**정규식 RegEx**

특정 문자 검색하기

|  |  |
| --- | --- |
| 정규식 | Ben |
| 문자열 | Hello, my name is Ben. Please Ben visit ben |
| 설명 | 보통 정규식은 처음으로 match일치하는 텍스트를 반환  두개의 일치가 만들어진걸까? 대다수 정규식 표현식 구현에서는 일치하는 목록을 모두 얻을수 있다(배열등)  일치된 결과물을 모두 담을 배열을 반환하고자 할 때 **g(global:전역)** 플래그를 사용한다. |

대소문자 다루기

|  |  |
| --- | --- |
| 정규식 | Ben |
| 문자열 | Hello, my name is Ben. Please Ben visit ben |
| 설명 | 자바스크립트 사용자들은 **i 플래그**를 사용해 대소문자 구별을 무시하고 검색할수있다. |

모든 문자 찾기

팁: .는 줄바꿈문자를 제외한 모든 문자와 일치한다

|  |  |
| --- | --- |
| 정규식 | sales. |
| 문자열 | sales1.xls  sales1.xls  orders3.xls  sales2.xls  sales3.xls  apac1.xls  europe2.xls  na1.xls  na2.xls  sa1.xls |
| 설명 | 정규표현식서는 특별한 문자들(혹은 문자 집합) 을 써 무엇을 검색할지 결정한다.  **마침표(.)** 문자는 아무 문자 하나와 일치한다  **여기서 사용한 sales. 라는 정규식은 sales로 시작하고 sales 바로뒤에 아무 문자가 하나 더붙는 파일명을 모두 찾는다. 파일 9개중에 3개가 이패턴과 일치한다.**  **마침표(.)는 어떠한 문자나 알파벳,숫자,심지어 문장 부호로 쓰인 마친표(.) 자체와도 일치한다** |

|  |  |
| --- | --- |
| 정규식 | .a. |
| 문자열 | sales1.xls  orders3.xls  sales2.xls  sales3.xls  apac1.xls  europe2.xls  na1.xls  na2.xls  sa1.xls |
| 설명 | **마침표(.) 여러 개 동시사용.**  **예를들어 .. 처럼 연속해 사용 하면 어떤 문자든 붙어 있는 문자 두개와 일치한다 아니면 서로 다른위치에 사용할수있다.** |

|  |  |
| --- | --- |
| 정규식 | .a.. |
| 문자열 | sales1.xls  orders3.xls  sales2.xls  sales3.xls  apac1.xls  europe2.xls  na1.xls  na2.xls  sa1.xls |
| 설명 | .a.. 도 .a.와 다르지 않다 우리는 마지막에 마침표가 붙는 글자를 일치시키고 싶었던 것인데 문자가 하나더 붙은 텍스트를 찾았다 . 어떻게해야 마침표를 의미하는 마침표(.)문자만 찾을수있을까??  아래서 확인하자 특수문자 찾기 ↓ |

특수문자 찾기

마침표(.)는 정규 표현식에서 특별한 의미가 있다. 따라서 여러분이 특수문자 마침표(.)를 찾으려한다면,

정규식 표현에게 특별한 의미의 마침표(.) 아니라 진짜 마침표(.) 를 찾고싶다고 알려줘야한다.

앞에 역슬래시(\)문자(메타문자)를 붙이면된다. 이것의 명칭은 secape 이스케이프 라고한다

|  |  |
| --- | --- |
| 정규식 | .a.\.xls |
| 문자열 | sales1.xls  orders3.xls  sales2.xls  sales3.xls  apac1.xls  europe2.xls  na1.xls  na2.xls  sa1.xls |
| 설명 | 역슬래시(\)를 메타문자이 며  (ecape 이스케이프 라고한다)  특수문자 역슬래시(\)를 찾고싶으면 [\\를](file:///\\를) 하면된다 |

