document.addEventListener("DOMContentLoaded", () => {

  const mainContent = document.querySelector("main");

  /\* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

   \* Función para cargar y mostrar contenido dinámico

   \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \*/

  function cargarContenidoDinamico(targetId, pushToHistory = true) {

    const sectionUrl = `${targetId}.html`; // Construir la URL del contenido dinámico

    fetch(sectionUrl)

      .then((response) => {

        if (!response.ok) throw new Error(`Error al cargar ${sectionUrl}`);

        return response.text();

      })

      .then((data) => {

        console.log(`Contenido cargado para ${targetId}`);

        // Verificar si ya existe un contenedor dinámico con este ID

        let dynamicContent = document.querySelector(`#${targetId}`);

        if (!dynamicContent) {

          dynamicContent = document.createElement("div");

          dynamicContent.id = targetId; // Asignar el ID correspondiente

          dynamicContent.classList.add("dynamic-content"); // Agregar clase para estilos

          mainContent.appendChild(dynamicContent);

        }

        dynamicContent.innerHTML = data; // Actualizar el contenido HTML cargado

        // Asignar navegación interna si existe un submenú dentro del contenido cargado

        const internalMenu = dynamicContent.querySelector(".sections");

        if (internalMenu) {

          assignInternalNavigation(dynamicContent); // Configurar la navegación interna

        }

        // Ocultar otras secciones

        document.querySelectorAll("main > section, main > div").forEach((section) => {

          section.classList.add("hidden");

          section.classList.remove("active");

        });

        // Mostrar el contenido cargado

        dynamicContent.classList.remove("hidden");

        dynamicContent.classList.add("active");

        dynamicContent.scrollIntoView({ behavior: "smooth" });

        // Registrar el estado en el historial si es necesario

        if (pushToHistory) {

          history.pushState({ targetId }, "", `#${targetId}`);

        }

      })

      .catch((error) =>

        console.error(`Error al cargar el contenido dinámico:`, error)

      );

  }

  /\* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

   \* Función para manejar submenús internos (menu .sections)

   \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \*/

  function assignInternalNavigation(container) {

    const internalLinks = container.querySelectorAll(".sections a[data-target]");

    internalLinks.forEach((link) => {

      link.addEventListener("click", (event) => {

        event.preventDefault();

        const targetId = link.getAttribute("data-target");

        const targetDiv = container.querySelector(`#${targetId}`); // Buscar dentro del contenedor dinámico

        if (targetDiv) {

          console.log(`Mostrando sección interna: ${targetId}`);

          // Ocultar todas las secciones internas del contenedor

          container.querySelectorAll("div[id^='section']").forEach((div) => {

            div.classList.add("hidden");

            div.classList.remove("visible");

          });

          // Mostrar la sección seleccionada

          targetDiv.classList.remove("hidden");

          targetDiv.classList.add("visible");

          // Manejar clases activas en los enlaces

          internalLinks.forEach((l) => l.classList.remove("active")); // Eliminar `active` de todos

          link.classList.add("active"); // Agregar `active` al enlace clicado

        } else {

          console.warn(`No se encontró la sección interna: ${targetId}`);

        }

      });

    });

    // Establecer el primer enlace como activo al cargar la página

    const firstLink = container.querySelector(".sections a[data-target]");

    if (firstLink) {

      firstLink.classList.add("active"); // Hacer el enlace inicial activo

    }

    // Mostrar la primera sección por defecto

    const firstSection = container.querySelector("div[id^='section']");

    if (firstSection) {

      firstSection.classList.remove("hidden");

      firstSection.classList.add("visible");

    }

  }

  /\* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

   \* Manejo de botones y enlaces del menú

   \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \*/

  document.querySelectorAll("nav a").forEach((link) => {

    link.addEventListener("click", (event) => {

      event.preventDefault();

      const href = link.getAttribute("href");

      if (href === "blog.html") {

        console.log(`Redirigiendo al blog: ${href}`);

        window.location.href = href;

        return;

      }

      const sectionId = href.replace("#", "");

      const sectionElement = document.getElementById(sectionId);

      if (sectionId === "inicio" || sectionId === "contacto") {

        console.log(`Mostrando sección existente: ${sectionId}`);

        mostrarSeccionInicial();

        updateActiveSectionInicio(sectionElement);

        sectionElement.scrollIntoView({ behavior: "smooth" });

        return;

