

# PHP for Base

Giảng viên: Bùi Quang Đăng

# Contents

1

Giới thiệu về JavaScript

2

Cơ bản về JavaScript

3

JavaScript Window

4

Bài tập

# GIỚI THIỆU VỀ JAVASCRIPT

# Giới thiệu Javascript

## ❖ Nội dung Javascript

```
function HienThiNgayThang()
{
document.getElementById("showdate").innerHTML=Date();
}
<html>
<body>
<h1>My First JavaScript</h1>
<p id="showdate">This is a paragraph.</p>
<button type="button" onclick=" HienThiNgayThang()">Hien thi
    ngay</button>
</body>
</html>
```

# Giới thiệu Javascript

## ❖ Javascript

- Là ngôn ngữ kịch bản trên web, sử dụng để thao tác với các phần tử HTML.
- Sử dụng để viết các hàm xử lý, kiểm tra dữ liệu,...
- Có thể viết javascript dưới dạng file độc lập hoặc trong thẻ `<script></script>` hoặc trong cặp thẻ `<body></body>`
- Ví dụ:
  - `x=document.getElementById("gioithieu")`  
`x.innerHTML="Xin chào";`
  - `document.getElementById("demo").style.color="#ff0000";`

# CƠ BẢN VỀ JAVASCRIPT

# Lập trình Javascript

## ❖ Javascript Output

- Sử dụng để thao tác với các phần tử của HTML
- Sử dụng phương thức getElementById để thao tác với một đối tượng có thuộc tính id
- Sử dụng phương thức document.write("noidung") để hiển thị text trên trang web.

Ví dụ:

- `document.getElementById("demo").innerHTML="Gioi thieu"`
- `document.write("Xin chao cac ban den voi Javascript !")`

# Lập trình Javascript

## ❖ Javascript Comments

- Trong javascript để chú thích sử dụng thẻ // hoặc cặp thẻ /\* và \*/

Ví dụ:

```
/*This is a function javascript*/
```

```
function ThôngBao()
```

```
{
```

```
    alert("Xin chào !"); //Dua ra thông báo
```

```
}
```



# Lập trình Javascript

## ❖ JavaScript Variables

- Trong javascript người ta sử dụng từ khóa “**var**” để khai báo biến

Ví dụ:

```
var hoten=“Nguyen Viet Dung”;
```

```
var hoten=“Nguyen Viet Dung”, gioitinh=“nam”, que=“HN”;
```

```
var x=2012;
```

```
var y =“Nam hien tai”;
```

```
var ketqua = x + y;
```

# Lập trình Javascript

## ❖ Data Types

- Các kiểu dữ liệu được sử dụng trong Javascript
- Sử dụng từ khóa new để khai báo các biến

### ■ Strings

- Sử dụng để thao tác với dữ liệu kiểu chuỗi, ký tự

Ví dụ: `var strCompany="Stanford";`

### ■ Numbers

- Sử dụng để thao tác với dữ liệu kiểu số

Ví dụ:

```
var x=48.00;
```

```
var y=25e3;    // 25000
```

```
var z=25e-3;   // 0.025
```

# Lập trình Javascript

## ❖ Data Types

### ■ Booleans

- Kiểu dữ liệu trả về true hoặc false

Ví dụ: `var x=true, y =false;`

### ■ Arrays

- Sử dụng để thao tác với một mảng trong Javascript

Ví dụ:

```
var arr=new Array();
```

```
arr[0]="Stanford";
```

```
arr[1]="Dao tao va phat trien cong nghe";
```

```
var arr = new Array("Stanford", "Day kinh nghiệm lập trình");
```

# Lập trình Javascript

## ❖ Data Types

### ■ JavaScript Objects

- Kiểu dữ liệu object bao gồm các thuộc tính được định nghĩa bằng cặp name và value.

Ví dụ: `var sv = {firstname:"Nguyen", lastname:"Anh", sex:"male"};`

- Sử dụng cú pháp sau để lấy giá trị từ object với thuộc tính tương ứng:

`sv.firstname;`

`sv["firstname"];`

# Lập trình Javascript

## ❖ Data Types

### ■ JavaScript Objects

- Có thể tạo ra đối tượng Object như sau:

```
svObj=new Object();  
svObj.firstname="Nguyen";  
svObj.lastname="Hung";  
svObj.age=30;  
svObj.eyecolor="blue";
```

# Lập trình Javascript

## ❖ Functions

- Sử dụng để viết một hàm xử lý bằng Javascript
- Cú pháp:

```
function [TenHam](var1, var2, var3)
{
    //Nội dung xử lý
}
```

