#Bài tập về nhà của bài 5:

\*Bài 1:

-Từ khoá final dùng để ngăn cản lớp con ghi đè lại các thuộc tính và phương thức của lớp cha

\*Bài 2:

-Trong một lớp mình có thể khai báo nhiều constructor trong một lớp

-Vì mỗi một constructor ta truyền khác các tham số thì sẽ tạo được nhiều constructor trong cùng một lớp

\*Bài 3:

-Khi lớp cha không cho phép lớp con kế thừa là khi lớp cha dùng final ,static ,private

\*Bài 4:

|  |  |
| --- | --- |
| Override | Overload |
| -Cho phép lớp con cung cấp và chiển khai cụ thể các phương thức và thuộc tính của lớp cha | -Cho phép tạo nhiều phiên bản có cùng phương thức ,mỗi phiên bản chấp nhận một danh sách đối tượng khác nhau nhắm thuận lợi cho việc gọi phương thức |
| -Thu thập tính đa hình runtime | -Để thu thập tính đa hình của compile time |
| -Khi ghi đè lại hành vi của lớp cha nhưng ta không muốn thay thế nó mà chỉ muốn bổ sung thì ta dùng super để gọi đến thành viên được kế thừa | -Kiểu dữ liệu và đối số truyền vào được thay đổi nếu kiểu dữ liệu khác(tự động ép kiểu )nếu giá trị của đối số không phù hợp với kiểu dữ liệu mà tham số định nghĩa |
| -Tham số phải trùng nhau ,kiểu giá trị trả về phải tương thích | -Cùng tên khác tham số  -Định nghĩa cùng kiểu trả về hoặc có thể khác kiểu trả về  -Có thể định nghĩa cùng hoặc khác access modifier  -Không được gọi là nạp chồng khi chỉ khác kiểu dưc liệu trả về hoặc access modifier |

|  |  |  |
| --- | --- | --- |
| Nội dung | Override | Overload |
| Hành vi | Thay đổi hành vi hiện tại của phương thức | Thêm hoặc mở rộng cho hành vi của phương thức |
| Đa hình | Thể hiện tính đa hình tại run time | Thể hiện tính đa hình tại compile time |
| Danh sách tham số | Danh sách tham số phải giống nhau | Danh sách tham số có thể khác nhau |
| Quyền truy cập | Phương thức ghi đè ở lớp con phải có quyền truy cập bằng hoặc lớn hơn phương thức được ghi đè ở lớp cha | Các phương thức nạp chồng có thẻ có quyên truy cập khác nhau |
| Giá trị trả về | Kiểu dữ liệu trả về bắt buộc phải giống nhau | Kiểu dữ liệu trả về có thể khác nhau |
| Phạm vi | Xảy ra giữa 2 class có quan hệ kế thừa | Xảy ratrong phạm vi cùng một class |