

# Factory Method

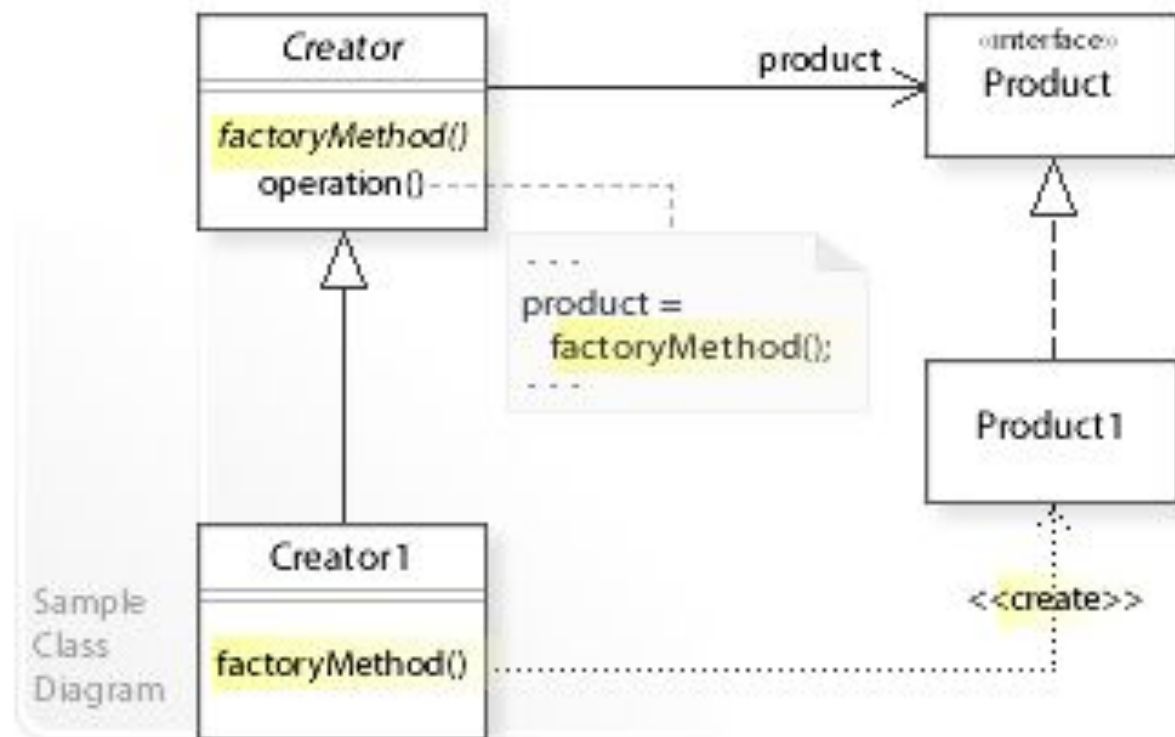
# Propósito

Permite crear objetos en una superclase, mientras deja a las subclasses “decidir” en tiempo de ejecución qué objeto específico crear.

Al evitar definir estáticamente cuando crear un objeto y cuando otro, elimina el acoplamiento entre clases.

Facilita el principio de “agregar código en vez de modificarlo”.





# Forma

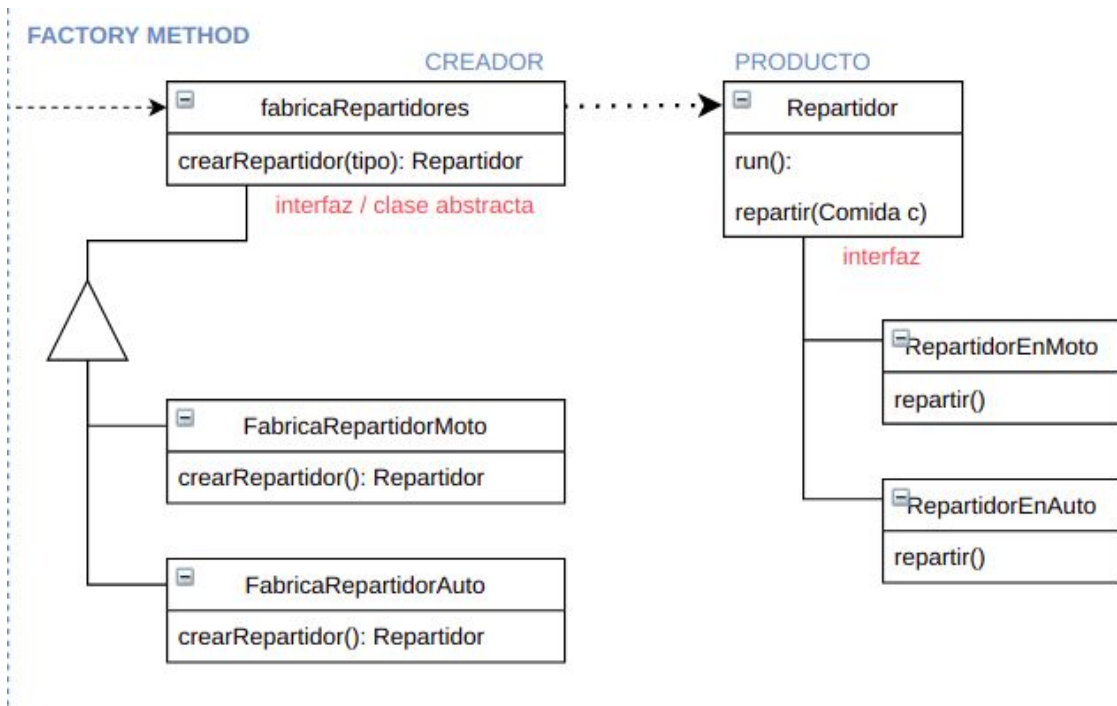
Define una clase abstracta constructora. En esta subclase se define una interfaz para la construcción de objetos tipo “Producto” pero no se implementa, las subclases redefinen ese método y cada una crea el Producto específico que le convenga.

Producto también es una clase abstracta con subclases concretas.

Esto evita definir “if else” o “switches” en la clase padre o la implementación a la hora de elegir un subtipo de producto en cada momento.



# Nuestra implementación



Con estas clases, los programas que utilicen las fabricas de repartidores no se tienen que preocupar de la lógica de crear un repartidor, ya que sea cual sea la fábrica, la interfaz será la misma y devolverá el repartidor que corresponde