

## TAREA 3: Mostrar texto oculto - Crear menú desplegable - Crear SlideShow

### MOSTRAR TEXTO OCULTO

Abra el archivo `/ejercicios/index.html` en el navegador. Realice el ejercicio utilizando el archivo `/ejercicios/js/blog.js`. La tarea es añadir alguna interactividad a la sección blog de la página:

- Al hacer click en alguno de los titulares del div `#blog`, se debe mostrar el párrafo correspondiente con un efecto de deslizamiento;
- Al hacer click en otro titular, se debe ocultar el párrafo mostrado con un efecto de deslizamiento y mostrar nuevamente el párrafo correspondiente también con un efecto de deslizamiento. Ayuda: No se olvide de utilizar el selector `:visible`.

```
Alberto Martínez Pérez *  
$(document).ready(function() {  
  $('#blog h3').click(function(e) {  
    e.preventDefault();  
    const paragraph = $(this).next('p'); // Selecciona el párrafo correspondiente al titular clicado  
    if (paragraph.is(':visible')) {  
      paragraph.slideUp(); // Si el párrafo está visible, lo oculta con un efecto de deslizamiento  
    } else {  
      $('#blog p:visible').slideUp(); // Oculta cualquier párrafo visible  
      paragraph.slideDown(); // Muestra el párrafo correspondiente con un efecto de deslizamiento  
    }  
  });  
});
```

Si ahora hacemos clic en los elementos `h3`, veremos el efecto de deslizamiento:

### Blog

- **Blog post 1**

Lorem ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

- **Blog post 2**

- **Blog post 3**

## CREAR UN MENÚ DESPLEGABLE

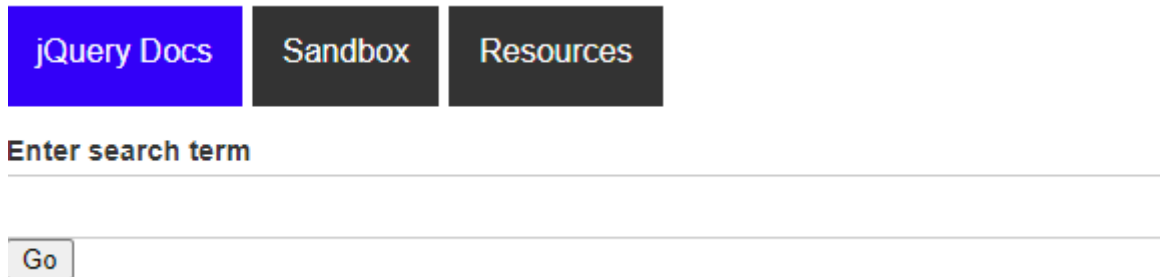
Abra el archivo `/ejercicios/index.html` en el navegador. Realice el ejercicio utilizando el archivo `/ejercicios/js/navigation.js`. La tarea es poder desplegar los ítems del menú superior de la página:

- Al pasar el puntero del ratón por encima de un ítem del menú, se debe mostrar su submenú en caso que exista;
- Al no estar más encima de un ítem, el submenú se debe ocultar.

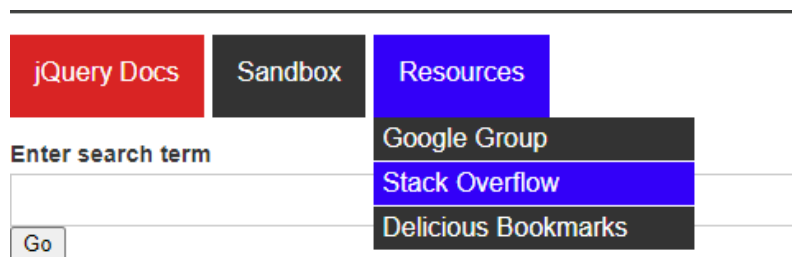
Para poder realizarlo, utilice el método `$.fn.hover` para añadir o remover una clase en el submenú para poder controlar si debe estar oculto o visible (El archivo `/ejercicios/css/styles.css` incluye una clase `"hover"` para este propósito).

```
Alberto Martínez Pérez *  
$(document).ready(function() {  
    // Seleccionar los elementos de navegación  
    const elementosNav = $('#nav li');  
    elementosNav.hover(mostrarSubMenu, ocultarSubMenu);  
    // Función para mostrar el submenú  
    1+ usages new *  
    function mostrarSubMenu() {  
        // Añade la clase hover  
        $(this).addClass('hover');  
        const elementosHijos = $(this).find('ul');  
        // Si existen elementos hijo  
        if (elementosHijos.length > 0) {  
            elementosHijos.slideDown();  
        }  
    }  
  
    // Función para ocultar el submenú  
    1+ usages new *  
    function ocultarSubMenu() {  
        // Elimina la clase hover  
        $(this).removeClass('hover');  
        const elementosHijos = $(this).find('ul');  
        // Si existen elementos hijo  
        if (elementosHijos.length > 0) {  
            elementosHijos.slideUp();  
        }  
    }  
});
```

Si ahora accedemos a la web y hacemos hover sobre los elementos del menú de navegación:



En el caso de tener una lista desplegable:



## CREAR UN SLIDESHOW

Abra el archivo `/ejercicios/index.html` en el navegador. Realice el ejercicio utilizando el archivo `/ejercicios/js/slideshow.js`. La tarea es añadir un slideshow a la página con JavaScript.