Ví dụ:

```
function GioiThieu()
{
    alert("Xin chao cac ban !");
}
```

# Lập trình Javascript

## ❖ Functions

- Hàm có kết quả trả về trong Javascript

Ví dụ:

```
function GioiThieu()  
{  
    var strGioiThieu="Stanford";  
    return strGioiThieu;  
}
```

```
function TinhTong(a,b)  
{  
    if (a>b) {return; }  
    x=a+b  
}
```

# Lập trình Javascript

## ❖ Functions

- Một số hàm thông dụng

Method	Description
<a href="#"><u>abs(x)</u></a>	Returns the absolute value of x
<a href="#"><u>acos(x)</u></a>	Returns the arccosine of x, in radians
<a href="#"><u>asin(x)</u></a>	Returns the arcsine of x, in radians
<a href="#"><u>atan(x)</u></a>	Returns the arctangent of x as a numeric value between -PI/2 and PI/2 radians
<a href="#"><u>atan2(y,x)</u></a>	Returns the arctangent of the quotient of its arguments
<a href="#"><u>ceil(x)</u></a>	Returns x, rounded upwards to the nearest integer
<a href="#"><u>cos(x)</u></a>	Returns the cosine of x (x is in radians)
<a href="#"><u>exp(x)</u></a>	Returns the value of E <sup>x</sup>
<a href="#"><u>floor(x)</u></a>	Returns x, rounded downwards to the nearest integer
<a href="#"><u>log(x)</u></a>	Returns the natural logarithm (base E) of x
<a href="#"><u>max(x,y,z,...,n)</u></a>	Returns the number with the highest value
<a href="#"><u>min(x,y,z,...,n)</u></a>	Returns the number with the lowest value
<a href="#"><u>pow(x,y)</u></a>	Returns the value of x to the power of y
<a href="#"><u>random()</u></a>	Returns a random number between 0 and 1
<a href="#"><u>round(x)</u></a>	Rounds x to the nearest integer
<a href="#"><u>sin(x)</u></a>	Returns the sine of x (x is in radians)
<a href="#"><u>sqrt(x)</u></a>	Returns the square root of x
<a href="#"><u>tan(x)</u></a>	Returns the tangent of an angle



# Lập trình Javascript

## ❖ Functions

- Một số hàm thông dụng

Property	Description
<u>Infinity</u>	A numeric value that represents positive/negative infinity
<u>NaN</u>	"Not-a-Number" value
<u>undefined</u>	Indicates that a variable has not been assigned a value

Function	Description
<u>decodeURI()</u>	Decodes a URI
<u>decodeURIComponent()</u>	Decodes a URI component
<u>encodeURI()</u>	Encodes a URI
<u>encodeURIComponent()</u>	Encodes a URI component
<u>escape()</u>	Encodes a string
<u>eval()</u>	Evaluates a string and executes it as if it was script code
<u>isFinite()</u>	Determines whether a value is a finite, legal number
<u>isNaN()</u>	Determines whether a value is an illegal number
<u>Number()</u>	Converts an object's value to a number
<u>parseFloat()</u>	Parses a string and returns a floating point number
<u>parseInt()</u>	Parses a string and returns an integer
<u>String()</u>	Converts an object's value to a string
<u>unescape()</u>	Decodes an encoded string

# Lập trình Javascript

## ❖ Functions

- Một số hàm thông dụng

Method	Description
<a href="#"><u>charAt()</u></a>	Returns the character at the specified index
<a href="#"><u>charCodeAt()</u></a>	Returns the Unicode of the character at the specified index
<a href="#"><u>concat()</u></a>	Joins two or more strings, and returns a copy of the joined strings
<a href="#"><u>fromCharCode()</u></a>	Converts Unicode values to characters
<a href="#"><u>indexOf()</u></a>	Returns the position of the first found occurrence of a specified value in a string
<a href="#"><u>lastIndexOf()</u></a>	Returns the position of the last found occurrence of a specified value in a string
<a href="#"><u>match()</u></a>	Searches for a match between a regular expression and a string, and returns the matches
<a href="#"><u>replace()</u></a>	Searches for a match between a substring (or regular expression) and a string, and replaces the matched substring with a new substring
<a href="#"><u>search()</u></a>	Searches for a match between a regular expression and a string, and returns the position of the match
<a href="#"><u>slice()</u></a>	Extracts a part of a string and returns a new string
<a href="#"><u>split()</u></a>	Splits a string into an array of substrings
<a href="#"><u>substr()</u></a>	Extracts the characters from a string, beginning at a specified start position, and through the specified number of character
<a href="#"><u>substring()</u></a>	Extracts the characters from a string, between two specified indices
<a href="#"><u>toLowerCase()</u></a>	Converts a string to lowercase letters
<a href="#"><u>toUpperCase()</u></a>	Converts a string to uppercase letters
<a href="#"><u>valueOf()</u></a>	Returns the primitive value of a String object

