This JavaScript code utilizes the jQuery library and provides functionality for a few interactive elements on a website, such as a spinner loader, sticky navigation bar, dropdown menus, back-to-top button, and carousels. Here's a breakdown of the functionality:

1. **Spinner Loader**:
   * The code sets a small delay (setTimeout function) to remove a loading spinner (#spinner) from the page once the document is ready.
   * It ensures that if an element with the ID spinner exists, the "show" class is removed to hide the spinner after the page loads.
2. **WOW.js Initialization**:
   * new WOW().init(); initializes the WOW.js library, which is typically used for scrolling animations, so elements can animate when they come into view.
3. **Sticky Navbar**:
   * The sticky navigation bar will adjust its position based on scroll position. If the user scrolls down more than 300 pixels, the navbar is shown (top: 0px). Otherwise, it's hidden by moving it upwards (top: -100px).
   * This adds a smooth appearance of the navbar when scrolling down the page.
4. **Dropdown Menu on Hover**:
   * This block modifies the behavior of the dropdown menu. If the window width is greater than or equal to 992 pixels, the dropdown menu is triggered on hover rather than on click.
   * When hovered over, the dropdown menu is shown, and when the mouse leaves, the menu hides again.
5. **Back-to-Top Button**:
   * A button with the class .back-to-top is shown when the user scrolls down more than 300 pixels.
   * Clicking the button smoothly scrolls the page back to the top over 1.5 seconds using the easeInOutExpo animation.
6. **Header Carousel**:
   * This block uses Owl Carousel (a jQuery plugin) to create an autoplaying carousel with transition speed of 1500ms. The carousel only displays one item at a time (items: 1) and includes navigation arrows.
   * The arrows are represented by icons from Bootstrap Icons (bi-chevron-left and bi-chevron-right).
7. **Testimonials Carousel**:
   * Another carousel specifically for testimonials, it also uses Owl Carousel but with a slightly faster speed (1000ms) and a center configuration that highlights the central testimonial.
   * The number of items shown changes based on screen width:
     + 1 item for small screens (below 768px),
     + 2 items for medium screens (768px and above),
     + 3 items for larger screens (992px and above).

This script helps in improving the website's user experience by adding smooth transitions, animations, and responsive elements, making it more engaging and visually appealing.

Sure! I’ll explain each line of the code for you.

javascript

Copy code

(function ($) {

"use strict";

This is an Immediately Invoked Function Expression (IIFE), which is a function that runs immediately when defined. It ensures that the code inside doesn’t interfere with other scripts on the page. The $ refers to the jQuery object, ensuring jQuery can be used safely within the function.

"use strict"; enables strict mode, which helps catch common coding mistakes and “unsafe” actions like using undeclared variables.

Spinner

javascript

Copy code

var spinner = function () {

setTimeout(function () {

if ($('#spinner').length > 0) {

$('#spinner').removeClass('show');

}

}, 1);

};

spinner();

var spinner = function () {...}: Defines a function spinner that will hide the spinner (loading indicator) on the page.

setTimeout(function () {...}, 1);: Sets a timeout of 1 millisecond to execute the anonymous function. It's delayed slightly to ensure the page is fully loaded.

if ($('#spinner').length > 0): Checks if an element with ID spinner exists on the page.

$('#spinner').removeClass('show');: If the spinner element exists, it removes the class show, which hides the spinner.

spinner();: This calls the spinner function, hiding the spinner immediately after the page load.

WOW.js Initialization

javascript

Copy code

new WOW().init();

new WOW().init();: Initializes the WOW.js library. WOW.js enables animations when scrolling down the page. This line activates the animations for any elements that are configured to use it.

Sticky Navbar

javascript

Copy code

$(window).scroll(function () {

if ($(this).scrollTop() > 300) {

$('.sticky-top').css('top', '0px');

} else {

$('.sticky-top').css('top', '-100px');

}

});

$(window).scroll(function () {...});: Adds an event listener for the scroll event on the window object. The function inside will run whenever the user scrolls the page.

$(this).scrollTop() > 300: Checks if the user has scrolled more than 300 pixels down the page.

$('.sticky-top').css('top', '0px');: If the page is scrolled more than 300 pixels, the element with class sticky-top is positioned at the top of the page (top: 0px), making it visible.

$('.sticky-top').css('top', '-100px');: If the page is scrolled less than 300 pixels, the element is hidden by moving it up off-screen (top: -100px).

