

FPT POLYTECHNIC



Bài 4: LẬP TRÌNH HẸNG I T HẸNG VÀ MÔ HÌNH BOM

- Cấu trúc i u khi n
- Cấu trúc l a ch n
 - L nh l a ch n n
 - L nh l a ch n kép
 - L nh a l a ch n
- Cấu trúc l p
 - L p không bi t tr c s l n l p
 - L p bi t tr c s l n l p
- Hàm

- Ph ng th c l p trình
- Ph ng th c l p trình h ng i t ng
 - Khái ni m i t ng, thu c tính và ph ng th c
 - T o i t ng
 - Thêm thu c tính và ph ng th c vào i t ng
- Khái ni m v l p
- nh ngh a l p, t o i t ng t l p
- Các thao tác v i i t ng trong l p
- Browser Object Model

PHƯƠNG THỨC LẬP TRÌNH

- Lập trình là gì? quy tắc và nguyên tắc trong cuộc sống
 - Bài toán tính toán phức tạp: Lập trình cho tên lửa bay vào vũ trụ
 - Bài toán logic: đưa ra quy tắc (dự báo thời tiết)
 - Bài toán quản lý trong các doanh nghiệp (phân mềm tính lương)



- Phong cách lập trình (programming paradigm) là cách thức qui tắc và nguyên tắc

- Có hơn 25 phân công thực tập
 - Mỗi phân công thực tập có quy định cho một vị trí
 - Mỗi sinh viên phân công thực tập khó đáp ứng trong thực tế phân công thực tập
 - Mỗi sinh viên phân công thực tập chỉ có thể nhận một nhóm sinh viên hay trong một thời gian ngắn
- Những phân công thực tập phải có:
 - Lịch trình học tập
 - Lịch trình học tập thành phần
 - Lịch trình cấu trúc
 - Lịch trình học tập nội dung
- Phân công thực tập học tập nội dung phát triển rèn luyện nhân lực

PH NG TH C L P TRÌNH H NG I T NG

- i t ng là t t c m i th trong cu c s ng (các v t, s v t)
 - Ví d i t ng: Qu bóng, cái bàn, ô tô, bông hoa, con ng i, nhà máy...
- M i i t ng có c tính và hành ng riêng
- Ý t ng ch o c a ph ng th c l p trình h ng i t ng: Mô ph ng cu c s ng th c trong l p trình
 - Trong cu c s ng có nh ng i t ng nh qu bóng, cái bàn... v i các c tính và hành ng riêng thì l p trình mô ph ng các i t ng ó v i các c tính và hành ng nh th



■ Mèo có nh ng c tính:

- Màu lông: tam th
- N ng: 2kg
- Móng: s c

■ Mèo có nh ng hành ng:

- B t chu t
- Li m lông

■ Trong l p trình: c tính c g i là thu c tính, hành ng c g i là ph ng th c



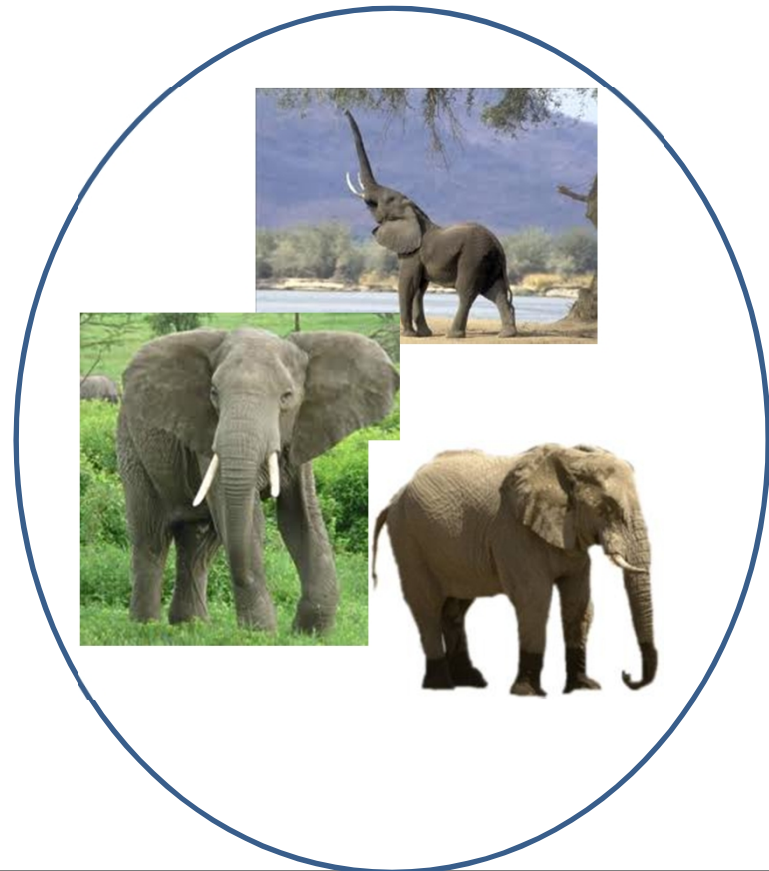
■ Voi có nh ng c tính:

- Màu da: nâu
- N ng: 2 t n
- Vòi: 1m

■ Voi có nh ng hành ng:

- Phun n c
- n c

- Các i t ng có cùng thu c tính và ph ng th c c gom l i thành m t l p
- Hay: L p nh ngh a t p h p các i t ng có cùng thu c tính và ph ng th c



- S d n g t khóa new

var *tendoituong* = new Object;

var meo = new Object;

var hoa = new Object;

- Thêm thuộc tính cho đối tượng

tendoituong.tenthuoctinh = giatri;

```
hoaDao.mau = "Hong";  
hoaDao.soCanh = "5";
```

- Truy cập thuộc tính của đối tượng

tendoituong.tenthuoctinh

```
alert(hoaDao.mau);  
alert(hoaDao.soCanh);
```

- Thêm hàm

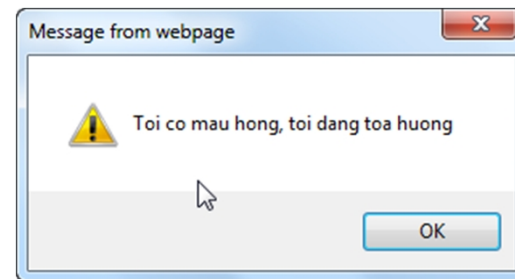
```
tendoituong.tenphuongthuc = function(){  
    //Viết mã cho hàm này  
}
```

```
hoaDao.toaHuong= function () {  
    alert("Toi co mau Hong, toi dang toa huong");  
}
```

- Gọi hàm

```
tendoituong.tenphuongthuc()
```

```
hoaDao.toaHuong();
```



- V n n y sinh: Gi s trong v n có nhi u lo i hoa, m i lo i hoa có màu s c, s cánh khác nhau

```
var hoaDao = new Object;  
hoaDao.mau = " Hong";  
hoaDao.soCanh = "5";  
hoaDao.toaHuong= function () {  
    alert("Toi co mau Hong, toi dang  
toa huong!");  
}  
var hoaHong = new Object;  
hoaHong.mau = " Do";  
hoaHong.soCanh = " 10";  
hoaHong.toaHuong= function () {  
    alert("Toi co mau Do, toi dang  
toa huong!");  
}
```

```
var hoaCuc = new Object;  
hoaCuc.mau = " Vang";  
hoaCuc.soCanh = " 20";  
hoaCuc.toaHuong= function () {  
    alert("Toi co mau vang, toi dang  
toa huong!");  
}  
var hoaLan = new Object;  
hoaLan.mau = "Tim";  
hoaLan.soCanh = "3";  
hoaLan.toaHuong= function () {  
    alert("Toi co mau tim, toi dang  
toa huong!");  
}
```

➔ T o m t khuôn m u chung (l p)

■ nh ngh a l p

```
function tenlop (tenbien1, tenbien2...) {
```

```
    tenthuoctinh1 = tenbien1;
```

```
    tenthuoctinh2 = tenbien2;
```

```
    tenphuongthuc = function() {
```

```
        //Vi t mã cho ph ng th c ây
```

```
    }
```

```
}
```

```
function Hoa(mauHoa, soCanhHoa){
```

```
    this.mau = mauHoa;
```

```
    this.soCanh = soCanhHoa;
```

```
    this.toaHuong = function() {
```

```
        alert("toi co mau " + this.mau + ", toi dang toa
```

```
        huong");
```

```
    }
```

```
}
```

■ T o i t ng

tenlop["tendoituong"] = new tenlop (giatri1, giatri2...)

