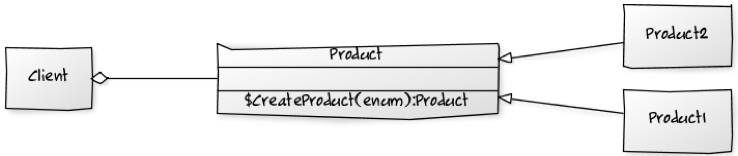
The **Factory pattern** **creates objects without exposing the instantiation logic to the client**. Refers to the newly created object through a common interface.

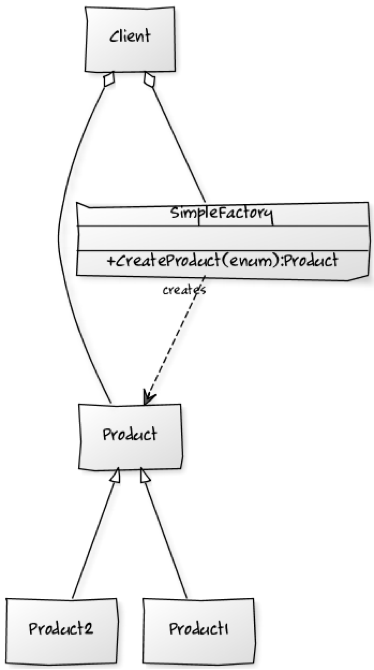
**Static Factory**

Static factory is a class with a **Static Method to produce various subtypes** of Product.



**Simple Factory**

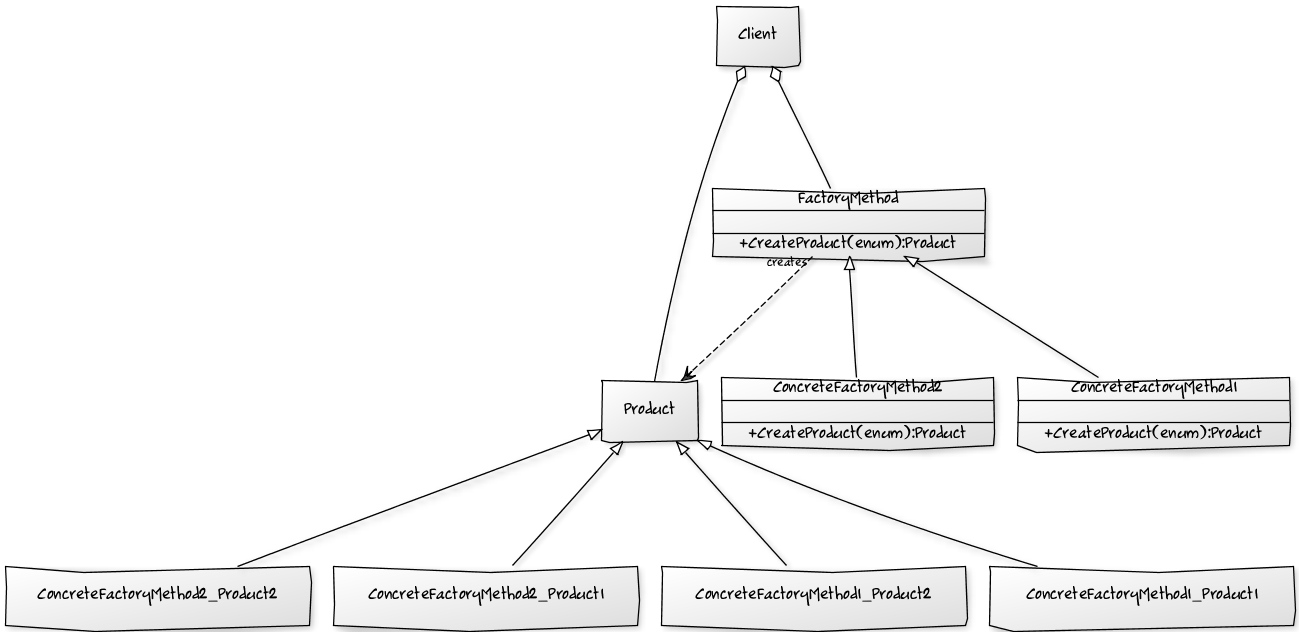
Simple factory is **a class that can produce various subtypes** of Product. (It is better than the Static Factory. When new types are added the base Product class does not need to be changed only the Simple Factory Class).



**Factory Method**

Factory Method **defines an interface for creating objects, but lets subclasses decide which class to instantiate**.

Contains one method to produce one type of product related to its type. (It is better than a Simple Factory because the type is deferred to a sub-class).



**Abstract Factory**

Abstract Factory **defines the interface for creating a family of related objects, without explicitly specifying their classes**. It is noticeably different from a Factory Method as it has more than one method of type it produces.

