Bài 4

|  |  |
| --- | --- |
| 1. Nhắc lại khái niệm OOP? Ưu và nhược điểm OOP so với POP? | * OOP (Object-Oriented Programming) là lập trình hướng đối tượng, xây dựng chương trình dựa trên các đối tượng (object), mỗi đối tượng bao gồm dữ liệu và các phương thức xử lý dữ liệu đó. * Ưu điểm :   + OOP : Tái sử dụng mã , dễ bảo trì , tổ chức dữ liệu rõ ràng  + POP : Đơn giản cho các chương trình nhỏ , dễ hiểu khi ít chức năng   * Nhược điểm :   + OOP : phức tạp hơn , tốn thời gian thiết kế  + POP : khó bảo trì với chương trình lớn , không hỗ trợ đóng gói hay kế thừa |
| 1. Nhắc lại 4 tính chất của OOP | + Encapsulation (Đóng gói): che giấu dữ liệu, chỉ cho phép truy cập qua các phương thức.  + Inheritance (Kế thừa): lớp con kế thừa thuộc tính/phương thức từ lớp cha.  + Polymorphism (Đa hình): cùng 1 phương thức có thể thực hiện khác nhau.  + Abstraction (Trừu tượng): ẩn chi tiết, chỉ lộ thông tin cần thiết. |
| 1. Phân biệt class và object | * Class : là khuôn mẫu , định nghĩa , không chiếm bộ nhớ => vidu : Car * Object : là thực thể được tạo ra từ class, chiếm bộ nhớ khi được khởi tạo => vidu : Car myCar = new Car (); |
| 1. Constructor là gì? Đặc điểm constructor trong Java | * Constructor : Là phương thức đặc biệt dùng để khởi tạo đối tượng. Nó trùng tên với class, không có kiểu trả về, và được gọi tự động khi tạo object. * Đặc điểm :   + Trùng tên với class.  + Không có kiểu trả về (void cũng không).  + Tự động gọi khi new.  + Có thể có nhiều constructor (dùng overloading).  + Nếu không khai báo, Java tạo constructor mặc định. |
| 1. Từ khóa this dùng để làm gì? | + Tham chiếu đến đối tượng hiện tại.  + Phân biệt giữa biến instance và biến tham số trùng tên.  + Dùng để gọi constructor khác trong cùng class (this(...)).  + Truyền đối tượng hiện tại vào method/constructor khác. |
| 1. Phân biệt constructor và method | * Constructor :   + Khởi tạo đối tượng, gán giá trị ban đầu.  + Phải trùng với tên lớp  + Không có kiểu trả về (kể cả void).  + Tự động gọi khi tạo đối tượng bằng new.  + Có thể dùng this() để gọi constructor khác.   * Method :   + Thực hiện hành vi hoặc chức năng cụ thể.  + Có thể đặt tên tùy ý.  + Có thể có kiểu trả về hoặc void.  + Phải gọi rõ ràng qua đối tượng hoặc lớp.  + Không dùng this() để gọi method khác. |
|  |  |