Study of Factory pattern.

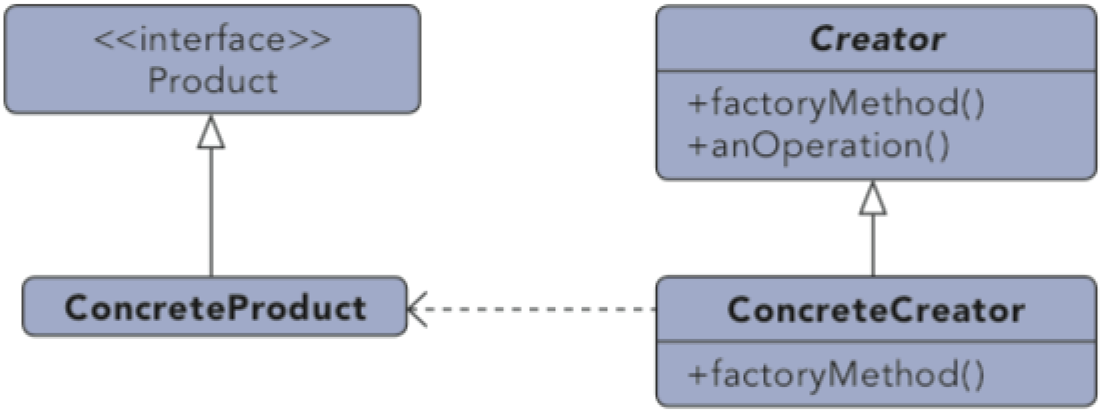
Factory pattern is a creational pattern. It returns instance of several possible classes depending of the data provided to it.

It specifies that the classes that it returns have the same parent class and methods but have different implementations in the subclass.

It defines an interface for creating an object, but lets the subclass decide which class to instantiate.

Factory pattern is specifically used when our application does not know what kind of object it wants to create at run-time but just want to get an object to get the work done.
It introduces loose coupling among classes.

Factory classes are useful when there are complex object creation steps involved. It ensures that these steps are centralized and not exposed to composing classes.



Product : It defines the interface of objects that are created by the factory method.
ConcreteProduct: It implements the Product interface.
Creator : It declares the factory method, which returns an object of type Product.
Creator relies on its subclasses to define the factory method so that it returns an instance of the appropriate ConcreteProduct
ConcreteCreator : It overrides the factory method to return an instance of a ConcreteProduct.