

Window Object

Window object được hỗ trợ trên mọi trình duyệt. Nó đại diện cho cửa sổ trình duyệt.

Tất cả [global Javascript object](#), function, biến (phải khai báo bằng **var** nha, không phải **let/const**) đều tự động trở thành thuộc tính (hoặc phương thức) của **window** object.

Ngay cả **document** object (của HTML DOM) cũng là một thuộc tính của **window** object

```
alert('Hello')  
// tương đương  
window.alert('Hello')
```

```
var gVar = 5  
alert(window.gVar)
```

```
window.document.getElementById('header')  
// tương đương  
document.getElementById('header')
```

Một số thuộc tính và phương thức của window object

- **window.innerHeight**: Chiều cao bên trong của cửa sổ trình duyệt (pixel)
- **window.innerWidth**: Chiều rộng bên trong của cửa sổ trình duyệt (pixel)
- **window.open()**: mở một cửa sổ
- **window.close()**: đóng cửa sổ hiện tại
- **window.resizeTo()**: resize cửa sổ hiện tại

Window location

window.location (location) object có thể được sử dụng để lấy thông tin địa chỉ url hiện tại và chuyển hướng trình duyệt sang 1 trang mới.

- **window.location.href** return về URL trang hiện tại
- **window.location.hostname** return tên domain
- **window.location.pathname** return pathmane của trang hiện tại
- **window.location.protocol** return giao thức trang web (http: hoặc https:)
- **window.location.assign()** chuyển hướng sang một trang mới

Ví dụ:

```
console.log(location.href) // https://www.xdevclass.com/phan-5-thao-tac-html-css-dom-va-browser-bom-voi-javascript
console.log(location.hostname) // www.xdevclass.com
console.log(location.pathname) // /phan-5-thao-tac-html-css-dom-va-browser-bom-voi-javascript
console.log(location.protocol) // https:
location.assign('https://www.google.com/') // Chuyển đến trang google
location.href = 'https://www.google.com/' // Tương đương location.assign()
```

