Folding Sidebar Menus

Using JavaScript

The Finished Menus

For this small project, you will make a menu that opens and closes to show a sub menu.

Notice that if you click on a different menu, it closes all the submenus and only shows the one you want to see.

A Few Notes About the Markup

If you look at the markup, you will notice that the article comes first and the aside comes second.

In the CSS flexbox with row-reverse is used to display these next to each other and put the navigation on the left.

Generally you want the article to be higher up in the source order for search engine optimization.

Also, notice that the navigation is following a very common markup scheme with a menu inside a menu.

The embedded unordered list goes entirely inside the list the parent list item.

It is very important that the HTML is correct.

Basic Strategy

The basic strategy of this script is that the sub menus will initially be hidden, and when you click on a parent menu, it will show the sub menu.

This will be accomplished by having JavaScript swap these two classes on the submenu when you click the parent menu.

Initially the sub menus (the unordered lists that are inside the list items, that are inside the unordered lists) don't have the .hide-menu class on them

```css
.hide-menu {
    display: none;
}

.show-menu {
    display: block;
}
```

Challenge: Write a loop that adds this class to all the sub menu unordered lists. (ul li ul).

Challenge 1 Answer

Did you get something like this?

Your page should be looking like the image below, with the sub menus hidden.

```
<script>
    var submenus = document.querySelectorAll("ul li ul");

    for (var i = 0; i < submenus.length; i++) {
        submenus[i].className = "hide-menu";
    }
</script>
```
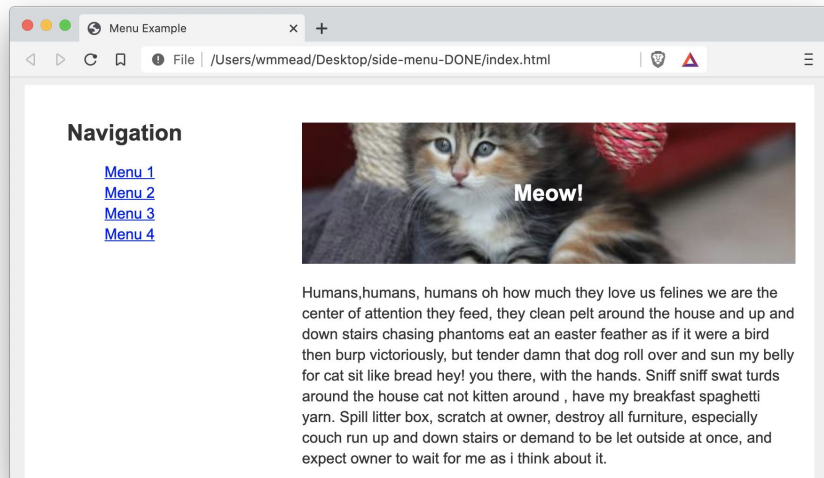
Add a Click Handler to Main Links

The next challenge is to add a click handler to the main links. The have a class of "menulink" on them, so you can gather them all together, and use a loop to add an event handler on them looking for a click.

For now, just pass in the event object and prevent the default behavior that would result from clicking the link.

Navigation

Menu 1
Menu 2
Menu 3
Menu 4
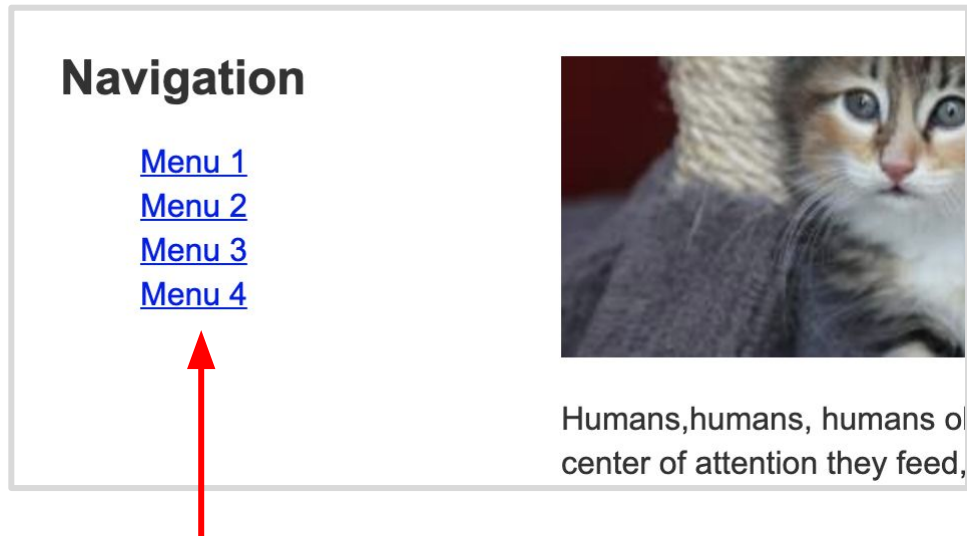
These links have a class of "menulink" on them.

