Factory để tạo Validator

*Basic validators cho các kiểu dữ liệu cơ bản như int, string, float,…*

**1. Cơ chế thông báo khi data không valid**

• Tổng hợp các thông báo => validator.validate(obj). Tạo 1 class như ValidateResult

• Có các cách khác nhau để thể hiện thông báo trên giao diện => Xem C2 của 2. Khi gọi onError.

**2. Thao tác thiết lập valid bằng code**

• Tự kiểm tra với code thêm vào

=> Validatable and Validator. Sử dụng *Visitor*

<https://lostechies.com/jimmybogard/2007/10/24/entity-validation-with-visitors-and-extension-methods/>

Dependency injection để lấy validator

C2: Có thể sử dụng AbstractValidator:

Ex:

**public** **class** **UserValidator** : AbstractValidator<User>

{

**public** UserValidator()

{

RuleFor(x => x.Name).NotNull();

}

}

<https://docs.fluentvalidation.net/en/latest/start.html>

• Template method. Tạo các RuleFor(x = x.Name)… và validate(T t) để trả về ValidationResult. Có thể thêm 1 function onError để trả về kết quả.

**3. Thao tác thiết lập valid tự động thông qua khai báo ràng buộc dữ liệu**

• Ví dụ sử dụng attributes trong .NET => annotation Java

**4. Có thể kết hợp các validation với nhau cho cùng kiểu dữ liệu**

• Ví dụ: vừa thiết lập kiểm tra chuỗi rỗng, chuỗi chỉ toàn ký tự,…

=> *Chain of responsibility*

Có builder để nối các validator lại thành chain.

Các validator thường sẽ implement generic nếu có thể (Như min, max và truyền vào 1 Comparator<> được định nghĩa trước))

**5. Hỗ trợ kết hợp regular expression**

=> Regex for string

**6. Cho phép tạo custom validation**

<https://blog.clairvoyantsoft.com/spring-boot-creating-a-custom-annotation-for-validation-edafbf9a97a4>

https://www.baeldung.com/javax-validation-method-constraints

**Validation framework được xây dựng dựa trên các lớp đối tượng với các phương thức phù hợp**

• Có thể sử dụng reflection hoặc attribute (C#) hoặc annotation (Java) để đọc các thông tin mô tả ràng buộc các trường dữ liệu.

• Xây dựng các lớp cơ bản hỗ trợ validation cho các kiểu dữ liệu cơ bản

• Xây dựng các lớp xử lý để thực hiện validation

# Các loại framework

FluentValidation

JSR380

Commons Validator

<https://github.com/hepisec/java-validation>

# Các loại Constraint:



Classes:

Chain Of Responsibility