**BÀI THU HOẠCH DAY 4**

* 1. **Tight couple :**

Khi một các object liên kết với nhau quá chặt chẽ khiến việc thay đổi logic hay một class bị lỗi sẽ ảnh hưởng tới toàn bộ các class khác;

* 1. **Loose couple:**

Giảm bớt sự phụ thuộc giữa các class với nhau bằng cách giao tiếp giữa các class qua interface. Dễ dàng hơn trong việc mở rộng code.

* 1. **Dependency Injection:**

Dependency hay dependent nghĩa là phụ thuộc vào hỗ trợ của một cái gì, việc gì đó. Ví dụ như nếu chúng ta phụ thuộc quá nhiều vào smartphone, thì có thể hiểu là chúng ta đã dependent lên smartphone.

Trước khi nói về dependency injection, hãy hiểu xem dependency trong lập trình nghĩa là gì trước

Khi mà class A sử dụng một số chức năng của class class B, thì có thể nói là class A có quan hệ phụ thuộc với class B

Trong java, trước khi ta có thể sử dụng method của class khác, ta phải khởi tạo một object của class đấy (hay A cần phải tạo 1 thực thể của B).

Vậy ta có thể hiểu, việc chuyển giao nhiệm vụ khởi tạo object đó cho một ai khác và trực tiếp sử dụng các dependency đó được gọi là dependency injection. Có nghĩa là nếu class A phụ thuộc vào B thì cần inject ( tiêm ) B vào A để class A có thể sử dụng class B đó.

Có 3 loại injection:

* **Constructor injection**: các dependency được cung cấp thông qua constructor của class.
* **Setter injection**: client tạo ra một setter method để các class khác có thể sử dụng chúng để cấp dependency.
* **Field injection**: các dependency được cung cấp thông qua trường của class..

**Lợi ích của dependency injection.**

* Giúp viết Unit test dễ dàng hơn.
* Giảm thiểu đc boilerplate code vì việc khởi tạo dependency đc làm bởi một component khác.
* Mở dụng dự án dễ dàng hơn.
* Giúp ích trong việc liên kết lỏng (loose coupling) giữa các thành phần trong dự án.