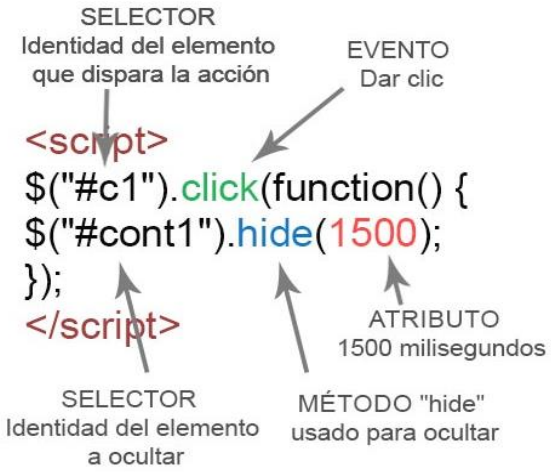
INSTITUTO DEL SUR

CURSO: **LENGUAJE DE PROGRAMACIÓN WEB III**

TEMA: **Selección de Elementos**

INSTITUTO DEL SUR

jQuery



Selección de Elementos

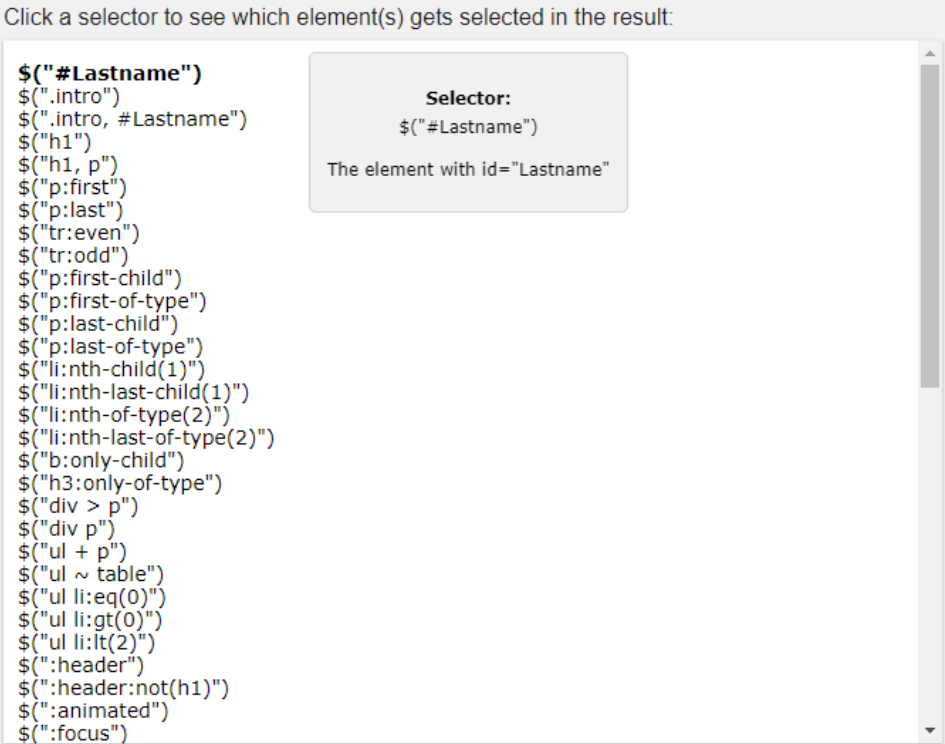
INSTITUTO DEL SUR

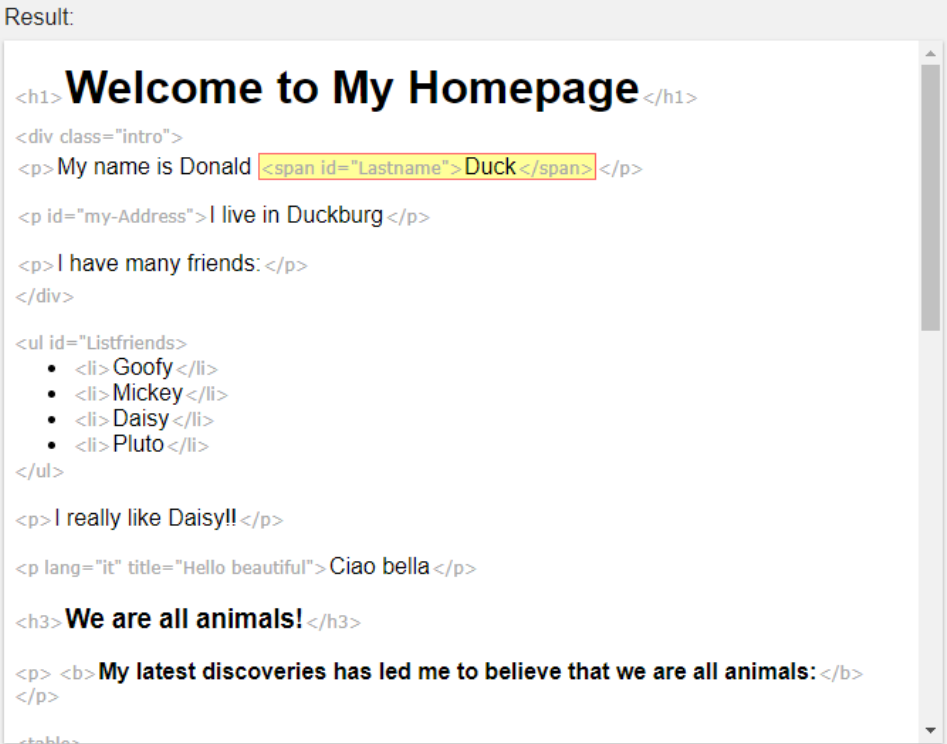
Selección de Elementos

- El concepto más básico de **jQuery** es el de *"**seleccionar algunos elementos y realizar acciones con ellos**"*. La biblioteca soporta gran parte de los selectores CSS3 y varios más no estandarizados.
- En **api.jquery.com/category/selectors** se puede encontrar una completa referencia sobre los selectores de la biblioteca.
- A continuación, se muestran algunas técnicas comunes para la selección de elementos:

Selección de Elementos

Utilice el **jQuery Selector Tester** para demostrar los diferentes selectores.
https://www.w3schools.com/jquery/trysel.asp





Selección de Elementos

Selección de elementos en base a su id

```
$('#myId’);
// notar que los IDs deben ser únicos por página
```

```
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/2.1.4/jquery.min.js">
</script>
<script type="text/javascript">
 $(document).ready(function() {
                $('#myId').click(function () {
                $('#myId').css("color","red");
 });
 });
 </script>
```

Selección de Elementos

Selección de elementos en base al nombre de clase

```
$('div.myClass');
// si se especifica el tipo de elemento,
// se mejora el rendimiento de la selección
```

```
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/2.1.4/jquery.min.js">
</script>
<script type="text/javascript">
 $(document).ready(function() {
                $('.myClass').click(function () {
                $('.myClass').css("color","blue");
 });
 });
 </script>
```

Selección de Elementos

Selección de elementos por su atributo

```
$('input[name=first_name]');

// tenga cuidado, que puede ser muy lento
```

```
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/2.1.4/jquery.min.js">
</script>
<script type="text/javascript">
 $(document).ready(function() {
                $('input').click(function () {
                $('input').css("color","green");

      $("input").attr ("value","Ingrese su nombre completo");
 });
 });
 </script>
```

INSTITUTO DEL SUR

Selección de Elementos

Pseudo-selectores

```
// selecciona el primer elemento <a> con la clase 'external'
$('a.external:first');

// selecciona todos los elementos <tr> impares de una tabla
$('tr:odd');

// selecciona todos los elementos del tipo input dentro del formulario #myForm
$('#myForm :input');

// selecciona todos los divs visibles
$('div:visible');

// selecciona todos los divs excepto los tres primeros
$('div:gt(2)');

// selecciona todos los divs actualmente animados
$('div:animated');
```