Hoa ["Dao"] = new Hoa ("Hong", "5");

Hoa ["Hong"] = new Hoa ("Do", "10");

Hoa ["Cuc"] = new Hoa ("Vang", "20");

Hoa ["Lan"] = new Hoa ("Tim", "3");

■ Truy c p n thu c tính và ph ng th c c a i t ng

tenlop["tendoituong"].tenthuoctinh

tenlop["tendoituong"].tenphuongthuc

Hoa ["Dao"].toaHuong();

Hoa ["Hong"].toaHuong();

Hoa ["Cuc"].toaHuong();

Hoa ["Lan"].toaHuong();

- L p qua các i t ng

```
for (var x in Hoa) {  
    Hoa[x].toaHuong();  
}
```

- Tìm m t i t ng trong thu c l p

```
if ("Dao" in Hoa) {  
    alert(" Doi tuong Dao da duoc tao");  
} else {  
    alert(" i t ng Dao chua duoc tao");  
}
```

- Thêm thu c tính cho i t ng thu c l p

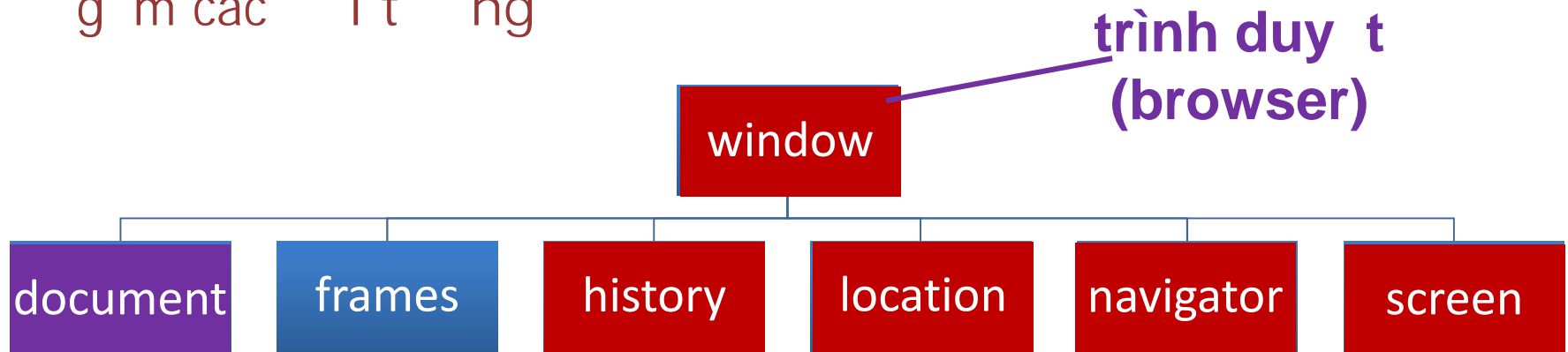
```
if ("Dao" in Hoa) {  
    Hoa["Dao"].bieuTuong = "Mua Xuan";  
}  
alert("Hoa dao bieu tuong cho " + Hoa["Dao"].bieuTuong);
```

BOM



Browser Object Model

- Browser Object Model là mô hình phân cấp hình cây gồm các đối tượng



- Các đối tượng cung cấp thuộc tính và phương thức cho lập trình viên JavaScript
- Mỗi đối tượng trình duyệt hỗ trợ các thuộc tính và phương thức khác nhau
 - Hiểu môi trường mà trình duyệt cung cấp viết mã JavaScript chạy trên môi trường trình duyệt

- Window là i t ng th hi n c a s hi n th hi n t i trên trình duy t
- M t s ph ng th c c a i t ng window ã c s d ng: `alert()`, `prompt()`, `confirm()`
- Các thu c tính và ph ng th c c a window có th g i tr c ti p ho c thông qua window

```
alert("Hi")
```

