

## pattern 예제 - 영문 3글자 국가코드 입력

※ 문자열이 3개의 알파벳인지 체크

```
<form action="/action_page.jsp">
  <label for="country_code">3글자 국가코드:</label>
  <input type="text" id="country_code" name="country_code" pattern="[A-Za-z]{3}" title="3글자 국가코드 입력"><br><br>
  <input type="submit">
</form>
```

### 설명 : pattern="[A-Za-z]{3}" 의미

3개의 알파벳으로 된 문자열인지 체크.

패턴 구성요소:

- **[A-Za-z]**: 대문자 알파벳 또는 소문자 알파벳 중 하나를 나타내는 패턴임. 대괄호([]) 안에 여러 문자를 넣어주면 해당 위치에 올 수 있는 문자를 제한할 수 있음. 대문자와 소문자를 모두 포함한 이유는, 대소문자를 구분하지 않고 검사하고자 하는 경우를 고려한 것임.
- **{3}**: 앞의 패턴이 세 번 반복되어야 함을 나타내는 패턴임. 중괄호({}) 안에 숫자를 넣어주면, 해당 패턴이 몇 번 반복되어야 하는지를 지정 가능함.

패턴 일치 예제:

- "abc"
- "Def"
- "xyz"
- "Qwe"

패턴 불일치 예제:

- "ab" (∵ 두 글자)
- "12A" (∵ 알파벳 외 다른 문자 포함)
- "ABCD" (∵ 네 글자)

패턴 활용: 3개의 알파벳으로 된 문자열 찾을 때 사용. (예)

- 3글자로 된 단어 검색
- 이름의 앞 글자 3개 추출
- 전화번호나 우편번호에서 지역코드 같은 세 글자 추출

패턴 단점

- 대소문자 구분 않기 때문에, 일부 상황에서는 원하는 결과 못 얻을 수도 있음. (이 경우, 대문자만, 또는 소문자만 찾는 패턴을 사용해야 함.)

## pattern 정의

<input> 태그 입력값 패턴 검사.

1. <input> 태그 중 pattern 속성 사용 가능 type 종류  
text, date, search, url, tel, email, password.

- 2.
- **title** 전역속성 사용해 사용자에게 패턴 설명 권장.
  - 패턴 체크는 서버에 폼데이터 제출된 후 이루어짐. (※ 입력 후 입력창에서 커서 떠날 때 체크하려면, JS 이벤트 이용해야 함.)

## pattern 구문

```
<input pattern="regexp">
```

[속성값]

*regex*

입력값 체크할 정규표현식 지정.

## pattern 예제 - 비밀번호 길이만 체크 입력

```
<form action="/action_page.jsp">
  <label for="pwd">패스워드:</label>
  <input type="password" id="pwd" name="pwd" pattern=".{8,}" title="최소 8글자
이상">
  <input type="submit">
</form>
```

### 설명 : **pattern=".{8,}"** 의미

문자열의 길이가 8자 이상인지 검사.

패턴 구성:

- **.** (마침표) - 어떤 문자 하나를 의미.
- **{min,max}** - 최소길이와 최대길이 지정. (예) "{3,5}"는 3자 이상 5자 이하 의미

패턴 일치 예제:

- password123

패턴 불일치 예제:

- pw123

패턴 활용:

- 비밀번호나 다른 보안 관련 정보 검증.

## pattern 예제 - 패스워드 길이와 (숫자/소문자/대문자) 조합까지 체크

```
<form action="/action_page.jsp">
  <label for="pwd">패스워드:</label>
  <input type="password" id="pwd" name="pwd" pattern="(?=.*\d)(?=.*[a-z])(?=.*[A-Z]).{8,}" title="최소 8글자 이상 (단, 숫자,소문자,대문자 각각 최소 1개씩 포함)">
  <input type="submit">
</form>
```

**설명 : pattern="(?=.\*\d)(?=.\*[a-z])(?=.\*[A-Z]).{8,}" 의미**

최소 8자 이상이며, 적어도 하나의 숫자, 소문자, 대문자가 포함하는지 체크.

패턴 구성:

- **(?=.\*\d)** : 숫자가 최소 1개 이상. (※ 세부설명: ① **(?=...)** : 전방 탐색(lookahead : 다음 문자열을 미리 확인) 의미. / ② **.\***는 임의의 문자열 의미. / ③ **\d**는 숫자 의미.)
- **(?=.\*[a-z])** : 소문자 최소 1개 이상 의미. (※ 세부설명: **[a-z]**는 소문자 나타내는 문자 범위)
- **(?=.\*[A-Z])** : 대문자 최소 1개 이상 의미. (※ 세부설명: **[A-Z]**는 대문자 나타내는 문자 범위)
- **.{8,}** : 최소 8자 이상 의미. (※ 세부설명: **.**(마침표)는 임의의 문자를 의미하며, **{8,}**는 이전 문자가 8번 이상 반복될 수 있음 의미.)

패턴 불일치 경우:

- 문자열이 8자 미만인 경우, 숫자, 소문자, 대문자 중 하나 이상이 빠져 있는 경우 등

## pattern 예제 - 이메일 유효성 체크

```
<form action="/action_page.jsp">
  <label for="email">이메일:</label>
  <input type="email" id="email" name="email" pattern="[a-z0-9._%+-]+@[a-z0-9.-]+\.[a-z]{2,}$">
  <input type="submit">
</form>
```

### 설명 : pattern="[a-z0-9.\_%+-]+@[a-z0-9.-]+\.[a-z]{2,}\$" 의미

이메일 주소의 로컬 파트 (= @기호 앞부분)가 알파벳 소문자, 숫자, 그리고 일부 기호로 이루어지며, 이메일 주소의 도메인 파트 (= @기호 뒷부분)가 알파벳 소문자, 숫자, 하이픈으로 이루어지며, 최상위 수준 도메인이 두 글자 이상의 알파벳 소문자로 끝나는지를 체크.