Dropdown Menu on Hover

javascript

Copy code

const $dropdown = $(".dropdown");

const $dropdownToggle = $(".dropdown-toggle");

const $dropdownMenu = $(".dropdown-menu");

const showClass = "show";

$(window).on("load resize", function() {

if (this.matchMedia("(min-width: 992px)").matches) {

$dropdown.hover(

function() {

const $this = $(this);

$this.addClass(showClass);

$this.find($dropdownToggle).attr("aria-expanded", "true");

$this.find($dropdownMenu).addClass(showClass);

},

function() {

const $this = $(this);

$this.removeClass(showClass);

$this.find($dropdownToggle).attr("aria-expanded", "false");

$this.find($dropdownMenu).removeClass(showClass);

}

);

} else {

$dropdown.off("mouseenter mouseleave");

}

});

const $dropdown = $(".dropdown");: Selects all elements with the class dropdown and stores them in the $dropdown variable.

const $dropdownToggle = $(".dropdown-toggle");: Selects all elements with class dropdown-toggle (button that triggers dropdowns).

const $dropdownMenu = $(".dropdown-menu");: Selects all elements with class dropdown-menu (the actual dropdown content).

const showClass = "show";: Defines the class show to be used later for showing the dropdown.

$(window).on("load resize", function() {...});: Adds an event listener for both the load and resize events. This ensures the following functionality works whenever the page is loaded or resized.

if (this.matchMedia("(min-width: 992px)").matches): Checks if the window width is at least 992px (desktop view). If so, dropdown menus will be activated on hover.

$dropdown.hover(function() {...}, function() {...});: Adds hover event listeners to dropdown elements. When the user hovers over the element, the first function is executed, and when the hover ends, the second function is executed.

$this.addClass(showClass);: Adds the class show to the dropdown, making it visible.

$this.find($dropdownToggle).attr("aria-expanded", "true");: Updates the aria-expanded attribute to true, indicating the dropdown is expanded (for accessibility).

$this.find($dropdownMenu).addClass(showClass);: Adds the show class to the dropdown menu, making it visible.

When the hover ends (mouseleave), the dropdown is hidden by removing the show class and updating the aria-expanded attribute to false.

$dropdown.off("mouseenter mouseleave");: For screen widths below 992px, it disables the hover effect by removing the event listeners.

Back to Top Button

javascript

Copy code

$(window).scroll(function () {

if ($(this).scrollTop() > 300) {

$('.back-to-top').fadeIn('slow');

} else {

$('.back-to-top').fadeOut('slow');

}

});

$('.back-to-top').click(function () {

$('html, body').animate({scrollTop: 0}, 1500, 'easeInOutExpo');

return false;

});

$(window).scroll(function () {...});: Adds a scroll event listener to the window object.

$(this).scrollTop() > 300: Checks if the page is scrolled more than 300 pixels.

$('.back-to-top').fadeIn('slow');: If it is, the "back-to-top" button fades in.

$('.back-to-top').fadeOut('slow');: If not, the button fades out.

$('.back-to-top').click(function () {...});: Adds a click event listener to the "back-to-top" button.

$('html, body').animate({scrollTop: 0}, 1500, 'easeInOutExpo');: Animates the page to scroll back to the top in 1500ms (1.5 seconds) with the easeInOutExpo easing function.

return false;: Prevents the default action of the button (i.e., prevents it from triggering any anchor link behavior).

Header Carousel

javascript

Copy code

$(".header-carousel").owlCarousel({

autoplay: true,

smartSpeed: 1500,

items: 1,

dots: false,

loop: true,

nav : true,

navText : [

'<i class="bi bi-chevron-left"></i>',

'<i class="bi bi-chevron-right"></i>'

]

});

$(".header-carousel").owlCarousel({...});: Initializes the Owl Carousel plugin for elements with class header-carousel.

autoplay: true: Enables autoplay so the carousel slides automatically.

smartSpeed: 1500: Sets the transition speed to 1500 milliseconds (1.5 seconds).

items: 1: Displays only one item at a time in the carousel.

dots: false: Disables the display of dots for navigation.

loop: true: Enables infinite looping of the carousel items.

nav: true: Enables the next/previous navigation buttons.

navText: [...]: Sets the HTML for the navigation buttons, which in this case are Bootstrap Icons (bi-chevron-left and bi-chevron-right).

Testimonials Carousel

javascript

Copy code

$(".testimonial-carousel").owlCarousel({

autoplay: true,

smartSpeed: 1000,

center: true,

margin: 24,

dots: true,

loop: true,

nav : false,

responsive: {

0:{

items:1

},

768:{

items:2

},

992:{

items:3

}

}

});

Similar to the header carousel, but this is for the testimonials section.

center: true: Centers the active testimonial.

margin: 24: Adds 24 pixels of space between each testimonial.

dots: true: Enables dot navigation for the carousel.

nav: false: Disables the next/previous navigation buttons.

responsive: {...}: Adjusts the number of items displayed based on the screen width.

0:{items:1}: For screen widths below 768px, only 1 item is displayed.

768:{items:2}: For screens 768px and wider, 2 items are displayed.

992:{items:3}: For screens 992px and wider, 3 items are displayed.

This code makes use of jQuery for event handling and DOM manipulation, along with external libraries like Owl Carousel and WOW.js to create an interactive and smooth user experience.