문자 집합으로 찾기

|  |  |
| --- | --- |
| 정규식 | [ns]a.\.xls |
| 문자열 | sales1.xls  orders3.xls  sales2.xls  sales3.xls  apac1.xls  europe2.xls  na1.xls  na2.xls  sa1.xls  usa1.xls  ca1.xls |
| 설명 | 정규표현식에는 메타 문자인 대괄호([])를 사용해 문자 집합을 표현한다.  집합에 속한 문자 가운데 하나가 일치한다. 집합에 속한 문자가 모두 일치할필요는 없다. |

|  |  |
| --- | --- |
| 정규식 | [Rr]eg[Ee]x |
| 문자열 | The phrase “regular expression” is often  Abbreviated as RegEx or regex |
| 설명 | 여기서 사용한 패턴에는 문자 집합이 두개있다 [Rr]은 R이나 r 이고 [Ee]는 E나 e이다.  이런식으로 RegEx와 regex둘다 일치시킬수 있다 하지만 REGEX와는 일치하지 않는다 |

문자 집합 범위 사용하기

|  |  |
| --- | --- |
| 정규식 | [ns]a[0123456789]\.xls |
| 문자열 | sales1.xls  orders3.xls  sales2.xls  sales3.xls  apac1.xls  europe2.xls  sam.xls  na1.xls  na2.xls  sa1.xls  ca1.xls |
| 설명 | 문자 집합[ ] 그집합 구성원중에 한문자라도 일치해야 한다(AND 가 아닌 OR이다) |

정규식 표현식을 사용할 때 여러분은 0~9 , A~Z 같이 문자들의 범위 지정하는일이 자주있을것이다.

단순하게 만들 때 정규표현식에서는 특별한 메타 문자인 하이픈(-)을 제공한다.

다음은 위와 같은 예제이지만 패턴에 범위를 지정하였다.

|  |  |
| --- | --- |
| 정규식 | [ns]a[0-9]\.xls |
| 문자열 | sales1.xls  orders3.xls  sales2.xls  sales3.xls  apac1.xls  europe2.xls  sam.xls  na1.xls  na2.xls  sa1.xls  ca1.xls |
| 설명 | 정규표현식에서는 특별한 메타 문자인 하이픈(-)을 제공한다.  다음은 위와 같은 예제이지만 패턴에 범위를 지정하였다.  [0123456789] = [0-9] |

범위 지정 항목

1. A-Z는 A부터Z사이에있는 모든 대문자와 일치
2. a-z는 a부터z사이에있는 모든 소문자와 일치
3. A-F는 A부터F사이에 있는 대문자와 일치
4. A-z는 아스키(ASCII)문자 A와 아스키문자 z 사이에 있는 모든 문자와 일치한다. 아스키 A-z는 [와 ^캐럿 같은 문자도 포함된다.(주의)

|  |  |
| --- | --- |
| 정규식 | #[0-9A-Fa-f] [0-9A-Fa-f] [0-9A-Fa-f] [0-9A-Fa-f] [0-9A-Fa-f] [0-9A-Fa-f] |
| 문자열 | <BODY BGCOLOR=”#336633” TEXT=”#FFFFFF”  MARGINWIDTH=”0” MARGINHEIGHT=”0” TOPMARGIN=”0”  LEFTMARGIN=”0”> |
| 설명 |  |

제외하고 찾기 ^

|  |  |
| --- | --- |
| 정규식 | [ns]a[^0-9]\.xls |
| 문자열 | sales1.xls  orders3.xls  sales2.xls  sales3.xls  apac1.xls  europe2.xls  sam.xls  na1.xls  na2.xls  sa1.xls  ca1.xls |
| 설명 | 모두 나열하기보다는 캐럿(^) 문자를 써 제외할 문자 집합을 지정한다  [^0-9] 는 모든 숫자를 제외하는 것을 찾는다. |

문자클래스

|  |  |
| --- | --- |
| 문자클래스 | 기능 |
| . | 하나의 문자를 지칭한다 |
| [ | 집합의 시작 |
| ] | 집합의 끝 |
| ^ | 뒤에문자 제외 |
| { | 구간 시작 |
| } | 구간 끝 |
| + | 하나이상 |
| \* | 없거나 하나이상 |
| ? | 없거나 특정 문자만 ?앞에 a? |

숫자메타문자

|  |  |
| --- | --- |
| 메타문자 | 기능 |
| \d | 숫자 하나([0-9] 와같다) |
| \D | 숫자를 제외한 문자 하나(^[0-9] 와같다); |
|  |  |
|  |  |
|  |  |
|  |  |

영숫자 메타문자

|  |  |
| --- | --- |
| 메타문자 | 기능 |
| \w | 대소문자와 밑줄을 포함하는 모든 영문자 ( [a-zA-Z0-9\_] 와같다 ) |
| \W | 대소문자와 밑줄을 포함하지않는 모든 영문자 ( [^a-zA-Z0-9\_] 와같다 ) |
|  |  |
|  |  |
|  |  |
|  |  |

공백 메타문자

|  |  |
| --- | --- |
| 메타문자 | 기능 |
| \s | 모든 공백 문자( [\f\n\r\t\v] 와같다) \b는 포함되지않는다 |
| \S | 모든 공백 문자제외문자( [^\f\n\r\t\v] 와같다) \b는 포함되지않는다 |
|  |  |
|  |  |
|  |  |
|  |  |

메타문자

|  |  |
| --- | --- |
| 메타문자 | 기능 |
| [\b] | 역스페이스 |
| \f | 페이지 넘김(form feed) |
| \n | 줄바꿈 |
| \r | 캐리지 리턴 |
| \t | 탭 |
| \v | 수직탭 |

메타문자 사용하기

|  |  |
| --- | --- |
| 정규식 | myArray[0] |
| 문자열 | Var myArray = new Array();  …  If (myArray[0] ==0){  …  } |
| 설명 | 위에서 사용한 [0]은 정규식에서의 집합을 의미한다  [ ] 문자는 정규식에서 집합을 정의하는 문자이기 때문이다.  myArray0 을 찾으면 찾을것이다. |

|  |  |
| --- | --- |
| 정규식 | myArray\[0\] |
| 문자열 | Var myArray = new Array();  …  If (myArray[0] ==0){  …  } |
| 설명 | 이스케이프 처리하여 [문자를 정말 문자로 보았다.  하지만 myArray[10] 은 찾지못한다 당연히 하나의 문자만 숫자로했으니. |

|  |  |
| --- | --- |
| 정규식 | myArray\[\d\] |
| 문자열 | Var myArray = new Array();  …  If (myArray[0] ==0){  …  } |
| 설명 | 이스케이프 처리하여 [문자를 정말 문자로 보았다.  하지만 myArray[10] 은 찾지못한다 당연히 하나의 문자만 숫자로했으니.  숫자메타문자를 썼다 |

|  |  |
| --- | --- |
| 정규식 | \\ |
| 문자열 | \home\ben\sales\ |
| 설명 | 이스케이프 처리 |

공백 및 엔터값처리

|  |  |
| --- | --- |
| 정규식 | \r\n\r\n |
| 문자열 | “101”,”Ben”,”Forta”  “102”,”Jim”,”James”  “103”,”Roberta”,”Robertson”  “104”,”Bob”,”Bobson” |
| 설명 | 엔터 두번 뛰어져있는 부분이 선택된다  \r\n은 줄바꿈과 캐리지 리턴의 조합과 일치한다 따라서 \r\n\r\n을 검색하면 줄끝이 연속해서 두번나오는 부분을 찾게된다 즉 레코드상에 빈공간을 찾는것이다 |

|  |  |
| --- | --- |
| 정규식 | \w\d\w\d\w\d |
| 문자열 | 11213  A1C2E3  48075  48237  M1B4F2  90046  H1H2H2 |
| 설명 | 여기서 \w와 \d를 조합한 패턴사용했는데 캐나다 우편번호만을 찾아냈다 |

16진수나 8진수 표현하기

|  |  |
| --- | --- |
| 16진수값 | 앞에 \x를 붙여 표시한다. 즉 \x0A(아스키문자10)은 줄바꿈 문자가되며 \n과 기능이같다. |
| 8 진수값 | 두자리나 세자리 정수 앞에 \0을 붙여 표시한다 즉 \011(아스키 문자9)은 탭문자이며, \t 와 기능이같다 |

컨트롤 문자

예를들어 \cZ는 Ctrl-Z와 일치한다. 실제로는 거의 안쓰인다.

포직스 문자 클래스 사용하기

메타문자와 여러문자 집합을 줄여 쓰는 방법에 대한 설명에는 포직스 POSIX

일부분 지원을안하는 엔진도있다. 자바스크립트는 지원안함.

포직스 문자 클래스

|  |  |
| --- | --- |
| 메타문자 | 기능 |
| [:alnum:] | 모든영숫자([a-zA-Z0-9] 와같다) |
| [:alpha:] | 모든 영문자 [a-zA-Z]와 같다. |
| [:blank:] | 빈칸(space)이나 탭문자 ([\t ]와같다) |
| [:cntrl:] | 아스키 제어문자(아스키 0번부터 31, 127번) |
| [:digit:] | 모든 한자리 숫자[0-9] 와같다 |
| [:graph:] | [:print:] 와 동일하나 빈칸(space)은 제외 |
| [:lower:] | 모든 소문자([a-z]와같다); |
| [:print:] | 출력 가능한 모든문자 |
| [:punct:] | [:alnum:]이나[:cntrl:] 가 포함 되지않는 모든문자 |
| [:space:] | 빈칸을 포함한 모든 공백 문자 [\f\n\r\t\v] 와같다 |
| [:upper:] | 모든 대문자[A-Z]와같다 |
| [:xdigit:] | 모든 16진수 숫자 [a-fa-F0-9]와같다 |

|  |  |
| --- | --- |
| 정규식 | #[[:xdigit:]] [[:xdigit:]] [[:xdigit:]] [[:xdigit:]] [[:xdigit:]] [[:xdigit:]] |
| 문자열 | <BODY BGCOLOR=”#336633” TEXT=”#FFFFFF”  MARGINWIDTH=”0” MARGINHEIGHT=”0” TOPMARGIN=”0”  LEFTMARGIN=”0”> |
| 설명 | [0-9A-Fa-f] 를 여섯 번 반복. |

반복찾기

|  |  |
| --- | --- |
| 정규식 | \w@\w\.\w |
| 문자열 | text@w.text |
| 설명 | \w 문자는 모든 영숫자 문자와 일치, 더불어 밑줄도 함께 찾는데.. 이메일을 찾을수 있다  하지만!!!!!!!!!! 앞뒤로 한글자씩밖에 들어는것만 가능하다 그러무로 쓸모없는.. |

문자 하나이상 찾기

|  |  |
| --- | --- |
| + | 문자 뒤에 더하기 (+) 문자를 붙이면된다  하나이상일 때 일치한다(최소한 하나와 일치하고, 없을 때는 일치하지 않는다. |
|  |  |
|  |  |
|  |  |
| [0-9+] 가 정확한게 아니라 [0-9]+ 가 정확한거다 | |

|  |  |
| --- | --- |
| 정규식 | \w+@\w+\.\w+ |
| 문자열 | Send personal email to [ben@forta.com](mailto:ben@forta.com). For questions  About a book use [support@forta.com](mailto:support@forta.com). Feel free to send  Unsolicited email to [spam@forta.com](mailto:spam@forta.com) (wouldn’t it be  Nice if it were that simple, huh?). |
| 설명 | 더하기(+) 메타문자 : 문자 그대로 더하기를 찾으려면 \+로 찾으면된다 |

|  |  |
| --- | --- |
| 정규식 | [\w.]+@ [\w.]+\.\w+ |
| 문자열 | Send personal email to [ben@forta.com](mailto:ben@forta.com). For questions  About a book use good.[support@forta.com](mailto:support@forta.com). Feel free to send  Unsolicited email to [spam@forta.com](mailto:spam@forta.com) (wouldn’t it be  Nice if it were that simple, huh?). |
| 설명 |  |

문자가 없는 경우나 하나 이상 연속하는 문자찾기

문자 없을수도 있고 있을수도있는.. 찾기

|  |  |
| --- | --- |
| \* | 메타 문자인 별표(\*)를 사용하면된다  집합이 없는경우 또는 하나 이상 연속하는 경우 일치한다. |

|  |  |
| --- | --- |
| 정규식 | [\w.]+@ [\w.]+\.\w+ |
| 문자열 | Hello [.ben@forta.com](mailto:.ben@forta.com) is my email address |
| 설명 | 제대로 검색은되지만 보통은 이메일 맨처음에 . 를 쓰지를 않는다는점이다. |

|  |  |
| --- | --- |
| 정규식 | \w+[\w.]\*@ [\w.]+\.\w+ |
| 문자열 | Hello [.ben@forta.com](mailto:.ben@forta.com) is my email address |
| 설명 | 이것이 찾고자 하는 텍스트다. |

문자가 없거나 하나인 문자 찾기

|  |  |
| --- | --- |
| ? | 물음표(?) 는 별표(\*)처럼 물음표(?)는 문자가 있는 경우 일치하고 문자가 없어도 일치하지만 별표(\*)와 달리 문자나 집합이 없거나 하나만 있는 경우 일치하며 하나이상은 일치하지 않는다. 즉, 물음표(?)는 문자 묶음 안에서 있는지 없는지 확실하지 않는 특정한 문자를 하나만 찾을 때 유용하다  **없거나 존재한다면 ? 앞에 있는 문자가 나와야한다**  https?:// 는 ? 는 자기문자 앞에 문자가 없거나 그문자가 하나만 있는 경우 일치한다.!! 여기서는 s다 |

|  |  |
| --- | --- |
| 정규식 | http://[\w./]+ |
| 문자열 | The URL is <http://www.forta.com/> , to connect  Securely use <https://www.forta.com/> instead. |
| 설명 |  |

|  |  |
| --- | --- |
| 정규식 | https?://[\w./]+ |
| 문자열 | The URL is <http://www.forta.com/> , to connect  Securely use <https://www.forta.com/> instead. |
| 설명 | https?:// 는 ? 는 자기문자 앞에 문자가 없거나 그문자가 하나만 있는 경우 일치한다.!! 여기서는 s다  https?://는 <http://나> <https://와> 일치하지만 그외에는 일치하지 않는다. |

공백 및 엔터값처리

|  |  |
| --- | --- |
| 정규식 | [\r]?\n[\r]?\n |
| 문자열 | “101”,”Ben”,”Forta”  “102”,”Jim”,”James”  “103”,”Roberta”,”Robertson”  “104”,”Bob”,”Bobson” |
| 설명 | [\r]?\n은 \r 이 있을경우에는 \r과 일치하고 \n과 반드시 일치한다. |

구간 지정하기 {}

기존 반복찾기로 찾기 힘든것들

1. 더하기(+)와 별표(\*)는 일치하는 문자 수에 제한이 없다. 문자가 최대 몇 개까지 일치하는지 정할수 없다.
2. 더하기(+),별표(\*),물음표(?)가 일치하는 문자 수의 최소값은 0이나 1이다 일치하는 문자수의 최소값을 명시적으로 정의할수 없다
3. 정확히 원하는 만큼만 일치하도록 문자 수를 정의할수 없다.

위문제를 해결하기위해 구간을 사용한다 구간은 중괄호 {}안에 표시한다.

정확한 구간찾기

|  |  |
| --- | --- |
| {숫자} | 구간 설정  {3}은 바로 앞에 있는 문자나 문자 집합이 세번 연속해서 일치하는지 확인한다, 만약 요소가 두개만 잇다면 패턴이 일치하지 않는다. |
| {숫자,숫자} | 구간설정 최소 최대값 정할수있다  {3,10}은 바로 앞에 있는 문자나 문자 집합이 최소3번부터 10번까지 일치시킨다는거다.  구간은 0부터 시작하기도한다. {0,3} 요소가 없는경우나 최대3번 일치함을 의미한다 |
| {숫자,} | 최소 구간찾기  {3,} 은 최소한 요소가 세번 일치함을 의미한다 최대값은 주어지지 않아서 최대는 없다. 무한  더하기(+)는 {1,}와 같다. |

|  |  |
| --- | --- |
| 정규식 | #[[:xdigit:]]{6} |
| 문자열 | <BODY BGCOLOR=”#336633” TEXT=”#FFFFFF”  MARGINWIDTH=”0” MARGINHEIGHT=”0” TOPMARGIN=”0”  LEFTMARGIN=”0”> |
| 설명 | [0-9A-Fa-f] 를 여섯 번 반복.  구간 설정 |

|  |  |
| --- | --- |
| 정규식 | \d{1,2}[-\/]\d{1,2}[-\/]\d{2,4} |
| 문자열 | 4/8/03  10-6-2004  2/2/2  01-01-01 |
| 설명 | 이건 54/67/9999 처럼 값이 틀려도 검사를 통과한다 따라서 사용할수없는것이다. |

|  |  |
| --- | --- |
| 정규식 | \d+: \$\d{3,}\.\d{2} |
| 문자열 | 1001: $496.80  1002: $1496.80  1003: $46.80  1004: $613.80  1005: $6.80  1006: $414.80  1007: $25.80 |
| 설명 | 이건 54/67/9999 처럼 값이 틀려도 검사를 통과한다 따라서 사용할수없는것이다. |