패턴 구성:

- **[a-z0-9.\_%+-]+**: 이메일 주소의 로컬 파트(local part: 로컬 파트는 이메일 주소에서 "@" 기호 앞에 있는 부분) 의미. 여기서는 알파벳 소문자(a-z), 숫자(0-9) 및 몇 가지 기호(.%+-)를 허용하며, 이러한 문자가 한 번 이상 나타날 수 있도록 "+" 기호가 사용됨.
- **@**: 이메일 주소에서 로컬 파트와 도메인 파트(domain part)를 구분하는 기호임.
- **[a-z0-9.-]+**: 이메일 주소의 도메인 파트 (Domain Part: 도메인 파트는 이메일 주소에서 "@" 기호 뒤에 오는 부분) 의미. 여기서는 알파벳 소문자(a-z), 숫자(0-9) 및 하이픈(-)을 허용하며, 이러한 문자가 한 번 이상 나타날 수 있도록 "+" 기호가 사용됨.
- **.**: 이 부분은 이메일 주소에서 도메인 파트의 최상위 수준 도메인(top-level domain)과 그 앞의 도메인 이름을 구분하는 마침표 기호임. 이 부분에서는 마침표가 직접 사용되지 않고 이스케이프 문자(\) 사용해 마침표를 문자 그대로 해석하도록 함.
- **[a-z]{2,}\$**: 이메일 주소의 최상위 수준 도메인 의미. "[a-z]{2,}"는 알파벳 소문자(a-z)로 이루어진 두 글자 이상의 문자열을 의미. 이 부분은 도메인 파트의 마지막 문자열이므로 "\$" 기호를 사용하여 문자열의 끝을 의미.

패턴 일치 예제:

- chunjae@chunjae.com

패턴 불일치 예제:

- chunjae@chunjae (∵ 최상위 도메인 누락)

패턴 활용:

- 사용자 이메일 받는 애플리케이션에서 이메일 주소 형식 맞는지 체크.

## pattern 예제 - 검색어 유효성 체크

```
<form action="/action_page.jsp">
  <label for="search">검색어 입력:</label>
  <input type="search" id="search" name="search" pattern="^[^\x22]+"
  title="Invalid input">
  <input type="submit">
</form>
```

### 설명 : pattern="^[^\x22]+" 의미

따옴표(= 작은따옴표 + 큰따옴표) 아닌 모든 문자 입력 허용.

패턴 구성:

- **^**: 대괄호 밖에서 사용 시, 문자열 시작 부분에서 시작됨 의미. 대괄호 안 처음에 사용 시, 부정(negation) 의미.
- **[^\x22]+**: 따옴표 제외한 모든 문자열을 의미. 여기서 "**^**" 기호는 부정(negation)을 나타내며, 이 패턴에 포함된 문자를 제외한 모든 문자를 의미. "**\x22**"는 따옴표를 나타내는

이스케이프 문자 의미. "+" 는 이 패턴에 포함된 문자가 한 번 이상 나타날 수 있도록 함 의미.

패턴 일치 예제:

- 천재닷컴 Chunjae.com

패턴 불일치 예제:

- 천재닷컴 'Chunjae.com'

패턴 활용:

- 입력된 데이터에 대해 따옴표 인용 부호 방지.
- 입력된 데이터에 따옴표 포함 안 되도록 강제.

## pattern 예제 - URL 유효성 체크

```
<form action="/action_page.jsp">
  <label for="site">사이트주소:</label>
  <input type="url" id="site" name="site" pattern="https?: /\.+" title="http:// 포함
해 입력">
  <input type="submit">
</form>
```

### 설명 : pattern="https?: /\.+" 의미

"http://" 또는 "https://"로 시작하는 URL 주소 허용하며, 이어지는 도메인 이름과 경로는 어떤 문자열이든 가능.

패턴 구성:

- http: URL 주소가 "http" 프로토콜 사용하는 경우 의미.

- **s?**: "s" 문자가 0회 또는 1회 포함되는 경우 의미. 즉, URL 주소가 "https" 프로토콜을 사용하는 경우에도 이 패턴에 부합.
- **://**: URL 주소의 프로토콜명과 나머지 URL 주소를 구분하는 구분자.
- **.+**: URL 주소의 도메인 이름과 경로 의미. (여기서 "."은 도메인 이름에서 서브 도메인을 구분하는 구분자로 사용되며, "+" 기호는 이 패턴에 포함된 문자가 한 번 이상 나타날 수 있도록 함. 따라서 이 패턴은 "http://" 또는 "https://" 다음에 나오는 모든 문자열을 URL 주소의 도메인 이름과 경로로 인식.)

#### 패턴 일치 예제:

- http://www.chunjae.com
- https://chunjae.com/lec/html5

#### 패턴 불일치 예제:

- ftp://www.chunjae.com (:: HTTP 또는 HTTPS 프로토콜 아님)
- www.chunjae.com (:: HTTP 또는 HTTPS 프로토콜 누락)
- chunjae.com (:: HTTP 또는 HTTPS 프로토콜 누락)

#### 패턴 활용:

- 입력 데이터가 URL 주소 형식에 부합하는지 체크에 사용