DesigParten là gì?

Khái niệm hơi trừu tượng và các khái niệm trong DesigParten trong java.

Yêu cầu cơ bản: Nắm được những phần của Java Core, liên quan đến UML, Diragram, trừu tượng, không phải dễ hiểu. nên đọc qua trước.

Nội dung: Loạt bài

Xác định được các class. Tối ưu các code đó

Có tính mở rộng, khả chuyển, có thể thay đổi trong một số trường hợp thực tế.

**Cách phân loại:**

Phạm vi theo class. Obj

Mục đích sử dụng: Trong quá trình tạo đối tượng - (creational), Cấu trúc của hệ thống – (Structural), cách thức chúng nó giao tiếp với nhau – (Behavioral).

**Bài 2: Java Design Patter Interface Abstract class:**cung cấp những service cho các class khác nhau nếu chúng extends từ class hoặc emplement từ interface. Cùng chung một cách gọi phuong thức nhưng cách gọi của chúng sẽ khác nhau.

VD: anh chàng gọi nước trà đá tại nơi làm việc và ở nhà.

Cùng khai báo những methob khi class muốn dùng phải xây dựng lại nó:

Abstract class: ngoài những phuong thức abstractclass còn có những phương thức không pahỉ là abstractclass. Tuỳ biến tuỳ chọn của client.

Mốt số bạn sẽ đặt câu hỏi khác nhau giữa interface và abstract class là gì. Abs có phương thức khai báo thân hàm và một số absmethod🡪 dùng có cái chung không cần thay đổi. extends chỉ được từ một class mà thôi. Nếu muốn extends nhiều thì ta lại chuyển hướng nó sang thành interface

Interface bắt override lại tất cả. elements nhiều interface.

**Bài 3: Private method. Accessor method, constant Data Manager:**

**Private method:** nếu một phương thức là private. Bản thân các phương thức của class đó mới được sử dụng. tăng tính bảo mật của class.

**Accessor Methods:** phương thức set/get của từng thuộc tính trong nó.

Cho phép chúng ta được truy cập vào thuộc tính thông qua những hàm set/get của nó. Không được truy cập vào trực tiếp thuộc tính trong nó🡪 tăng tính bảo mật trong một class.

**Constant Data Manager**: tạo một class đặt những giá trị không đổi. static final là hằng số tĩnh. Không cần new class mà dùng chính cái tên đó rồi chầm đến từng thuộc tính trong đó.

Bài 4: Factory method:

Phần bắt đầu cho những điều khó trong chương này.

Bai 5: Singleton:

Tìm hiểu về desiger singleton:

khi ta muốn chia sẻ tài nguyên. Thì có nhiều người truy cập máy in. tài nguyên đó cần được đồng bộ hoá. Tài nguyên chia sẻ cho tất cả mọi người. tại một thời điểm có một đối tượng được sử dụng. nó sẽ public cho toàn bộ hệ thống.

Cẩn một đối tượng chỉ trong chính nó.

VD: logger

Để tồn tại duy nhất một đối tượng. dùng từ khoá private cho hàm khởi tạo thì đồng thời nó không cho phép khởi tạo đổi tượng nào khác ngoài từ class nào khác. Nó chỉ tạo duy nhất một đổi tượng tại 1 thời điểm.

Khởi tạo connect đến Database. Hạn chế bao nhiêu tạo đến thì có thể hạn chế những singleton này.

Họ sẽ kết hợp nhiều desiger Patter với nhau.

Cách sử dụng của Singleton: