**TÍNH ĐÓNG GÓI (Encapsulation)**

Tính chất này nhằm bảo về đối tượng không bị truy cập từ code bên ngoài vào để thay đổi giá trị các thuộc tính hoặc truy cập trực tiếp.

Tính chất này giúp tăng tính bảo mật cho đối tượng và tránh tình trạng dữ liệu bị hư hỏng ngoài ý muốn.

* Tính chất này được thể hiện qua các bài: access modifier, setters, getters.

**TÍNH TRỪU TƯỢNG (Abstraction)**

Là một tiến trình chỉ nói ra tính năng của người dung, bỏ qua những chi tiết bên trong, các khái niệm sẽ được định nghĩa trong quá trình phát triển. Tính trừu tượng cho phép người lập trình tập trung cốt lỗi cần thiết của đối tượng thay vì quan tâm sự phức tạp bên trong hoặc cách nó hoạt động

* Tính chất này được thể hiện qua các bài: abstract class, interface

**TÍNH KẾ THỪA (Inheritance)**

Tính kế thừa cho phép xây dựng một lớp mới (lớp Con), kế thừa và tái sử dụng các thuộc tính, phương thức dựa trên lớp cũ (lớp Cha) đã có trước đó.

Lớp con sẽ kế thừa tất cả các thành phần của lớp cha, nhờ sự chia sẻ này mới có thể mở rộng những đặc tính sẵn có mà không cần định nghĩa lại (khả năng tái sử dụng).

* Tính chất này được thể hiện qua các bài: Inheritance, overriding, overloading

**TÍNH ĐA HÌNH (Polymorphism)**

Tính đa hình có thể nói luôn tồn tại song song với tính kế thừa.

Khi có nhiều lớp con kế thừa lớp cha nhưng có những tính chất khác nhau thì cũng gọi là đa hình.

Những tác vụ trong cùng một đối tượng có thể thể hiện nhiều cách khác nhau cũng gọi là đa hình.

* Tính chất này được thể hiện qua các bài: Interface, overriding, overloading