




UNIVERSIDAD DE SEVILLA
Escuela Técnica Superior
de Ingeniería Informática
Departamento de Lenguajes
y Sistemas Informáticos

Validación con Expresiones Regulares

*Grupo de Ingeniería del Software y Bases de Datos
Departamento de Lenguajes y Sistemas Informáticos
Universidad de Sevilla
febrero 2017*

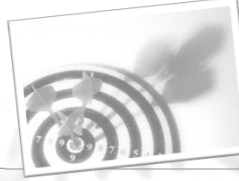
© Departamento de Lenguajes y Sistemas Informáticos, 2017



UNIVERSIDAD DE SEVILLA
Escuela Técnica Superior
de Ingeniería Informática
Departamento de Lenguajes
y Sistemas Informáticos

Validación con Expresiones Regulares

- **Objetivos del tema**
 - Entender el concepto de **expresión regular**.
 - Entender la necesidad de las expresiones regulares en la **validación** de formularios.
 - Ser capaz de **utilizar** las expresiones regulares para validar formularios en **cliente** y en **servidor**.




febrero 2017

Introducción a la Ingeniería del Software y a los Sistemas de Información

1

© Departamento de Lenguajes y Sistemas Informáticos, 2017




UNIVERSIDAD DE SEVILLA
Escuela Técnica Superior
de Ingeniería Informática
Departamento de Lenguajes
y Sistemas Informáticos

Validación con Expresiones Regulares

- ¿Qué es una expresión regular?
 - Una expresión regular define un **patrón** que puede **buscarse** o **reemplazarse** dentro de cadenas de texto.
 - Ejemplo: `/abd/i` (i indica que no distinga entre mayúsculas y minúsculas)
 - "La asignatura de **ABD**"
 - "**Abd**ominales"
 - "Scully, esto es un caso claro de **abducción**"*

*La agente especial Dana Scully, junto con su compañero el agente Fox Mulder, es uno de los personajes principales de la serie *Expediente X* emitida en la década de 1990.




1. Introducción
2. HTML 5 y PHP
3. Expresiones regulares en JS
4. Definición de patrones
5. Ejemplos

febrero 2017

Introducción a la Ingeniería del Software y a los Sistemas de Información

2



UNIVERSIDAD DE SEVILLA
Escuela Técnica Superior
de Ingeniería Informática
Departamento de Lenguajes
y Sistemas Informáticos

Validación con Expresiones Regulares


- ¿Para qué sirve una expresión regular?
 - Las principales utilidades de las expresiones regulares en el desarrollo de aplicaciones web son:
 - Validación** de datos de formularios, tanto en cliente como en servidor.
 - Ejemplo: validar direcciones de correo electrónico, números de teléfono, códigos postales, etc.
 - Manipulación de texto mediante el **reemplazo** de expresiones.
 - Ejemplo: asegurar que el SQL que generamos es correcto (') → (").

1. Introducción
2. HTML 5 y PHP
3. Expresiones regulares en JS
4. Definición de patrones
5. Ejemplos

febrero 2017

Introducción a la Ingeniería del Software y a los Sistemas de Información

3



UNIVERSIDAD DE SEVILLA
Escuela Técnica Superior
de Ingeniería Informática
Departamento de Lenguajes
y Sistemas Informáticos

- 1. Introducción
- 2. HTML 5 y PHP
- 3. Expresiones regulares en JS
- 4. Definición de patrones
- 5. Ejemplos

Validación con Expresiones Regulares


- Expresiones regulares en HTML 5
 - En HTML 5:

```
<input type="text"
      pattern = "[ 0-9]{5}"
      title="Código postal"
/>
```
 - Este patrón es una expresión regular que indica una cadena de 5 dígitos.
 - La mayor parte de los nuevos controles de HTML 5 usan de manera implícita patrones predefinidos, por ejemplo, para comprobar un correo electrónico, una URL bien construida, etc.