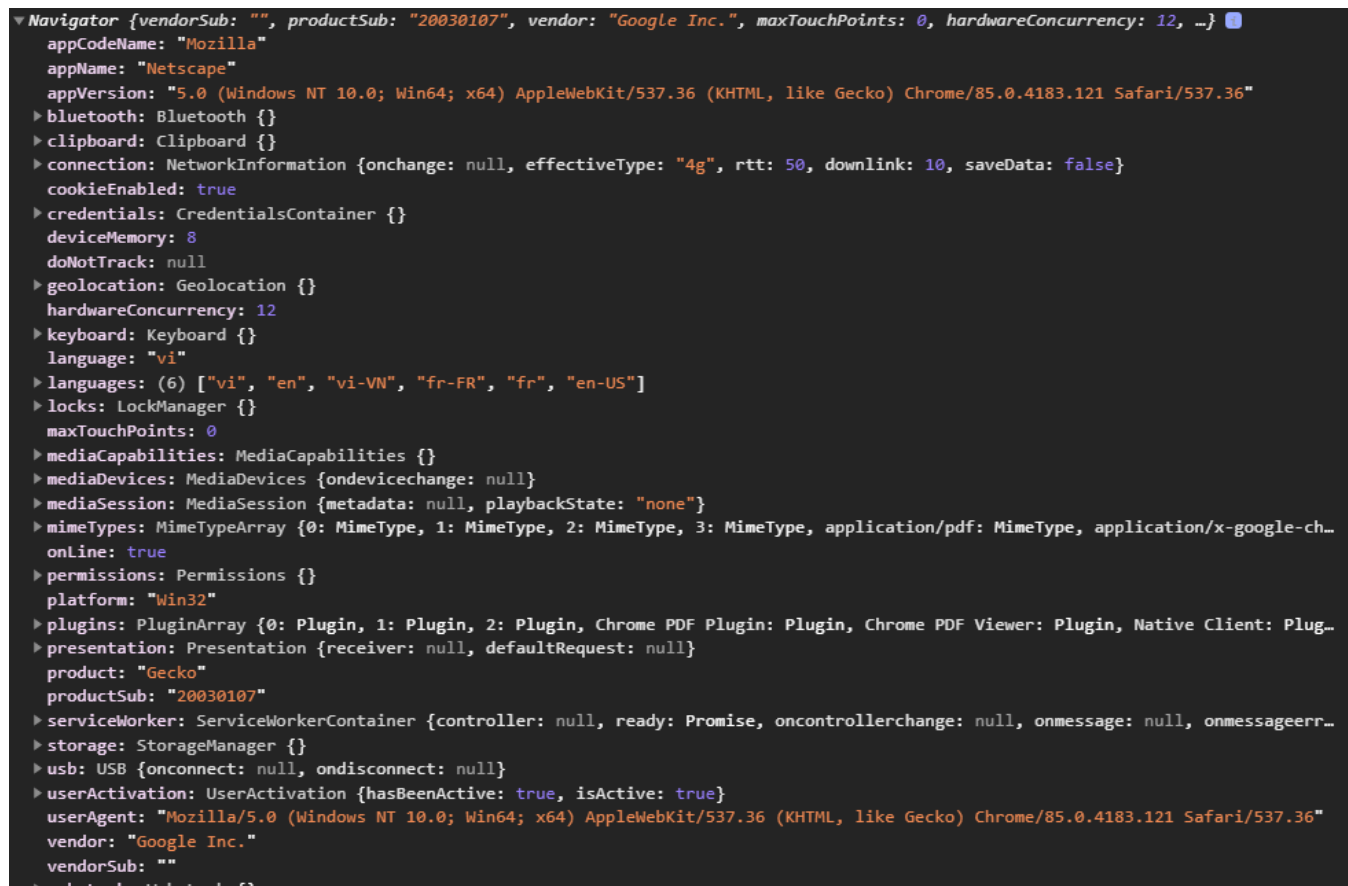
Window Navigator

`window.navigator` có thể cho ta biết thông tin trình duyệt của người dùng

Ví dụ

- `navigator.language`: return ngôn ngữ trình duyệt
- `navigator.userAgent`: return **user-agent header** được gửi bởi trình duyệt đến server

Nói chung thì **navigator** còn cực nhiều thuộc tính và phương thức, liệt kê không hết luôn, mọi người có thể thao khảo hình dưới



```
▼ Navigator {vendorSub: "", productSub: "20030107", vendor: "Google Inc.", maxTouchPoints: 0, hardwareConcurrency: 12, ...}
  appCodeName: "Mozilla"
  appName: "Netscape"
  appVersion: "5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/85.0.4183.121 Safari/537.36"
  ▶ bluetooth: Bluetooth {}
  ▶ clipboard: Clipboard {}
  ▶ connection: NetworkInformation {onChange: null, effectiveType: "4g", rtt: 50, downlink: 10, saveData: false}
  cookieEnabled: true
  ▶ credentials: CredentialsContainer {}
  deviceMemory: 8
  doNotTrack: null
  ▶ geolocation: Geolocation {}
  hardwareConcurrency: 12
  ▶ keyboard: Keyboard {}
  language: "vi"
  ▶ languages: (6) ["vi", "en", "vi-VN", "fr-FR", "fr", "en-US"]
  ▶ locks: LockManager {}
  maxTouchPoints: 0
  ▶ mediaCapabilities: MediaCapabilities {}
  ▶ mediaDevices: MediaDevices {ondevicechange: null}
  ▶ mediaSession: MediaSession {metadata: null, playbackState: "none"}
  ▶ mimeType: MimeTypeArray {0: MimeType, 1: MimeType, 2: MimeType, 3: MimeType, application/pdf: MimeType, application/x-google-ch...
    online: true
  ▶ permissions: Permissions {}
  platform: "Win32"
  ▶ plugins: PluginArray {0: Plugin, 1: Plugin, 2: Plugin, Chrome PDF Plugin: Plugin, Chrome PDF Viewer: Plugin, Native Client: Plug...
  ▶ presentation: Presentation {receiver: null, defaultRequest: null}
  product: "Gecko"
  productSub: "20030107"
  ▶ serviceWorker: ServiceWorkerContainer {controller: null, ready: Promise, oncontrollerchange: null, onmessage: null, onmessageerr...
  ▶ storage: StorageManager {}
  ▶ usb: USB {onconnect: null, ondisconnect: null}
  ▶ userActivation: UserActivation {hasBeenActive: true, isActive: true}
  userAgent: "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/85.0.4183.121 Safari/537.36"
  vendor: "Google Inc."
  vendorSub: ""
  ▶ wakeLock: WakeLock {}
```

Window History

`window.history` object chứa lịch sử của trình duyệt

Một số phương thức phổ biến:

- `history.back()` dùng để trở về trang trước
- `history.forward()` dùng để đi đến trang tiếp theo