과하게 일치하는 상황 방지하기

범위선택

|  |  |
| --- | --- |
| .\* | 처음매칭된거와 마지막 매칭된거의 사이를 모두 선택한다. |

|  |  |
| --- | --- |
| 정규식 | <[Bb]>.\*</[Bb]> |
| 문자열 | This offer is not available to customers  Living in <B>AK</B> and <B>HI</B> |
| 설명 | .\*는 처음나온 <B> 부터 마지막에 나온 </B> 사이에 있는 모든 텍스트와 일치하기 때문에, AK</B> and <B>HI가 일치한 것이다. 우리가 원하는 텍스트를 포함하긴하지만 찾으려 하지 않는 텍스트도 포함했다. |

바로 별표(\*) 와 더하기(+)같은 메타 문자가 탐욕적(greedy)이기때문인데 이는 가능한 한가장 큰 덩어리를 찾으려 한다는 뜻이다.

이런 메타 문자는 찾으려는 텍스트를 앞에서부터 찾는게 아니라, 텍스트 마지막에서 시작해 거꾸로 찾는다. 의도적으로 수량자(quantifier)를 탐욕적으로 설계했기 때문이다. 하지만 만약 우리가 탐욕적 일치를 원하지 않는다면 어떻게 해야할까?

탐욕적 수량자를 게으른(lazy)수량자로 바꿔 이문제를 해결한다 ‘게으른’ 이라고 부르는 이유는 문자가 최소 일치하기 때문이다 게으른 수량자는 기존 수량자 뒤에 물음표(?)를 붙여서 표현한다.

|  |  |
| --- | --- |
| 탐욕적 수량자 | 게으른 수량자 |
| \* | \*? |
| + | +? |
| {n,} | {n,}? |

|  |  |
| --- | --- |
| 정규식 | <[Bb]>.\*?</[Bb]> |
| 문자열 | This offer is not available to customers  Living in <B>AK</B> and <B>HI</B> |
| 설명 | .\*? 게으른 수량자이다 먼저 AK만 일치시켰고 뒤있어 <B>HI</B>를 찾아 두부분을 따로 일치시켰다 |

정리

정규 표현 패턴이 지닌 진짜 능력은 반복 찾기를 할 때 뚜렸하게 나타난다 이장에서는 더하기(+) 하나이상일치

와 별표(\*) 없거나 하나이상있는경우 일치 , 물음표(?) 없거나 하나인경우 일치 로 반복 찾기를 수행하는 법을 소개했다 검색조건을 더 구체적으로 지정하고자 구간으로 정확한 반복 횟수나 최소값 최대값을 정한다 방금말한 수량자들은 탐욕적이여서 너무 넓은 범위와 일치할수있다 이를 방지하고자 게으른 수량자를 사용한다

위치찾기(position matching)

|  |  |
| --- | --- |
| 정규식 | cat |
| 문자열 | The cat scattered his food all over the room. |
| 설명 | cat 패턴은 cat이 있는 부분과 모두 일치한다.cat을 dog로 취환할려고 하는데 scattered 에있는 cat이 검색이 된다면 말도 안되는 소리다. |

단어 경계 지정하기

|  |  |
| --- | --- |
| \b | 단어경계다 단어 경계라는 이름에서 유추할수있듯 \b는 단어의 시작이나 마지막을 일치시킬 때 사용한다. |

|  |  |
| --- | --- |
| 정규식 | \bcat\b |
| 문자열 | The cat scattered his food all over the room. |
| 설명 | 그럼 정확히 \b는 무엇과 일치하는것일까? 정규표현식 엔진은 영어를 비롯해 어떤언어도 이해하지 못한다 따라서 무엇이 단어인지도 알지못한다 \b 는 일반적으로 단어의 일부로 사용하는 문자(영숫자 문자, 밑줄,\w와 일치하는 문자) |

특정한 문자 형태와 일치시키기