febrero 2017

Introducción a la Ingeniería del Software y a los Sistemas de Información

4



UNIVERSIDAD DE SEVILLA
Escuela Técnica Superior
de Ingeniería Informática
Departamento de Lenguajes
y Sistemas Informáticos

- 1. Introducción
- 2. HTML 5 y PHP
- 3. Expresiones regulares en JS
- 4. Definición de patrones
- 5. Ejemplos

Validación con Expresiones Regulares

- Expresiones regulares en PHP*
 - `preg_match()`:

```
if ( preg_match("/[ 0-9]{5}/", "41012") )
    echo "Es un código postal.";
else
    echo "No es un código postal.";
```
 - También puede utilizarse para recuperar partes de la cadena que satisfagan la expresión regular:


```
if ( preg_match("/[ 0-9]{5}/", "CP:41012", $cp) )
    echo "El código postal es $cp[ 0].";
else
    echo "No se incluye ningún código postal.";
```

* Puedes encontrar más información relativa al uso avanzado de estas funciones en php.net y w3schools.

febrero 2017

Introducción a la Ingeniería del Software y a los Sistemas de Información

5



Escuela Técnica Superior
de Ingeniería Informática
Departamento de Lenguajes
y Sistemas Informáticos

- 1. Introducción
- 2. HTML 5 y PHP
- 3. Expresiones regulares en JS
- 4. Definición de patrones
- 5. Ejemplos

Validación con Expresiones Regulares

- Expresiones regulares en PHP*
 - `filter_var()`:


```
if ( filter_var( "cadena", FILTER_VALIDATE_EMAIL ) )
    echo "La cadena contiene un email válido.";
else
    echo "La cadena no contiene un email válido.";
```
 - Alguno de los filtros predefinidos más usados:

| Filtro | Descripción |
|-------------------------------------|---|
| <code>FILTER_VALIDATE_INT</code> | Si la cadena es un entero. |
| <code>FILTER_VALIDATE_FLOAT</code> | Si la cadena es un real. |
| <code>FILTER_VALIDATE_URL</code> | Si la cadena es una URL válida. |
| <code>FILTER_VALIDATE_EMAIL</code> | Si la cadena es un email válido. |
| <code>FILTER_VALIDATE_IP</code> | Si la cadena es una IP válida. |
| <code>FILTER_VALIDATE_REGEXP</code> | Si la cadena es una expresión regular válida. |

* Puedes encontrar más información relativa al uso avanzado de estas funciones en php.net y w3schools.

febrero 2017
Introducción a la Ingeniería del Software y a los Sistemas de Información
6




Escuela Técnica Superior
de Ingeniería Informática
Departamento de Lenguajes
y Sistemas Informáticos

- 1. Introducción
- 2. HTML 5 y PHP
- 3. Expresiones regulares en JS
- 4. Definición de patrones
- 5. Ejemplos

Validación con Expresiones Regulares



febrero 2017
Introducción a la Ingeniería del Software y a los Sistemas de Información
7



UNIVERSIDAD DE SEVILLA
Escuela Técnica Superior
de Ingeniería Informática
Departamento de Lenguajes
y Sistemas Informáticos

1. Introducción
2. HTML 5 y PHP
3. Expresiones regulares en JS
4. Definición de patrones
5. Ejemplos

Validación con Expresiones Regulares

- **Expresiones regulares en JavaScript**
 - Las expresiones regulares en JavaScript pueden declararse de dos formas distintas:


```
var er = /patrón/[ opciones]

// o bien...

var er = new RegExp( "patrón", [ "opciones" ] );
```

- Las opciones son opcionales y pueden ser:
 - *i*: no distingue entre mayúsculas y minúsculas.
 - *g*: búsqueda global en toda la cadena; no para en la primera coincidencia, las encuentra todas. Es útil para reemplazar texto.

febrero 2017
Introducción a la Ingeniería del Software y a los Sistemas de Información
8



UNIVERSIDAD DE SEVILLA
Escuela Técnica Superior
de Ingeniería Informática
Departamento de Lenguajes
y Sistemas Informáticos