INSTITUTO DEL SUR

Elección de selectores

- La elección de buenos selectores es un punto importante cuando se desea mejorar el rendimiento del código. Una pequeña especificidad — por ejemplo, incluir el tipo de elemento (como div) cuando se realiza una selección por el nombre de clase — puede ayudar bastante. Por eso, es recomendable darle algunas "pistas" a jQuery sobre en que lugar del documento puede encontrar lo que desea seleccionar. Por otro lado, demasiada especificidad puede ser perjudicial.
- Un selector como #miTabla thead tr th.especial es un exceso, lo mejor sería utilizar #miTabla th.especial.

INSTITUTO DEL SUR

Elección de selectores

- jQuery ofrece muchos selectores basados en atributos, que permiten realizar selecciones basadas en el contenido de los atributos utilizando simplificaciones de expresiones regulares.

// encontrar todos los **<a>** cuyo atributo **rel** terminan en "**algo**"

$("a[rel$=algo']");

- Estos tipos de selectores pueden resultar útiles pero también ser muy lentos.
- Cuando sea posible, es recomendable realizar la selección utilizando IDs, nombres de clases y nombres de etiquetas.

INSTITUTO DEL SUR

Comprobar selecciones

- Una vez realizada la selección de los elementos, querrá conocer si dicha selección entregó algún resultado. Para ello, pueda que escriba algo así:

```
if ($('div.foo')) { ... }
```

- Sin embargo esta forma no funcionará. Cuando se realiza una selección utilizando $(), siempre es devuelto un objeto, y si se le evalúa, éste siempre devolverá true. Incluso si la selección no contiene ningún elemento, el código dentro del bloque if se ejecutará.

INSTITUTO DEL SUR

Comprobar selecciones

- En lugar de utilizar el código mostrado, lo que se debe hacer es preguntar por la cantidad de elementos que posee la selección que se ejecutó. Esto es posible realizarlo utilizando la propiedad JavaScript **length**.
- Si la respuesta es 0, la condición evaluará falso, caso contrario (más de 0 elementos), la condición será verdadera.
- **Evaluar si una selección posee elementos:**

```
if ($('div.foo').length) { ... }
```

INSTITUTO DEL SUR

Guardar selecciones

- Cada vez que se hace una selección, una gran cantidad de código es ejecutado. jQuery no guarda el resultado por si solo, por lo tanto, si va a realizar una selección que luego se hará de nuevo, deberá salvar la selección en una variable.
- Guardar selecciones en una variable

```
var $divs = $('div’);
```

- Una vez que la selección es guardada en la variable, se puede utilizar en conjunto con los métodos de jQuery y el resultado será igual que utilizando la selección original.

Refinamiento y filtrado de selecciones

- A veces, puede obtener una selección que contiene más de lo que necesita; en este caso, es necesario refinar dicha selección. jQuery ofrece varios métodos para poder obtener exactamente lo que desea.

- **Refinamiento de selecciones**

```
$('div.foo').has('p');          // el elemento div.foo contiene elementos <p>

$('h1').not('.bar');            // el elemento h1 no posee la clase 'bar'

$('ul li').filter('.current');  // un ítem de una lista desordenada que posee la clase 'current'

$('ul li').first();             // el primer ítem de una lista desordenada

$('ul li').eq(5);               // el sexto ítem de una lista desordenada
```

Selección de elementos de un formulario

- jQuery ofrece varios **pseudo-selectores** que ayudan a encontrar elementos dentro de los formularios, éstos son especialmente útiles ya que dependiendo de los estados de cada elemento o su tipo, puede ser difícil distinguirlos utilizando selectores CSS estándar.