ho c

```
window.alert("Hi")
```

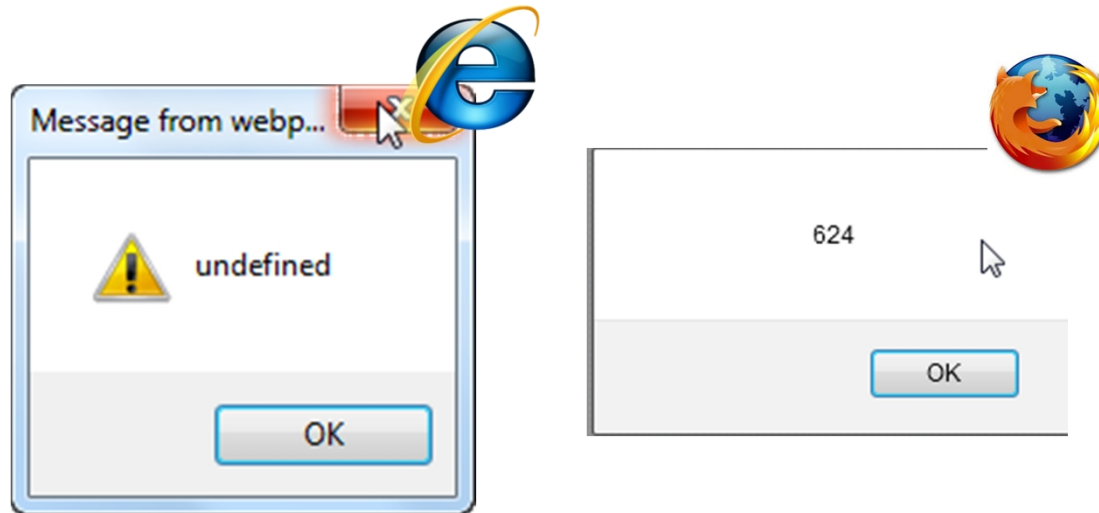
Thuộc tính	Giải thích
closed	Có giá trị là True khi cửa sổ được đóng
defaultStatus	Thiết lập văn bản mặc định trên thanh trạng thái của trình duyệt
name	Thiết lập hoặc trả về tên của cửa sổ
opener	Tham chiếu đến cửa sổ tạo ra cửa sổ hiện tại
status	Dùng để thiết lập văn bản trên thanh trạng thái khi người dùng di chuột trên một thành phần trên cửa sổ (ví dụ khi người dùng di chuột lên link)
innerHeight	Thiết lập hoặc trả về chiều cao phần nội dung của cửa sổ
document	Trả về đối tượng document của cửa sổ

Tham khảo thêm trang

http://www.w3schools.com/jsref/obj_window.asp

để tìm thêm nhiều thuộc tính khác

```
alert(window.innerHeight);
```



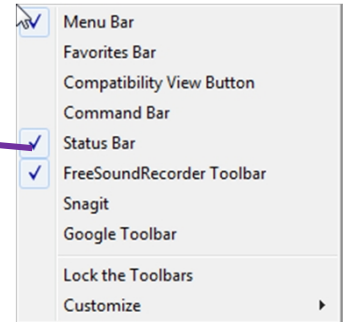
- Firefox hỗ trợ thuộc tính này trong khi IE không hỗ trợ
- Tham khảo trang w3school về bài tập trình duyệt nào hỗ trợ thuộc tính này

Demo s d ng thu c tính defaultStatus

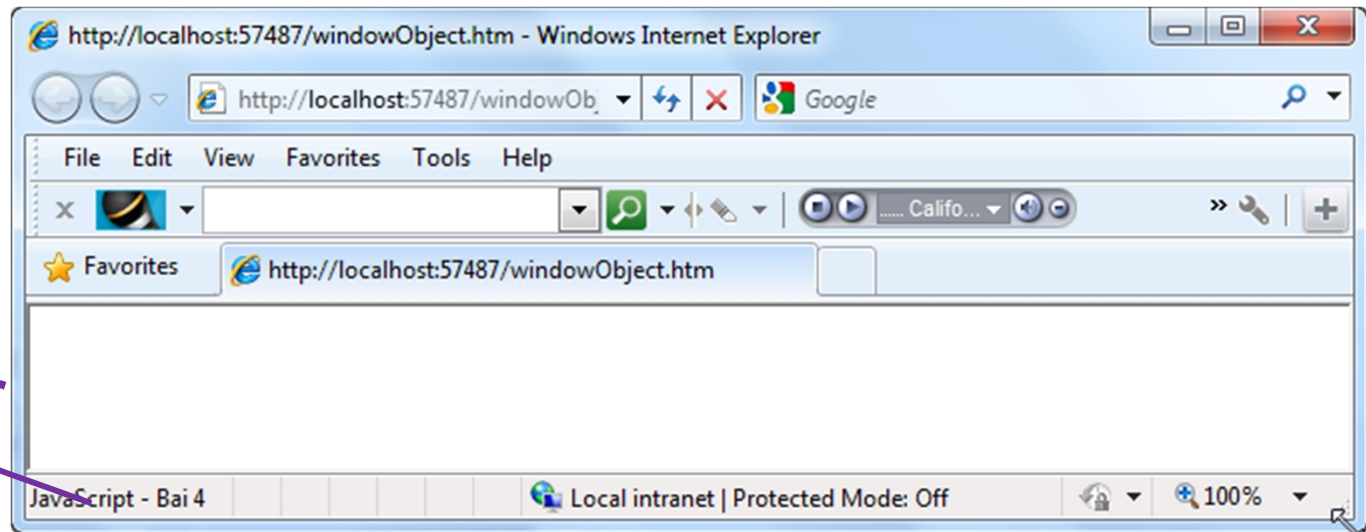
- i v i IE, hi n th Status bar ph i ch n "Status bar"