# Lập trình Javascript

## ❖ Operators

- Các toán tử hay sử dụng trong ngôn ngữ Javascript như sau:

Operator	Description	Example	Result of x	Result of y
+	Addition	<code>x=y+2</code>	7	5
-	Subtraction	<code>x=y-2</code>	3	5
*	Multiplication	<code>x=y*2</code>	10	5
/	Division	<code>x=y/2</code>	2.5	5
%	Modulus (division remainder)	<code>x=y%2</code>	1	5
++	Increment	<code>x=++y</code>	6	6
		<code>x=y++</code>	5	6
--	Decrement	<code>x=--y</code>	4	4
		<code>x=y--</code>	5	4

Trong đó:  $y = 5$

# Lập trình Javascript

## ❖ Operators

- Các toán tử hay sử dụng trong ngôn ngữ Javascript như sau:

Operator	Example	Same As	Result
=	<code>x=y</code>		<code>x=5</code>
+=	<code>x+=y</code>	<code>x=x+y</code>	<code>x=15</code>
-=	<code>x-=y</code>	<code>x=x-y</code>	<code>x=5</code>
*=	<code>x*=y</code>	<code>x=x*y</code>	<code>x=50</code>
/=	<code>x/=y</code>	<code>x=x/y</code>	<code>x=2</code>
%=	<code>x%=y</code>	<code>x=x%y</code>	<code>x=0</code>

Trong đó: `x=10`, `y = 5`

# Lập trình Javascript

## ❖ Operators

- Các toán tử hay sử dụng trong ngôn ngữ Javascript như sau:

Operator	Description	Comparing	Returns
==	is equal to	<code>x==8</code>	<i>false</i>
		<code>x==5</code>	<i>true</i>
===	is exactly equal to (value and type)	<code>x==="5"</code>	<i>false</i>
		<code>x===5</code>	<i>true</i>
!=	is not equal	<code>x!=8</code>	<i>true</i>
!==	is not equal (neither value nor type)	<code>x!== "5"</code>	<i>true</i>
		<code>x!==5</code>	<i>false</i>
>	is greater than	<code>x&gt;8</code>	<i>false</i>
<	is less than	<code>x&lt;8</code>	<i>true</i>
>=	is greater than or equal to	<code>x&gt;=8</code>	<i>false</i>
<=	is less than or equal to	<code>x&lt;=8</code>	<i>true</i>

Trong đó: `x=5`

# Lập trình Javascript

## ❖ If...Else Statements

- Khối lệnh điều kiện trong Javascript
- Cú pháp:

```
If (dieu_kien)
{
    //Nội dung xử lý nếu điều kiện đúng
}
Else
{
    //Nội dung xử lý nếu điều kiện kiểm tra sai
}
```

# Lập trình Javascript

## ❖ Switch Statement

- Sử dụng cấu trúc này khi có nhiều điều kiện cần so sánh.

Ví dụ:

```
var day=new Date().getDay();
switch (day)
{
case 0:
    x="Chủ Nhật";
    break;
case 1:
    x="Thứ Hai";
    break;
}
```

# Lập trình Javascript

## ❖ For Loop

- Sử dụng để thực hiện một đoạn lệnh có tính lặp.
- **for**
  - Vòng lặp thực hiện một khối lệnh

Ví dụ:

```
for (var i=0,len=cars.length; i<len; i++)  
{  
  document.write(cars[i] + "<br>");  
}
```

- **for...in**
  - Sử dụng để duyệt các đối tượng kiểu object



# Lập trình Javascript

## ❖ For Loop

- **for...in**

Ví dụ:

```
var person={firstname:"Le",lastname:"Hung",age:25};
```

```
for (x in person)
{
    txt=txt + person[x];
}
```

# Lập trình Javascript

## ❖ For Loop

### ■ while

- Sử dụng lặp thực hiện một công việc có điều kiện kiểm tra.

Ví dụ:

```
while (i<5)
{
  x=x + "Đây là số " + i + "<br />";
  i++;
}
```

# Lập trình Javascript

## ❖ For Loop

### ■ do...while

- Sử dụng lặp thực hiện một công việc có điều kiện kiểm tra.

