

유효성검사





유효성검사



유효성검사

유효성검사

- 사용자가 입력한 데이터가 양식에 맞는지 검사하는 것
 - 1) 비밀번호와 비밀번호 확인의 값이 같은지 확인
 - 2) 아이디 또는 비밀번호의 복잡도가 만족하는지
- 유효성검사를 통해서 맞지 않으면 `false`를 리턴하여 `submit`동작을 하지 않게 처리





정규표현식



정규표현식

정규표현식(Regular Expression)

- 사전적 의미로 특정한 규칙을 가진 문자열의 집합
- 주로 프로그래밍 언어나 Text Editor 등에서 문자열의 검색과 치환을 위한 용도로 사용
- 일반 조건문으로 다소 복잡할 수 있는 패턴을 정규표현식을 이용하면 간단하게 표현 가능
- 간단하게 표현 하는 만큼 가독성이 떨어져 표현식을 숙지해야 함
- 확장성에 따라 다양한 정규표현식이 존재하지만 기본적인 것은 비슷함



정규표현식

Meta 문자

- 정규표현식에서 사용하는 기호
- 어떠한 의미가 있는 문자

표현식	설명
$\wedge x$	문자열의 시작을 표현 \rightarrow x로 시작하는 문자
$x\$$	문자열의 끝을 표현 \rightarrow x로 끝나는 문자
\cdot	임의의 한문자 \rightarrow 모든 단일문자를 의미
$x+$	앞의 문자 1회 이상 반복 \rightarrow x가 1번 이상 반복
x^*	앞의 문자 0회 이상 반복 \rightarrow x가 0번 이상 반복
$x?$	앞의 문자가 존재하거나 존재하지 않을 때 사용



정규표현식

Meta 문자

- 정규표현식에서 사용하는 기호
- 어떠한 의미가 있는 문자

표현식	설명
(xyz)	()안의 내용을 그룹화
[xyz]	[]사이에 존재하는 문자들 중 하나에 일치
[^xyz]	not을 표현하며 xyz를 제외한 문자를 의미
[x-z]	범위를 표현하며 x~z 사이의 문자를 의미
x{n}	반복을 표현하며 x문자가 n번 반복
x{n,}	반복을 표현하며 x문자가 n번 이상 반복
x{n,m}	반복을 표현하며 x문자가 최소 n번 이상 최대 m번 이하



정규표현식

Meta 문자

- 정규표현식에서 사용하는 기호
- 어떠한 의미가 있는 문자

표현식	설명
<code>\d</code>	숫자를 의미
<code>\D</code>	숫자가 아닌 것을 의미
<code>\s</code>	공백을 의미
<code>\S</code>	공백이 아닌것을 의미
<code>\w</code>	word를 표현하며 알파벳+숫자+_ 중의 한 문자임을 의미
<code>\W</code>	non word를 표현하며 알파벳+숫자+_ 가 아닌 문자를 의미

※ 정규표현식의 메타문자는 명시된 것 이외에도 다양함



정규표현식

Flag

- 정규표현식을 사용할 때 Flag를 사용하지 않으면 문자열에 대해서 검색을 한번만 처리하고 종료
- Flag는 사용해도 되고 사용하지 않아도 됨

표현식	설명
g	Global → 문자열 내의 모든 패턴을 찾음
i	Ignore case → 대상 문자열에 대해서 대/소문자를 식별하지 않는 것을 의미
m	Multi Line → 대상 문자열이 다중 라인의 문자열인 경우에도 검색하는 것을 의미

※ 정규표현식의 Flag는 명시된 것 이외에도 다양함



정규표현식

javascript에서의 정규표현식 사용

- 정규 표현식 메소드를 이용하여 검사

메소드	설명
test()	일치하는 문자열을 검사할 때 사용(true/false 반환)
exec()	일치하는 문자열을 찾을 때 사용(정보를 가진 배열 반환)
replace()	일치하는 문자열을 찾아 대체할 때 사용



정규표현식

javascript에서의 정규표현식 사용예

```
function regTest(){  
    //id가 input1 인 input태그의 value값  
    var str = document.getElementById("input1").value;  
    //정규표현식 작성  
    //    /정규표현식/flag; 형식으로 작성  
    // a로 시작하는지 확인하는 정규표현식 작성  
    var regExp = /^a/;  
    //test메소드를 사용하여 정규표현식을 검사(true/false 반환)  
    regExp.test(str);  
  
}
```