Selección de elementos de un formulario

- :button Selecciona elementos <button> y con el atributo type='button'
- :checkbox Selecciona elementos <input> con el atributo type='checkbox'
- :checked Selecciona elementos <input> del tipo checkbox seleccionados
- :disabled Selecciona elementos del formulario que están deshabitados
- :enabled Selecciona elementos del formulario que están habilitados
- :file Selecciona elementos <input> con el atributo type='file'
- :image Selecciona elementos <input> con el atributo type='image'
- :input Selecciona elementos <input>, <textarea> y <select>
- :password Selecciona elementos <input> con el atributo type='password'
- :radio Selecciona elementos <input> con el atributo type='radio'
- :reset Selecciona elementos <input> con el atributo type='reset'
- :selected Selecciona elementos <options> que están seleccionados
- :submit Selecciona elementos <input> con el atributo type='submit'
- :text Selecciona elementos <input> con el atributo type='text'

Selección de elementos de un formulario

Utilizando pseudo-selectores en elementos de formularios

```
// obtiene todos los elementos inputs dentro del formulario #myForm

$('#myForm :input');
```

Ejemplos prácticos del uso de JQuery

Efectos y animaciones

Ocultar

- En el primero de ellos al dar clic en el botón "**Ocultar bloque**" usamos el evento "**click()**" para ocultar con la función "**hide()**" el bloque de ejemplo que posee el identificador "cont1". Usamos el siguiente código:

```
<script>
$("#c1").click(function() {
$("#cont1").hide(1500);
});
</script>
```

Ejemplos prácticos del uso de JQuery

Efectos y animaciones

Mostrar

```
//Mostrar bloque
$("#c2").click(function() {
$("#cont1").show("slow");
});
```

Ejemplos prácticos del uso de JQuery

Efectos y animaciones

Ocultar/Mostrar

```
//Ocultar / Mostrar bloque
$("#c1b").click(function() {
$("#cont1").toggle(1500);
},function(){
$("#cont1").toggle(1500);
});
```

Ejemplos prácticos del uso de JQuery

Efectos y animaciones

Cambiar de tamaño

```
//Cambiar tamaño
$("#c3").click(function() {
$("#cont1").animate({fontSize:'2.4em',width:400,padding:24}, 2500);
});
```

Ejemplos prácticos del uso de JQuery

Efectos y animaciones

Ocultar con Efecto

```
//Ocultar con FadeOut
$("#c4").click(function() {
$("#cont1").fadeToggle(3000);
});
```

Ejemplos prácticos del uso de JQuery

Efectos y animaciones

Mostrar con Efecto

```
//Mostrar con FadeIn
$("#c5").click(function() {
$("#cont1").fadeIn(3000);
});
```

Ejemplos prácticos del uso de JQuery

Efectos y animaciones

Mover

```
//Mover
$("#c6").click(function(){

                $("#cont1").animate({opacity: "0.1", left: "+=400",fontSize:'1em',width: "200"}, 1200)
                .animate({opacity: "0.4", top: "+=160", height: "20", width: "80",fontSize:'0.5em'}, "slow")
                .animate({opacity: "1", left: "0", width: "260"}, "slow")
                .animate({top: "0",width: "260",fontSize:'1.2em'}, "fast")
                .slideUp()
                .slideDown(1800)
                return false;

        });
```

Ejemplos prácticos del uso de JQuery

Efectos y animaciones

Cambiar Estilo CSS

```
//Cambiar estilo CSS
$("#c7").click(function() {
$("#cont1").css({'border':'4px solid #b7e5ff','color':'#cc3333','font-weight':'bold','background-color':'#ffffff'});
});
```

Ejemplos prácticos del uso de JQuery

Efectos y animaciones

Copiar y escribir texto y HTML con JQuery

```
//Copia texto del H1 y lo escribe en cont3
$("#c12").click(function() {
texto1=$("h1").text();
$("#cont3").text(texto1);
});
```