Ví dụ:

```
do
{
  x=x + "Đây là số " + i + "<br />";
  i++;
}
while (i<5);
```

# Lập trình Javascript

## ❖ JavaScript Try...Catch

- Sử dụng để bắt ngoại lệ khi có lỗi xảy ra trong JS

Cú pháp:

```
try
{
    //Nội dung xử lý
}
catch(err)
{
    //Xử lý khi xảy ra lỗi
}
```

# JAVASCRIPT WINDOW

# Lập trình Javascript

## ❖ Javascript Window

- Đối tượng window được hỗ trợ ở tất cả các trình duyệt.
- Các biến, các phương thức sử dụng toàn cục trong đối tượng window.
- **Window Size**
  - Sử dụng để lấy width và height của trình duyệt (không bao gồm scroll và toolbar)
  - **window.innerHeight**: Lấy chiều cao của trình duyệt
  - **window.innerWidth**: Lấy chiều dài của trình duyệt

# Lập trình Javascript

## ❖ Javascript Window

- Với Internet Explorer 8, 7, 6, 5 có thể sử dụng thuộc tính sau:

`document.documentElement.clientHeight`

`document.documentElement.clientWidth`

Hoặc

`document.body.clientHeight`

`document.body.clientWidth`

# Lập trình Javascript

## ❖ Javascript Window

- Một số phương thức hay sử dụng:
  - `window.open()` - Sử dụng để mở một cửa sổ window
  - `window.close()` - Để đóng một cửa sổ window hiện thời
  - `window.moveTo()` - Sử dụng để di chuyển một cửa sổ window hiện thời
  - `window.resizeTo()` - Sử dụng để thay đổi độ lớn của một cửa sổ window hiện thời



# Lập trình Javascript

## ❖ Window Screen

- Sử dụng đối tượng **window.screen** để lấy kích thước màn hình
  - screen.availWidth - Lấy chiều rộng màn hình
  - screen.availHeight - Lấy chiều cao màn hình

Ví dụ:

```
<script>
```

```
document.write("Available Width: " + screen.availWidth);
```

```
document.write("Available Height: " + screen.availHeight);
```

```
</script>
```

# Lập trình Javascript

## ❖ Window Location

- Sử dụng đối tượng **window.location** để lấy địa chỉ (URL) của trang hiện tại và di chuyển trình duyệt đến một trang mới
  - location.hostname: Trả về domain của web
  - location.path: Trả về đường dẫn của trang hiện thời
  - location.port : Trả về port web (80 or 443)
  - location.protocol: Trả về giao thức (http:// or https://)
  - location.assign(): Mở một document

# Lập trình Javascript

## ❖ Window Location

Ví dụ:

```
<script>
function newPage()
{
    document.write(location.href);
    document.write(location.pathname);
    window.location.assign("http://www.stanford.com.vn")
}
</script>
```

# Lập trình Javascript

## ❖ Window History

- Sử dụng **window.history** để lấy thông tin về lịch sử của trình duyệt
  - **history.back()** – Sử dụng để lấy thông tin click back của trình duyệt
  - **history.forward()** – Lấy thông tin khi click forward của trình duyệt

Ví dụ:

```
function goHistory()  
{  
    document.write(window.history.back());  
    document.write(window.history.forward());  
}
```

# Lập trình Javascript

## ❖ JavaScript Popup

- Sử dụng trong Javascript để hiển thị thông báo, bao gồm: Alert box, Confirm box, and Prompt box.

- **Alert box:**

Ví dụ:

```
<script>
  function myFunction()
  {
    alert("Xin chao cac ban !");
  }
</script>
```

# Lập trình Javascript

## ❖ JavaScript Popup

- **Confirm box:**

Ví dụ:

```
function myFunction()  
{  
    var r=confirm("Press a button");  
    if (r==true)  
    {  
        x="You pressed OK!";  
    }  
    else  
    {  
        x="You pressed Cancel!";  
    }  
}
```

# Lập trình Javascript

## ❖ JavaScript Popup

### ■ Prompt Box:

- Sử dụng để hiển thị thông báo và cho người dùng nhập thông tin vào.

Ví dụ:

```
var name=prompt("Nhập tên của bạn vào","Không nhập");  
if (name!=null && name!="")  
{  
  x="Xin chào " + name + "! Bạn có khỏe không ?";  
}
```

Chú ý: Trong nội dung thông báo để xuống dòng dùng ký tự: \n



# Thank You !