1. Introducción
2. HTML 5 y PHP
3. Expresiones regulares en JS
4. Definición de patrones
5. Ejemplos

Validación con Expresiones Regulares

- **Uso de expresiones regulares en JavaScript**
 - Las expresiones regulares en JavaScript pueden utilizarse mediante métodos de las clases **String** o **RegExp**.
- **Métodos de la clase String***
 - **cadena.match(er)**: devuelve *null* si no encuentra el patrón definido por *er* dentro de la cadena; si lo encuentra, devuelve un array con información sobre la coincidencia (*input*, *index*, *lastIndex*).
 - **cadena.search(er)**: devuelve -1 si no encuentra el patrón definido por *er* dentro de la cadena; si lo encuentra, devuelve el índice en el que empieza la coincidencia.

*Otro método de *String* relacionado con expresiones regulares es *split*.

febrero 2017
Introducción a la Ingeniería del Software y a los Sistemas de Información
9



UNIVERSIDAD DE SEVILLA
Escuela Técnica Superior
de Ingeniería Informática
Departamento de Lenguajes
y Sistemas Informáticos

- 1. Introducción
- 2. HTML 5 y PHP
- 3. Expresiones regulares en JS
- 4. Definición de patrones
- 5. Ejemplos

Validación con Expresiones Regulares


- **Uso de expresiones regulares en JavaScript**
 - Las expresiones regulares en JavaScript pueden utilizarse mediante métodos de las clases **String** o **RegExp**.
- **Métodos de la clase String***
 - **cadena.replace(er, texto)**: devuelve una copia de la cadena en la que las coincidencias del patrón definido por *er* han sido reemplazadas por el *texto* indicado.
 - Si la opción *g* está activa, reemplaza todas las coincidencias, si no, sólo la primera.

*Otro método de *String* relacionado con expresiones regulares es *split*.

febrero 2017

Introducción a la Ingeniería del Software y a los Sistemas de Información

10



UNIVERSIDAD DE SEVILLA
Escuela Técnica Superior
de Ingeniería Informática
Departamento de Lenguajes
y Sistemas Informáticos

- 1. Introducción
- 2. HTML 5 y PHP
- 3. Expresiones regulares en JS
- 4. Definición de patrones
- 5. Ejemplos


Validación con Expresiones Regulares

- **Uso de expresiones regulares en JavaScript**
 - Las expresiones regulares en JavaScript pueden utilizarse mediante métodos de las clases **String** o **RegExp**.
- **Métodos de la clase RegExp**
 - **er.test(cadena)**: devuelve *true* si encuentra alguna coincidencia del patrón dentro de la cadena o *false* en caso contrario.
 - **er.exec(cadena)**: devuelve *null* si no encuentra el patrón dentro de la cadena; si lo encuentra, devuelve un array con información sobre la coincidencia (*input*, *index*, *lastIndex*).

febrero 2017

Introducción a la Ingeniería del Software y a los Sistemas de Información

11



UNIVERSIDAD DE SEVILLA
Escuela Técnica Superior
de Ingeniería Informática
Departamento de Lenguajes
y Sistemas Informáticos

1. Introducción
2. HTML 5 y PHP
3. Expresiones regulares en JS
4. Definición de patrones
5. Ejemplos


Validación con Expresiones Regulares

- **Equivalencias de métodos**
 - Los métodos de expresiones regulares de **String** y de **RegExp** son equivalentes:
 - `er.exec(cadena) == cadena.match(er)`
 - `er.test(cadena) == (cadena.search(er) != -1)`

febrero 2017

Introducción a la Ingeniería del Software y a los Sistemas de Información

12



UNIVERSIDAD DE SEVILLA
Escuela Técnica Superior
de Ingeniería Informática
Departamento de Lenguajes
y Sistemas Informáticos

