■ 정규식

정규식은 특별히 정한 기호를 사용하고 있는 문자열의 패턴으로 슬래시(/정규식/)로 구분한 문자들의 집합이다. 정규식은 일반문자와 특별한 의미를 가지는 문자를 조합하여 구성하며 문자열 중에서 원하는 문자를 검색하거나 검색 후 교체하기 위해서 사용되는 표현식이다.

1. 정규식을 사용하는 메소드

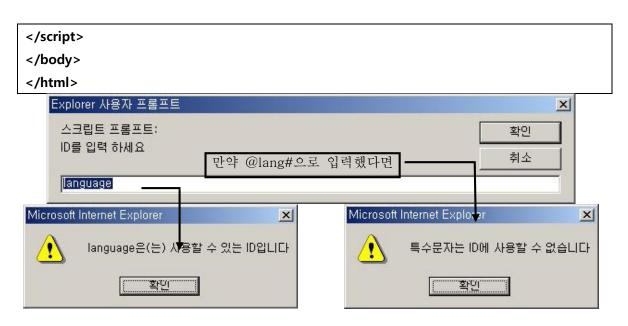
정규식은 슬래시와 슬래시 사이에 오는 문자들의 집합으로서 그 자체로 의미를 가지고 있다. 정규식은 search(), match(), replace(), split() 메소드의 매개변수로 사용될 경우에 이 메소드들은 더욱 강력한 힘을 발휘하게 된다. match(), split() 메소드는 검색된 결과를 array 객체로 리턴하게 된다.

- search(string) 메소드

search(string) 메소드의 리턴값			
문자열이 String로 매치될	문자열 중에서 매개변수와 매치되는 첫 번째 문자열의 시작		
때	index값		
String과 매치되지 않을 때	매치되는 문자열이 없을 때는 -1을 리턴한다.		

search() 메소드는 매개변수와 일치하는 문자열이 있는지 검사하는 메소드이다

```
<html>
<head> <title> 문자열을 이용한 search() 메소드 </title></head>
<body>
<script type="text/javascript">
      var special = "~!@#$%^&*( )_=+|₩₩{}[];:\"\""<>,?/";
                                                      // 특수문자 일람
      var str = prompt("ID를 입력 하세요", "language");
                                                      // 사용자가 입력한 ID
      var bool = false;
                                                       // Boolean 초기값
      for(i=0; i<str.length; i++){
                                               // c는 ID중에서 index가 i인 문
             var c = str.charAt(i);
자
             if(special.search(c) >= 0){
                                              // 특수문자 중에 c가 있으면
true
                    bool = true;
                    break;
             }
      }
      if(bool)
          alert("특수문자는 ID에 사용할 수 없습니다")
      else
          alert( str + "은(는) 사용할 수 있는 ID입니다")
```



```
<html>
<head> <title> 문자열을 이용한 search() 메소드 </title></head>
<body>
<script type="text/javascript">
      var reg = /\text/javascript">
      var reg = /\text/javascript">
      var str = prompt("ID를 입력 하세요", "language");      // 사용자가 입력한 ID

if(str.search(reg)>=0)
      alert("ID는 영문자와 숫자만 사용할 수 있습니다.")
      else
      alert( str + "은(는) 사용할 수 있는 ID입니다")
</script>
</body>
</html>
```

- match(string) 메소드

match(string) 메소드의 리턴값			
"문자열"	매개변수와 일치하는 첫 번째 문자열을 배열변수가 하나인 Array 객체로		
	리턴한다. 리턴되는 Array 객체의 길이는 1로써 첫 번째 매치된 문자열이		
	배열변수의 index 0에 할당된다. 일치되는 내용이 없을 때는 null을 리턴한		
	다.		
/정규식/g	매개변수와 일치하는 첫 번째 문자열을 배열변수가 하나인 Array객체로 리		
	턴한다. 만일 정규식에 g속성이 사용되면 첫 번째 문자열 뿐 아니라 일치		
	되는 모든 문자열을 순서대로 Array 객체의 배열변수에 할당한다. 이 때		
	배열의 길이는 매치되는 문자열의 수와 같다. 매치되는 문자열이 없을 때		

는 null을 리턴한다.