Ch n
Status bar

window.defaultStatus= "JavaScript - Bai 4"



Status bar



- Phiên b n 8 c a Firefox không h tr default status bar

Phương thức	Giải thích
focus()	Chuyển focus đến cửa sổ
blur()	Bỏ focus đến cửa sổ
close()	Đóng cửa sổ
open()	Mở cửa sổ
print()	Thực hiện chức năng in
moveTo()	Sử dụng để chuyển cửa sổ về vị trí xác định
resizeTo()	Sử dụng để thay đổi kích thước cửa sổ về vị trí xác định

Tham khảo thêm trang

http://www.w3schools.com/jsref/obj_window.asp

biết thêm nhiều phương thức khác

- Sử dụng hàm mở cửa sổ mới và hiển thị

window.open(url, ten, dactinh)

- url: url của trang web
- ten: tên cửa sổ mới
- dactinh: các tính năng mà cửa sổ mới có (mỗi trình duyệt sẽ hỗ trợ một tập các tính năng riêng)

**window.open("http://www.google.com.vn/", "timkiem",
"menubar = yes, width = 800, height = 600")**

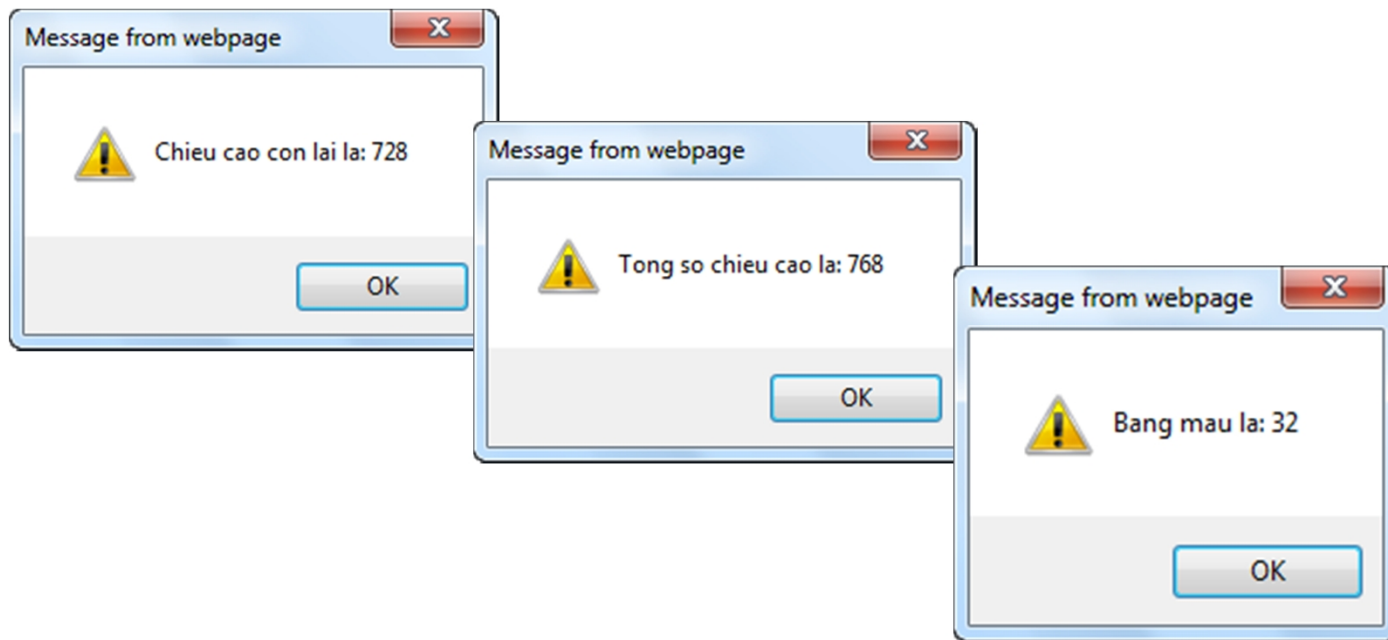
Chú ý:

- Cần sử dụng cách này khi cần thiết vì trình duyệt có thể disable javascript
- Có thể sử dụng thẻ `<a>` thay thế

- Màn hình i truy c p s d ng màn hình có phân gi i khác nhau, kích th c khác nhau, d i màu khác nhau...
 ➔ Ng i l p trình ph i n m c thông tin này hi n th nh phù h p, hi n th trang web có kích th c phù h p...
- i t ng screen cung c p thu c tính l y thông tin v màn hình c a ng i truy c p

Thuộc tính	Giải thích
availHeight	Trả về chiều dài của màn hình (trừ kích thước của window taskbar)
availWidth	Trả về chiều rộng của màn hình (trừ kích thước của window taskbar)
height	Trả về chiều dài của màn hình
width	Trả về chiều rộng của màn hình
pixelDepth	Trả về độ phân giải của màn hình
colorDepth	Trả về bảng màu để hiển thị ảnh

```
alert("Chiều cao còn lại là: " + screen.availHeight);  
alert("Tổng số chiều cao là: " + screen.height);  
alert("Bảng màu là: " + screen.colorDepth);
```



- Mỗi trình duyệt có cách thức thi hành mã JavaScript riêng
- Có thể cùng thức hi n m t ch c n ng, nh ng i v i t ng trình duyệt, c n ph i vi t các o n mã khác nhau
 - ➔ C n bi t thông tin v trình duyệt i vi t mã JavaScript phù h p
- i t ng Navigator cung c p các thông tin v trình duyệt

Thuộc tính	Giải thích
appName	Trả về mã của trình duyệt
appVersion	Trả về tên của trình duyệt
cookieEnabled	Trả về phiên bản của trình duyệt
platform	Xác định xem Cookie có được bật hay không
	Trả về nền tảng mà trình duyệt được biên dịch

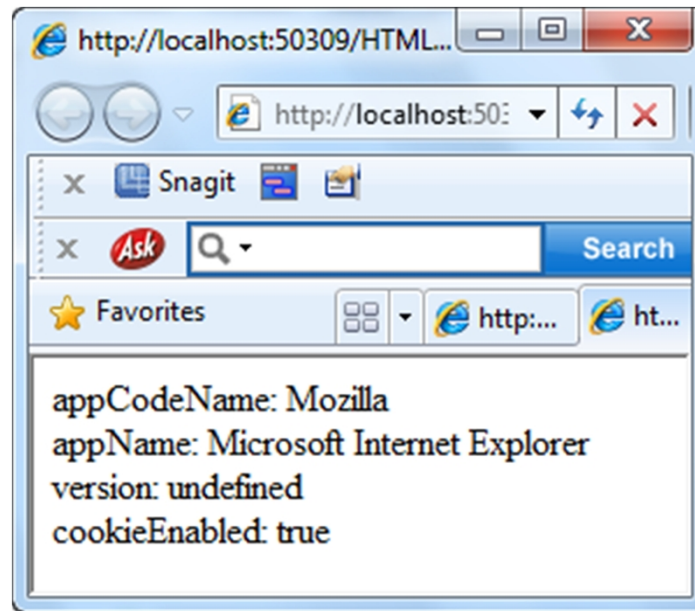
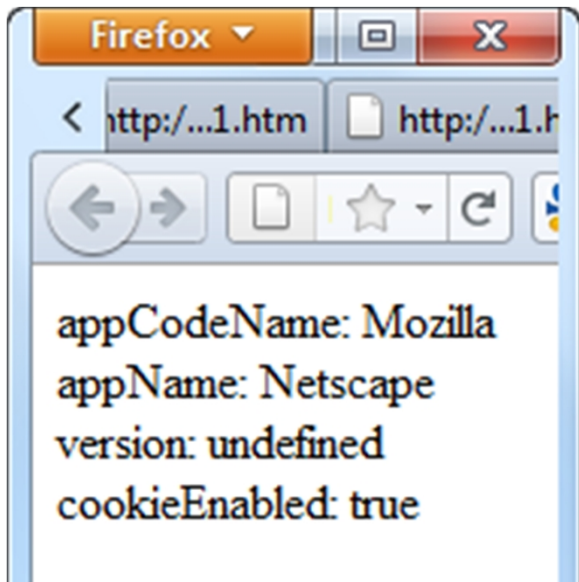
Phương thức	Giải thích
javaEnabled()	Xác định xem trình duyệt có kích hoạt Java hay không