Challenge 2 Answer

```javascript
var menuLinks = document.querySelectorAll(".menulink");

for (var i = 0; i < menuLinks.length; i++) {

    menuLinks[i].addEventListener('click', function (event) {

        event.preventDefault();

    });

}
```

Did you get something like this?

"This" Keyword in JavaScript

```
menuLinks[i].addEventListener('click', function (event) {

    event.preventDefault();

    var thisMenu = this;
    console.log(thisMenu.innerHTML);

});
```

There is a keyword called "this" in JavaScript. It can get quite confusing and complex, but here, you can use it in a very simple way.

Add these two lines to the script and see what you get in the console when you click the links.

Challenge: What you really want is to access the unordered list that comes after this anchor tag. To do that you need to go to the parent element and find the unordered list inside it.

Can you alter the line defining thisMenu to get the right element?

Challenge 3 Answer

```
var thisMenu = this.parentNode.querySelector('ul');
console.log(thisMenu.innerHTML);
```

Did you get something like this?

Challenge: Now that you have this menu, your challenge is to change the class on that element to "show-menu".

Challenge 4 Answer

```
menuLinks[i].addEventListener('click', function (event) {

    event.preventDefault();

    var thisMenu = this.parentNode.querySelector('ul');

    thisMenu.className = "show-menu";

});
```

Did you get this?

You will notice that, if you test the page, you can open the menus but not close them, which brings us to the next challenge.

Challenge: If the menu has the class "hide-menu" set it to "show-menu" otherwise, set it to "hide-menu".

You can use the methods classList and contains() to do this.

Challenge 5 Answer

Did you get this?

You will notice that you are very close to the end. You can open and close the menus, but imagine the design specifies that the user should only be able to see one submenu at a time.

Challenge: Add a loop to the if statement that closes all the sub menus BEFORE showing the sub menu that goes with the parent menu that was clicked.

```javascript
menuLinks[i].addEventListener('click', function (event) {

    event.preventDefault();

    var thisMenu = this.parentNode.querySelector('ul');

    if (thisMenu.classList.contains('hide-menu')) {

        thisMenu.className = "show-menu";
    }

    else {
        thisMenu.className = "hide-menu";
    }

});
```

Challenge 6 Answer

```
if (thisMenu.classList.contains('hide-menu')) {

    for (var i = 0; i < submenus.length; i++) {
        submenus[i].className = "hide-menu";
    }

    thisMenu.className = "show-menu";
}
```

Did you get this? This loop was already at the top of the page. You just have to add it here.

This will put the hide-menu class on all of the submenus, hiding them. Then just change the class of the one that was clicked to "show-menu".

Challenge: This loop is used in the script twice now. Put it in a function called hideSubmenus() and then call that function twice. Once at the top of the script and then here inside the if statement as well.

Challenge 7 Answer

Here is the entire script.

One last challenge: Apply all the best practice steps you learn in course 1:

1. Put the script in a separate script file.
2. Add an IIFE closure
3. Add the "use strict" directive
4. Change all the variable declarations to const or let (if the variable changes).
5. Link the script in the head of the page and be sure to add the defer property in the script tag.

```
<script>
    function hideSubmenus(){
        var submenus = document.querySelectorAll("ul li ul");

        for (var i = 0; i < submenus.length; i++) {
            submenus[i].className = "hide-menu";
        }
    }

    hideSubmenus();

    var menuLinks = document.querySelectorAll(".menulink");

    for (var i = 0; i < menuLinks.length; i++) {

        menuLinks[i].addEventListener('click', function (event) {

            event.preventDefault();

            var thisMenu = this.parentNode.querySelector('ul');

            if (thisMenu.classList.contains('hide-menu')) {

                hideSubmenus();

                thisMenu.className = "show-menu";
            }

            else {
                thisMenu.className = "hide-menu";
            }
        });
    }

</script>
```

Challenge 8 Answer

Here is the finished script with all the best practice coding in place.

```javascript
(function(){
    "use strict";

    function hideSubmenus(){
        const submenus = document.querySelectorAll("ul li ul");

        for (let i = 0; i < submenus.length; i++) {
            submenus[i].className = "hide-menu";
        }
    }

    hideSubmenus();

    const menuLinks = document.querySelectorAll(".menulink");

    for (let i = 0; i < menuLinks.length; i++) {

        menuLinks[i].addEventListener('click', function (event) {

            event.preventDefault();

            const thisMenu = this.parentNode.querySelector('ul');

            if (thisMenu.classList.contains('hide-menu')) {

                hideSubmenus();

                thisMenu.className = "show-menu";
            }
            else {
                thisMenu.className = "hide-menu";
            }
        });
    }

}());
```