

자바스크립트로 홈페이지를 다양하게

이번호부터는 홈페이지에 보다 다양한 기능을 가능하게 해주는 자바스크립트에 대해 자세한 내용을 다루도록 한다. 지금부터 준비하는 자바스크립트에 대한 강좌는 프로그램에 대해 특별한 경험이 없는 독자들을 대상으로 프로그램에 대한 기본적인 마인드와 자바스크립트의 핵심적인 구조 및 활용까지를 다루어 볼 것이다.

황희정/넥스텔 멀티미디어연구소



목 차

1. 스크립트언어와 자바스크립트의 개념 (1회)
2. 자바스크립트의 규칙 과 기본 문법 (1회)
- 3.4. 자바스크립트 실전 활용 (2회)
5. 향상된 자바스크립트 기법 (3회)
6. 쿠키와 자바 스크립트 (4회)

최

근 자바스크립트의 활용이 높아졌지만 비교적 이해하기도 쉽고 소스가 노출되므로, 다른 사이트에서 만들어진 내용을 그대로 복사해 이용하는 경우가 많이 발생하고 있다. 하지만, 자바스크립트 역시 하나의 프로그램을 짜는 것이므로, 프로그램에 대한 기본적인 마인드를 가지고 있다면, 보다 쉽고 다양한 응용 프로그램을 제작할 수 있을 것이다.

스크립트언어와 자바스크립트

스크립트언어, '곧바로 실행된다'

일반적으로 스크립트언어는 라인 단위로 번역되어 곧바로 실행되는 프로그래밍 언어로 흔하게 접할 수 있는 도스의 Autoexec.bat 및 각종 배치 파일들과 매킨토시 계열에서 사용되는 Apple Script를 들 수 있다. 이밖에 CGI(Common Gateway Interface)에서 많이 사용되고 있는 Perl, tcl/tk 와 같은 언어들도 대표적인 스크립트 언어라

고 할 수 있다. 그러면, 스크립트 언어의 구조와 실행 방식에 대해 살펴보기로 한다.

스크립트언어의 예를 들면, 다음과 같다.

(autoexec.bat)

```
loadhigh C:\WINDOWS\nlsfunc.exe C:\WINDOWS\country.sys
SET PATH=C:\Cafe\Bin;C:\Cafe\Java\Bin;%PATH%;c:\win_util\trumpet
SET CLASSPATH=C:\PROGRA~1\NETSCA~1\COMMUN~1\PROGRAM\JAVA
SET CLASSPATH=.;%CLASSPATH%
```

(shell script)

```
JAVA_HOME = dirname $PRG/..
if { -z "$JAVA_HOME" } ; then
    export JAVA_HOME
    JAVA_HOME=$J_HOME
fi
CLASSPATH="$({CLASSPATH-})"
if { -z "${CLASSPATH}" } ; then
    CLASSPATH="$J_HOME/classes:$J_HOME/lib/classes.zip"
else
    CLASSPATH="$CLASSPATH:$J_HOME/classes:$J_HOME/lib/classes.zip"
fi
export CLASSPATH
```

```

(perl)
#####
# Get Data Number Subroutine

sub get_number {
  open(NUMBER, "$basedir/$datafile");
  $num = <NUMBER>;
  close(NUMBER);
  if ($num == 99999) {
    $num = "1";
  }
  else {
    $num++;
  }
}

```

스크립트언어의 실행과정

스크립트언어의 실행과정은 다음과 같다.

1. 텍스트로 된 소스 프로그램.
2. 인터프리터에 의해 로딩(자바스크립트의 경우 넷스케이프 브라우저가 인터프리터의 기능을 수행한다).
3. 인터프리터는 라인단위로 소스를 읽으면서 해당 명령을 수행한다.

스크립트언어의 장점은 첫째, 비교적 구현하기가 쉽다. 둘째, 별도의 개발툴이 필요하지 않기 때문에 개발에 따른 소요비용을 줄일 수 있다. 셋째, 컴파일 등이 필요하지 않기 때문에 실행 속도가 빠르다는 것 등이 있다.

객체지향, '재활용을 높인다'

객체지향 프로그래밍(Object Oriented Programming)은 프로그램의 재활용을 높이고, 동일한 기능을 구현하기 위해 소요되는 시간 및 기술적 투자의 낭비를 줄이기 위해 고안된 프로그램 기법으로 상당히 오래전부터 논의되어온 프로그래밍 기법이다. 현재 대표적인 객체지향 프로그래밍언어로는 LISP, C++ 계열 과 최근에 관심이 모아지고 있는 자바(Java)등이 있는데, 이들의 기본적인 개념은 같다고 볼 수 있다.

객체(Object)는 다음과 같은 특징이 있다.