1. Introducción
2. HTML 5 y PHP
3. Expresiones regulares en JS
4. Definición de patrones
5. Ejemplos


Validación con Expresiones Regulares

- **Caracteres y meta-caracteres**
 - Los patrones de las expresiones regulares pueden contener tanto caracteres como **meta-caracteres**.
 - Existen dos **meta-caracteres** especialmente relevantes:
 - `^` : que representa el comienzo de una cadena
 - `$` : que representan el final de una cadena

febrero 2017

Introducción a la Ingeniería del Software y a los Sistemas de Información

13



UNIVERSIDAD DE SEVILLA
Escuela Técnica Superior de Ingeniería Informática
Departamento de Lenguajes y Sistemas Informáticos


1. Introducción
2. HTML 5 y PHP
3. Expresiones regulares en JS
4. Definición de patrones
5. Ejemplos

Validación con Expresiones Regulares

- Caracteres y meta-caracteres
 - **^** y **\$** Permiten comprobar que una cadena **coincide exactamente** con el patrón de una expresión regular en lugar de comprobar solamente si lo contiene.
 - Para usar como **literales** aquellos caracteres que se usan en la sintaxis de los **meta-caracteres** es necesario precederlos de la barra ****.
 - Los meta-caracteres son:

`\ ^ $ * + ? { } . () | []`

febrero 2017
Introducción a la Ingeniería del Software y a los Sistemas de Información
14



UNIVERSIDAD DE SEVILLA
Escuela Técnica Superior de Ingeniería Informática
Departamento de Lenguajes y Sistemas Informáticos


1. Introducción
2. HTML 5 y PHP
3. Expresiones regulares en JS
4. Definición de patrones
5. Ejemplos

Validación con Expresiones Regulares

- Ejemplos de uso de meta-caracteres

| Sintaxis | Descripción |
|------------------------------------|--|
| ^E | <i>E</i> al comienzo de la cadena. |
| E\$ | <i>E</i> al final de la cadena. |
| E* | Repetición de <i>E</i> cero o más veces. |
| E+ | Repetición de <i>E</i> una o más veces. |
| E? | <i>E</i> cero o una vez. |
| E{n} | <i>n</i> repeticiones de <i>E</i> . |
| E{n,} | <i>n</i> o más repeticiones de <i>E</i> . |
| E{n,m} | Entre <i>n</i> y <i>m</i> repeticiones de <i>E</i> . |
| . | Cualquier carácter excepto "\n". |
| E₁ E₂ | E ₁ o E ₂ . |

febrero 2017
Introducción a la Ingeniería del Software y a los Sistemas de Información
15



Escuela Técnica Superior
de Ingeniería Informática
Departamento de Lenguajes
y Sistemas Informáticos

1. Introducción
2. HTML 5 y PHP
3. Expresiones regulares en JS
4. Definición de patrones
5. Ejemplos


Validación con Expresiones Regulares

- Ejemplos de uso de meta-caracteres

| Sintaxis | Descripción |
|---|---|
| <code>[...]</code> | Conjunto de caracteres, p.e. <code>[aeiou]</code> . |
| <code>[^...]</code> | Negado del conjunto de caracteres, p.e. <code>[^aeiou]</code> . |
| <code>[c₁-c₂]</code> | Rango de caracteres, p.e. <code>[a-z]</code> , <code>[A-Z]</code> , <code>[A-Za-z]</code> . |
| <code>[^c₁-c₂]</code> | Negado del rango de caracteres. |
| <code>(...)</code> | Agrupación de expresiones.* |
| <code>\b, \B</code> | Límite de palabra, negado de límite de palabra. Sirve para encontrar palabras enteras, no trozos de palabras. |
| <code>\d, \D</code> | Dígito, negado de dígito. Equivalen a <code>[0-9]</code> , <code>[^0-9]</code> . |
| <code>\s, \S</code> | Espacio (incluyendo <code>\t</code> , <code>\n</code> , etc.), negado de espacio. |
| <code>\w, \W</code> | Letra, dígito o subrayado, negado de lo mismo. Equivalen a <code>[A-Za-z0-9_]</code> , <code>[^A-Za-z0-9_]</code> . |