- replace(string1, string2) 메소드

replace(string1, string2) 메소드의 리턴값			
"문자열"	문자열 중에서 매개변수 string1과 매치되는 첫 번째 문자열을 두 번째 매		
	개변수 string2로 교체하여 새로운 문자열을 리턴한다. 매치되는 문자열이		
	없을 때는 String 객체 자체(원본 문자열)을 리턴한다.		
/정규식/g	정규식의 g속성이 사용되면 매치되는 첫 번째 문자열뿐만 아니라 모든 문		
	자열을 두 번째 매개변수로 교체하여 교체된 문자열을 리턴한다. 매치되는		
	문자열이 없을 때는 원본 문자열을 리턴한다.		

☞ 메타문자

정규식에서 사용하는 메타문자는 일반문자와 달리 의미를 가지고 있는 문자들을 말한다. 여기에는 단독으로 의미를 가지는 것(\$, ^, *, +, |, ?)들과 그룹으로 묶여서 의미를 가지는 것([], {}, ())들이 있다.

	정규식에 사용되는 메타문자와 단축 문자클래스 의미		
[]		[abc]는 문자 "a","b","c" 중에 하나의 문자와 매치되는 것.	
[^]		[^abc]는 "a","b","c"를 제외한 임의의 한 문자와 매치되는 것	
{}		{}앞에 있는 항목이 반복될 때 반복횟수를 지정하는 것	
()		여러 항목을 묶어서 단일 단위로 사용할 수 있도록 하는 것	
		a b는 a나 b중에 하나와 매치된다.	
₩b		단어와 단어 사이에 경계에 매치되어 해당 단어만을 정확하게 매	
		치시킨다.	
₩B		단어의 경계가 아닌 단어의 위치에 매치된다.	
₩d	[0-9]	0부터 9까지의 숫자 중에 하나와 매치된다.	
₩D	[^0-9]	숫자(0~9)를 제외한 임의의 문자에 매치된다.	
₩s		임의의 공백문자(₩t, ₩n, ₩r)에 매치된다. (공백을 의미하지만 출력	
		되지 않는 문자들)	
₩S		공백문자가 아닌 임의의 한 문자에 매치된다.	
₩w	[A-Za-z0-9]	알파벳과 숫자에 매치되는 임의의 한 문자(알파벳과 \d)	
₩W	[^A-Za-z0-9]	알파벳과 숫자가 아닌 임의의 한 문자	
^		문자열 행의 시작부분과 매치된다.	
\$		문자열 행의 끝부분과 매치한다.	
•		줄바꿈 문자을 제외한 "모든 문자" 하나에 매치된다.	
?		앞에 있는 항목과 1회 이하로 매치된다.	
+		앞에 있는 항목과 1회 이상 매치된다.	
*		앞에 있는 항목과 0회 이상 매치된다.	