1. 객체는 데이터와 데이터 사이에서 상호 작용하는 일련의 절차들로 구성된 소프트웨어의 집합이다.
2. 객체를 보다 현실적으로 나타내기 위해, 상태(Property)와 동작

(Method)이라는 특징적인 요소들을 가지고 있다(())는 자바스크립트 기준용어).

3. 우리가 주위에 보고 있는 모든 것들은 객체가 될 수 있다(사람, 자동차, 시계).
4. 자바 스크립트도 객체지향 언어라고 할 수 있다.

예를 들어, 자동차라는 객체가 있다고 하자. 그렇다면 이 객체에는 어떤 속성(상태)이 있을까? 먼저 색상을 들 수 있다. 또한, 바퀴의 개수, 모양, 좌석수, 각종 옵션의 장착 여부가 여기에 해당한다. 또한, 자동차는 앞으로 가는 자동차와 뒤로 가는 자동차, 회전하는 자동차 등의 동작을 가지고 있다. 이처럼 객체는 우리가 표현하고자 하는 특정 사물을 자연스럽게 접근할 수 있는 대상이라고 생각할 수 있다.

자바스크립트를 예로 들면, 'window' 라는 객체는 윈도우 이름, 윈도우 메뉴바, 스크롤바, 오픈할 URL 등등의 속성을 가지고 있으며, open(), close() 등의 함수(동작)를 가지고 있다.

자바스크립트, 'HTML의 한계를 극복'

자바 스크립트는 넷스케이프사에서 만든 스크립트 언어로 HTML의 한계를 벗어나 프로그램적인 요소를 지원하게 함으로써, 기존에 자바 혹은 다른 플러그인(Plug-in)을 사용해야 했던 다양한 기능들을 빠른 속도로 지원할 수 있도록 설계된 언어이다.

최초에 자바스크립트는 라이브 스크립트로 불리웠으며, 실제로 자바와 관련이 있기 보다는 자바언어와 비슷한 문법으로 이루어졌고 사용자에게 인지도를 높이기 위해 자바스크립트로 이름이 바뀌게 된 것이다. 현재 자바스크립트는 넷스케이프 2.0 이상과 IE3.0 이상에서 모두 지원되는데, 넷스케이프 브라우저에는 브라우저 고유기능과 관계되는 추가적인 기능을 자바스크립트로 지원하고 있다.

자바스크립트의 장점으로는 다음 네가지를 들 수 있다.

1. HTML로 구성된 홈페이지와 조화롭게 응용이 가능하다.
2. 브라우저를 컨트롤 할 수 있는 다양한 방법이 제공된다.
3. 자바 애플릿 혹은 다른 플러그인 보다 빠른 처리속도를 나타낸다.
4. 브라우저 업체에서 새롭게 제공하는 오브젝트를 통해 새로운 기능을 쉽게 적용할 수 있다.

그러나, 자바스크립트의 단점으로 제기되는 것으로 ▲ 스크립트 언어이다. ▲내장된 오브젝트만 사용할 수 있으므로, 브라우저 제조업체에서 새로운 오브젝트를 만들 때까지 기다려야 한다는 것 등이 있다.

자바스크립트의 규칙과 기본문법

프로그램의 기본 '기초를 튼튼하게'

이번에는 자바스크립트를 보다 잘 활용하기 위해 프로그래밍에 대한 기본적인 개념을 이해하는 시간이다. 모든 일은 기초가 튼튼해야 하는 법...

프로그램 작성 과정은 다음과 같다.

1. 어떤 기능을 가진 프로그램을 짤 것인가?
2. 프로그램의 인자는 무엇이며 리턴값은 무엇인가?
3. 프로그램의 플로우를 개괄적으로 작성.
4. 코딩.
5. 프로그램 혹은 프로그램내에서 다른 함수의 호출시점?(이벤트)
6. 디버깅

일반적으로 프로그램은 위와 같은 절차를 거쳐 작성된다고 볼 수 있다. 가장 단순하게 표현한 것이다. 그러면, 이제 위와 같은 내용을 중심으로 실제 자바스크립트 프로그램 작성과정을 살펴보기로 한다.

자바스크립트의 기본 구성 '실제'

자바스크립트도 하나의 프로그램이므로 위에서 언급했던 프로그램의 일반적인 제작과정과 비슷한 절차를 거치게 된다. 그러면, 실제로 한 상황을 예로 자바스크립트를 작성하기 위한 준비절차와 작성하는 과정을 살펴보기로 한다. 이 예에서는 프로그램의 인자와 리턴값은 필요없다.

어떤 기능을 가진 프로그램을 짤 것인가?

홈페이지에서 도움말을 보여주기 위해 버튼을 누르면 새로운 윈도우가 나타나고 거기에 도움말이 적힌 HTML화일이 표시되면서 로딩되며, 브라우저의 하단에는 도움말을 호출했다는 정보를 출력하도록 하며, 도움말 하단에는 윈도우를 닫는 버튼이 있어 이 버튼을 누르면 윈도우가 닫히게 된다.

