

CÂU HỎI THƯỜNG GẶP KHI PHÒNG VẤN HƯỚNG ĐỐI TƯỢNG TRONG PHPP

Câu 1. Lập trình hướng đối tượng (OOP) là gì ? Tại sao lại sử dụng OOP mà ko phải lập trình khác? Điểm mạnh của nó là gì ?

a. OOP là phương pháp lập trình qui tất cả về 1 đối tượng, khi cần chỉ việc lấy ra sử dụng.

b.php là phần mềm mã mở được tất cả người sử dụng trên thế giới phát triển, mỗi người phát triển 1 hướng, nên OOP được sử dụng để tập hợp tất cả qui về 1 mối, Điểm mạnh của OOP là kế thừa, cho phép sử dụng lại những gì đã có trước đó và phát triển thêm dựa trên những cái đã có sẵn.

Câu 2. Làm thế nào để truy cập trong phương thức và thuộc tính?

- Để truy cập tới 1 phương thức hoặc thuộc tính ta dùng từ khóa \$this trở tới phương thức hoặc thuộc tính cần truy cập.

Câu 3. Thế nào là thuộc tính ? Phương thức ?

- Thuộc tính là những biến được khởi tạo trong 1 lớp có kèm theo cơ chế.

- Phương thức là những hàm được khởi tạo trong 1 lớp có kèm theo cơ chế

Câu 4. Làm thế nào để sử dụng class?

- Khởi tạo 1 đối tượng thể hiện lại class đó.

Câu 5. Có các cơ chế nào ? Nói rõ các cơ chế. Private khác protected chỗ nào ?

Có 3 cơ chế: Public, private, protected.

- Public: cho phép truy cập và sử dụng ở phạm vi trong class và ngoài class và nó hỗ trợ sự kế thừa

- Protected: Không thể sử dụng đối tượng để truy cập tới các phương thức và thuộc tính, nếu trong 1 lớp chúng ta có thể truy cập được tới phương thức và thuộc tính đó, có hỗ trợ sự kế thừa

- Private: Giống như Protected, nhưng không cho phép sự kế thừa.

Câu 6. Hàm khởi tạo là gì ? Tại sao phải dùng hàm khởi tạo? Đối lập với hàm khởi tạo là gì ? __destruct đc gọi khi nào ?

- Hàm khởi tạo __construct() là hàm khi chúng ta tiến hành khởi tạo một đối tượng thì ngay lập tức hàm khởi tạo được gọi ra.

Tại sao phải dùng hàm khởi tạo. Dùng hàm khởi tạo để khởi tạo những phương thức và thuộc tính cần thiết khi 1 đối tượng được khởi tạo.

Đối lập với hàm khởi tạo là hàm hủy khởi tạo __destruct(), Hàm hủy khởi tạo được gọi khi kết thúc một kịch bản php

Câu 7. Thế nào là kế thừa? Việc kế thừa diễn ra theo hướng nào ? Kế thừa đc bao nhiêu lớp ?

- Kế thừa: khi 1 lớp con kế thừa từ 1 lớp cha. thì lớp con có thể sử dụng lại các phương thức và thuộc tính của lớp cha, ngoại trừ những phương thức và thuộc tính mang cơ chế Private
- Việc kế thừa diễn ra theo 1 chiều và 1 lớp chỉ kế thừa được 1 lớp và lớp cha có thể có nhiều lớp con kế thừa.

Câu 8. Hiện tượng ghi đè là gì ? Hậu quả của việc ghi đè là gì ? Làm gì để khắc phục việc ghi đè ? Tầm vực của parent là bao nhiêu cấp ? Làm sao để chống việc ghi đè?

- Hiện tượng ghi đè là khi lớp cha và lớp con có cùng tên 1 phương thức (trùng tên phương thức).
- Hậu quả của việc ghi đè nó sẽ phá hủy cấu trúc ban đầu của lớp cha.
- Để khắc phục hiện tượng ghi đè chúng ta dùng từ khóa parent::
- Tầm vực của parent: chỉ được 1 cấp kế cận với nó. Muốn parent nhiều cấp thì phải có tính bắc cầu
- Để chống việc ghi đè chúng ta đặt từ khóa final đằng trước phương thức đó.

Câu 9. Final là gì ? diễn giải nó như thế nào ? Nêu ra các chức năng của final.

- Final là qui ước phương thức cuối cùng hoặc 1 class cuối cùng. Final dùng để chống việc ghi đè hoàn toàn và chống sự kế thừa

Câu 10. Nếu sử dụng phương thức private có chống đc việc ghi đè ko ?

- Không vì cơ chế private chỉ chống sự kế thừa.

Câu 11. Thuộc tính tĩnh là gì? Tác dụng là gì ? Khác thuộc tính thường chỗ nào?Làm sao để truy xuất thuộc tính tĩnh ? Thuộc tính tĩnh khác hằng chỗ nào ?

- Thuộc tính tĩnh là thuộc tính được qui ước từ khóa **static** đứng trước tên 1 thuộc tính.
- Tác dụng là thường dùng để huy động hoặc lặp lại hành động lặp lại nhiều lần nên đỡ tốn tài nguyên.
- Thuộc tính tĩnh khác thuộc tính thường ở chỗ không cần khởi tạo đối tượng vẫn có thể truy xuất được.
- Để truy xuất thuộc tính tĩnh dùng từ khóa self::tenthuoctinh
- Thuộc tính tĩnh khác hằng ở chỗ là giá trị có thể thay đổi, trong khi đó hằng là có giá trị cố định.

Câu 12. Clone là gì ? Sử dụng clone như thế nào ?

- Clone là nhân bản 1 đối tượng đã có sẵn(không phải khởi tạo đối tượng mới)
- Sử dụng clone khi ta không muốn nó làm thay đổi giá trị của lớp ban đầu

Câu 13. Lớp trừu tượng là gì ? dùng abstract để làm gì ? Nhược điểm của abstract là gì? Có cách nào để khắc phục nhược điểm đó ko ?

- Lớp trừu tượng là lớp chứa các phương thức trừu tượng, các lớp sử dụng lớp trừu tượng phải thực thi các phương thức trừu tượng của lớp trừu tượng đó.

- Dùng Abstract để quản lý dự án(vừa và nhỏ).

- Nhược điểm là mang tính đơn hình(chỉ được kế thừa từ 1 lớp). Để khắc phục ta dùng giao diện(interface).

Câu 14. Interface là gì ? Công dụng của nó ?

- Interface là giao diện, mang tính đa hình cho phép sử dụng 1 hay nhiều giao diện

- Công dụng là để quản lý dự án có qui mô lớn.

Câu 15. Phân biệt sự khác và giống nhau của interface và abstract là gì ?

- Giống nhau là để dùng để quản lý dự án

- Khác nhau: Abstract mang tính đơn hình, chỉ kế thừa được 1 lớp(bản chất là 1 lớp). còn Interface mang tính đa hình, có thể sử dụng 1 hay nhiều giao diện

Câu 16. 1 lớp có thể sử dụng abstract và interface cùng lúc ko ? - được

Câu 17. 1 lớp có thể sử dụng 2 giao diện có cùng tên hàm ko? - Không

Câu 18. Lazy loading technology là gì ?

- là công nghệ tự động nạp 1 file khi khởi tạo 1 đối tượng(tên file trùng tên lớp)