|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **1- Nhắc lại khái niệm về OOP? Ưu và nhược điểm OOP so với POP?** | - OOP( Object Oriented Programming) Lập trình hướng đối tượng là đưa các đối tượng ngoài thực tế vào lập trình.   |  |  |  | | --- | --- | --- | | **Loại** | **POP (Procedural Oriented Programming)** | **OOP( Object Oriented Programming)** | | **Ưu điểm** | - Đơn giản dễ hiểu  - Hiệu suất tốt  - Phù hợp với các tác vụ nhỏ , sript nhanh . | - Tổ chức rõ ràng  - Tái sử dụng mã  - Tính mở rộng cao  - Che giấu thông tin  - Tăng khả năng bảo trì và bảo mật | | **Nhược điểm** | - Khó bảo trì , mở rộng  - Khó tái sử dụng code  - Không che giấu dữ liệu tốt | - Học phức tạp hơn  - Tốn nhiều tài nguyên hơn  - Không cần thiết cho trương trình nhỏ | |
| **2- Nhắc lại 4 tính chất của OOP?** | **1- Tính bao gói (encapsulation) :**  Che giấu thuộc tính bên trong của đối tượng và không cho bên ngoài truy cập trự tiếp, chỉ có thể truy cập thông qua các phương thức được cho phép.  + setter thay đổi thuộc tính của đối tượng  + getter lấy thông tin của dối tượng  **2- Tính kế thừa (inheritance) :**  Lớp con có thể kế thừa lại những đặc điểm và hành vi của lớp cha mà không cần phải khai báo lại . Và lớp con có thể mở rộng thêm thuộc tính và hành vi riêng.  **3- Tính đa hình(polymorphism) :**  Cùng một đối tượng sẽ có các thể hiện khác nhau tùy thuộc vào ngữ cảnh . Cùng một phương thức sẽ có các cách triển khai khác nhau tùy thuộc vào các lớp.  **4- Tính trừu tượng (abtraction) :**  Bỏ qua các đặc điểm và hành vi không cần thiết , chỉ quan tâm đến đặc điểm và hành vi cần giải quyết . Chỉ quan tâm đến dữ liệu đầu vào và kết quả đầu ra , không quan tâm đến tiến trình bên trong . |
| **3- Phân biệt class và object?** | |  |  | | --- | --- | | **Object** | **Class** | | Là thể hiện cụ thể , có thể tạo từ class hoặc trực tiếp bằng { } .  Thể hiện cụ thể bằng ojaclass | Là khuôn mẫu để tạo Object , giúp tổ trức code tốt hơn  Khuôn mẫu 🡺 quy định đặc điểm của hành vi chung 1 nhóm các đối tượng | |
| **4- Contructor là gì ? Đặc điểm của contructor trong java?** | - Contructor là một phương thức đặc biệt giúp khởi tạo đối tượng  - Contructor phải có tên trùng với tên của lớp  - Một lớp có thể có nhiều contructor  - Nếu không khai báo contructor cho lớp thì mặc định lớp có một contructor không có tham số  - Không có kiểu dữ liệu trả về  - Được gọi khi dùng từ khóa new  - Có contructor mặc định  - Một contructor có thể gọi lại một con tructor khác bằng từ khóa this(điều kiện phải đặt lên đầu trước đối tượng ) |
| **5- Từ khóa this dùng để làm gì?** | Từ khóa this trong java được dùng để truy cập đến thuộc tính hoặc hành vi trong class  + Phân biệt biến instance với tham số truyền vào  + Gọi contructor khác trong cùng class  + Truy cập phương thức hoặc biến instance trong class  + Truyền đối tượng hiện tại vào phương thức khác  + Trả về chính dối tượng hiện tại (chanining) |
| **6- Phân biệt contructor và method?** | |  |  |  | | --- | --- | --- | | **Đặc điểm** | **Contructor** | **Method (Phương thức )** | | **Mục đích** | Dùng để khởi tạo đối tượng | Dùng để thực hiện hành động/xử lý logic | | **Tên** | Phải trùng tên với class | Có thể đặt tùy ý  (theo quy tắc đặt tên ) | | **Kiểu trả về** | Không có kiểu trả về  (không dùng void,int,…) | Phải có , có thể là void,int,String,… | | **Tự động gọi khi tạo đối tượng ?** | Có , gọi tự động khi dùng new | Không, chỉ gọi khi bạn gọi trực tiếp | | **Kế thừa (inherit)?** | Không được kế thừa | Có thể kế thừa và override(ghi đè) | | **Gọi bằng** | Gọi khi dùng new | Gọi bằng cách object.methodName() | |