프로그램의 플로우

1. 새로운 윈도우를 오픈하는 함수 호출 - 메뉴바 등이 나타나지 않게 한다.
2. 브라우저 하단의 상태창에 도움말 호출이라는 메시지를 출력한다.

3. 도움말 HTML파일에 버튼을 만들고 버튼을 클릭하면 윈도우를 닫는 함수를 호출하는 부분을 추가한다.

필요한 오브젝트와 프로퍼티(Property)의 정리

1. 윈도우를 오픈 : window.open("URL", "윈도우 이름", "윈도우에 대한 각종 옵션")
2. 브라우저 하단에 메시지 출력 : window.status = "출력 메시지"
3. 버튼 클릭시 도움말 윈도우 닫음
: <form><input type=button onClick=window.close()></form>

코딩 (main.html)

```
<html>
<head>
<script language=javascript>

function helpwin() {
    window.open("help.html", "help", "toolbar=0,directorie=0,
        menubar=0,scrollbars=0, width=200, height=250");
    window.status = "도움말 윈도우 오픈중..";
}

</script>
</head>

<body bgcolor=#ffffff>
<H3> 자바스크립트 테스트입니다.</H3>
<BR><P>
<form>
<input type=button onClick="helpwin()" value="도움말">
</form>
```

(help.html)

```
<html>
<body bgcolor=#ffffff>
<H3> 도움말 입니다.</H3>
<BR><P>
<font color=#0000ff>
1. 이것은 도움말 이지요?<P>
2. 하지만 정말로 도움말 일까요?<P>
3. 그래도 일단은 믿어 봅시다.<P>
</font>
<HR>
<form>
<input type=button onClick=window.close() value="닫을래">
</form>
</body>
</html>
```

자바스크립트의 기본구성

```

<HTML>
<HEAD>
<Script Language=JavaScript>
자바스크립트 메소드
.....
</Script>
</HEAD>
<BODY>
<... Event=Method 호출>
</BODY>
</HTML>


```

Object, Property, Method

자바스크립트는 프로그래머로 치자면 부족한 점이 많지만, 어쨌든 객체지향적인 프로그램 구조를 가지고 있다. 따라서, 자바스크립트에도 객체지향적 프로그램의 기본은 객체와 객체의 상태를 나타내는 프로퍼티, 그리고 동작을 나타내는 메소드(메소드는 함수라고도 함)라는 것이 있다. 자바 혹은 C++ 등의 객체지향 프로그램에서는 클래스라고 하는 객체를 사용자가 정의해 사용할 수도 있고, 기존의 객체를 상속받아 수정된 객체를 이용할 수도 있다.

하지만, 자바스크립트는 그러한 기능은 없고, 단지 정의되어 있는 객체만을 이용할 수 있다. 예를 들어, 사용자가 자바스크립트로 어떠한 기능을 구현하고자 한다면, 최소한 구현하고자 하는 내용을 지원하는 객체와 객체의 속성이 존재해야 한다.

위의 예에서처럼 새로운 윈도우를 열고, 윈도우의 메뉴를 표시하지 않고, HTML화일을 로딩하는 일련의 과정은 자바스크립트에 'window' 라는 객체와 동작에 해당되는 open()이라는 함수, 그리고 윈도우객체가 가지는 크기, 메뉴, 스크롤바 등등의 속성이 정의되어 있기 때문에 사용가능한 것이다. 따라서, 자바스크립트를 보다 잘 활용하려면 자바스크립트에서 지원하는 객체와 기본 함수 그리고, 객체들의 속성에 대해 잘 알고 있어야 한다.

다음에는 자바스크립트의 기본 문법을 중심으로 보다 자세한 실질적인 내용을 다루어 보도록 한다. 

자바스크립트 함수

abs	forward	round
acos	getDate	select
alert	getDay	setDate
anchor	getHours	setHours
asin	getMinutes	setMinutes
atan	getMonth	setMonth
back	getSeconds	setSeconds
big	getTime	setTimeout
blink	getTimezoneOffset	setTime
blur	getYear	setYear
bold	go	sin
ceil	indexOf	small
charAt	isNaN	sqrt
clear	italics	strike
clearTimeout	lastIndexOf	sub
click	link	submit
close (document)	log	substring
close (window)	max	sup
confirm	min	tan
cos	open (document)	toGMTString
escape	open (window)	toLocaleString
eval	parse	toLowerCase
exp	parseFloat	toUpperCase
fixed	parseInt	unescape
floor	pow	UTC
focus	prompt	write
fontcolor	random	writeln
fontSize		

자바스크립트 객체

```

window
|
+--parent, frames, self, top
|
+--location
|
+--history
|
+--document
|
+--forms
| |
| elements (text fields, textarea, checkbox, password
|         radio, select, button, submit, reset)
+--links
|
+--anchors

```