



# Abstract Factory

# *Abstract Factory*

O Abstract Factory resolve o problema de criar famílias inteiras de produtos sem especificar suas classes concretas.

O Abstract Factory define uma interface para criar todos os produtos diferentes, deixando a criação real do produto para classes fábrica concretas. Cada tipo de fábrica implementada corresponde a uma determinada variedade de produtos.

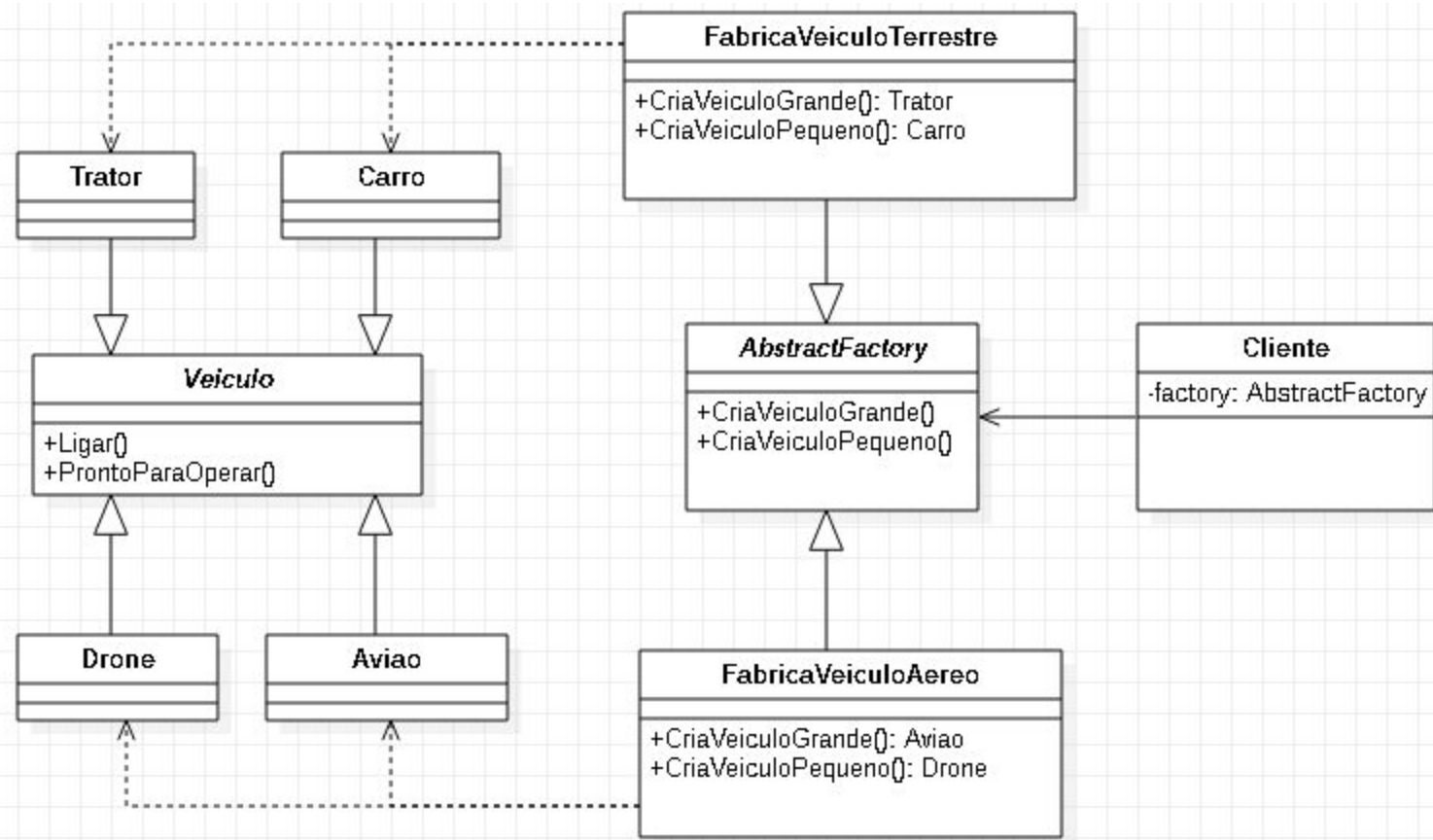
O código cliente chama os métodos de criação de um objeto fábrica em vez de criar produtos diretamente. Como uma fábrica corresponde a uma única variante de produto, todos os seus produtos serão compatíveis.

# Vantagens

- **Isolamento de classes:** O padrão Abstract Factory ajuda a controlar as classes de objetos que um aplicativo cria. Como uma fábrica encapsula a responsabilidade e o processo de criação de objetos de produto, ela isola os clientes das classes de implementação. Os clientes manipulam as instâncias por meio de suas interfaces abstratas. Os nomes das classes de produtos são isolados na implementação da fábrica de concreto; eles não aparecem no código do cliente.
- Útil quando o cliente não sabe que tipo irá criar.
- Consistencia entre os objetos.
- Isola o código do cliente das classes concretas.
- Facilita a troca de famílias de objetos.

# Desvantagem

- **Modificar interface:** Qualquer modificação na interface vai forçar a modificação na classe e nas sub-classes.
- **Extender interface:** A extensão da interface para produzir novos produtos não é simples, visto que a interface define quais produtos serão criados, e para adaptar ou modificar devido ao novo produto vai ocorrer a desvantagem citada acima.
- Complexidade extra durante o início da criação do código.



Perguntas

Obrigado