## Factory Pattern

## Factory Pattern là gì?

Factory Method Design Pattern, hay còn được gọi là Factory Pattern là một trong những Pattern phổ biến trong [lập trình hướng đối tượng](https://chiasekinang.com/lap-trinh-huong-doi-tuong/).

Factory Pattern thuộc nhóm Creational Design Pattern (mẫu khởi tạo). Vì nhiệm vụ của mẫu này là quản lý và trả về các đối tượng theo yêu cầu, cung cấp một trong những cách tốt nhất để tạo một đối tượng. Giúp cho việc khởi tạo đổi tượng một cách linh hoạt hơn.

Factory pattern là một trong những mẫu thiết kế được sử dụng nhiều nhất trong Java.

Trong Factory Pattern, chúng ta tạo đối tượng mà không để lộ logic tạo đối tượng ở phía người dùng và tham chiếu đến đối tượng mới được tạo ra bằng cách sử dụng một interface chung.

## Factory Design Pattern có lợi ích gì?

* Giúp việc khởi tạo các Objects mà che giấu đi xử lí logic của việc khởi tạo đó. Người dùng không biết logic thực sự được khởi tạo bên dưới phương thức factory.
* Mẫu thiết kế này cho phép các lớp con chọn kiểu đối tượng cần tạo.
* Nó thúc đẩy sự liên kết lỏng lẻo bằng cách loại bỏ sự cần thiết phải ràng buộc các lớp cụ thể vào code. Nghĩa là code chỉ tương tác với interface hoặc lớp abstract, để nó sẽ làm việc với bất kỳ lớp nào implements interface đó hoặc extends lớp abstract.
* Factory Pattern giúp giảm sự phụ thuộc giữa các module: cung cấp 1 hướng tiếp cận với Interface thay vì các implement. Giúp chuơng trình độc lập với những lớp cụ thể mà chúng ta cần tạo 1 **đối tượng**, code ở phía client sẽ không bị ảnh hưởng khi thay đổi logic ở factory hay sub class.
* Việc mở rộng code dễ dàng hơn: khi cần mở rộng, chỉ việc tạo ra những sub class và implement thêm vào factory method.
* Dễ dạng quản lý life cycle của các Object được tạo bởi **Factory Method Pattern**.
* Thống nhất về mặt naming convention: giúp cho các developer có thể hiểu về cấu trúc source code.

## Áp dụng vào đâu ?

* “Factory method” thường được sử dụng trong việc thiết kế toolkit hay **framework**, đoạn mã của framework cần thiết phải tạo ra một đối tượng là những lớp con tương ứng tăng tính mềm dẻo trong sử dụng framework đó.
* Cách sử dụng, khi nào bạn nên sử dụng mẫu thiết kế Factory Method

## Ưu điểm và nhược điểm của Factory Method Pattern

### Ưu điểm

• Factory Method Pattern giúp hạn chế sự phụ thuộc giữa creator và concrete products.

• Factory Method Pattern giúp gom các đoạn code tạo ra product vào một nơi trong chương trình, nhờ đó giúp dễ theo dõi và thao tác.

• Với Pattern này, chúng ta có thể dễ dàng mở rộng, thêm những đoạn code mới vào chương trình mà không cần phá vỡ các đối tượng ban đầu .

### Nhược điểm

Ngoài những ưu điểm nên trên thì mẫu thiết kế này cũng có nhược điểm như sau:

• Mã code có thể trở nên phức tạp hơn mức bình thường do đòi hỏi phải sử dụng nhiều class mới có thể cài đặt được pattern này.

## Nên sử dụng Factory Pattern khi nào?

Dựa vào những lợi ích của việc sử dụng Factory pattern mà ta sẽ dùng chúng với một số mục đích sau:

• Khi tạo ra 1 cách mới trong việc khởi tạo Object => Tất nhiên là vậy.

• Che giấu xử lý logic của việc khởi tạo => Trong trường hợp bạn đang muốn viết 1 thư viện để cho người khác sử dụng.

• Giảm bớt sự phụ thuộc => Dễ dàng cho việc mở rộng trong trường hợp bạn chưa biết chắc số lượng các đối tượng là đã đủ cho bài toán của mình chưa.

## Mục đích sử dụng

Tại sao chúng ta lại phải dùng Factory pattern thay cho việc khởi tạo đối tượng thông thường? Dùng Factory pattern có lợi ích gì?

Factory pattern đưa ra 1 ý tưởng mới cho việc khởi tạo các instance phù hợp với mỗi yêu cầu từ phía Client. Sử dụng Factory pattern sẽ có những ưu điểm sau:

* Tạo ra 1 cách mới trong việc khởi tạo Object thông qua 1 interface chung.
* Khởi tạo các Objects mà che giấu đi xử lí logic của việc khởi tạo đấy.
* Giảm sự phụ thuộc giữa các module, các logic với các class cụ thể. Mà chỉ phụ thuộc vào interface hoặc abstract class.

Ví dụ mẫu thiết kế Factory Method