Tham khảo thêm trang

http://www.w3schools.com/jsref/obj_navigator.asp

biết thêm nhiều phương thức khác

```
document.write("appCodeName: " + navigator.appCodeName + "<br>");  
document.write("appName: " + navigator.appName + "<br>");  
document.write("version: " + navigator.version + "<br>");  
document.write("cookieEnabled: " + navigator.cookieEnabled);
```

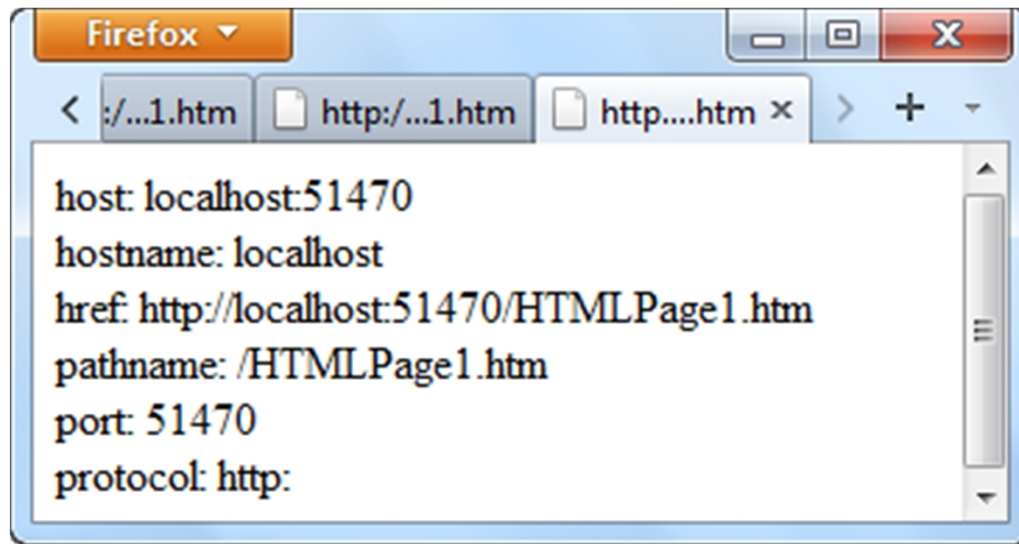


■ Các thuộc tính và URL hiển thị

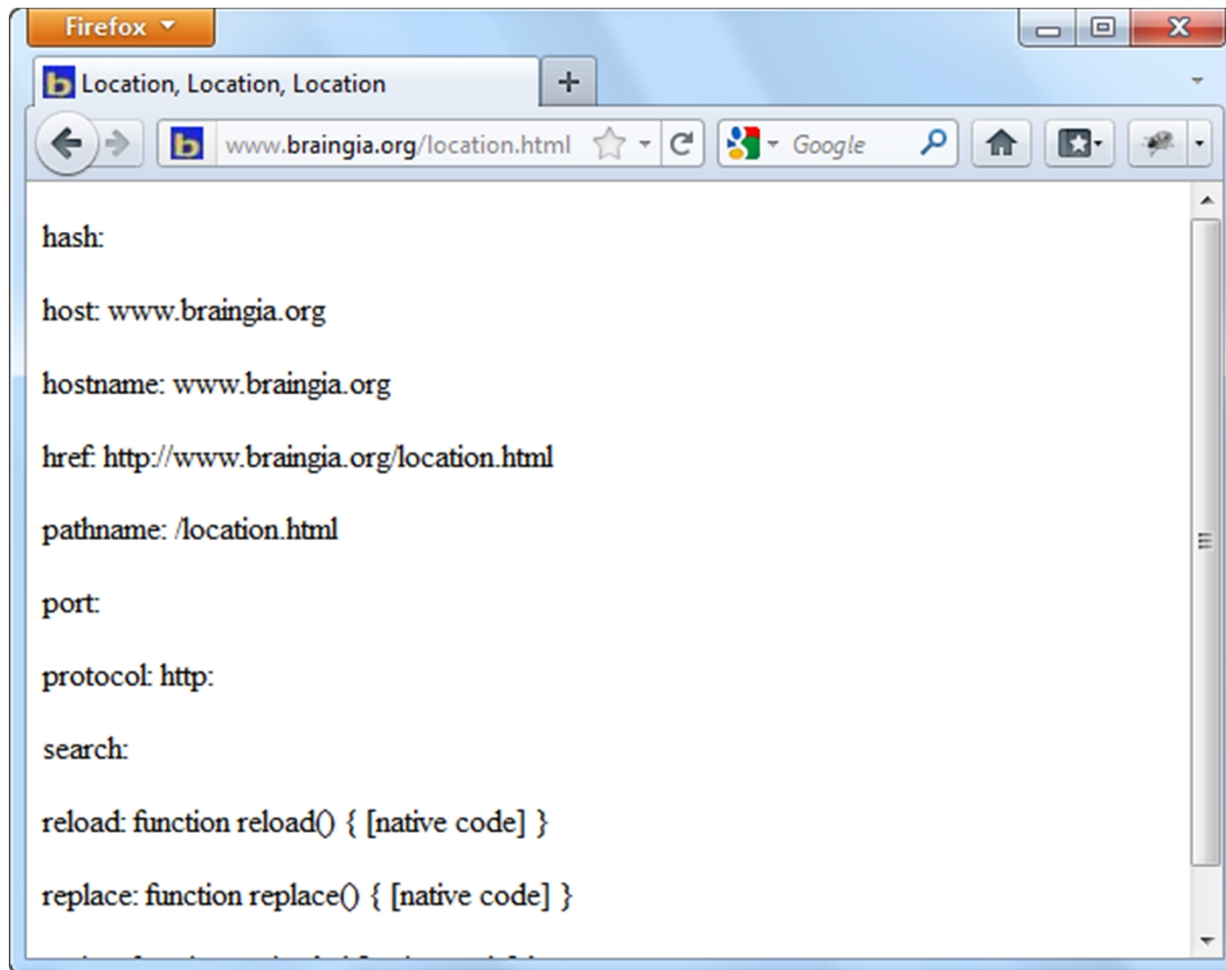
Thuộc tính	Giải thích
host	Trả về tên host và cổng của URL
hostname	Trả về tên host
href	Trả về toàn bộ URL
pathname	Trả về tên đường dẫn của URL
port	Trả về cổng mà server sử dụng cho URL
protocol	Trả về protocol của URL