      }

      console.log(`Cargando contenido dinámico desde: ${sectionId}`);

      cargarContenidoDinamico(sectionId);

    });

  });

  document.querySelectorAll(".button").forEach((button) => {

    button.addEventListener("click", () => {

      const targetId = button.getAttribute("data-target");

      const sectionElement = document.getElementById(targetId);

      if (sectionElement) {

        console.log(`Mostrando sección desde botón: ${targetId}`);

        updateActiveSectionInicio(sectionElement);

        sectionElement.scrollIntoView({ behavior: "smooth" });

      } else {

        console.log(`Cargando contenido dinámico desde botón para: ${targetId}`);

        cargarContenidoDinamico(targetId);

      }

    });

  });

  /\* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

   \* Manejo de botones "Atrás"

   \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \*/

  window.addEventListener("popstate", (event) => {

    if (event.state && event.state.targetId) {

      const targetId = event.state.targetId;

      console.log(`Regresando al contenido dinámico: ${targetId}`);

      cargarContenidoDinamico(targetId, false); // Cargar sin registrar en el historial

    } else {

      console.log("Regresando a la página inicial");

      mostrarSeccionInicial(false); // Mostrar la sección inicial

    }

  });

  /\* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

   \* Función para mostrar la sección inicial

   \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \*/

  function mostrarSeccionInicial(pushToHistory = true) {

    document.querySelectorAll("main > section, main > div").forEach((section) => {

      section.classList.add("hidden");

      section.classList.remove("active");

    });

    const defaultSection = document.getElementById("inicio");

    defaultSection.classList.remove("hidden");

    defaultSection.classList.add("active");

    if (pushToHistory) {

      history.replaceState(null, "", "#inicio");

    }

  }

  /\* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

   \* Efecto Parallax

   \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \*/

  document.addEventListener("scroll", () => {

    const parallaxElements = document.querySelectorAll(".parallax");

    parallaxElements.forEach((element) => {

      const speed = 0.5;

      const offset = window.scrollY \* speed;

      element.style.backgroundPositionY = `${offset}px`;

    });

  });

  /\* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

   \* Manejo del carrusel

   \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \*/

  function initCarousel() {

    let currentIndex = 0;

    let interval;

    function moveCarousel(direction) {

      const items = document.querySelectorAll(".carousel-item");

      const totalItems = items.length;

      items[currentIndex].classList.remove("active");

      currentIndex = (currentIndex + direction + totalItems) % totalItems;

      items[currentIndex].classList.add("active");

      const inner = document.querySelector(".carousel-inner");

      inner.style.transform = `translateX(-${currentIndex \* 100}%)`;

    }

    function startAutoSlide() {

      interval = setInterval(() => {

        moveCarousel(1);

      }, 5000);

    }

    function stopAutoSlide() {

      clearInterval(interval);

    }

    const carousel = document.querySelector(".carousel");

    if (carousel) {

      carousel.addEventListener("mouseenter", stopAutoSlide);

      carousel.addEventListener("mouseleave", startAutoSlide);

    }

    startAutoSlide();

  }

  initCarousel();

  /\* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

   \* Manejo del menú móvil

   \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \*/

  const menuToggle = document.querySelector(".menu-toggle");

  const mobileMenu = document.querySelector(".menu-container.mobile-menu .menu");

  if (menuToggle && mobileMenu) {

    menuToggle.addEventListener("click", () => {

      mobileMenu.classList.toggle("active");

    });

    const menuLinks = mobileMenu.querySelectorAll("a");

    menuLinks.forEach((link) => {

      link.addEventListener("click", () => {

        mobileMenu.classList.remove("active");

      });

    });

  }

  /\* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

   \* Inicialización de la página

   \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \*/

  if (window.location.hash) {

    const targetId = window.location.hash.replace("#", "");

    console.log(`Cargando desde hash inicial: ${targetId}`);

    cargarContenidoDinamico(targetId, false);

  } else {

    mostrarSeccionInicial(true);

  }

});