

JS



PROFE VIDERMID

Programación en Java Script

(Parte 5)

Facilitador: Ing. Esp. Vidermid Sánchez



@vidermid



+584147106623



@ingenieriadigitalsc



+584147464801



¿Qué son?

También conocidas como Regex, son una especie de formulas que realizan búsquedas de coincidencias en cadenas de caracteres, utilizan unas reglas y conjuntos de operadores con ciertos dígitos para filtrar las coincidencias. Es decir, secuencia de caracteres que conforman un patrón de búsqueda.

¿Cuál es su uso?

RegExp ó expresión regular es un patrón de caracteres, el patrón se utiliza para realizar funciones de "búsqueda y reemplazo" de coincidencia de patrones en el texto.

En JavaScript, un objeto RegExp es un patrón con propiedades y métodos .

Documentación oficial:

https://developer.mozilla.org/es/docs/Web/JavaScript/Guide/Regular_expressions

Herramienta de testeo para Regex:

<https://regexr.com/>

Coincidencias Basicas:

- .** - Cualquier Caracter, excepto nueva linea
- \d** - Cualquier Digitos (0-9)
- \D** - No es un Dígito (0-9)
- \w** - Caracter de Palabra (a-z, A-Z, 0-9, _)
- \W** - No es un Caracter de Palabra.
- \s** - Espacios de cualquier tipo. (espacio, tab, nueva linea)
- \S** - No es un Espacio, Tab o nueva linea.

Conjuntos de Caracteres:

- []** - Caracteres dentro de los brackets
- [^]** - Caracteres que NO ESTAN dentro de los brackets

Modificadores:

- g** - Realiza una coincidencia global (encuentre todas las coincidencias en lugar de detenerse después de la primera)
- i** - Realiza coincidencias sin distinción de mayús. ó minús.
- m** - Realizar coincidencias multilínea

Limites:

- \b** - Limite de Palabra
- \B** - No es un Limite de Palabra
- ^** - Inicio de una cadena de texto
- \$** - Final de una cadena de texto

Grupos:

- ()** - Grupo
- |** - Uno u otro

Cuantificadores:

- *** - 0 o Más
- +** - 1 o Más
- ?** - 0 o Uno
- {3}** - Numero Exacto
- {3,4}** - Rango de Numeros (Minimo, Maximo)

Texto de pruebas para de regex

Vidermid Sanchez
414 160 81 01
vidermid_@hotmail.com
www.vidermid.com

Martha Zambrano
414-709-11-17
marthica@cantv.com
<https://www.lamartha.zambrano.com.ve>
<http://alejandro.com.mx>

Kassandra Lopez
424 310 92 17
kassi@gmail.com
mipaginakassi.net

<https://regexr.com/>

Ejemplos de regex

```
/[0-9]{3}(\ |-)[0-9]{3}(\ |-)[0-9]{2}(\ |-)[0-9]{2}/g
```

```
/[a-z\ a-z\ ]/
```

```
/[a-z\ a-z\ ]/g
```

```
/[a-z\ a-z\ ]/gi
```

```
/[a-z0-9!#$%&'*+/?^_`{|}~-]+(?:\.[a-z0-9!#$%&'*+/?^_`{|}~-]+)*@(?:[a-z0-9](?:[a-z0-9-]*[a-z0-9])?\.)+[a-z0-9](?:[a-z0-9-]*[a-z0-9])?/g
```

```
/[(http(s)?):\//(\www\.)?a-zA-Z0-9@:~#%_]{2,256}\.[a-z]{2,6}\b([-a-zA-Z0-9@:~#%_+~#?&//=]*)/ig
```

```
const valida=() => {  
  let xcor_cli=document.getElementById('cor_cli').value  
  alert(`tu correo es: ${xcor_cli}`);  
  let regExp=/[a-z0-9!#$%&'*+/?^_`{|}~-]+(?:\.[a-z0-9!#$%&'*+/?^_`{|}~-]+)*@(?:[a-z0-9](?:[a-z0-9-]*[a-z0-9])?\.)+[a-z0-9](?:[a-z0-9-]*[a-z0-9])?/g;  
  let valida=regExp.test(xcor_cli);  
  if(valida){  
    alert(`La dirección ${xcor_cli} es un correo Valido`);  
  }else{  
    alert(`La dirección ${xcor_cli} No es un correo Valido`);  
  }  
}
```


Estilos para personalizar validación

```
.invalida{  
  border: solid 1px red;  
  box-shadow: 0 0 10px red;  
}  
.error{  
  color:red;  
  display:block;  
}
```

```
const elemento_cor = document.querySelector("[name=cor_cli]");

const validaCompoVacio=(evento)=>{
  const Campo=evento.target;
  const valorCampo=evento.target.value;

  if(valorCampo.length===0){
    Campo.classList.add("invalida");
    Campo.nextElementSibling.classList.add("error");
    Campo.nextElementSibling.innerText=`${Campo.name} no puede estar vacio`;
  }else{
    Campo.classList.remove("invalida");
    Campo.nextElementSibling.classList.remove("error");
    Campo.nextElementSibling.innerText=" ";
  }
};

elemento_cor.addEventListener("blur", validaCompoVacio);
```



```
const elementoCorreo = document.querySelector("[name=cor_cli]");
```

```
elementoCorreo.addEventListener("blur", function(evento){  
  const CampoCorreo=evento.target;  
  const valorCampoCorreo=evento.target.value;  
  if(valorCampoCorreo.length===0){  
    CampoCorreo.classList.add("invalida");  
    CampoCorreo.nextElementSibling.classList.add("error");  
    CampoCorreo.nextElementSibling.innerText="Valor agregado no es valido";  
  }else{  
    CampoCorreo.classList.remove("invalida");  
    CampoCorreo.nextElementSibling.classList.remove("error");  
    CampoCorreo.nextElementSibling.innerText=" ";  
  }  
});
```

antes de la etiqueta ----->beforebegin
al inicio de la etiqueta ----->afterbegin
antes de cerrar la etiqueta ----->beforeend
al terminar la etiqueta ----->afterend