1. Mover el elemento `#slideshow` a la parte superior de la página;
2. Escribir un código que permita mostrar los ítems de forma cíclica, mostrando un ítem por unos segundos, luego ocultándolo con un efecto `fade out` y mostrando el siguiente con un efecto `fade in`;
3. Una vez llegado al último ítem de la lista, comenzar de nuevo con el primero;

Para un desafío mayor, realice un área de navegación por debajo del slideshow que muestre cuantas imágenes existen y en cual se encuentra (ayuda: `$.fn.prevAll` puede resultar útil).

```
Alberto Martínez Pérez *  
$(document).ready(function() {  
    // Variable para el temporizador  
    let timeout = false;  
  
    // Variable para el modo manual de navegación  
    let modoManual = false;  
  
    // Selecciona el slideshow y lo mueve al contenedor principal  
    const slideshow = $('#slideshow').prependTo('#main');  
  
    // Oculta todos los elementos li del slideshow  
    const elementos = slideshow.find('li').hide();  
  
    // Obtiene el total de elementos  
    const total = elementos.length;  
  
    // Crea un div para el contador y lo inserta después del slideshow  
    const contador = $('<div/>').insertAfter(slideshow);  
  
    // Crea un div para los controles y lo inserta después del slideshow  
    const controles = $('<div/>').insertAfter(slideshow);  
});
```



```
// Crea un botón 'anterior' y lo añade a los controles
const btnAnterior = $('<input/>', {
  type: 'button',
  value: 'Anterior'
}).appendTo(controles);

// Crea un botón 'siguiente' y lo añade a los controles
const btnSiguiente = $('<input/>', {
  type: 'button',
  value: 'Siguiente'
}).appendTo(controles);

// Función para actualizar el contador con el número de foto actual
1+ usages new *
function actualizarContador(num) {
  contador.text('Foto ' + num + ' de ' + total);
}

// Función para obtener el elemento a mostrar, según la dirección (anterior/siguiente)
1+ usages new *
function obtenerElemento($item, trav) {
  const elementoDevuelto = $item[trav]();
  return elementoDevuelto.length ?
    elementoDevuelto :
    elementos[(trav == 'next') ? 'first' : 'last']();
}
```



```
// Función para mostrar un elemento, con fade in/out y llamada a callbackFade
1+ usages  👤 Alberto Martínez Pérez *
function mostrarElemento($elementoActual, $elementoAMostrar) {
    $elementoAMostrar = $elementoAMostrar || obtenerElemento($elementoActual, trav: 'next');

    $elementoActual.fadeOut(500, function() {
        $elementoAMostrar.fadeIn(500, callbackFade);
    });
}

// Función de callback para el fade, maneja la lógica de cambio automático de elementos
1+ usages  👤 Alberto Martínez Pérez *
function callbackFade() {
    if (modoManual) { return; }

    const $this = $(this);
    const siguiente = obtenerElemento($this, trav: 'next');
    const num = $this.prevAll().length + 1;

    // Actualiza el contador con el número de foto actual
    actualizarContador(num);

    // Establece el temporizador para mostrar el siguiente elemento en 5 segundos
    timeout = setTimeout( handler: function() {
        mostrarElemento($this, siguiente);
    }, timeout: 5000);
}
```

```
// Función para manejar el click en los botones 'anterior' y 'siguiente'
1+ usages  ✎ Alberto Martínez Pérez *
function manejarClickBoton(e) {
    clearTimeout(timeout);
    modoManual = true;

    const elementoActual = elementos.filter(':visible');
    const elementoAMostrar = obtenerElemento(elementoActual, e.data.direccion);

    mostrarElemento(elementoActual, elementoAMostrar);
    const num = elementoAMostrar.prevAll().length + 1;
    actualizarContador(num);
}

// Eventos clic para los botones 'anterior' y 'siguiente', con manejo de dirección
btnAnterior.bind('click', { direccion: 'prev' }, manejarClickBoton);
btnSiguiente.bind('click', { direccion: 'next' }, manejarClickBoton);


// Muestra el primer elemento del slideshow con fade in y llama a callbackFade
elementos.eq(0).fadeIn(500, callbackFade);
});
```

Al acceder a la web veremos un slideshow en la parte superior:

# jQuery Fundamentals

[jQuery Docs](#)
[Sandbox](#)
[Resources](#)

## Vegetables



Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

[Anterior](#)
[Siguiente](#)

Foto 2 de 3