**1.Why getter and setter methods are evil**

+)Vi phạm Tính Đóng Gói:

- Làm Lộ Trạng Thái Nội Bộ: Getter và setter thường làm lộ các trường dữ liệu bên trong của một đối tượng, cho phép các lớp bên ngoài thay đổi trực tiếp chúng. Điều này vi phạm tính đóng gói, vì các lớp khác có thể truy cập hoặc thay đổi các hoạt động nội bộ của đối tượng thay vì tương tác qua các hành vi có ý nghĩa.

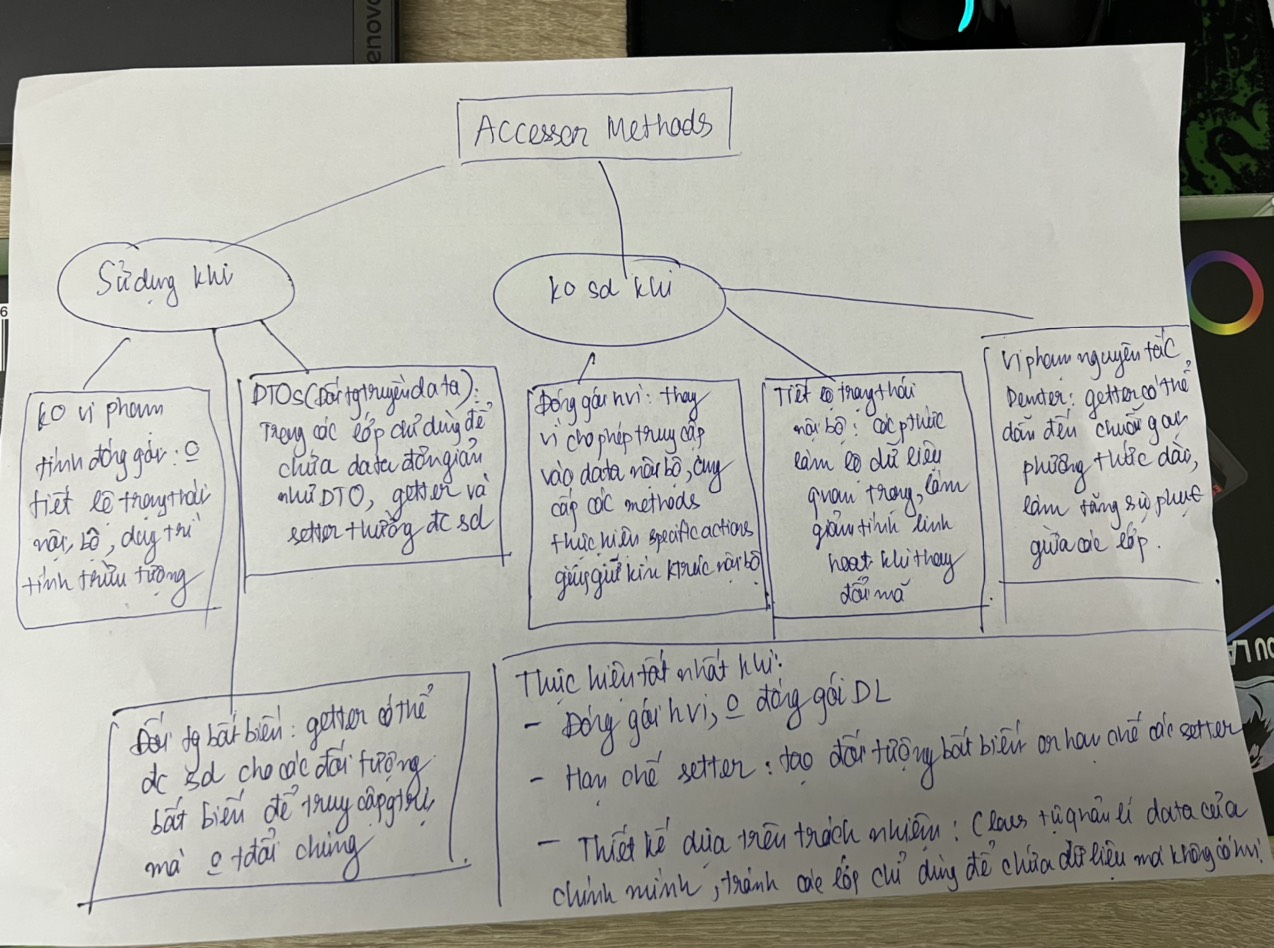
- Phá Vỡ Tính Trừu Tượng: Các đối tượng nên đại diện cho hành vi hơn là chỉ chứa dữ liệu. Khi sử dụng getter và setter, một đối tượng có thể trở thành một "kho chứa dữ liệu", khiến các thực thể bên ngoài kiểm soát hành vi của nó, làm suy yếu tính đóng gói.

+)Mã Thủ Tục trong Hướng Đối Tượng:

Getter và setter thường dẫn đến mã theo phong cách lập trình thủ tục trong mã hướng đối tượng.Điều này có thể khiến mã của bạn đi ngược lại với nguyên tắc hướng đối tượng, nơi mà đối tượng nên đóng gói cả dữ liệu và hành vi.

+)Vi Phạm Nguyên Tắc Demeter:

Getter thường dẫn đến chuỗi gọi phương thức (vídụ:object.getA().getB().getC()), tạo ra sự phụ thuộc chặt chẽ và vi phạm nguyên tắc Demeter, khuyến nghị nên giảm thiểu việc biết quá nhiều về các đối tượng khác. Việc này dẫn đến cấu trúc mã khó bảo trì và cứng nhắc.



**2. If you create a constructor method to build a DVD by title then create a constructor method to build a DVD by category. Does JAVA allow you to do this?**

-> Đúng vậy, Java cho phép bạn có nhiều constructor trong một lớp, một khái niệm được gọi là *nạp chồng constructor*. Điều này có nghĩa là bạn có thể tạo các constructor khác nhau với danh sách tham số duy nhất trong cùng một lớp.