nuestra aplicación sea flexible y sencilla de mantener. Con el uso de este patrón, podremos centralizar la creación de instancias de vehículos, lo que facilitará la incorporación de nuevos tipos de vehículos en el futuro sin necesidad de modificar el código existente. Además, el patrón 'Factory' promoverá un diseño modular y organizado, lo que contribuirá a la escalabilidad y la claridad del código en nuestro proyecto de gestión de vehículos.

