Theo định nghĩa lập trình thì callback là một phần trong đoạn code được thực thi và nó được truyền vào giống như một tham số của một đoạn mã khác

Trong JavaScript cũng như jQuery thì code sẽ được thực thi từng dòng (line by line). Tuy nhiên với các hiệu ứng, những dòng tiếp theo của code vẫn có thể được thực thi mặc dù hiệu ứng không hoàn thành hoặc chưa hoàn thành. Và để ngăn điều này xảy ra, chúng ta sẽ tạo một function **callback**

Mình lấy ví dụ về cú pháp trong bài hiệu ứng ẩn hiện sử dụng jQuery như sau :

$(*selector*).hide(*speed,****callback***);

Để hiểu rõ hơn về phương thức thực thi code từng dòng (line by line) trong jquery và chức năng của callback, mình sẽ đưa ra 2 ví dụ

Ví dụ 1 : tham số ***callback*** theo cú pháp trên là một function sẽ được thực thi (executed) sau khi hiệu ứng ẩn 1 đoạn văn hoàn thành

$("button").click(function(){  
  $("p").hide("slow",function(){  
    alert("Đoạn văn đã được ẩn");  
  });  
});

Còn đây là ví dụ khác, nhìn mã code thì có vẻ rất giống với ví dụ 1 ở trên, đều là hiệu ứng ẩn 1 đoạn văn và thông báo "Đoạn văn đã được ẩn" sẽ xuất hiện. Nhưng bạn hãy thử xem nó khác nhau như thế nào nhé

$("button").click(function(){  
  $("p").hide(1000);  
  alert("Đoạn văn đã được ẩn");  
});

Sự khác nhau là rất rõ ràng khi bạn thực hiện 2 ví dụ trên.  
  
- Ở ví dụ 1 chúng ta sẽ thấy hiệu ứng ẩn 1 đoạn văn sẽ xảy ra trước và khi hoàn thành hiệu ứng thì đoạn code thông báo mới được thực thi  
  
- Còn ở ví dụ 2, chúng ta sẽ dùng hiệu ứng ẩn với đoạn văn trong khoảng thời gian là 1 giây (1000 milisecond). Nhưng đoạn code thông báo alert("Đoạn văn đã được ẩn"); sẽ được thực thi ngay sau khi jQuery chạy qua dòng code $("p").hide(1000); mà không chờ đợi 1 giây sau mới thực hiện