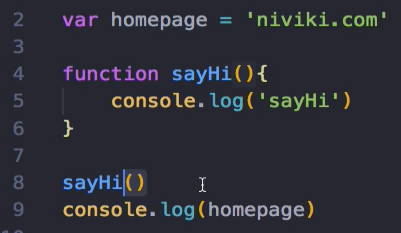
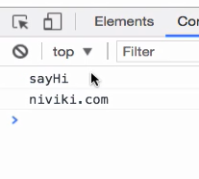
1. Hoisting



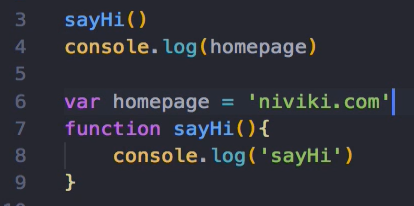
1. Gồm 2 giai đoạn: Khởi tạo (Chuẩn bị bộ nhớ), Thực thi.
2. Ví dụ:
   1. Ví dụ 1:



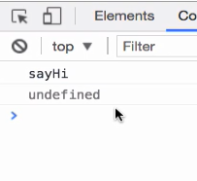
Kết quả:



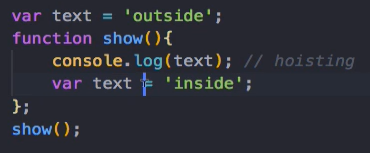
* 1. Ví dụ 2:



Kết quả:



Tại sao undefined?

* Vì thực thi gồm 2 giai đoạn:
  + Chuẩn bị bộ nhớ:
    - Nếu là biến: tạo bộ nhớ cho biến & set undefined cho ô nhớ đó.
    - Nếu là hàm: tạo bộ nhớ cho hàm & gán luôn body của hàm vào bộ nhớ.
  + Thực thi:
    - Sau khi thực hiện gán xong ở bước chuẩn bị bộ nhớ
    - Bước thực thi: chạy từ trên xuống
      * Khi gọi sayHi() thì ô nhớ đã có body của hàm thì thực thi sẽ ra kết quả bình thường.
      * Khi gọi console.log(homepage) thì lại ra undefined vì ở bước chuẩn bị bộ nhớ nó không thực hiện phép gán (biến=...) do coder tạo ra mà nó chỉ set kiểu dữ liệu cho biến là undefined nên không có body hoặc giá trị để thực thi. Phải khi chạy đến dòng gán giá trị thì lúc đó mới có giá trị để xuất :vv
    - Global Execution Context sẽ chạy đầu tiên, Các Execution Context chỉ chạy khi được gọi và hoisting diễn ra ở cả Execution Context và Global Execution Context
      * 
      * Global Execution Context sẽ chạy đầu tiên
      * Khi invoke (thực thi) hàm show(), Khi đó, Execution Context của show được chạy & xảy ra hoisting nên khi truy cập giá trị biến text, kết quả là undefined vì chỉ mới tạo bộ nhớ chứ chưa được gán giá trị cho ô nhớ đó.