| **Tiêu chí** | **Overloading (Nạp chồng)** | **Overriding (Ghi đè)** |
| --- | --- | --- |
| **Khái niệm** | Định nghĩa nhiều phương thức cùng tên nhưng khác nhau về danh sách tham số trong cùng một lớp. | Cung cấp cài đặt lại cho một phương thức đã có ở lớp cha bằng cách định nghĩa lại ở lớp con. |
| **Thời điểm quyết định** | Quyết định tại thời điểm biên dịch (compile-time), tức là dựa trên tham số gọi phương thức. | Quyết định tại thời điểm chạy (run-time) dựa trên đối tượng thực sự của lớp. |
| **Sự liên quan đến đối tượng** | Không liên quan đến tính đa hình động; chủ yếu áp dụng trong cùng một lớp, không nhất thiết phải có mối quan hệ kế thừa. | Liên quan chặt chẽ đến kế thừa và đa hình; yêu cầu có mối quan hệ cha-con giữa các lớp. |
| **Số lượng tham số** | Các phương thức phải có danh sách tham số khác nhau (về số lượng, kiểu dữ liệu hoặc thứ tự các tham số). | Phương thức ghi đè phải có cùng số lượng tham số, cùng kiểu dữ liệu và cùng thứ tự như phương thức trong lớp cha. |
| **Mục đích sử dụng** | Tăng tính linh hoạt và khả năng sử dụng lại của lớp bằng cách cho phép phương thức xử lý các kiểu dữ liệu khác nhau. | Cho phép lớp con thay đổi hoặc mở rộng hành vi của lớp cha, đáp ứng yêu cầu nghiệp vụ cụ thể của lớp con. |

**Giải thích thêm:**

* **Overloading** chủ yếu hỗ trợ khả năng xử lý các tình huống khác nhau dựa trên đầu vào khác nhau, giúp mã nguồn trở nên rõ ràng và dễ quản lý. Vì quyết định gọi phương thức nào được xác định ở thời điểm biên dịch nên Overloading còn được gọi là *static polymorphism*.
* **Overriding** giúp cài đặt lại hành vi của phương thức từ lớp cha trong lớp con, tạo nên tính đa hình (polymorphism) cho đối tượng. Việc này cho phép các đối tượng của lớp con phản hồi một cách đặc trưng khi được gọi thông qua tham chiếu của lớp cha, và quyết định phương thức nào được thực thi chỉ được xác định tại thời điểm chạy.