Singleton Design Pattern

đây là một thuật ngữ chỉ một lớp chỉ có thể được khởi tạo một lần và cung cấp một điểm truy cập toàn cục đến phiên bản đó.

Trong ngôn ngữ lập trình Java, có một vài cách thực hiện tiêu chuẩn của mẫu Singleton. Dưới đây là một số cách thường sử dụng để thực hiện một Singleton.

public class SingletonClass {

private static SingletonClass instance = null;

private SingletonClass() {

}

public static SingletonClass getInstance() {

if (instance == null) {

instance = new SingletonClass();

}

return instance;

}

}

Trong ví dụ trên, chúng ta viết một lớp với một phương thức tạo một phiên bản mới của lớp nếu không có phiên bản nào tồn tại. Nếu một phiên bản đã được tạo, constructor đơn giản trả về một tham chiếu đến đối tượng đó. Đây là cách thực hiện đơn giản nhất của mẫu thiết kế Singleton trong Java.

Lưu ý rằng constructor của lớp SingletonClass được làm riêng tư, để đảm bảo rằng không có cách nào khác để khởi tạo lớp. Ví dụ này được biết đến là khởi tạo lười biếng - có nghĩa là nó giới hạn việc tạo ra phiên bản cho đến khi được yêu cầu lần đầu tiên.

https://springframework.guru/gang-of-four-design-patterns/singleton-design-pattern/