Phương thức	Giải thích
assign()	Load document mới
reload()	Load lại document hiện tại

```
document.write("host: " + location.host + "<br>");  
document.write("hostname: " + location.hostname + "<br>");  
document.write("href: " + location.href + "<br>");  
document.write("pathname: " + location.pathname + "<br>");  
document.write("port: " + location.port + "<br>");  
document.write("protocol: " + location.protocol + "<br>");
```



- Vào trang <http://www.braingia.org/location.html>



```
<html >
<head>
<script type="text/javascript">
    function newDoc() {
        window.location.assign("http://www.w3schools.com")
    }
</script>
</head>
<body>
<input type="button" value="Load new document"
onclick="newDoc()" />
</body>
</html>
```

- Chứa thông tin về các URL đã từng truy cập

Thuộc tính	Giải thích
length	Trả về số lượng URL trong danh sách History

Phương thức	Giải thích
back()	Load URL trước đó trong danh sách History
forward()	Load URL sau đó trong danh sách History
go()	Load URL cụ thể từ History

- nhúng hàm trong thẻ JavaScript

```
function goBack() {  
    history.back();  
}
```

```
function goNext() {  
    history.forward();  
}
```

- Gọi hàm

```
<p><a href = "#" onclick="goBack()">Back</a></p>  
<p><a href = "#" onclick="goNext()">Next</a></p>
```

- [Xem mã trong tệp m.c “Ví dụ History Object”](#)

- Có r t nhi u ph ng th c l p trình. M i ph ng th c phù h p cho m t m c ích riêng. Ph ng th c l p trình h ng i t ng c phát tri n r ng rãi nh t
- M i i t ng có các thu c tính và ph ng th c riêng
- Các i t ng có các thu c tính và ph ng th c gi ng nhau thu c cùng m t l p
- Browser Object Module là t p h p các i t ng c xây d ng s n giúp l p trình viên thao tác v i trình duy t
- M i trình duy t h tr mô hình BOM theo các cách khác nhau nên l p trình viên c n ph i tìm hi u sâu v trình duy t vì t mã ch y trên nhi u trình duy t

- Trình duyệt là cửa sổ để hiển thị nội dung của trang web.
- Trình duyệt web có các thành phần con là document, frames, history, location, navigator, screen
- Trình duyệt document là nội dung của trang web
- Trình duyệt history chứa thông tin về các url đã từng truy cập
- Trình duyệt location chứa thông tin về url hiện tại
- Trình duyệt navigator chứa thông tin về trình duyệt
- Trình duyệt screen chứa thông tin về màn hình

XIN CHÀO!