Ejemplos prácticos del uso de JQuery

Efectos y animaciones

Copiar y escribir texto y HTML con JQuery

```
//Copia texto del H1 y lo escribe en cont3
$("#c12").click(function() {
texto1=$("h1").text();
$("#cont3").text(texto1);
});

//Copia HTML del H1 y lo escribe en cont4
$("#c13").click(function() {
html1=$("h1").clone();
$("#cont4").html(html1);
});
```

Ejemplos prácticos del uso de JQuery

Efectos y animaciones

Usar contenedor o bloque como evento

En el siguiente ejemplo usamos como evento en vez de un enlace o un botón, el mismo contenedor o bloque. Usamos tres métodos encadenados, hide(), delay() y show(), para que al dar clic encima se oculte el bloque. Vuelve a aparecer lentamente después de dos segundos.

```
//Toca para ocultarme
$("#cont2").click(function() {
$(this).hide().delay(1500).show(1500);
});
```

Ejemplos prácticos del uso de JQuery

Efectos y animaciones

Usar evento hover

Otros de los eventos muy utilizados es hover, que permite efectuar cualquier acción al situar el cursor del ratón encima de un elemento. Por ejemplo al situarlo encima de este párrafo o bloque de texto, se muestra su fondo de color amarillo y retorna a su color predeterminado al retirarlo de encima. Para eso usamos la clase definida hover con la función addClass(), a continuación removeClass().

Hover al igual que click acepta dos funciones.

```
//Fondo amarillo
$(".yellow").hover(
        function() { $(this).addClass("hover"); },
        function() { $(this).removeClass("hover");
}
);
```

Ejemplos prácticos del uso de JQuery

Efectos y animaciones

Usar eventos del teclado

Otros eventos que se emplean mucho en las páginas son los que desencadenan acciones en el teclado. En el siguiente ejemplo usamos los eventos **keyup()** y **keydown(),** para contar la cantidad de caracteres en el cuadro de texto y mostrarlos.

Contador de caracteres

```
//Contador de caracteres
$('textarea').keyup(updateCount);
$('textarea').keydown(updateCount);
function updateCount() {
    var cs = $(this).val().length;
    $('#characters').text(cs);
}
```

Ejemplos prácticos del uso de JQuery

Efectos y animaciones

Cargar y mostrar archivos con JQuery y Ajax

JQuery posee varios métodos de Ajax que permiten cargar contenido de forma asíncrona, sin tener que recargar la página. Es de mucha utilidad para facilitar a los lectores cargar contenido adicional, cuando sea necesario.

A continuación se muestran dos ejemplos.

En el primero de ellos usamos un botón para cargar y mostrar contenido de otra página de este sitio, usando el método load().

```
//Cargar contenido externo
$("#c8").click(function() {
$("#noticias").load("../facebook/problemas-simbolos.php");
});

//Ocultar
$("#c9").click(function() {
$("#noticias").hide(1500);
});
```

Ejemplos prácticos del uso de JQuery

Efectos y animaciones

Cargar y mostrar archivos con JQuery y Ajax

Con el siguiente botón usamos el método get() para cargar datos de otra página y mostrarlo en un cuadro de alerta.
En este ejemplo es el código fuente de esta misma página, aunque también puede ser de cualquier otra página de internet.

```
//GET (Ver codigo fuente)
$("#c10").click(function(){
   $.get('https://norfipc.com/codigos/jquery-ejemplos-practicos-usar-paginas-web.php', function(data, status){
       alert("Data: " + data + "\nStatus: " + status);
   });
});
```

Ejemplos prácticos del uso de JQuery

Efectos y animaciones

Cargar y mostrar archivos con JQuery y Ajax

En dispositivos móviles el cuadro es muy pequeño y reducido.
Por último usando el siguiente botón podemos saber la versión de JQuery que empleamos.

```
//Version de JQuery
$("#c11").click(function() {
var version=$().jquery;
alert("Estas usando la version: " +version+" de JQuery");
});
```