*La agrupación de expresiones permite usos avanzados mediante el objeto global *RegExp*.

febrero 2017
Introducción a la Ingeniería del Software y a los Sistemas de Información
16



Escuela Técnica Superior
de Ingeniería Informática
Departamento de Lenguajes
y Sistemas Informáticos

1. Introducción
2. HTML 5 y PHP
3. Expresiones regulares en JS
4. Definición de patrones
5. Ejemplos

Validación con Expresiones Regulares

- Ejemplos de expresiones regulares para validación
 - Número de teléfono nacional (sin espacios)
 - Ejemplo: 954556817
 - Expresión regular: `/^\d{9}$/`
 - Número de teléfono internacional
 - Ejemplo: (+34)954556817
 - Expresión regular: `/^\(\+\d{2,3}\)\d{9,10}$/`
 - Fecha con formato DD/MM/AAAA
 - Ejemplo: 09/01/2006
 - Expresión regular: `/^\d{2}/\d{2}/\d{4}$/`
 - Código postal
 - Ejemplo: 41012
 - Expresión regular: `/^\d{5}$/`

febrero 2017
Introducción a la Ingeniería del Software y a los Sistemas de Información
17

UNIVERSIDAD DE SEVILLA
Escuela Técnica Superior de Ingeniería Informática
Departamento de Lenguajes y Sistemas Informáticos

- 1. Introducción
- 2. HTML 5 y PHP
- 3. Expresiones regulares en JS
- 4. Definición de patrones
- 5. Ejemplos

Validación con Expresiones Regulares

- Ejemplos de expresiones regulares para sustitución
 - Sustitución de apóstrofes en sentencias SQL
 - Ejemplo: "c/ O'Donell, 23" → "c/ O''Donell, 23"
 - Expresión regular: `cadena = cadena.replace("'", "''");`

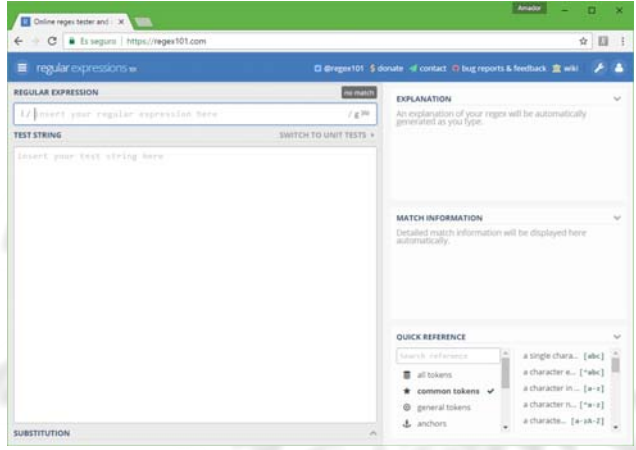
febrero 2017
Introducción a la Ingeniería del Software y a los Sistemas de Información
18

UNIVERSIDAD DE SEVILLA
Escuela Técnica Superior de Ingeniería Informática
Departamento de Lenguajes y Sistemas Informáticos


- 1. Introducción
- 2. HTML 5 y PHP
- 3. Expresiones regulares en JS
- 4. Definición de patrones
- 5. Ejemplos

Validación con Expresiones Regulares

- Para probar expresiones regulares online
 - <https://regex101.com/>
 - <http://regexr.com/>




febrero 2017
Introducción a la Ingeniería del Software y a los Sistemas de Información
19




UNIVERSIDAD DE SEVILLA
Escuela Técnica Superior
de Ingeniería Informática
Departamento de Lenguajes
y Sistemas Informáticos

Validación con Expresiones Regulares

- Comentarios, sugerencias, ...



Amador Durán Toro
amador@us.es
Departamento de Lenguajes y Sistemas Informáticos
E.T.S. Ingeniería Informática, Universidad de Sevilla, España



febrero 2017

Introducción a la Ingeniería del Software y a los Sistemas de Información

20

© Copyright de Amador Durán Toro, 2011.