|  |  |
| --- | --- |
| SOLID là gì ? | Bao gồm:   * Single Responsibility (nguyên tắc Đơn trách nhiệm) * Open/Closed (nguyên tắc Mở/Đóng) * Liskov Substitution (nguyên tắc Thay thế Liskov) * Interface Segregation (nguyên tắc Phân tách Interface) * Dependency Inversion (nguyên tắc Đảo ngược phụ thuộc) |
| Ý nghĩa của 5 nguyên lý trong SOLID? | **Single responsibility principle**  Một lớp chỉ nên đảm nhiệm một trách nhiệm duy nhất (Nghĩa là chỉ có thể sửa đổi lớp đó với 1 lý do duy nhất).  **O - Open/closed principle**  Có thể thoái mái mở rộng một lớp, nhưng không được sửa đổi bên trong lớp đó.  **L - Liskov substitution principle**  Trong một chương trình, các đối tượng của lớp con có thể thay thế đối tượng của lớp cha mà không làm thay đổi tính đúng đắn của chương trình.  **I - Interface segregation principle**  Thay vì dùng một interface lớn, ta nên tách thành nhiều interface nhỏ, với nhiều mục đích cụ thể.  **D - Dependency inversion principle**  Các module cấp cao không nên phụ thuộc vào các modules cấp thấp. Cả hai nên phụ thuộc vào abstraction.  Abstraction không nên phụ thuộc vào chi tiết, mà ngược lại (Các lớp giao tiếp với nhau thông qua interface, không phải thông qua triển khai.) |