- 예제

파일명 : check.js

```
function CheckForm(f){
var r1 = /[0123456789]/g; //숫자 0에서 부터 9까지의 문자 클래스
/* var r1 = /[0-9]/
   var r1=/₩d/
*/
 var r2=/[^a-zA-Z0-9]/q; //영문자와 숫자를 제외한 모든 문자
//var r2=/₩W/
uname=f.user_name.value
 b=false;
  if (uname.replace(/\forall s/g,"")=="") {
       alert("이름을 입력하세요.");
      f.user_name.focus();
       return false;
 }
  for (i=0;i<uname.length;i++){
         var c=uname.charAt(i)
      if(r3.search(c)>=0 || c.search(r1)>=0){ //찾지 못하였을 때 반환하는 값:-1
       b=true;
               break;
     }
 }
  if(b){
      alert("이름은 문자만 입력하셔야 합니다.")
     f.user_name.value="";
     f.user_name.focus();
     return false;
 }
  var strjumin1 = f.jumin1.value;
  var strjumin2 = f.jumin2.value;
  var j1 = / \forall d\{6\}/g;
  var j2 =/\foralld{7}/g;
  if (strjumin1.replace(/lambda s/g,"")=="" || strjumin1.match(j1)==null) {
      alert("주민등록번호 6자리를 입력하지 않았거나 숫자만 가능합니다.");
          f.jumin1.value=""
      f.jumin1.focus();
      return false;
```

```
/*for (var i=0;i<strjumin1.length;i++){
        val1=strjumin1.charAt(i)
   if (val1 < "0" || val1 > "9") {
          alert("주민등록번호는 숫자만 가능합니다.");
              f.jumin1.value="";
      f.jumin1.focus();
          return false;
   }
}*/
if (strjumin2.replace(/\psis,g,"")=="" || strjumin2.match(j2)==null) {
    alert("주민등록번호 7자리를 입력하지 않았거나 숫자만 가능합니다.");
        f.jumin2.value=""
    f.jumin2.focus();
    return false;
}
var jumin= strjumin1+strjumin2
var m=parseInt(jumin.charAt(12))
var sum=0
 for (i=0, j=2; i<=11; i++){
   sum=sum+parseInt(jumin.charAt(i))*j;
  j=j+1
       if (j=10)
         j=2
 }
var namuji = sum % 11;
var gumsa = 11 - namuji;
var check_no = gumsa % 10;
if (check_no != m) {
   alert("주민등록번호를 잘못입력하셨습니다. 다시 입력해 주세요.");
   f.jumin1.value="";
   f.jumin2.value="";
   f.jumin1.focus();
       return false;
```

```
id = f.user_id.value
 if (id.replace(/\foralls/g,"")=="" || id.search(r2)>=0 ) {
       alert("아이디를 입력되지 않았거나 아이디는 영문자와 숫자로 구성되어야 합니다.");
           f.user_id.value=""
       f.user_id.focus();
       return false;
  }
 if (id.substr(0,1).match(r1)){
         alert("아이디의 첫 글자에는 숫자가 올 수 없습니다.");
         f.user_id.value=""
     f.user_id.focus();
     return false;
}
function check_focus(f) {
var strfocus = f.jumin1.value.length;
if(strfocus == 6)
   f.jumin2.focus();
}
function id_check(){
window.open("idcheck.html","","width=300,height=100")
```

파일명: style.css

```
body, td, textarea {font-size:9pt; font-family:굴림;}
.back{background:#fff2ff}
.ta{border:1px solid #CCCCCC; width:520; }
.t0{border-bottom:1px solid #CCCCCC;font-weight:bold;text-align:center;
    background-color:#fafafa }
.t1{border-bottom:1px solid #CCCCCC;border-right:1px solid #CCCCCC;
    text-align:center; background-color:#fafafa; width:100}
.t2{border-bottom:1px solid #CCCCCC;}
.t3{border-right:1px solid #CCCCCC;text-align:center;background-color:#fafafa }
.in{border:1px solid #DADADA; font-size:9pt; font-family:굴림;}
.bu{border:1px solid #CCCCCC; background-color:#fafafa; width:50; height:20;
    font-size:9pt; text-align:center}
.in1{border-width:0px 0px 1px 0px;border-style:solid; border-color:#DADADA;
    font:9pt 굴림; height:18}
```

```
.d{border:1px solid #CCCCC;width:540; padding:10 }
a{ font-size:9pt; text-decoration:none; color:darkblue }
```

파일명: form.html

```
<html>
 <head><title>실명확인</title>
 k rel="stylesheet" type="text/css" href="style.css">
 <script language="javascript" src="check.js">
 </script>
</head>
<body><center><br><b>[회원 실명 확인]</b>
<form name="regform" method="post" onsubmit="return CheckForm(document.regform)" >
이용자 본인의 이름 및 주민등록번호를 정확히 입력해주시기
    바랍니다.
 이름
    <input type=text size=28 name="user_name" maxlength=5
  style="ime-mode:active;" class="in">  [공백없이 입력하세요]
 주민등록번호
    <input type=text size=12 maxlength=6 name="jumin1"
   class="in" onKeyUp="check_focus(this.form)"> - 
  <input type=password size=12 maxlength=7 name="jumin2" class="in">
 아이디
    <input type=text size=28 maxlength=10 name="user_id" class="in">
     <input type="button" value="아이디검색" class="bu" onfocus="blur()" style="width:80"
     onclick="id_check()">
 <br>
<div style="width:520;">
 <input type="submit" value="등록" class="bu" onfocus="blur()">&nbsp;&nbsp;
 <input type="reset" value="취소" class="bu" onfocus="blur()">
</div>
```

</form>
</center>
</body></html>

